



# ESCOLA NAVAL



talant de bi-faire

Departamento de Ciências do Mar

Haroun Fernandes Tlemçani

*Elaboração de um Almanaque Náutico*

*Desenvolvimento de uma rotina informática para obtenção de coordenadas celestes*

Dissertação para obtenção do grau de Mestre em Ciências Militares Navais, na  
especialidade de Marinha



Alfeite

2020





# ESCOLA NAVAL

talant de bi-faire



Haroun Fernandes Tlemçani

## *Elaboração de um Almanaque Náutico*

*Desenvolvimento de uma rotina informática para obtenção de coordenadas celestes*

Dissertação para obtenção do grau de Mestre em Ciências Militares Navais, na  
especialidade de Marinha

**Orientação de:** CMG Costa Canas

**Coorientação de:** CFR Lourenço Gorricha

O aluno mestrando,

Os orientadores,

---

ASPOF Fernandes Tlemçani

---

CMG Costa Canas/CFR Lourenço Gorricha

**Alfeite**

**2020**



“No man is an island entire of itself; every man  
is a piece of the continent, a part of the main;  
if a clod be washed away by the sea, Europe  
is the less, as well as if a promontory were, as  
well as any manner of thy friends or of thine  
own were; any man's death diminishes me,  
because I am involved in mankind.  
And therefore never send to know for whom  
the bell tolls; it tolls for thee. “

John Donne



Ao meu avô Para, ao meu camarada Fonseca e ao meu amigo Filipe.



## Agradecimentos

O elaborar de uma dissertação de mestrado é um processo longo e demorado. Exige paciência, ambição e persistência. Para conseguir o meu melhor desempenho no realizar de um projecto desta dimensão comecei, como sabem os mais próximos de mim, um longo processo de auto-análise e investigação sobre os temas da psicologia e em particular da motivação. O que é? Como se gera? Pode ser trabalhada? Nesta procura, acabei por descobrir que inúmeros factores podem ter influência neste processo. O que comemos, o que pensamos, como dormimos, o que conseguimos, como conseguimos, onde estamos, com quem estamos. Tentei potenciar a melhoria das condições em todos os aspectos referidos para maximizar a minha motivação, mas é sobre o último que me debruçarei neste texto.

Na minha investigação descobri, que a forma como os outros nos vêem (principalmente, aqueles que nos conhecem melhor e que estão mais próximos de nós) tem uma grande influência sobre a forma como nos vemos a nós próprios. Neste sentido percebi que havia pessoas à minha volta que acreditavam nas minhas capacidades. Tentei manter-me próximo dessas pessoas. Mostrar-lhes, quando possível, o quão importante é para mim a confiança que me transmitem, os conselhos que me dão, a ajuda nunca negada, o esforço que também eles fizeram, e a paciência para comigo que por vezes faltava a mim próprio.

No meu interesse pela psicologia percebi que nos podemos mudar a nós próprios. Esta mudança começa de dentro para fora, somos nós que a iniciamos, e, se tivermos sorte, é acompanhada de fora para dentro. Tentei fazer a minha parte, e não há agradecimento suficiente para quem fez a segunda.

Espero que este texto os faça ver o que sinto.

Ao meu orientador, CMG Costa Canas pela confiança que me transmitiu, por não ter receio de me propor um projeto ambicioso, pela ajuda constante, paciência e dedicação inigualáveis;

Ao Tenente Moreira da Costa que durante o exercer das suas funções como comandante de companhia do 4º ano, me fez querer mais que ninguém, melhorar enquanto militar;

A todos os envolvidos na viagem de cruzeiro de Verão 2019 a bordo do NRP Zarco, que me mostraram o valor do respeito mútuo e bom ambiente quando a bordo de um navio;

À câmara de oficiais do NRP Bérrio, por ter sido a melhor experiência possível que poderia ter a bordo do meu primeiro navio, numa altura em que bem precisava de prespetivas positivistas sobre o meu futuro motivando-me para continuar a perseguir os meus objetivos;

Ao CFR Lourenço Gorricha pela disponibilidade demonstrada para auxiliar o realizar desta dissertação;

Ao CMG Franco Facada pela disponibilidade que prontamente ofereceu e pelo gosto pela navegação astronómica que me conseguiu transmitir nas suas aulas, levando-me a procurar um tema nesta área;

Ao meu amigo Gonçalo Horta, pela sua sincera amizade e por partilhar do meu gosto por esta área, e pela motivação daí gerada;

Aos meus amigos Rodrigo Delaunay, Afonso Luís, Ricardo Nunes e Rúben Parreira por anos de boa convivência e apoio incondicional;

Aos meus amigos Leonardo Duarte e Ana Beatriz Marques por me fazerem acreditar em mim próprio e na possibilidade de melhoria, em prol da felicidade, que agora sei que toda a gente tem;

À minha irmã Inês, por ser uma pequena luz sempre acesa na minha vida;

Aos meus pais, que me deram tudo,

Obrigado.

## Resumo

A presente dissertação aborda os conceitos teóricos e algoritmos necessários para o desenvolvimento de uma rotina de obtenção de coordenadas celestes. Estes são conceitos relacionados com o cálculo astronómico, particularmente da posição dos astros. Os dados obtidos são vários e de vários tipos tais como: Ascensão Reta, ângulo horário em Greenwich, declinação, ângulo sideral, hora da passagem meridiana, horas de nascimento e ocacos de astros, horas de crepúsculos, fracção iluminada do disco da Lua, alturas e azimutes para diferentes posições geográficas, equação de tempo, paralaxe horizontal, semi-diâmetro, entre outros.

Estes conceitos e algoritmos foram postos em prática, criando a dita ferramenta informática que foi, posteriormente, testada em comparação com ferramentas de referência já existentes. Os resultados obtidos foram discutidos comprovando a eficácia dos ditos algoritmos para aplicação no contexto da navegação astronómica.

Palavras-chave: Navegação Astronómica, Coordenadas Celestes, Almanaque Náutico, AstroNavTH2020



## **Abstract**

The present dissertation addresses the theoretical concepts and algorithms necessary for the development of a routine for obtaining celestial coordinates. These are concepts related to astronomical calculation specifically of the positions of celestial bodies. The obtained data are numerous and of several kinds such as: Right Ascension, Greenwich Hour Angle, declination, sidereal angle, meridian passage time, rise and sunset times, twilight hours, illuminated fraction of the Moon's disk, heights and azimuths for different geographical positions, time equation, horizontal parallax, semi-diameter, among others.

The concepts and algorithms were put into practice, creating a computer tool that was subsequently tested against existing reference tools. The results obtained were discussed proving the effectiveness of the mentioned algorithms for application in the context of astronomical navigation.

Keywords: Astronomical Navigation, Celestial Coordinates, Nautical Almanac, AstroNavTH2020



# Índice

Epígrafe .....	I
Dedicatória.....	III
Agradecimentos.....	V
Resumo.....	VII
Abstract .....	IX
Índice .....	XI
Lista de Abreviaturas e Acrónimos .....	XVII
Índice de Figuras.....	XIX
Índice de Tabelas .....	XXI
Introdução.....	1
Enquadramento Geral.....	1
Âmbito, Objetivos e Metodologia da Dissertação .....	1
Pertinência do Tema.....	2
Estrutura da Dissertação.....	3
Capítulo 1. Enquadramento Teórico.....	5
1.1 Introdução ao Cálculo Astronómico.....	5
1.1.1. A Importância da Astronomia na Navegação Astronómica.....	5
1.1.2. A Esfera Celeste.....	6
1.1.3. Coordenadas Celestes.....	9
1.1.4. A Contagem do Tempo .....	12
1.1.5. Precessão .....	16
1.1.6. As épocas de referência.....	16
1.1.7. Nutação e Obliquidade .....	17
1.1.8. Correções necessárias .....	19
1.2 Necessidade de Precisão.....	23
1.3 A Teoria VSOP.....	23
Capítulo 2. Algoritmos .....	25
Algoritmo 1 - Cálculo do $\Delta T$ .....	25
Algoritmo 2 - Conversão para Dia Juliano.....	27

Algoritmo 3 - Interpolação de 2º Grau.....	29
Algoritmo 4 - Efeitos da Nutação .....	30
Algoritmo 5 - Cálculo do Tempo Sideral em Greenwich (TSG).....	31
Algoritmos para Obtenção de Coordenadas do Sol.....	32
Algoritmo 6 – Algoritmo da Nota Técnica.....	33
Algoritmo 7 - Algoritmo de Meeus.....	37
Algoritmo 8 - Cálculo da Posição Aparente de Estrelas .....	38
Algoritmo 9 - Cálculo da Posição Aparente da Lua .....	45
Algoritmo 10 - Cálculo da Fração Iluminada da Lua.....	49
Algoritmo 11 - Cálculo da Posição Aparente de Planetas .....	49
Algoritmo 12 - Conversão de Coordenadas Eclípticas para Equatoriais.....	53
Algoritmo 13 - Cálculo de Nascimento, Ocaso e PM.....	54
Algoritmo 14 - Cálculo das Horas dos Crepúsculos.....	57
Algoritmo 15 - Cálculo de Horas Favoráveis para Observações ao Crepúsculo.....	58
Algoritmo 16 - Cálculo de Alturas e Azimutes.....	58
Algoritmo 17 - Cálculo de Sugestão de Observações.....	59
Capítulo 3. Descrição da Ferramenta .....	61
3.1 AstroNavTH2020.....	61
Funcionalidade Almanaque.....	62
Funcionalidade Planeamento de Observações .....	64
Capítulo 4. Discussão de Resultados .....	67
4.1 Sol.....	68
4.2 Lua.....	69
4.3 Vénus .....	70
4.4 Marte .....	71
4.5 Jupiter.....	72
4.6 Saturno.....	73
4.7 Polar .....	74
4.8 S.Octantis.....	75
4.9 Outras estrelas.....	76
Atria.....	76

Miaplacidus .....	77
Kochab .....	78
Alnilam .....	79
Bellatrix .....	80
Procyon .....	81
4.10 Estudo Geral.....	82
Conclusões .....	85
Referências Bibliográficas .....	87
Apêndice A – Código do Programa.....	1
Menu Inicial .....	1
Funcionalidade Almanaque .....	2
Funcionalidade Planejamento de Observações .....	14
Terra .....	24
Sol – Algoritmo 6.....	28
Sol – Algoritmo 7.....	31
Venus .....	35
Marte .....	37
Jupiter .....	39
Saturno.....	40
$\Delta T$ .....	42
Estrelas .....	43
Dia Juliano .....	45
TSG .....	46
Efeitos da Nutação .....	47
Obliquidade Média.....	48
Nascimento , Ocaso e PM.....	48
Crepúsculos.....	52
Efeito da Aberração.....	55
Eclípticas para Equatoriais .....	55
Horizontais.....	56
Interpolação .....	57

Magnitudes .....	57
Heliocêntricas de Venus.....	59
Heliocêntricas de Marte .....	64
Heliocêntricas de Jupiter.....	69
Heliocêntricas de Saturno .....	74
Horas Favoráveis .....	79
Funções Auxiliares .....	80
angle2str .....	80
ascensaoreta0.....	85
declinacao0 .....	86
deg2hours.....	87
Escolha de Estrelas.....	87
Matrizes Lua .....	90
Matrizes Venus.....	96
Matrizes Marte .....	99
Matrizes Jupiter.....	105
Matrizes Saturno .....	116
Matrizes Terra .....	129
Conversão de graus decimais .....	133
Vetor Beta0.....	133
Vetor_estrelas.....	136
Vetor_LAMBDA0.....	137
Vetor_MIU .....	140
Vetor_MIU_linha .....	142
Tabela .....	144
Ler dados introduzidos .....	145
Gráfico - Planeamento .....	147
Definição de variáveis para o gráfico.....	149
Sol gráfico .....	149
Lua Gráfico.....	149
Venus Gráfico .....	150

Marte Gráfico .....	150
Jupiter Gráfico .....	151
Saturno Gráfico.....	151
Saturno Anéis .....	152
Estrela gráfico .....	153
Rotinas de Cálculo dos desvios relativos ao NavPac.....	153
Estudo_desvios.....	153
docs_datas.....	156
desvios .....	158
desvios_sem_polares.....	159
astros_navpac .....	159
Pagina_desvios .....	161
navpac.....	161
gerar data.....	162
Documentos de texto com dados do NavPac.....	163



## Lista de Abreviaturas e Acrónimos

**AR** – Ascensão Reta

**AstroNavTH2020** – software desenvolvido no âmbito desta dissertação de mestrado

**Dec.** -Declinação

**GDH** – Grupo Data-Hora

**GHA** – Ângulo Horário em Greenwich

**GMT** – *Greenwich Mean Time*

**GPS** – *Global Positioning System*

**GUI** – *Graphical User Interface*

**HP** – Paralaxe Horizontal

**IERS** –*International Earth Rotation and Reference Systems Service*

**J2000.0** – época de referência

**JD** – Dia Juliano

**JDE** – *Julian Day Ephemeris*

**LHA** - Ângulo Horário no Lugar

**Nav32** – software de referência desenvolvido por Omar Reis, comumente usado como ferramenta de auxílio para a navegação astronómica.

**NavPac** - software de referência desenvolvido por UKHO, comumente usado como ferramenta de auxílio para a navegação astronómica.

**PM** – Passagem Meridiana

**PNC** – Polo Norte Celeste

**PSC** – Polo Sul Celeste

**Q** – Equador Celeste

**SD** – Semi-Diâmetro

**TAI** - *Temps Atomic International*

**TSG** – Tempo Sideral em Greenwich

**TT** – *Terrestrial Time*

**UKHO** - *United Kingdom Hydrographic Office*

**USNO** – *United States Naval Observatory*

**UT** – *Universal Time*

**VSOP** - *Variations Séculaires des Orbites Planétaires*

## Índice de Figuras

Figura 1- Representação esquemática do segmento de circunferência constituída pelos pontos dos quais se observa determinado astro com uma altura igual. ....	6
Figura 2 - O modelo da esfera celeste.....	7
Figura 3– Representação da eclíptica, meridiano e paralelo celestes .....	8
Figura 4 - Coordenadas Horizontais.....	9
Figura 5 - Coordenadas Equatoriais .....	10
Figura 6– Coordenadas Eclípticas Geocêntricas .....	11
Figura 7- Coordenadas Horárias.....	11
Figura 8 - Esquemática do conceito de Sol médio .....	13
Figura 9– Gráfico da diferença entre TT e UT ao longo dos últimos quatro séculos .....	14
Figura 10 - Representação da obliquidade do eixo de rotação da Terra .....	18
Figura 11– Esquemática do Semi-Diâmetro.....	20
Figura 12– Anomalia média e verdadeira.....	21
Figura 13– Aberração .....	22
Figura 14 - Diferença de Aberração Sol Vs Estrelas.....	23
Figura 15 – Dados e previsões do USNO e IERS para o $\Delta T$ de 2000 a 2050.....	27
Figura 16– Fluxograma do algoritmo de conversão para Dia Juliano.....	28
Figura 17 – Fluxograma do algoritmo de cálculo dos efeitos da nutação .....	31
Figura 18 – Fluxograma do algoritmo de cálculo do Tempo Sideral em Greenwich.....	32
Figura 19 – Fluxograma do algoritmo 6 de cálculo de dados do Sol.....	37
Figura 20 – Fluxograma do algoritmo de cálculo das posições aparentes das estrelas .....	44
Figura 21 – Fluxograma do algoritmo de cálculo da posição da Lua .....	48
Figura 22 – Fluxograma do algoritmo de cálculo dos planetas .....	52
Figura 23 - Fluxograma do algoritmo de conversão de coordenadas eclípticas para equatoriais .....	54
Figura 24 – Fluxograma do algoritmo de cálculo de Nascimento, Ocaso e Passagem Meridiana.....	57
Figura 25 – Menu inicial do AstroNavTH2020.....	62
Figura 26 – Interface da funcionalidade Almanaque .....	62
Figura 27 – Apresentação dos painéis, mostradores e botões da funcionalidade Almanaque.....	63

<b>Figura 28 – Funcionalidade de Planeamento de observações- Botão “Céu Visível” .....</b>	<b>64</b>
<b>Figura 29 – Funcionalidade Planeamento de Observações – Botão “Azimutes do Sol” .....</b>	<b>65</b>
<b>Figura 30 – Funcionalidade Planeamento de Observações- Botão “Sugestão deObservações” .....</b>	<b>66</b>

## Índice de Tabelas

Tabela 1 - Resumo dos sistemas de coordenadas relevantes .....	12
Tabela 2 – $\Delta T$ em segundos para os diferentes anos .....	26
Tabela 3 – Tabela de $\lambda_0$ e $\beta_0$ para diferentes estrelas referentes à época J2000 .....	40
Tabela 4 – Extrato das tabelas do cálculo da posição da Lua .....	46
Tabela 5 – Extrato de uma das tabelas da teoria VSOP87 .....	50
Tabela 6 - Lista dos dos astros disponíveis no AstroNavTH2020.....	63
Tabela 8 - Resultados da comparação para o Sol com o Algoritmo 7 .....	68
Tabela 8 – Resultados da comparação para o Sol com o Algoritmo 6 .....	68
Tabela 9 – Resultados da comparação para a Lua.....	69
Tabela 10 - – Resultados da comparação para Vénus .....	70
Tabela 11 – Resultados da comparação para Marte .....	71
Tabela 12 – Resultados da comparação para Jupiter .....	72
Tabela 13 - Resultados da comparação para Saturno.....	73
Tabela 14 - Resultados da comparação para a Polar.....	74
Tabela 15 - Resultados da comparação para S.Octantis .....	75
Tabela 16 Resultados da comparação para a Atria .....	76
Tabela 17 - Resultados da comparação para a Miaplacidus .....	77
Tabela 18 - Resultados da comparação para a Kochab .....	78
Tabela 19 - Resultados da comparação para a Alnilam .....	79
Tabela 20- Resultados da comparação para aBellatrix .....	80
Tabela 21- Resultados da comparação para aProcyon.....	81
Tabela 22 – Astros com maiores desvios em cada data .....	82
Tabela 23 – Astros (excepto Polar e S.Octantis) com maiores desvios em cada data.....	83
Tabela 24 – Astros (excepto Polar e S.Octantis) com maiores desvios em cada data usando o algoritmo 7 para o Sol .....	84



## **Introdução**

### **Enquadramento Geral**

A navegação astronómica é a prática de determinação da posição que permite a um indivíduo conhecer a sua posição baseando-se em observações astronómicas (Umland, 2001), capítulo 1. É uma arte antiga que foi desenvolvida ao longo do tempo. Já na época dos descobrimentos os astros eram frequentemente usados como referência para o posicionamento e seleção de rumos.

Há relatos de navegadores portugueses do século XV que evidenciam o conhecimento de conceitos astronómicos como altura do Polo Elevado bem como relatos da sua medição para efeito de determinação da latitude. (Reis, 1997), pp. 47 Muitos dos instrumentos utilizados ao longo da história na prática da navegação astronómica são, de facto, invenção portuguesa. Muitos conhecimentos desta arte chegam hoje até nós oriundos desse período, não só da prática propriamente dita, mas também, conhecimentos teóricos de astronomia. A navegação astronómica está intrinsicamente ligada ao ramo da astronomia diretamente relacionado com o cálculo da posição dos astros e os seus movimentos.

Nos tempos modernos, com o evoluir da tecnologia surgiram várias ferramentas de computação de dados astronómicos que visam automatizar dos cálculos. No contexto da conciliação do poder destas novas tecnologias e visando a utilidade prática para a navegação astronómica, surgiram ferramentas como o nav32 de Omar Reis ou o NavPac (fornecido pelo United Kingdom Hydrographic Office), que tornaram o processo de cálculo de coordenadas celestes e navegação astronómica mais eficiente.

### **Âmbito, Objetivos e Metodologia da Dissertação**

Em termos académicos esta área é tratada na unidade curricular de Navegação III pertencente ao Mestrado Integrado em Ciências Militares Navais da classe de Marinha (4º semestre curricular).

O desenvolvimento da ferramenta informática daqui em diante designada por: “AstroNavTH2020” baseou-se nos conhecimentos da área nuclear de astronomia e integrou conceitos da Análise Matemática, Análise Numérica, Álgebra Linear e Programação em Matlab.

Os objetivos estabelecidos para esta dissertação foram os seguintes:

- Aprofundar o conhecimento na área da navegação astronómica.
- Desenvolver uma ferramenta informática com utilidade prática em duas vertentes:
  - Navegação astronómica operacional através do cálculo de coordenadas de astros.
  - Propósitos didáticos.

Os objetivos propostos enquadram-se num programa mais abrangente que visa o processo da navegação astronómica como um todo. Além desta dissertação, o programa completa-se com o trabalho desenvolvido no âmbito de outra dissertação do mesmo ciclo de estudos que visa particularmente a obtenção do ponto astronómico.

A metodologia usada consistiu numa abordagem sequencial, das seguintes fases:

- Aprofundamento teórico sobre os conceitos necessários
- Reflexão sobre a possibilidade de melhoria das ferramentas já existentes
- Estudo e escolha dos algoritmos adequados aos objetivos delineados
- Programação em Matlab dos algoritmos escolhidos
- Comparação com programas atuais de referência e discussão de resultados

Como é característico do método científico, a aplicação de uma metodologia implica uma avaliação das conclusões e uma reanálise dos pressupostos em cada uma das etapas, implicando avanços e retrocessos nos passos acima descritos.

### **Pertinência do Tema**

Com o evoluir da tecnologia moderna, a navegação astronómica cai cada vez mais em desuso. Hoje em dia é utilizada quase exclusivamente como forma de redundância, uma alternativa de última instância para sistemas de Geoposicionamento. De facto, sistemas como o GPS permitem inúmeras vantagens em relação a este método de navegação. A facilidade de utilização, celeridade, rigor, são exemplos de algumas delas. O que torna então, este método, mercedor de atenção no presente?

Como bem se sabe, os aparelhos eletrónicos podem falhar. É de extrema importância, principalmente numa marinha de guerra que se garanta a redundância até ao limite da

capacidade material e humana. Como tal, garantir que se consegue navegar, mesmo no pior cenário é de facto relevante para a formação de marinheiros capazes.

Surgem, conseqüentemente, várias razões justificativas da necessidade de um conhecimento aprofundado desta área do saber:

- Garantir, como referido, a redundância.
- Diminuir ao máximo a dependência de sistemas exteriores ao nosso conhecimento próprio. Saber (hoje mais que nunca) é poder.
- Possibilidade de automatizar os procedimentos da navegação astronómica, tornando-a, com os devidos meios necessários, uma alternativa mais viável em termos de celeridade e rigor.
- Possibilidade de exploração/inação de novas tecnologias conciliadas com este método.

É indo ao encontro dos pontos acima mencionados que se enquadra esta dissertação que tem como um dos seus objetivos apresentar uma ferramenta operacional de cálculos náuticos/astronómicos, para além de fomentar o conhecimento aprofundado desta área, para todos os navegantes.

No presente é prática comum, praticar-se/ensinar-se a navegação astronómica com recurso a um almanaque náutico ou a um *software* equivalente. Nesta dissertação são explorados métodos que permitem contornar a própria utilização do almanaque, calculando as coordenadas necessárias recorrendo a algoritmos descritos passo por passo.

### **Estrutura da Dissertação**

Regra geral, a estrutura da dissertação prevista inicialmente no Plano de Dissertação foi cumprida. Contudo, houve necessidade de uma reorganização dos capítulos para uma melhor adequação ao trabalho efetivamente realizado.

Assim, no Capítulo 1. - Enquadramento Teórico, são apresentados os conceitos teóricos necessários, com uma introdução ao cálculo astronómico salientando a sua importância na navegação astronómica e explicando os seus princípios básicos.

No segundo capítulo – Algoritmos, são descritos os principais algoritmos do cálculo dos parâmetros astronómicos.

No Capítulo 3. – Descrição da Ferramenta, é feita, tal como o nome do capítulo indica, uma sucinta apresentação da interface da ferramenta informática desenvolvida, AstroNavTH2020, evidenciando as suas potencialidades. A ferramenta representa, no fundo, a parte fundamental que se propôs realizar. Para uma completa compreensão da dimensão do trabalho realizado, este capítulo deverá ser lido na presença do *software* AstroNavTH2020 desenvolvido.

No capítulo seguinte – Discussão de Resultados, são discutidas as comparações realizadas entre o software AstroNavTH2020 e os programas habitualmente utilizados em navegação astronómica NavPac e nav32. Os resultados apresentados são bastante favoráveis ao AstroNavTH2020, sendo demonstrada a validade do trabalho realizado.

Por último é realizada uma conclusão onde se abordam as limitações encontradas ao longo da realização do projecto, perspectivas de trabalho futuro e, por fim, algumas considerações finais.

## **Capítulo 1. Enquadramento Teórico**

### **1.1 Introdução ao Cálculo Astronómico**

Várias civilizações ao longo da história valorizaram o conhecimento das posições e movimentos dos astros utilizando-o para os mais diversos propósitos científicos, práticos, religiosos (Evans & Jones, 2000). As observações registadas de civilizações como os babilónios providenciaram informação muito valiosa para vários grandes astrónomos ao longo da história. (Brown, 2002). Os modelos concebidos foram gradualmente melhorados até à atualidade. Hoje em dia os modelos utilizados permitem o cálculo das posições dos astros nas suas órbitas com rigor do segundo do grau ao segundo (de tempo). (Meeus, 1992).

#### **1.1.1. A Importância da Astronomia na Navegação Astronómica**

Para que algum objeto possa ser usado como referência para a navegação é necessário conhecer a sua posição. Assim acontece na navegação costeira com marcas conspícuas tal como na navegação astronómica com os corpos celestes. (Hofmann-Wellenhof, Legat, & Wieser, 2003), pp. 33. Para conseguir praticar a navegação astronómica é então necessário conhecer a posição em que se encontra determinado astro no momento em que se faz a sua observação.

Sabendo a posição do astro, é possível ao observador, medindo a altura à qual o vê, determinar uma circunferência das suas posições possíveis sobre a superfície da Terra, pois somente sobre essa circunferência o astro é visível com uma altura correspondente à altura medida (ver figura 1).

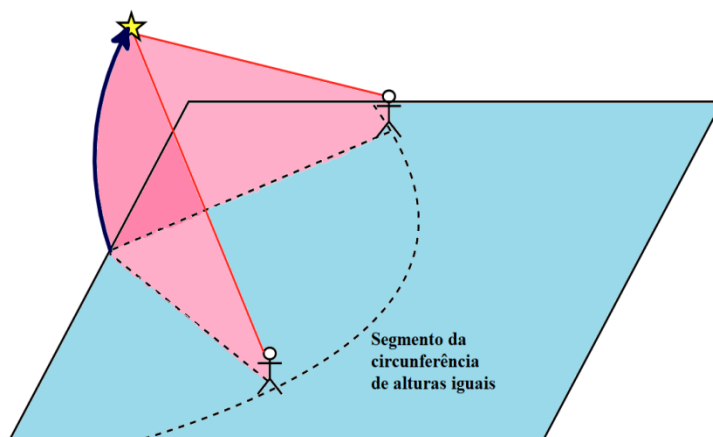


Figura 1- Representação esquemática do segmento de circunferência constituída pelos pontos dos quais se observa determinado astro com uma altura igual.

Usando várias medições de alturas de diferentes astros, é teoricamente possível cruzar estas circunferências, à semelhança do processo de cruzamento de linhas de posição em navegação costeira, e obter assim a nossa posição (Umland, 2001), capítulo 4..

Porém, como veremos mais à frente, o método utilizado é na verdade mais complexo devido às habitualmente demasiado grandes dimensões destas circunferências. Ainda assim, para o referido método, o posicionamento dos astros é fundamental.

### 1.1.2. A Esfera Celeste

A representação dos corpos celestes e respetivas posições no contexto da navegação astronómica é feita utilizando um referencial geocentrico assumindo uma Terra imóvel. Envolvendo a Terra representa-se uma esfera maior denominada esfera celeste (ver Figura 2) na qual podem ser representados os diferentes astros. (Fraknoi, Morrison, & Wolff, 2016), pp. 33.

Logicamente, este modelo não corresponde à realidade (o universo não é geocêntrico) mas simplifica os cálculos no âmbito da navegação astronómica, baseando-se no conceito de movimento aparente e posição aparente dos astros (movimento relativo), considerando a terra estacionária e os astros móveis.

A posição aparente de um astro é a posição em que um observador vê o astro. Esta posição pode diferir da posição verdadeira em que o astro se encontra por diversos efeitos causados por fenómenos como a paralaxe ou a aberração. (Meeus, 1992), pp. 149-156.

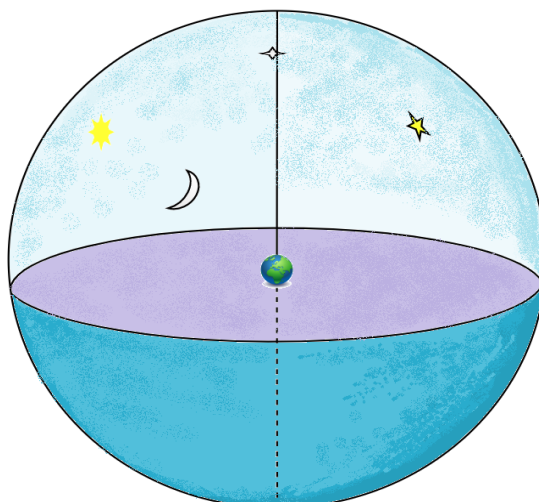


Figura 2 - O modelo da esfera celeste

Segundo (Meeus, 1992), pp. 149 no caso das estrelas, define-se a sua posição aparente como a posição da estrela na esfera celeste observada por um observador situado no centro da Terra em movimento. (Ver aberração).

O movimento aparente de um astro é a trajectória que um observador à superfície da terra vê o astro descrever assumindo que a Terra permanece estacionária.

Na face interior da superfície da esfera celeste podem ser representados os astros assim como os respetivos movimentos aparentes, ao longo de determinados períodos de tempo. Por exemplo, o movimento de translação da terra pode ser traduzido neste modelo através da eclíptica, representado assim por uma trajectória descrita pelo Sol ao longo do ano na esfera celeste.

Nesta esfera podem ser representados a nível celeste conceitos equivalentes da geografia terrestre tais como: Equador celeste (Q), Polos celestes (PNC e PSC), Meridianos e paralelos. (Umland, 2001), capítulo 3.

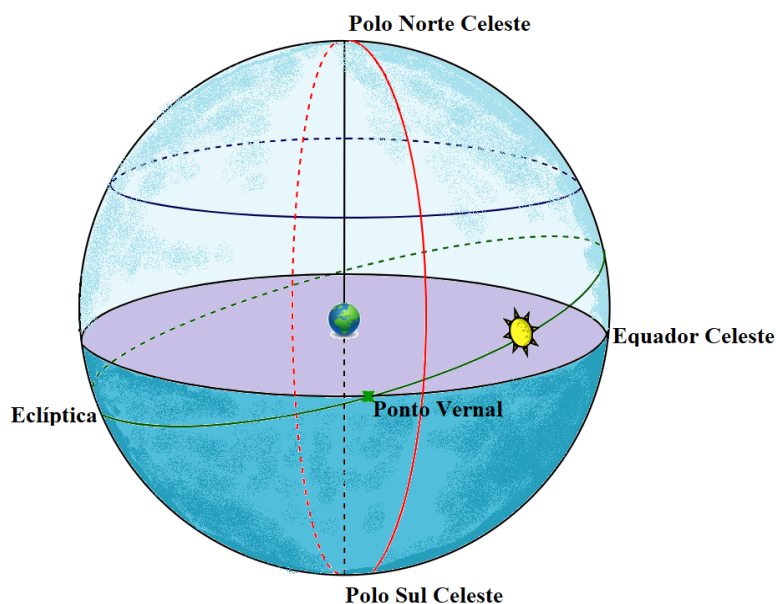


Figura 3– Representação da eclíptica, meridiano e paralelo celestes

Da intersecção da eclíptica com o equador celeste surgem dois pontos correspondentes aos equinócios de Primavera e Outono. O ponto do equinócio da Primavera é também denominado de ponto Vernal (ver figura 3). Este não é influenciado pelos movimentos de rotação e translação da Terra sendo assim usado como referência para posicionamento de estrelas e planetas.(Umland, 2001), capítulo 3.

Os astros podem ser classificados em fixos (estrelas) e errantes (Sol, Lua e planetas). Esta classificação baseia-se no tipo de movimento observado.

Devido ao facto de as estrelas se encontrarem extremamente longe da Terra relativamente aos astros errantes, o observador tem a sensação de que estas se deslocam segundo o mesmo padrão de movimento de cerca de 15° por hora.

Este movimento corresponde à rotação da Terra, pouco ou nada alterando as suas posições relativas entre si.

Já nos errantes, é possível verificar em curtos espaços de tempo as consequências do movimento de translação da Terra e do movimento próprio dos mesmos. Cada um destes astros assume assim uma trajectória peculiar. (Fraknoi et al., 2016), pp. 39.

### 1.1.3. Coordenadas Celestes

Para proceder ao posicionamento dos astros podem ser usados diferentes referenciais conforme o nosso objetivo.

Quando observamos astros, registamos as suas coordenadas horizontais (relativas ao plano do Horizonte).

De um Almanaque Náutico retiramos coordenadas Equatoriais e horárias mais usuais no contexto da prática da navegação astronómica para posicionamento dos astros. Existem ainda as coordenadas Eclípticas que serão usadas para efeito de cálculos astronómicos.

#### Coordenadas Horizontais

O sistema de Coordenadas Horizontais é constituído pelas coordenadas altura e azimute (ver Figura 4).

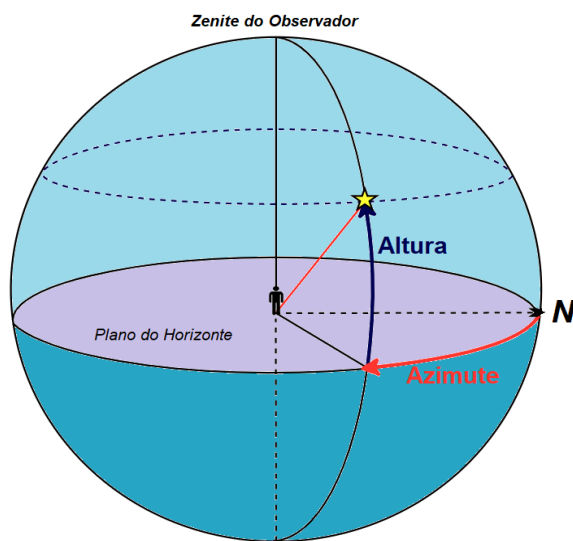


Figura 4 - Coordenadas Horizontais

Tem como plano de referência o horizonte do observador e é usado para referenciar um astro na prática.. O observador pode medir diretamente os valores das coordenadas usando os instrumentos adequados (Umland, 2001), capítulo 6.

#### Coordenadas Equatoriais

**Ascensão Reta ( $\alpha$ )** – Distância angular de um determinado ponto medida no sentido Oeste-Este ao longo do equador celeste que vai do ponto vernal até ao meridiano do astro.

Constitui, juntamente com a declinação, conforme figura 5, o sistema (celeste) de coordenadas equatoriais. (Vincent, 1976), pp. 17.

**Declinação ( $\delta$ )**- Juntamente com a ascensão reta, é uma das duas coordenadas do sistema de coordenadas equatoriais. Consiste na distância angular medida, ao longo do meridiano do astro, do astro para Norte ou para Sul, entre o Equador Celeste, e o astro em questão.

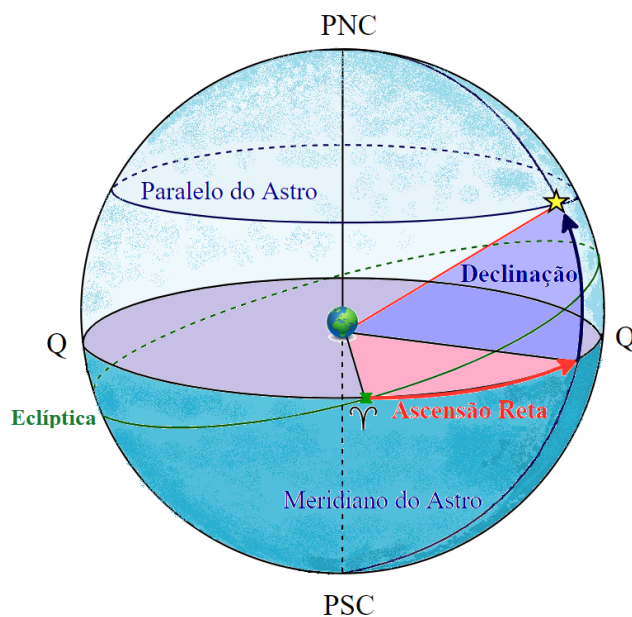


Figura 5 - Coordenadas Equatoriais

### Coordenadas Eclípticas

As coordenadas Eclípticas são semelhantes às coordenadas equatoriais usando porém o plano da eclíptica como referencial e mantendo o ponto vernal como o ponto 0. As coordenadas constituintes do sistema são a longitude e latitude eclípticas. A **longitude eclíptica ( $\lambda$ )** é medida ao longo da eclíptica partindo do ponto vernal enquanto a latitude eclíptica é medida perpendicularmente ao plano da eclíptica. (Vincent, 1976), pp. 37 e (Meeus, 1992), pp. 217.

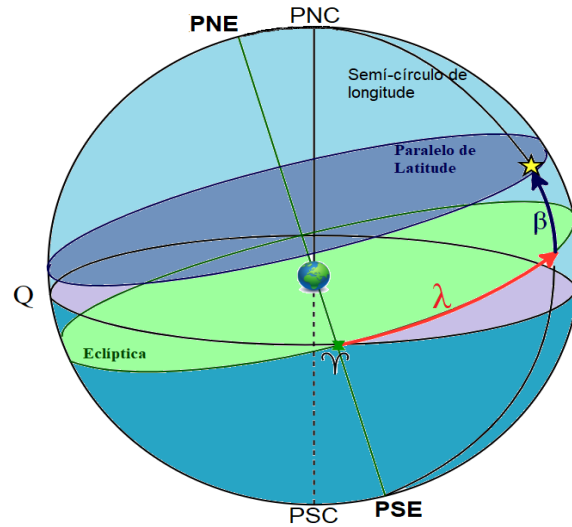


Figura 6– Coordenadas Eclípticas Geocêntricas

Estas coordenadas podem ainda dividir-se em Eclípticas Geocêntricas (têm a Terra como origem do referencial) conforme representado na Figura 6. ou Eclípticas Heliocêntricas (têm o Sol como origem do referencial).

### Coordenadas Horárias

Também chamadas em astronomia como o segundo sistema de coordenadas equatorial por ter como plano de referência o equador (Vincent, 1976), pp. 17 este sistema tem como coordenadas constituintes o ângulo horário e a distância polar, esta última derivada da declinação.

**Ângulo Horário Greenwich (GHA)** de um determinado astro corresponde ao ângulo medido ao longo de qualquer plano paralelo ao equador, entre o meridiano de Greenwich e o meridiano do astro.

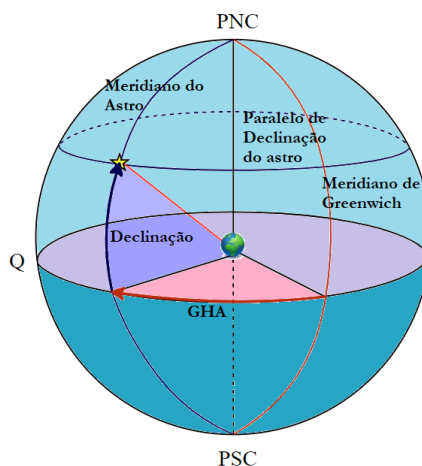


Figura 7- Coordenadas Horárias

**Ângulo Horário no Lugar (LHA)** de um astro corresponde ao ângulo medido ao longo de qualquer plano paralelo ao Equador, entre um determinado astro e o meridiano do observador.

Sintetizam-se no quadro seguinte os diferentes sistemas de coordenadas celestes abordados.

Tabela 1 - Resumo dos sistemas de coordenadas relevantes

Sistema de Coordenadas	Origem do referencial	Plano de referência	Coordenadas	Direcção 0°
Horizontais	Observador	Horizonte do observador	Altura e Azimute	Ponto Cardeal Norte/Sul do Horizonte
Equatoriais	Centro da Terra (geocêntricas)	Equador Celeste	Ascensão Reta e Declinação	Ponto Vernal
	Centro do Sol (Heliocêntricas)			
Eclípticas	Centro da Terra (geocêntricas)	Eclíptica	Longitude e Latitude Eclípticas	Ponto Vernal
	Centro do Sol (Heliocêntricas)			
Horárias	Centro da Terra	Equador Celeste	Ângulo Horário e Distância Polar	Ponto Vernal

#### 1.1.4. A Contagem do Tempo

Desde cedo a humanidade reconheceu a possibilidade de contagem do tempo pela periodicidade do movimento dos Astros. Há indícios de que já no paleolítico existiriam medições temporais baseadas no movimento da Lua. (Rudgley, 1999)

O dia é normalmente associado ao movimento de uma rotação de 360° da Terra em torno do seu próprio eixo. A questão que se coloca é então: como saber quando é que a rotação de 360° se concluiu (é necessário um ponto de referência).

Um dia é então definido como o período (P) compreendido entre duas passagens consecutivas de um determinado astro pelo meridiano do observador.

Em astronomia considera-se o tempo sidereal medido a partir do movimento das estrelas (do ponto Vernal) e o tempo solar, medido a partir do movimento do sol.

Porém ambos estes pontos de referência se revelaram não completamente eficazes por várias razões:

- No tempo sideral, há uma retrogradação do ponto Vernal de cerca de 0,125" por ano, devido ao movimento de precessão. Isto leva a que o dia definido pela passagem do vernal duas vezes consecutivas num mesmo meridiano, seja na verdade ligeiramente inferior ao período de rotação de 360° da Terra.
- A utilização do Sol como ponto de referência mostrava-se também ela falível pois a terra desloca-se com diferentes velocidades ao longo da eclíptica para além das consequências da própria inclinação do eixo da Terra que levam a que a projecção do Sol sobre o equador se movimente com velocidades inconstantes.

### A Equação de Tempo

Por forma a obter um parâmetro fidedigno de referência temporal, criou-se então um astro imaginário denominado de Sol médio que se movimentaria sobre o equador circular com uma velocidade angular constante. E o dia seria definido como o intervalo compreendido entre duas passagens consecutivas deste Sol médio.

À discrepância temporal variável ao longo do ano entre o tempo medido pelo Sol médio e o tempo medido pelo Sol verdadeiro dá-se o nome de **equação de tempo**, e esta pode ser calculada conforme descrito no algoritmo 6.

O Sol médio origina a escala de tempo conhecida como UT, usada hoje em dia como o tempo corrente no quotidiano da vida civil. Definiu-se assim esta escala de tempo assumindo a hora como dia/24, o minuto como hora/60 e o segundo como minuto/60. O dia teria então 86400 segundos.

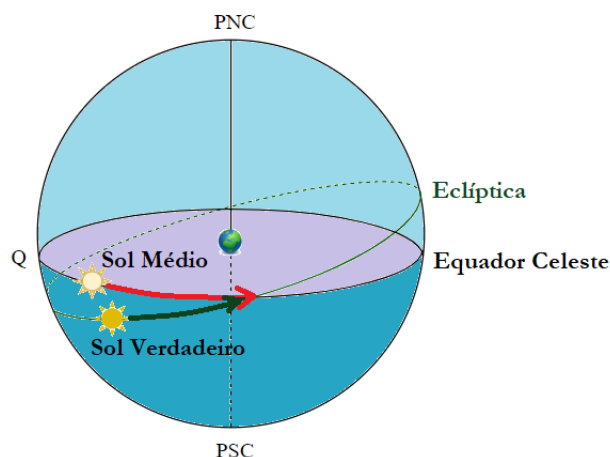


Figura 8 - Esquematização do conceito de Sol médio

## Escalas de Tempo

No entanto, em astronomia, percebeu-se que nem esta escala de tempo seria regular por outro problema ainda: O período de rotação da terra não é constante, acelerando e desacelerando irregularmente ao longo do tempo, com tendência a longo prazo para a desaceleração. (Stephenson & Morrison, 1995), pp. 167-169.

Hoje em dia, para efeitos de cálculo astronómico usam-se escalas de tempo dinâmicas por forma a eliminar a irregularidade da variação da rotação da terra obtendo assim uma unidade mais constante de medição de tempo. (Meeus, 1992; Stephenson & Morrison, 1995)

Resumindo: A escala UT mede o tempo baseando-se numa unidade de tempo dependente do movimento de rotação da Terra. Para efeito de previsão do movimento dos astros, esta escala movimento revelou-se irregular pelo que se criaram escalas de tempo mais constantes através de tecnologia atómica (ex. *Terrestrial Time*) ou pelo movimento observado de outros astros com movimentos mais regulares (ex. *Ephemeris Time*).

Sendo que a tendência a longo prazo do UT é atrasar-se em relação a formas mais regulares de medição de tempo, a correção para uma escala de tempo uniforme consiste na soma de um  $\Delta T$  aplicável ao período em causa.

Na figura 9 (Stephenson & Morrison, 1995), pp. 168 estão representadas as variações de

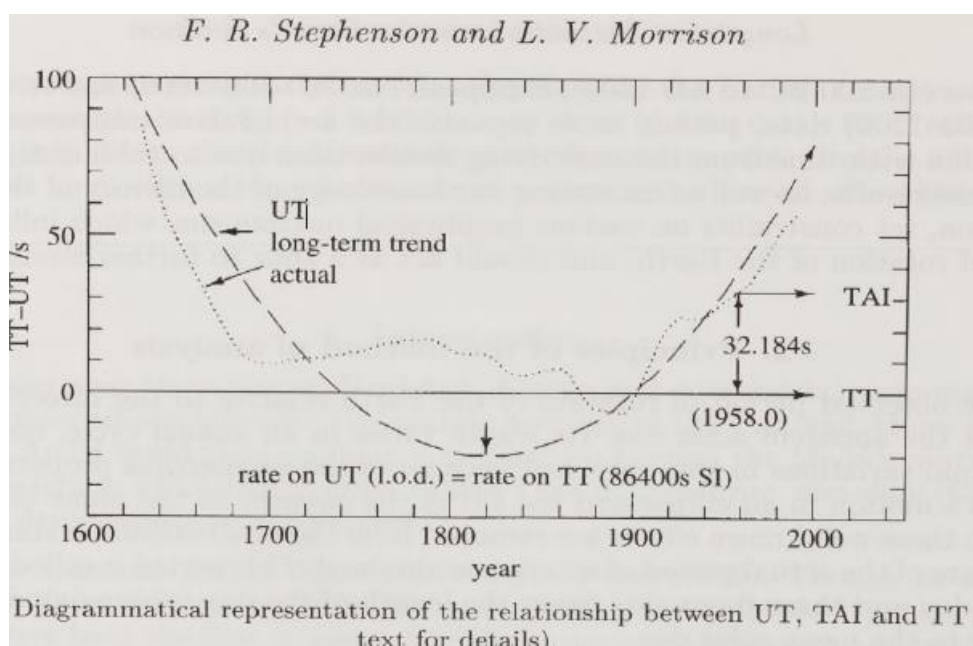


Figura 9– Gráfico da diferença entre TT e UT ao longo dos últimos quatro séculos

UT em relação à escala uniforme TT ao longo do tempo. A escala de tempo TAI (Temps Atomic International), é também uma escala de tempo uniforme, que se mantém sempre 32 segundos adiantada em relação ao TT devido à diferença existente em 1958 entre UT e a escala antecessora do TT.

### **Dia Juliano**

Sendo assim necessário conhecer o tempo decorrido entre as duas datas. Para isto usa-se o conceito de Dia Juliano.

Suponhamos que se pretende calcular o número de dias ocorridos entre duas datas (UT) Digamos por exemplo 30 Outubro de 1996 e 28 Março de 2020. O cálculo deste valor é complicado devido à variação da duração dos meses.

Para simplificação dos cálculos astronómicos usa-se uma contagem contínua de dias desde o início do período juliano usado em astronomia e *software* para calcular facilmente o número de dias ocorridos entre dois eventos. Um dia Juliano tem 24 horas e um século juliano tem 36525 dias.

Para calcular o período decorrido entre duas datas basta calcular os dias julianos correspondentes a cada uma e subtrair à mais recente o valor da mais antiga.

No exemplo referido calcularíamos então o dia juliano para cada uma das datas:

- 30/10/1996 corresponde ao dia juliano 2450386
- 28/03/2020 corresponde ao dia juliano 2458936

Subtraindo, obtém-se o valor de 8550 dias decorridos entre as duas datas.

Uma vez que um século tem 36525 dias, pode-se dividir o valor obtido por 36525 e obter assim o valor correspondente em séculos julianos.

Dado um intervalo de tempo  $t$  em séculos julianos, para o efeito de cálculos astronómicos precisos, pode converter-se este intervalo para um intervalo  $T$  correspondente numa escala de tempo uniforme através duma equação do tipo:

$$T = t + \Delta T(t)$$

Em que  $\Delta T(t)$  pode estar tabelado ou ser obtido através de equações em função de  $t$ .

### 1.1.5. Precessão

A precessão é a mudança de orientação contínua do eixo de rotação de um corpo celeste. Neste contexto abordar-se-á em particular o movimento de precessão da Terra que leva à gradual alteração da orientação do seu eixo num ciclo de aproximadamente 26000 anos.

Segundo (Meeus, 1992), pp. 131, devido a este movimento, o Polo Norte Celeste (presentemente situado perto da estrela Polar, roda lentamente em torno do polo da Eclíptica no período referido acima, e, conseqüentemente, o ponto vernal (sendo a intersecção do equador celeste com a eclíptica regride cerca de 50" por ano ao longo da eclíptica.

Para além disto, o próprio plano da eclíptica não se encontra imóvel no espaço. Devido à atração gravitacional exercida pelos restantes planetas sobre a Terra, este plano roda lentamente sobre uma "linha de nós" (do inglês: *line of nodes*), sendo que roda presentemente com uma velocidade de 47" por século.

Sendo a eclíptica e o equador os planos de referência dos sistemas de coordenadas eclípticas e equatoriais, devido à precessão, as coordenadas das estrelas estão em constante alteração.

Assim, catálogos de estrelas dão-nos as coordenadas tabeladas para as chamadas épocas de referência, tais como: 1900.0, 1950.0 ou 2000.0.

### 1.1.6. As épocas de referência

Para calcular alguns dados astronómicos usam-se as chamadas épocas que servem como referência. Uma época corresponde a uma data precisa no tempo para a qual determinados valores de certos astros estão tabelados.

Uma vez que as coordenadas das estrelas estão em constante alteração devido à precessão e ao movimento próprio das mesmas, as coordenadas são expressas relativamente a uma determinada época e a posição do ponto vernal nessa época (equinócio da época).

O referencial do equinócio usado é frequentemente o equinócio médio o que significa que se exclui o efeito da nutação no cálculo da sua posição, sendo assim uma referência menos exacta.

A época J2000.0 que se encontra presentemente em uso, corresponde ao momento 2451545.0 (TT) em dias Julianos ou seja 1 Janeiro de 2000 às 0h TT que corresponde ao momento UT 1 janeiro às 11:58:55.816.

### 1.1.7. Nutação e Obliquidade

A **nutação**, descoberta pelo astrónomo britânico James Bradley em 1728, é uma oscilação periódica do eixo de rotação da terra em torno da sua posição “média”. Devido à nutação, o polo instantâneo de rotação da Terra oscila em torno de um Polo médio que avança (com a precessão) em torno do polo da eclíptica.

A nutação é provocada por uma inclinação de  $5,1^\circ$  no plano da órbita da Lua em relação à eclíptica, que muda de orientação ao longo de 18 anos num movimento de precessão. Assim a força exercida pela Lua sobre a Terra é durante nove anos de maior e depois nove anos de menor intensidade do que a média.

Segundo (Meeus, 1992), pp.143, este movimento pode ser descrita por um somatório de termos periódicos. O maior destes tem um periodo de cerca de 18.6 anos, no entanto, existem outros termos periódicos significativos com menores períodos que podem ser tidos em conta caso se pretenda uma maior exactidão.

Nesta dissertação, no entanto, não se utilizaram estes termos uma vez que é possível aproximar os valores pretendidos, nutação na longitude e nutação na obliquidade, sem recorrer aos mesmos.

A nutação é convenientemente dividida em duas componentes: uma paralela e outra perpendicular à eclíptica.

A componente ao longo da eclíptica é denotada por  $\Delta\psi$  e chama-se Nutação em Longitude; esta afecta a longitude celeste de todos os astros. A componente perpendicular à eclíptica é denotada por  $\Delta\varepsilon$  e chama-se Nutação em Obliquidade, sendo que afecta a obliquidade do equador em relação à eclíptica.

A nutação não afecta a Latitude de corpos celestes.

Os valores de  $\Delta\psi$  e  $\Delta\varepsilon$  são usados para o cálculo das posições aparentes de corpos celestes, bem como para o cálculo do tempo sideral aparente.

Ambas estas componentes são dependentes da longitude eclíptica do nó ascendente da órbita da Lua ( $\Omega$ ).

A **obliquidade da eclíptica**, ou inclinação do eixo de rotação, é o ângulo entre o plano do equador celeste e a eclíptica.

Distingue-se entre obliquidade média e obliquidade verdadeira pelo facto da segunda, ao contrário da primeira, considerar o efeito da nutação.

A obliquidade média pode ser obtida através da fórmula:

$$\varepsilon_0 = 23^{\circ}26'21''.448 - 46''.8150T - 0''.00059 T^2 + 0''.001813 T^3$$

em que T é o intervalo de tempo medido em séculos julianos a partir da época J2000.0.

A obliquidade verdadeira pode ser obtida somando ao valor de  $\varepsilon_0$  a nutação em obliquidade:

$$\varepsilon = \varepsilon_0 + \Delta\varepsilon$$

A obliquidade do eixo da Terra varia entre cerca de 22.1 e 24.5 ao longo de um ciclo

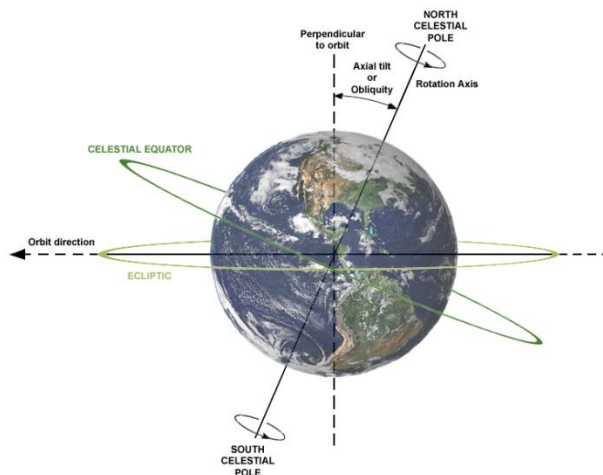


Figura 10 - Representação da obliquidade do eixo de rotação da Terra

de aproximadamente 41000 anos de duração.

Ver Figura 10 (<https://commons.wikimedia.org/wiki/File:AxialTiltObliquity.png>).

### 1.1.8. Correções necessárias

#### Paralaxe

A correção da paralaxe é, no contexto da navegação astronómica necessária para o Sol, Lua, Vénus e Marte. Segundo (Gameiro, 1964), pp. 70, a correção da paralaxe é na verdade aplicável ao calcular a altura de um astro e depende:

- da distância a que se encontra o mesmo
- da altura do mesmo, sendo máxima no horizonte e nula no zénite.

Chama-se paralaxe horizontal ( $\pi$ ) ao valor da paralaxe quando o astro se encontra no horizonte. Esta é dada por:

$$\pi = \frac{8''.794}{\Delta}$$

onde  $\Delta$  corresponde à distância do astro à Terra em unidades astronómicas.

### Semi-Diâmetro

Semi-Diâmetro – Ao efectuar uma medição de altura do Sol, da Lua, e por vezes de alguns planetas, é necessário para efeito de exactidão medir a altura do limbo superior ou inferior. Ou seja: para a medição de uma altura com um sextante é muito mais fácil alinhar uma linha definida (limbo superior ou inferior) com a linha do horizonte. Conhecendo o semi-diâmetro pode ser efetuada a correção para saber a verdadeira altura do objeto (a altura do seu centro). Assim, torna-se necessária uma correção adicional correspondente à distância angular entre estes limbos e o centro do astro.

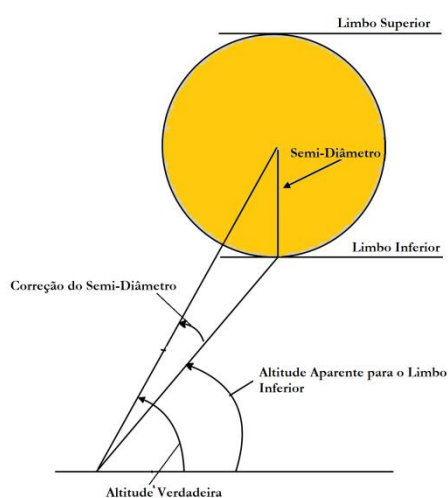


Figura 11– Esquematisação do Semi-Diâmetro

Segundo (Meeus, 1992), pp. 389-392 o semi-diâmetro para Sol e planetas é dado por :

$$s = \frac{s_0}{\Delta}$$

em que  $s_0$  é o valor de semi-diâmetro correspondente à distância de 1 unidade astronómica, e  $\Delta$  corresponde à distância do astro à Terra em unidades astronómicas. O valor de  $\Delta$  pode ser calculado conforme descrito no algoritmo 11 do próximo capítulo.

Para o Sol pode considerar-se

$$s_0 = 15'59''.63$$

Para Vénus

$$s_0 = 8.34''$$

Para Marte

$$s_0 = 4.68''$$

Para a Lua pode considerar-se a seguinte expressão para o semi-diâmetro:

$$\sin s = k \sin \pi$$

onde  $k$  corresponde ao rácio entre o raio da Lua e o raio da Terra para o qual se pode usar o valor aproximado 0.2725 e em que  $\pi$  corresponde ao valor da paralaxe horizontal da Lua.

### Equação do Centro

Pela segunda lei de Kepler, sabe-se que numa órbita elíptica um corpo percorre as diferentes fases da sua órbita com velocidades diferentes. Este fenómeno acentua-se com o acentuar da excentricidade da órbita. Em astronomia chama-se **anomalia média**, à distância angular que o corpo teria do pericentro (periélio), caso se movesse sobre uma órbita circular com o mesmo período orbital. (Meeus, 1992), pp. 193-207.

A anomalia verdadeira corresponde ao verdadeiro ângulo que o corpo dista do periélio, visto que, a órbita, de facto, não é circular, mas sim elíptica. Esta pode ser calculada se conhecida a anomalia média e a equação do centro.

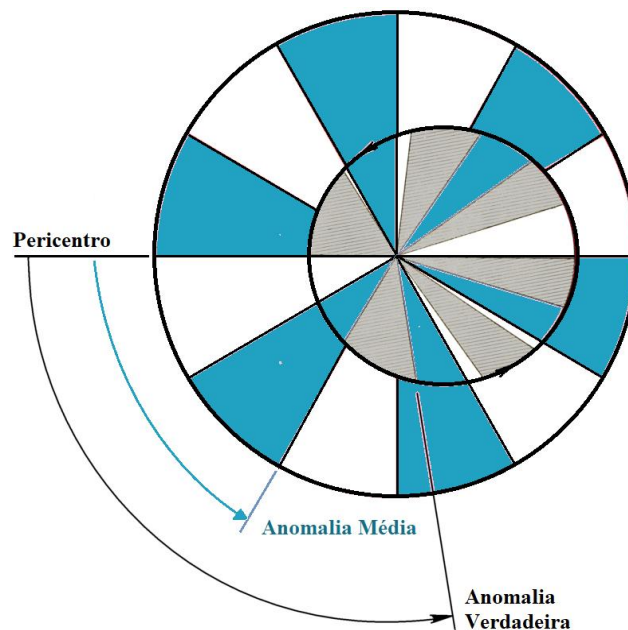


Figura 12– Anomalia média e verdadeira

A **equação do centro** é a diferença angular entre a verdadeira posição de um corpo sobre a sua órbita e a sua posição caso a órbita fosse circular e o seu movimento uniforme. Por outras palavras, pode definir-se a equação do centro como sendo a diferença entre a anomalia verdadeira e a anomalia média. O cálculo da equação do centro depende da excentricidade da órbita e da anomalia média.

Somando a equação do centro à longitude eclíptica média obtém-se a longitude eclíptica verdadeira.

### Aberração

A **aberração** é uma diferença verificada entre a posição verdadeira do sol e a posição aparente em que o vemos causada pela variação da incidência dos raios de luz na sequência do próprio movimento de translação da Terra. (Meeus, 1992), pp. 151.

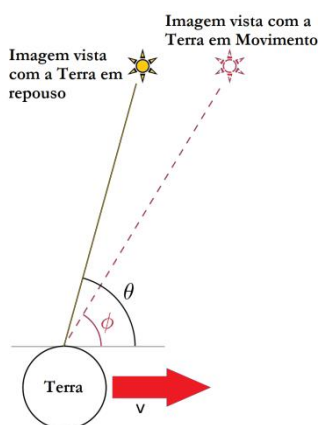


Figura 13– Aberração

Um exemplo elucidativo para explicar este fenómeno é o vento relativo sentido nos navios. Existe o verdadeiro e um relativo, o navio apenas capta o relativo pois o sensor não considera o seu próprio movimento.

No caso da aberração sofrida pela luz emitida pelas estrelas irá haver consequências diferentes da sofrida pelo Sol devido às diferentes consequências do movimento da Terra na forma como vemos o Sol ou as estrelas.

Assim para o seu cálculo é importante saber em que instante da eclíptica se encontra a Terra, ou seja, é necessário determinar a longitude eclíptica verdadeira do Sol.

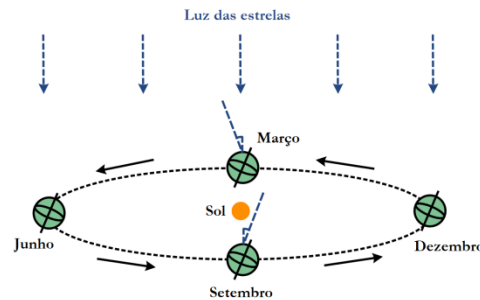


Figura 14 - Diferença de Aberração Sol Vs Estrelas

Somando o efeito da aberração e da nutação à Longitude Eclíptica do verdadeira do Sol obtêm-se a longitude eclíptica aparente.

## 1.2 Necessidade de Precisão

Nas várias etapas do processo da navegação astronómica, são calculados e medidos diversos dados. É importante saber com que precisão devem estes ser calculados ou medidos para que se mantenha a eficácia do processo.

Idealmente deve então ser-se suficiente preciso na medida em que se maximize a eficiência sem prejuízo da eficácia.

Qual é então a precisão necessária para o efeito?

A precisão usada será da ordem da décima do minuto do grau, pois é a fornecida pelo almanaque náutico, portanto já comprovadamente eficaz ao longo de anos e é também a precisão com que são medidas as alturas dos astros com recurso aos sextantes comuns.

## 1.3 A Teoria VSOP

A teoria VSOP (Do francês: Variations Séculaires des Orbites Planétaires) é um modelo matemático desenvolvido inicialmente em 1982 que descreve variações de longo prazo nas órbitas dos planetas do sistema solar. Os modelos mais antigos consideravam apenas a força gravitacional exercida entre o Sol e cada planeta por si só, o que previa órbitas elípticas keplerianas invariáveis.

A exactidão dos cálculos ficava assim comprometida uma vez que estas elipses são afetadas por pequenas alterações na forma e na orientação, causadas pelas forças exercidas pelos planetas uns sobre os outros.

O modelo inicial chama-se VSOP82, sendo que várias versões foram elaboradas posteriormente.

Na ferramenta elaborada no âmbito desta dissertação, foi utilizada uma versão adaptada do modelo VSOP87, conforme descrito em (Meeus, 1992), pp. 217-227.

Esta versão do modelo fornece diretamente as coordenadas heliocêntricas (Longitude, Latitude e Distância ao Sol).

## Capítulo 2. Algoritmos

Analisou-se o documento *Nautical Almanac Office Technical Note*, bem como o livro *Astronomical Algorithms* de Johan Meeus, que contêm as instruções que serviram de base para o elaborar dos algoritmos do programa.

O documento referido apresenta-nos 4 algoritmos de cálculos astronómicos com diferentes objectivos, dos quais serão abordados os seguintes: Algoritmo de Conversão para Dia Juliano, Algoritmo para obtenção de Efemérides do Sol, Algoritmo de cálculo das posições aparentes de Estrelas.

No documento, o calcular dos valores pretendidos nos diferentes algoritmos está dividido em passos que serão aqui abordados um por um, do ponto de vista da algoritmização, por forma a compreender de que forma são obtidos estes valores.

Quanto ao livro, a sua estrutura e disposição dos algoritmos foi adaptada para sintetizar os algoritmos usados na construção do programa.

### Algoritmo 1 - Cálculo do $\Delta T$

Conforme descrito no capítulo 1, a velocidade de rotação da Terra varia de forma não linear ao longo do tempo, levando a alguma imprevisibilidade das suas alterações futuras. Conforme referido isto gera uma diferença mutável ao longo do tempo entre a escala UT e a escala TT (que é uma escala de tempo uniforme). A esta diferença expressa normalmente em segundos dá-se o nome de  $\Delta T$ .

No entanto, observando a tendência de longo prazo da alteração desta velocidade de rotação é possível construir modelos que permitam não errar significativamente (do ponto de vista do rigor necessário para a navegação astronómica).

Ponderou-se a utilização de diversos algoritmos, sendo que o utilizado de facto, cruza informação de várias fontes, por forma a tentar maximizar o rigor deste cálculo para as mais variadas datas. Segundo (United-Kingdom-Hydrographic-Office, 2010) pode usar-se para efeitos da navegação astronómica as fórmulas:

$$\Delta T(t) = (28.43 + 4.525t + 1.404t^2) \times 10^{-8}$$

para datas entre 390 A.C e 948 D.C. e a fórmula

$$\Delta T(t) = 0.808(t - 2)^2 \times 10^{-8}$$

para qualquer outra data.

Em que  $t$  é o intervalo em séculos julianos, contado a partir da Época de referência J2000 até à data em causa.

Segundo (United-Kingdom-Hydrographic-Office, 2010) um erro de 2 minutos na utilização do  $\Delta T$  provoca um erro de apenas 0.1' no cálculo da ascensão reta, ao utilizar o método desta fonte no cálculo da mesma.

No entanto, para maximizar o rigor dos cálculos, procurou-se usar valores tabelados, corrigidos à décima do segundo. Para datas entre 1950 e 1999, usaram-se os valores fornecidos por (Meeus, 1992), pp. 79, dispostos na tabela abaixo.

Tabela 2 –  $\Delta T$  em segundos para os diferentes anos

year	$\Delta T$	year	$\Delta T$	year	$\Delta T$	year	$\Delta T$	year	$\Delta T$
1620	+121	1700	+7	1780	+16	1860	+7.7	1940	+24.3
1622	112	1702	7	1782	16	1862	7.3	1942	25.3
1624	103	1704	8	1784	16	1864	6.2	1944	26.2
1626	95	1706	8	1786	16	1866	5.2	1946	27.3
1628	88	1708	9	1788	16	1868	2.7	1948	28.2
1630	+82	1710	+9	1790	+16	1870	+1.4	1950	+29.1
1632	77	1712	9	1792	15	1872	-1.2	1952	30.0
1634	72	1714	9	1794	15	1874	-2.8	1954	30.7
1636	68	1716	9	1796	14	1876	-3.8	1956	31.4
1638	63	1718	10	1798	13	1878	-4.8	1958	32.2
1640	+60	1720	+10	1800	+13.1	1880	-5.5	1960	+33.1
1642	56	1722	10	1802	12.5	1882	-5.3	1962	34.0
1644	53	1724	10	1804	12.2	1884	-5.6	1964	35.0
1646	51	1726	10	1806	12.0	1886	-5.7	1966	36.5
1648	48	1728	10	1808	12.0	1888	-5.9	1968	38.3
1650	+46	1730	+10	1810	+12.0	1890	-6.0	1970	+40.2
1652	44	1732	10	1812	12.0	1892	-6.3	1972	42.2
1654	42	1734	11	1814	12.0	1894	-6.5	1974	44.5
1656	40	1736	11	1816	12.0	1896	-6.2	1976	46.5
1658	38	1738	11	1818	11.9	1898	-4.7	1978	48.5
1660	+35	1740	+11	1820	+11.6	1900	-2.8	1980	+50.5
1662	33	1742	11	1822	11.0	1902	-0.1	1982	52.2
1664	31	1744	12	1824	10.2	1904	+2.6	1984	53.8
1666	29	1746	12	1826	9.2	1906	5.3	1986	54.9
1668	26	1748	12	1828	8.2	1908	7.7	1988	55.8
1670	+24	1750	+12	1830	+7.1	1910	+10.4	1990	+56.9
1672	22	1752	13	1832	6.2	1912	13.3	1992	58.3
1674	20	1754	13	1834	5.6	1914	16.0	1994	60.0
1676	18	1756	13	1836	5.4	1916	18.2	1996	61.6
1678	16	1758	14	1838	5.3	1918	20.2	1998	63.0
1680	+14	1760	+14	1840	+5.4	1920	+21.1		
1682	12	1762	14	1842	5.6	1922	22.4		
1684	11	1764	14	1844	5.9	1924	23.5		
1686	10	1766	15	1846	6.2	1926	23.8		
1688	9	1768	15	1848	6.5	1928	24.3		
1690	+8	1770	+15	1850	+6.8	1930	+24.0		
1692	7	1772	15	1852	7.1	1932	23.9		
1694	7	1774	15	1854	7.3	1934	23.9		
1696	7	1776	16	1856	7.5	1936	23.7		
1698	7	1778	16	1858	7.6	1938	24.0		

Para datas entre 2000 e 2050 foram usados os dados e previsões disponibilizados por USNO (United States Naval Observatory) e IERS (International Earth Rotation and Reference Systems Service).

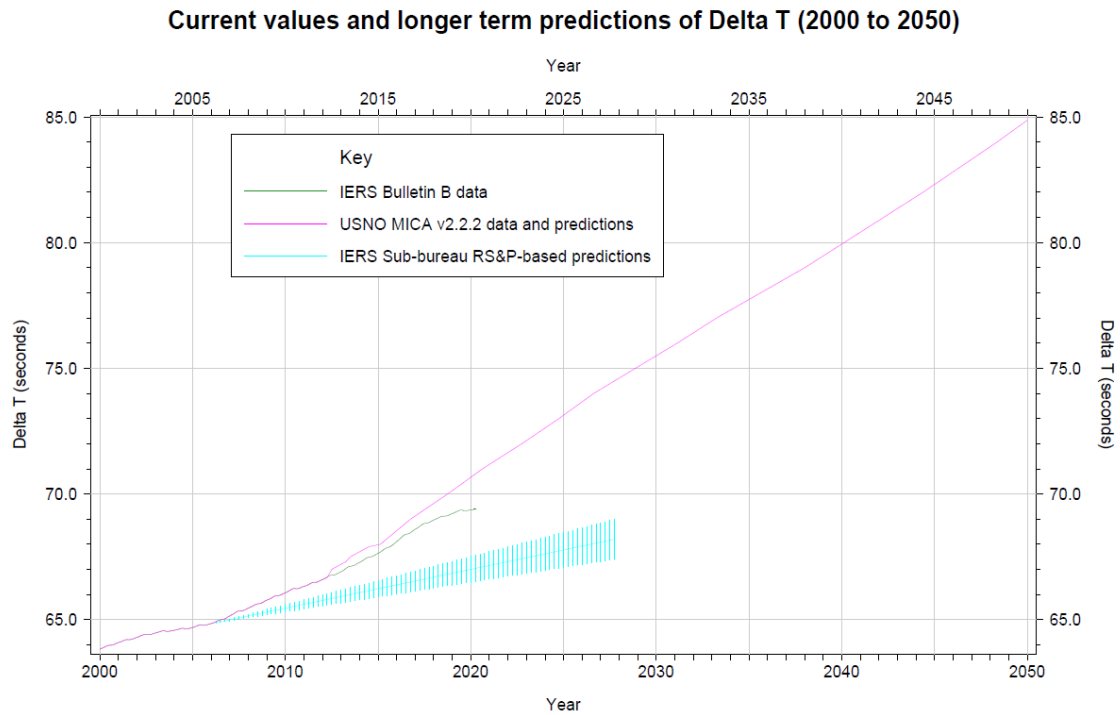


Figura 15 – Dados e previsões do USNO e IERS para o  $\Delta T$  de 2000 a 2050

Compilou-se então estas fontes de informação criando um algoritmo que usa:

- A tabela de (Meeus, 1992), para datas de 1950 a 2000.
- Uma média das previsões USNO e IERS, para o período 2000 a 2050.
- As fórmulas de (United-Kingdom-Hydrographic-Office, 2010) para os restantes períodos.

### Algoritmo 2 - Conversão para Dia Juliano

Este processo converte datas do calendário gregoriano ou Juliano às 0h UT devolvendo-nos o Dia Juliano correspondente para essa exacta data.

Deve inicialmente considerar a data para a qual pretendemos calcular o Dia juliano no formato Y-M-D (Ano, mês, Dia).

STEP 1 – Separação das datas anteriores ou posteriores ao mês de fevereiro

- Obter os parâmetros intermédios y e m:
  - Se  $M > 2$ , considerar  $y=Y$  e  $m=M-3$ ,
  - se  $M \leq 2$  considerar  $y=Y$  e  $m=M+9$

STEP 2 – Calcular o valor intermédio J

$$J = [365 \cdot 25(y + 4712)] + [30.6m + 0.5] + 59 + D - 0.5$$

STEP 3 – Determinar G, correção necessária para datas Gregorianas

- Se a data que se quer converter for Gregoriana definir

$$G = 38 - [3 [49 + y/100]/4]$$

- Se a data for Juliana considerar

$$G=0$$

STEP 4 – Calcular o Dia Juliano JD

$$JD = J + G$$

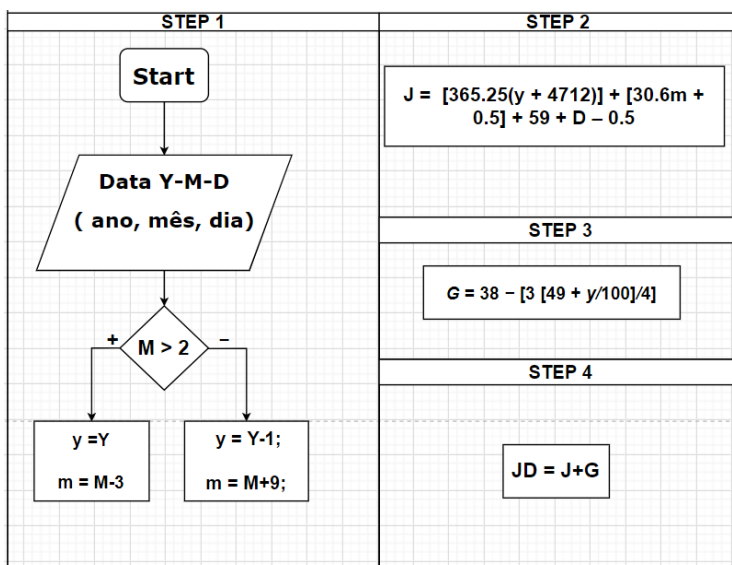


Figura 16– Fluxograma do algoritmo de conversão para Dia Juliano

Para obter o dia Juliano correspondente a um dado instante no tempo deve obter-se o JD correspondente ao dia e somar-lhe as horas do momento em questão divididas por 24. Por exemplo: 30/10/2020 às 14:30 UT . O dia referido corresponde ao JD = 2459152.5.

$$2459152.5 + 14.5/24 = 2459153.10417.$$

Caso se quisesse obter o JDE, a data que se pretende converter já deverá estar em TT. Como referido no capítulo 1, para converter de UT para TT, basta somar o  $\Delta T$  correspondente à data.

### Algoritmo 3 - Interpolação de 2º Grau

Os almanaques náuticos, astronómicos, ou outras publicações contêm tabelas que dispõem algumas quantidades y para valores equidistantes de um argumento x. Por exemplo: damos y como a ascensão reta e x são os diferentes dias do ano às 0<sup>h</sup>.

Chamamos interpolação ao processo que nos permite encontrar valores para instantes, quantidades, etc., intermédios aos fornecidos por uma tabela.

A tabela, no entanto, não tem de ser obrigatoriamente retirada de uma publicação. Um programa informático pode gerar dados sob esta forma.

Nesta ferramenta incluiu-se um algoritmo de interpolação para 3 valores tabelados.

Dados os três valores  $y_1, y_2, y_3$  da função y correspondentes aos objectos  $x_1, x_2$  e  $x_3$ .

STEP 1 – Calcular as diferenças a e b

$$a = y_2 - y_1$$

$$b = y_3 - y_2$$

STEP 2 – Calcular a diferença c

$$c = b - a$$

STEP 3 – Considerando n o objecto para o qual se pretende interpolar o valor de y, calcular y.

$$y = y_2 + \frac{n}{2} (a + b + nc)$$

#### Algoritmo 4 - Efeitos da Nutação

Este algoritmo permite-nos calcular a nutação em longitude e nutação na obliquidade. Estes são cálculos necessários para efectuar as correcções para obtenção das posições aparentes de corpos celestes.

A precisão do mesmo é de 0.5" para  $\Delta\psi$  e 0.1" para  $\Delta\varepsilon$ .

STEP 1- Calcular a longitude do nó ascendente da órbita média da Lua a partir do equinócio médio da data

$$\Omega = 125.04452 - 1934.136261 T + 0.0020708 T^2 + T^3/450000$$

STEP 2- Calcular as longitudes médias do Sol (L) e da Lua (L')

$$L = 280.46665 + 36000.7698 T$$

$$L' = 218.3165 + 481267.8813 T$$

(Remover Múltiplos de 360)

STEP 3- Obter a nutação na longitude  $\Delta\psi$  e nutação na obliquidade  $\Delta\varepsilon$ .

$$\Delta\psi = -17.20'' \sin(\Omega) - (1.32'' \sin(2 * L)) - 0.23'' \sin(2L') + 0.21'' \sin(2\Omega)$$

$$\Delta\varepsilon = 9.20'' \cos \Omega + 0.57'' \cos 2L + 0.10'' \cos 2L' - 0.09'' \cos 2\Omega$$

Quando necessário esta fórmula deve ser reescrita e aplicada em graus multiplicando os valores apresentados em segundos de arco por 3600.

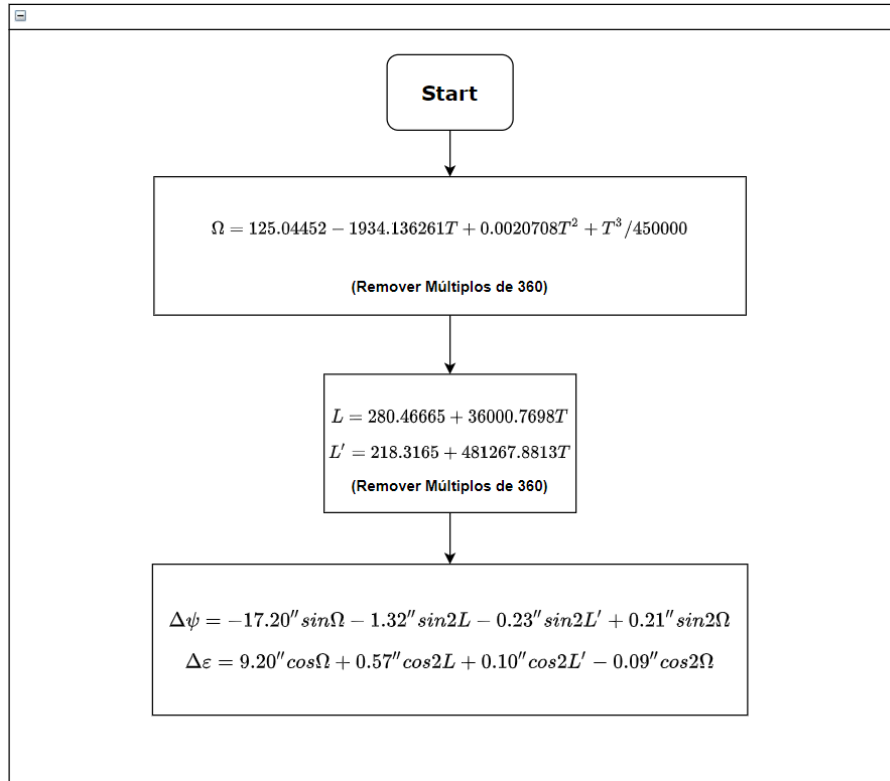


Figura 17 – Fluxograma do algoritmo de cálculo dos efeitos da nutação

### Algoritmo 5 - Cálculo do Tempo Sideral em Greenwich (TSG)

STEP 1- Calcular o JDE do instante pretendido.

STEP 2- Calcular o intervalo de tempo T decorrido desde a época de referência J2000.0.

Divide-se por 36525 para obter o resultado em séculos julianos de 36525 dias.

$$T = \frac{JDE - 2451545.0}{36525}$$

STEP 3- Calcular o TSG médio ( $\theta_0$ )

$$\theta_0 = 280.46061837 + 360.98564736629(JD - 2451545.0) + 0.000387933 T^2 - T^3/38710000$$

STEP 4- Calcular nutação na longitude  $\Delta\psi$ , nutação na obliquidade  $\Delta\varepsilon$ , e obliquidade média  $\varepsilon_0$ .

$$\varepsilon_0 = 23^\circ 26' 21.448'' - 46.8150'' T - 0.00059'' T^2 + 0.001813'' T^3$$

STEP 5- Obter a obliquidade verdadeira

$$\varepsilon = \varepsilon_0 + \Delta\varepsilon$$

STEP 6- Adicionar os efeitos da nutação e precessão para obter o TSG aparente:

$$TSG \text{ Aparente} = \theta_0 + \Delta\psi \cos \varepsilon$$

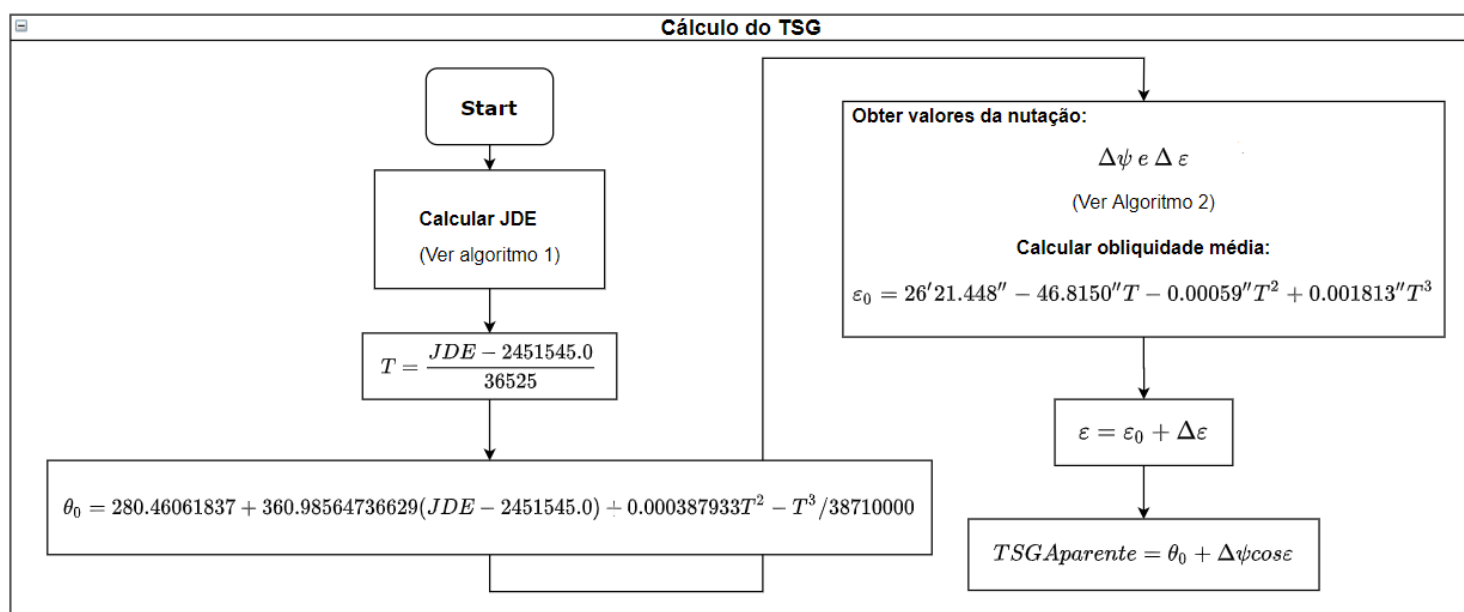


Figura 18 – Fluxograma do algoritmo de cálculo do Tempo Sideral em Greenwich

### Algoritmos para Obtenção de Coordenadas do Sol

Abaixo são apresentados dois algoritmos distintos para cálculo da posição do Sol.

O primeiro, elaborado segundo a nota técnica do almanaque náutico (United-Kingdom-Hydrographic-Office, 2010), é menos rigoroso, sendo que aproxima por fórmulas menos exactas os valores pretendidos nos vários passos. É, ainda assim, suficientemente rigoroso para a prática da navegação astronómica.

O segundo, elaborado de acordo com (Meeus, 1992), capítulo 25, baseia-se na teoria VSOP87, o que lhe confere uma exactidão bastante elevada e mais que suficiente do ponto

de vista da navegação astronómica. As diferenças consequentes da utilização dos dois algoritmos para o mesmo efeito são abordadas no capítulo 4.

### **Algoritmo 6 – Algoritmo da Nota Técnica**

Este algoritmo permite calcular a Ascensão reta, declinação e semi-diâmetro com uma exactidão de 0.6' durante o período de 2000 A.C até 2200 D.C.

STEP 1- Calcular Dia Juliano correspondente à data

- Converter data civil (Ano-Mes-Dia) em Dia juliano às 0h UT.
- Calcular  $D = JD + h/24 - 2451545 \cdot 0$  em que h é a hora do momento expressa em horas apenas i.e: 0730=7,5
- Calcular  $t = D/36525$ , onde t é o intervalo de 1 Janeiro 2000 Às 12h UT em séculos julianos de 36525 dias.

STEP 2- Conversão de Escala de tempo

- Converter para o intervalo de tempo correspondente T numa escala de tempo Uniforme (Tempo Dinâmico) a partir de:

$$T = t + \Delta T(t)$$

sendo:

$$\Delta T(t) = (28.43 + 4.525 t + 1.404 t^2) \times 10^{-8}$$

entre 390 A.C e 948 D.C, e

$$\Delta T(t) = 0.808(t - 2)^2 \times 10^{-8}$$

em qualquer outra data

Nota: Esta é a fórmula de  $\Delta T$  apresentada em (United-Kingdom-Hydrographic-Office, 2010). Para planetas usou-se um método distinto, apesar de a diferença entre os dois métodos não ser significativa no contexto da navegação astronómica.

STEP 3 - Cálculo dos argumentos solares:

- Calcular a Longitude eclíptica média da data

$$\lambda m = 280.46645 + 36000.76975 T + 0.0003132 T^2$$

Retirar múltiplos de  $360^\circ$  ao resultado obtido

- Obter a anomalia média da data

$$G = 357.529 + 35999.050 29 T$$

Retirar múltiplos de  $360^\circ$  ao resultado obtido

- Obter a obliquidade média da eclíptica pela expressão aproximada:

$$\varepsilon_0 = 23.4393 - 0.01301 T - 0.0000001 T^2 + 0.0000006 T^3$$

- Determinar a equação do centro

$$C = (1.9147 - 0.00482 T - 0.000015 T^2) \sin G + 0.01999 \sin 2G$$

STEP 4 – Determinar o efeito da nutação e obter os valores da longitude eclíptica aparente e obliquidade verdadeira

- Calcular o valor  $\Omega$  da longitude eclíptica do nó ascendente da órbita da Lua para determinar o efeito da nutação.

$$\Omega = 125.045 - 1934.136 T$$

- Obter o valor da obliquidade verdadeira da eclíptica somando à obliquidade média o efeito da nutação.

$$\varepsilon = \varepsilon_0 + 0.0026 \cos \Omega$$

- Aplicar a correcção do centro, a aberração e o efeito da nutação para obter  $\lambda$ , a longitude eclíptica aparente para a data:

$$\lambda = \lambda m + C - 0.0057 - 0.0048 \sin \Omega$$

Retirar múltiplos de  $360^\circ$  ao resultado obtido

STEP 5 - Converter para ascensão reta aparente e declinação da data ( $\alpha$ ,  $\delta$ ) e obter o semi-diâmetro (SD).

- Calcular os seguintes valores auxiliares:

$$x = \cos \lambda,$$

$$y = \cos \varepsilon \sin \lambda,$$

$$z = \sin \varepsilon \sin \lambda$$

- Calcular o valor auxiliar  $A$

$$A = \tan^{-1} y/x$$

Ou

$$A = \tan^{-1}(\cos \varepsilon \tan \lambda)$$

- Determinar a Ascensão Reta  $\alpha$ 
  - Se  $x < 0$  então  $\alpha = A + 180^\circ$

- Se  $x > 0$  e  $y < 0$  então  $\alpha = \mathcal{A} + 360^\circ$
- Para outras condições de  $x$  e  $y$ ,  $\alpha = \mathcal{A}$
- Obter a declinação

$$\delta = \sin^{-1} z$$

*ou*

$$\delta = \sin^{-1} (\sin \varepsilon \sin \lambda)$$

- Calcular o Semi-Diâmetro

$$SD = 0.2666 / (1 - 0.017 \cos G)$$

STEP 6 - Calcular Ângulo Horário em Greenwich GHA e equação de tempo E

$$GHA = 100.4606 + 36000.76998 t + 0.000387 t^2 + 15h - 0.0048 \sin \Omega \cos \varepsilon_0 - \alpha$$

Remover múltiplos de  $360^\circ$ .

Para efeito de conservação de exactidão é recomendável separar o termo  $t$  em dois elementos  $36000t + 0.76998t$ . Deve retirar-se os múltiplos de  $360^\circ$  ao primeiro termo antes de adicionar o segundo.

A equação de tempo E é calculada a partir de:

$$E = GHA - (15h - 180^\circ). \text{ Se } E > 10^\circ \text{ então } E = E - 360^\circ$$

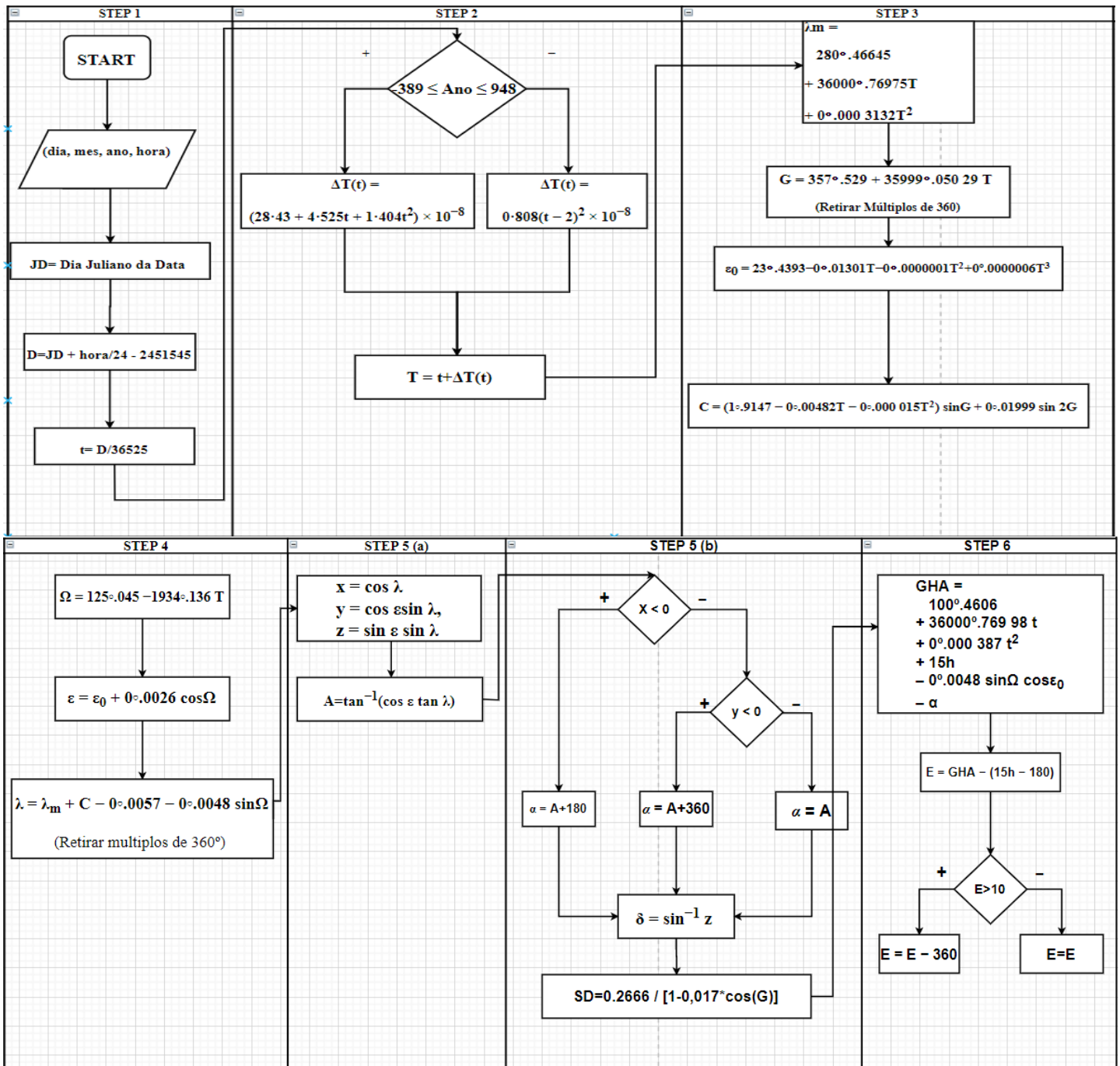


Figura 19 – Fluxograma do algoritmo 6 de cálculo de dados do Sol

### Algoritmo 7 - Algoritmo de Meus

Este algoritmo visa calcular a ascensão reta e declinação do Sol pra qualquer data UT inserida. A sua exactidão é da ordem de 1" entre os anos 2000 A.C e 6000 D.C.

STEP 1 - Calcular o JDE do instante pretendido.

STEP 2 - Calcular o intervalo de tempo  $T$  decorrido desde a época de referência J2000.0 em séculos julianos.

$$T = \frac{JD - 2451545.0}{36525}$$

STEP 3 - Calcular as coordenadas da Eclípticas Heliocêntricas da Terra ( $L$ ,  $B$  e  $R$ ) conforme descrito no algoritmo 12.

STEP 4 - Calcular a Longitude geocêntrica do Sol ( $\odot$ ) e Latitude Geocêntrica ( $\beta$ )

- $\odot = L + 180$
- $\beta = -B$

STEP 5 - Calcular a nutação na obliquidade ( $\Delta\psi$ ) e somar a  $\odot$ . (Ver algoritmo 4)

$$\odot = \odot + \Delta\psi$$

STEP 6- Aplicar o efeito da aberração

$$\odot = \odot - \frac{20''.4898}{R}$$

STEP 7- Converter para coordenadas equatoriais ( $AR$  e  $\delta$ ) usando o algoritmo 12.

STEP 8- Calcular o TSG (algoritmo 5) e calcular o GHA:

$$GHA = TSG - AR$$

### Algoritmo 8 - Cálculo da Posição Aparente de Estrelas

Este algoritmo permite calcular a posição aparente de estrelas para navegação usando as suas longitudes e latitudes eclípticas e as suas taxas de variação seculares em relação ao equinócio e época J2000. (United-Kingdom-Hydrographic-Office, 2010)

STEP 1- Calcular Dia Juliano correspondente à data

- Converter data civil (Ano-Mes-Dia) em Dia juliano às 0h UT.
- Calcular  $D = JD + h/24 - 2451545.0$  em que  $h$  é a hora do momento expressa em horas apenas i.e: 0730=7,5

- Calcular  $t = D/36525$ , onde  $t$  é o intervalo de 1 Janeiro 2000 Às 12h UT em séculos julianos de 36525 dias.

STEP 2 - Calcular as Longitudes  $\lambda_1$  e Latitudes  $\beta_1$  Eclípticas de cada estrela para a época e equinócio médio J2000.0

- Para cada estrela, resolver as equações

$$\lambda_1 = \lambda_0 + \mu t$$

$$\beta_1 = \beta_0 + \mu' t$$

Em que  $\lambda_0$  e  $\beta_0$  são as longitudes e latitudes relativas à eclíptica e equinócio médios J2000.0 e  $\mu$  e  $\mu'$  são as taxas de variação destes valores por século.

$\lambda_0$ ,  $\beta_0$ ,  $\mu$  e  $\mu'$  podem ser obtidos, para cada estrela, na tabela da página seguinte retirada de (United-Kingdom-Hydrographic-Office, 2010).

Tabela 3 – Tabela de  $\lambda_0$  e  $\beta_0$  para diferentes estrelas referentes à época J2000

No.	Name		$\lambda_0$	$\mu$	$\beta_0$	$\mu'$
1	Acamar	$\theta$ Eri	23 <sup>o</sup> -2723	-0 <sup>o</sup> -00152	-53 <sup>o</sup> -7402	+0 <sup>o</sup> -00112
2	Achernar	$\alpha$ Eri	345-3117	+0-00285	-59-3783	-0-00275
3	Acrux	$\alpha$ Cru	221-8701	-0-00047	-52-8787	-0-00070
4	Adhara	$\varepsilon$ CMa	110-7630	+0-00025	-51-3602	+0-00010
5	Aldebaran	$\alpha$ Tau	69-7892	+0-00104	-5-4674	-0-00550
6	Alioth	$\varepsilon$ UMa	158-9334	+0-00417	+54-3188	+0-00194
7	Alkaid	$\eta$ UMa	176-9331	-0-00430	+54-3880	-0-00230
8	Al Na'ir	$\alpha$ Gru	315-9070	+0-00184	-32-9133	-0-00536
9	Alnilam	$\varepsilon$ Ori	83-4636	-0-00002	-24-5064	-0-00007
10	Alphard	$\alpha$ Hya	147-2792	-0-00074	-22-3825	+0-00067
11	Alphecca	$\alpha$ CrB	222-2959	+0-00568	+44-3236	-0-00118
12	Alpheratz	$\alpha$ And	14-3085	+0-00162	+25-6804	-0-00575
13	Altair	$\alpha$ Aql	301-7765	+0-01939	+29-3035	+0-00733
14	Ankaa	$\alpha$ Phe	345-4938	-0-00100	-40-6331	-0-01237
15	Antares	$\alpha$ Sco	249-7623	-0-00007	-4-5699	-0-00061
16	Arcturus	$\alpha$ Boo	204-2337	-0-00768	+30-7363	-0-06288
17	Atria	$\alpha$ TrA	260-8962	+0-00123	-46-1513	-0-00075
18	Avior	$\varepsilon$ Car	173-1294	-0-00250	-72-6798	-0-00013
19	Bellatrix	$\gamma$ Ori	80-9464	-0-00032	-16-8161	-0-00037
20	Betelgeuse	$\alpha$ Ori	88-7547	+0-00080	-16-0270	+0-00026
21	Canopus	$\alpha$ Car	104-9614	+0-00308	-75-8239	+0-00076
22	Capella	$\alpha$ Aur	81-8579	+0-00126	+22-8643	-0-01191
23	Deneb	$\alpha$ Cyg	335-3293	+0-00029	+59-9061	-0-00002
24	Denebola	$\beta$ Leo	171-6176	-0-01153	+12-2669	-0-00849
25	Diphda	$\beta$ Cet	2-5835	+0-00673	-20-7836	-0-00191
26	Dubhe	$\alpha$ UMa	135-1975	-0-00239	+49-6802	-0-00343
27	Elnath	$\beta$ Tau	82-5750	+0-00037	+5-3851	-0-00491
28	Eltanin	$\gamma$ Dra	267-9687	-0-00080	+74-9223	-0-00055
29	Enif	$\varepsilon$ Peg	331-8850	+0-00090	+22-0999	-0-00029
30	Fomalhaut	$\alpha$ PsA	333-8604	+0-00716	-21-1357	-0-00802
31	Gacrux	$\gamma$ Cru	216-7397	+0-00737	-47-8312	-0-00543
32	Gienah	$\gamma$ Crv	190-7256	-0-00449	-14-5009	-0-00128
33	Hadar	$\beta$ Cen	233-7925	-0-00036	-44-1375	-0-00076
34	Hamal	$\alpha$ Ari	37-6625	+0-00364	+9-9651	-0-00569
35	Kaus Australis	$\varepsilon$ Sgr	275-0787	-0-00106	-11-0519	-0-00346
36	Kochab	$\beta$ UMi	133-3195	-0-00112	+72-9876	-0-00088
37	Markab	$\alpha$ Peg	353-4857	+0-00125	+19-4060	-0-00182
38	Menkar	$\alpha$ Cet	44-3201	-0-00091	-12-5856	-0-00197
39	Menkent	$\theta$ Cen	222-3086	-0-00873	-22-0800	-0-01871
40	Miaplacidus	$\beta$ Car	211-9692	-0-01254	-72-2357	-0-00329
41	Mirfak	$\alpha$ Per	62-0810	+0-00051	+30-1255	-0-00084
42	Nunki	$\sigma$ Sgr	282-3853	+0-00026	-3-4495	-0-00156
43	Peacock	$\alpha$ Pav	293-8176	-0-00041	-36-2677	-0-00244
44	Pollux	$\beta$ Gem	113-2156	-0-01700	+6-6842	-0-00436
45	Procyon	$\alpha$ CMi	115-7855	-0-01504	-16-0196	-0-03143
46	Rasalhague	$\alpha$ Oph	262-4487	+0-00459	+35-8352	-0-00609
47	Regulus	$\alpha$ Leo	149-8292	-0-00648	+0-4649	-0-00222
48	Rigel	$\beta$ Ori	76-8295	-0-00003	-31-1228	-0-00007
49	Rigil Kentaurus	$\alpha$ Cen	239-4793	-0-13521	-42-5959	-0-02399
50	Sabik	$\eta$ Oph	257-9696	+0-00084	+7-1978	+0-00275
51	Schedar	$\alpha$ Cas	37-7838	+0-00105	+46-6222	-0-00157
52	Shaula	$\lambda$ Sco	264-5858	+0-00007	-13-7884	-0-00079
53	Sirius	$\alpha$ CMa	104-0816	-0-01524	-39-6053	-0-03492
54	Spica	$\alpha$ Vir	203-8414	-0-00075	-2-0545	-0-00118
55	Suhail	$\lambda$ Vel	161-1877	-0-00116	-55-8708	+0-00011
56	Vega	$\alpha$ Lyr	285-3164	+0-01403	+61-7328	+0-00709
57	Zubenelgenubi	$\alpha^2$ Lib	225-0827	-0-00226	+0-3330	-0-00267
58	Polaris	$\alpha$ UMi	88-5676	+0-00098	+66-1014	-0-00118
59	$\sigma$ Octantis	$\sigma$ Oct	271-8706	+0-00118	-65-8402	-0-00042

STEP 3 - Corrigir a longitude e latitude considerando o efeito da aberração

- Determinar a longitude eclíptica do Sol
  - $\lambda_{\odot} = 280.460 + 36000.770 t$
  - retirar múltiplos de  $360^{\circ}$
- Calcular a longitude eclíptica corrigida
  - $\lambda_2 = \lambda_1 - 0.0057 \cos(\lambda_1 - \lambda_{\odot}) / \cos \beta_1$
- Calcular a Latitude corrigida
  - $\beta_2 = \beta_1 + 0.0057 \sin(\lambda_1 - \lambda_{\odot}) \sin \beta_1$

STEP 4 – Aplicar a precessão

- Calcular os parâmetros intermédios a, b e c respeitantes ao cálculo da alteração da posição do ponto vernal (equinócio médio) desde J2000 até à data

$$a = 1.39697 t + 0.000309 t^2$$

$$b = 0.0131 t - 0.00001 t^2$$

$$c = 5.1236 + 0.2416 t$$

- Obter os valores da longitude e latitude eclípticas para a data considerada a precessão

$$\beta_3 = \beta_2 + b \sin(\lambda_2 + c)$$

$$\lambda_3 = \lambda_2 + a - b \cos(\lambda_2 + c) \tan \beta_3$$

STEP 5 – Aplicar o efeito da nutação para obter as longitude e latitude eclípticas aparentes e o valor da obliquidade verdadeira da eclíptica

- Calcular o valor  $\Omega$  da longitude eclíptica do nó ascendente da órbita da Lua para determinar o efeito da nutação.

$$\Omega = 125.045 - 1934.136 T$$

- Obter o valor da obliquidade média da eclíptica para a data

$$\varepsilon_0 = 23.4393 - 0.0130 t$$

- Obter o valor da obliquidade verdadeira da eclíptica somando à obliquidade média o efeito da nutação.

$$\varepsilon = \varepsilon_0 + 0.0026 \cos \Omega$$

- Obter a longitude e latitude eclípticas verdadeiras

$$\lambda = \lambda_3 - 0.0048 \sin \Omega$$

$$\beta = \beta_3$$

STEP 6 – Converter para a Ascensão Reta e Declinação da data

- Calcular os valores auxiliares x, y, z

$$x = \cos \beta \cos \lambda$$

$$y = \cos \varepsilon \cos \beta \sin \lambda - \sin \varepsilon \sin \beta$$

$$z = \sin \varepsilon \cos \beta \sin \lambda + \cos \varepsilon \sin \beta$$

- Calcular o valor auxiliar  $A$

$$A = \tan^{-1} y/x$$

- Determinar a Ascensão Reta  $\alpha$ 
  - Se  $x < 0$  então  $a = A + 180^\circ$
  - Se  $x > 0$  e  $y < 0$  então  $a = A + 360^\circ$
  - Para outras condições de  $x$  e  $y$ ,  $a = A$

- Obter a declinação

$$\delta = \sin^{-1} z$$

STEP 7 – Calcular o GHA

$$\text{GHA} = 100^\circ.4606 + 36000^\circ.77005 t + 0^\circ.000388 t^2 + 15h - 0^\circ.0048 \sin\Omega \cos \epsilon - \alpha$$

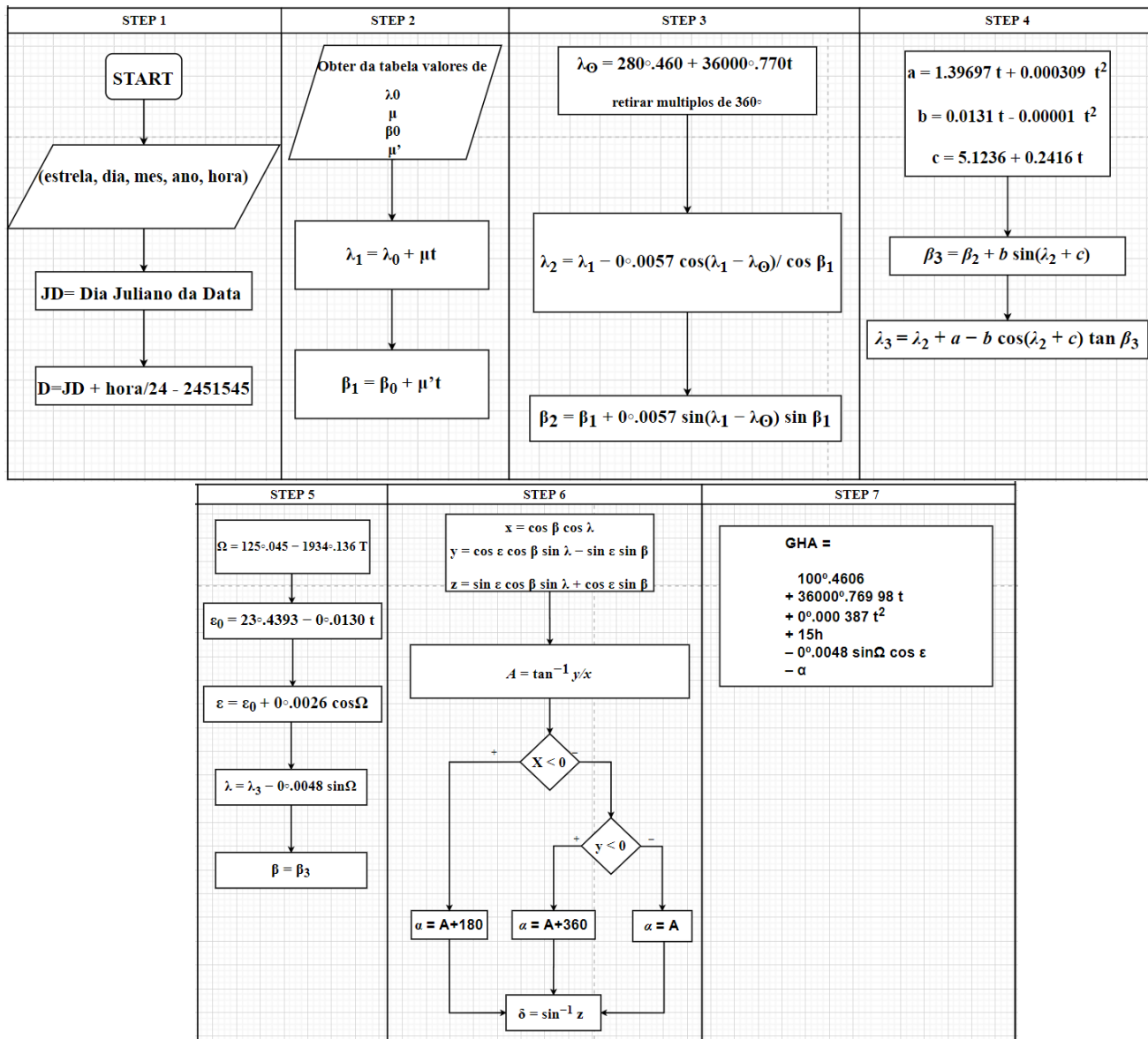


Figura 20 – Fluxograma do algoritmo de cálculo das posições aparentes das estrelas

### Algoritmo 9 - Cálculo da Posição Aparente da Lua

Para calcular com exactidão a posição da Lua para um dado instante, é necessário considerar centenas de termos periódicos da sua longitude, latitude e distância. No entanto para o rigor pretendido, pode limitar-se esta quantidade aos termos periódicos mais importantes. A exactidão dos resultados será, aproximadamente, de 10" na longitude da Lua e 4" na sua Latitude.

De acordo com (Meeus, 1992), apresenta-se o seguinte algoritmo para cálculo da posição da Lua:

STEP 1- Calcular o tempo T em séculos Julianos decorridos entre a a época de referência J2000 e a data para a qual se pretende calcular as coordenadas.

STEP 2- Calcular a longitude média e enlogação média da Lua, anomalia média do Sol e da Lua, e argumento de latitude da Lua referentes ao equinócio médio da data, tendo em conta o termo constante de efeito Luz-Tempo.(-0".70):

$$L' = 218.3164477 + 481267.88123421 T) - 0.0015786 T^2 + T^3/538841 - T^4/65194000$$

$$D = (297.8501921 + 445267.1114034 T - 0.0018819 T^2 + T^3/545868 - T^4/113065000)$$

$$M = (357.5291092 + 35999.0502909 T - 0.0001536 T^2 - T^3/24490000)$$

$$M' = (134.9633964 + + 477198.8675055 T + 0.0087414 T^2 + T^3/69699) - T^4/14712000$$

$$F = (93.2720950 + 483202.0175233 T - 0.0036539 T^2 - T^3/3526000) - T^4/863310000$$

STEP 3 – Calcular os argumentos A1, A2 e A3:

$$A1 = 119.75 + 131.849 T$$

$$A2 = 53.09 + 479264.290 T$$

$$A3 = 313.45 + 481266.484 T$$

STEP 4 – Calcular os somatórios  $\Sigma_l$  e  $\Sigma_r$  de acordo com os coeficientes das tabelas abaixo.

Estas tabelas, estão parcialmente transcritas na Tabela 4 e a sua versão completa deve ser consultada em (Meeus, 1992).

TABLE 47.A (cont.)

Argument				$\Sigma_l$	$\Sigma_r$
Multiple of D M M' F				Coefficient of the sine of the argument	Coefficient of the cosine of the argument
0	1	2	0	-2120	5751
0	2	0	0	-2069	
2	-2	-1	0	2048	-4950
2	0	1	-2	-1773	4130
2	0	0	2	-1595	
4	-1	-1	0	1215	-3958
0	0	2	2	-1110	
3	0	-1	0	-892	3258
2	1	1	0	-810	2616
4	-1	-2	0	759	-1897
0	2	-1	0	-713	-2117
2	2	-1	0	-700	2354
2	1	-2	0	691	
2	-1	0	-2	596	
4	0	1	0	549	-1423
0	0	4	0	537	-1117
4	-1	0	0	520	-1571
1	0	-2	0	-487	-1739
2	1	0	-2	-399	
0	0	2	-2	-381	-4421
1	1	1	0	351	
3	0	-2	0	-340	
4	0	-3	0	330	
2	-1	2	0	327	
0	2	1	0	-323	1165
1	1	-1	0	299	
2	0	3	0	294	
2	0	-1	-2		8752

TABLE 47.B

Periodic terms for the latitude of the Moon ( $\Sigma_b$ ).  
The unit is 0.000001 degree.

Argument				$\Sigma_b$	Argument				$\Sigma_b$
Multiple of D M M' F				Coefficient of the sine of the argument	Multiple of D M M' F				Coefficient of the sine of the argument
0	0	0	1	5128122	0	0	1	-3	777
0	0	1	1	280602	4	0	-2	1	671
0	0	1	-1	277693	2	0	0	-3	607
2	0	0	-1	173237	2	0	2	-1	596
2	0	-1	1	55413	2	-1	1	-1	491
2	0	-1	-1	46271	2	0	-2	1	-451
2	0	0	1	32573	0	0	3	-1	439
0	0	2	1	17198	2	0	2	1	422
2	0	1	-1	9266	2	0	-3	-1	421
0	0	2	-1	8822	2	1	-1	1	-366
2	-1	0	-1	8216	2	1	0	1	-351
2	0	-2	-1	4324	4	0	0	1	331
2	0	1	1	4200	2	-1	1	1	315
2	1	0	-1	-3359	2	-2	0	-1	302
2	-1	-1	1	2463	0	0	1	3	-283
2	-1	0	1	2211	2	1	1	-1	-229
2	-1	-1	-1	2065	1	1	0	-1	223
0	1	-1	-1	-1870	1	1	0	1	223
4	0	-1	-1	1828	0	1	-2	-1	-220
0	1	0	1	-1794	2	1	-1	-1	-220
0	0	0	3	-1749	1	0	1	1	-185
0	1	-1	1	-1565	2	-1	-2	-1	181
1	0	0	1	-1491	0	1	2	1	-177
0	1	1	1	-1475	4	0	-2	-1	176
0	1	1	-1	-1410	4	-1	-1	-1	166
0	1	0	-1	-1344	1	0	1	-1	-164
1	0	0	-1	-1335	4	0	1	-1	132
0	0	3	1	1107	1	0	-1	-1	-119
4	0	0	-1	1021	4	-1	0	-1	115
4	0	-1	1	833	2	-2	0	1	107

Tabela 4 – Extrato das tabelas do cálculo da posição da Lua

O argumento de cada seno (para  $\Sigma_l$  e  $\Sigma_b$ ) e coseno (para  $\Sigma_r$ ) será combinação linear dos 4 argumentos fundamentais D, M, M', F. Por exemplo, o argumento da oitava linha da tabela 47.A será 2D-M-M'. Assim, as contribuições desta linha para  $\Sigma_l$  e  $\Sigma_r$  serão, respectivamente:

$$+57066 \sin (2D-M-M')$$

e

$$-152138 \cos (2D-M-M').$$

No entanto, os termos cujos argumentos contenham o ângulo  $M$ , dependem da excentricidade da eclíptica que se encontra no presente a diminuir com o tempo. Por essa razão a amplitude desses termos é, na verdade, variável. Para considerar este efeito devem multiplicar-se os termos cujo argumento contenha  $M$  ou  $-M$  por  $E$ , e os termos que contenham  $2M$  ou  $-2M$  por  $E^2$  onde:

$$E = 1 - 0.002516 T - 0.0000074 T^2.$$

Por exemplo, o oitavo termo da longitude será então:

$$+57066 E \sin(2D - M - M').$$

STEP 5- Somar os aditivos necessários aos somatórios:

Aditivo a  $\sum_l$  :

$$+3958 \sin A_1$$

$$+1962 \sin(L^2 - F)$$

$$+318 \sin A_2$$

Aditivo a  $\sum_b$ :

$$- 2235 \sin L'$$

$$+ 382 \sin A_3$$

$$+ 175 \sin(A_1-F)$$

$$+ 175 \sin(A_1+F)$$

$$+127 \sin(L'-M')$$

$$-115 \sin(L'+M')$$

STEP 6- Calcular a longitude eclíptica  $\lambda$ , latitude eclíptica  $\beta$  e distância ao Sol  $\Delta$ :

$$\lambda = L' + \frac{\sum l}{1000000}$$

$$\beta = \frac{\sum b}{1000000}$$

$$\Delta = 385000.56 + \frac{\sum r}{1000}$$

Para obter o valor da paralaxe horizontal (HP):

$$HP = \sin^{-1} \frac{6378.14}{\Delta}$$

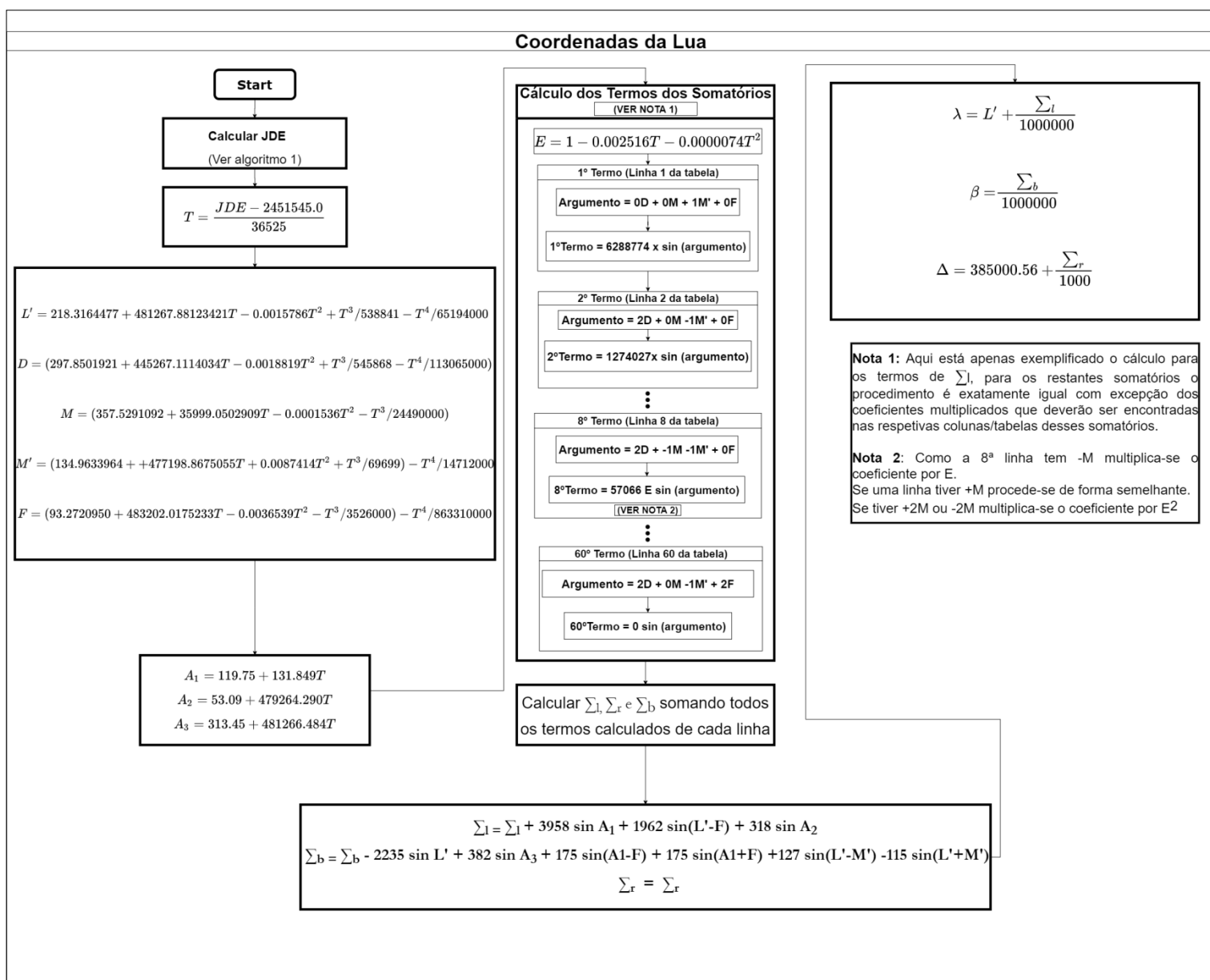


Figura 21 – Fluxograma do algoritmo de cálculo da posição da Lua

### Algoritmo 10 - Cálculo da Fração Iluminada da Lua

A fração iluminada do disco da Lua ( $k$ ) depende da enlongação selenocêntrica da Terra a partir do Sol ( $i$ ). Por outras palavras: depende do ângulo, observado a partir da Lua, entre a Terra e o Sol.

**STEP 1** – Calcular  $\psi$ , enlongação geocêntrica da Lua em relação ao Sol

$$\cos \psi = \sin \delta_0 \sin \delta + \cos \delta_0 \cos \delta \cos(\alpha_0 - \alpha)$$

onde  $\alpha_0, \delta_0$  são as ascensão reta e declinação geocêntricas do Sol, e  $\alpha, \delta$  são a ascensão reta e declinação geocêntricas da Lua.

**STEP 2** – Calcular  $i$

$$\tan i = \frac{R \sin \psi}{\Delta - R \cos \psi}$$

onde  $R$  é a distância Terra-Sol e  $\Delta$  a distância Terra-Lua (Ver algoritmos 7 e 9)

**STEP 3** – Calcular  $k$

$$k = \frac{1 + \cos i}{2}$$

### Algoritmo 11 - Cálculo da Posição Aparente de Planetas

O presente algoritmo dará as posições aparentes dos planetas de acordo com a teoria VSOP87 conforme descrito em (Meeus, 1992). A Longitude eclíptica será dada por  $L$ .  $B$  corresponderá à Latitude Eclíptica e  $R$  será a distância ao Sol.

As coordenadas  $L, B$  e  $R$  são coordenadas Eclípticas Heliocêntricas. É necessário proceder a uma conversão para obter as coordenadas Eclípticas Geocêntricas e posteriormente as Equatoriais pretendidas. Será descrito apenas o algoritmo para cálculo da posição de Venus de acordo com o método descrito, sendo que os restantes planetas são calculados da mesma forma.

**STEP 1** - Calcular o JDE correspondente ao instante pretendido.

STEP 2 - Calcular o intervalo T em milénios julianos desde a época J2000.0 ao instante pretendido.

$$T = \frac{JDE - 2451545.0}{365250}$$

STEP 3 - Deverão ser consultadas as tabelas da teoria VSOP87 para cada série (L0, L1, L2, L3, L4, L5, etc. para as longitudes), (B0, B1, B2, etc. para as latitudes) e R0, R1, etc. para a distância ao Sol. Para a versão completa desta tabelas, consultar (Meeus, 1992). Na tabela abaixo é apresentado um extrato, como exemplo.

Tabela 5 – Extrato de uma das tabelas da teoria VSOP87

VENUS	L1	1	1021 352 943 053	0	0
		2	95 708	2.464 24	10 213.285 55
		3	14 445	0.516 25	20 426.571 09
		4	213	1.795	30 639.857
		5	174	2.655	26.298
		6	152	6.106	1 577.344
		7	82	5.70	191.45
		8	70	2.68	9 437.76
		9	52	3.60	775.52
		10	38	1.03	529.69
		11	30	1.25	5 507.55
		12	25	6.11	10 404.73
VENUS	L2	1	54 127	0	0
		2	3 891	0.345 1	10 213.285 5
		3	1 338	2.020 1	20 426.571 1
		4	24	2.05	26.30
		5	19	3.54	30 639.86
		6	10	3.97	775.52
		7	7	1.52	1 577.34
		8	6	1.00	191.45
VENUS	L3	1	136	4.804	10 213.286
		2	78	3.67	20 426.57
		3	26	0	0
VENUS	L4	1	114	3.141 6	0
		2	3	5.21	20 426.57
		3	2	2.51	10 213.29
VENUS	L5	1	1	3.14	0
VENUS	B0	1	5 923 638	0.267 027 8	10 213.285 546 2
		2	40 108	1.147 37	20 426.571 09
		3	32 815	3.141 59	0
		4	1 011	1.089 5	30 639.856 6
		5	149	6.254	18 073.705
		6	138	0.860	1 577.344
		7	130	3.672	9 437.763
		8	120	3.705	2 352.866
		9	108	4.539	22 003.915
VENUS	B1	1	513 348	1.803 643	10 213.285 546
		2	4 380	3.386 2	20 426.571 1
		3	199	0	0
		4	197	2.530	30 639.857
VENUS	B2	1	22 378	3.385 09	10 213.285 55
		2	282	0	0
		3	173	5.256	20 426.571
		4	27	3.87	30 639.86
VENUS	B3	1	647	4.992	10 213.286
		2	20	3.14	0
		3	6	0.77	20 426.57
		4	3	5.44	30 639.86

Cada termo de cada série da Longitude eclíptica é dado por:

$$L_n = A \cos (B + C T)$$

Por exemplo a série L4 de vénus será dada por:

$$114 \cos(3.1416 + 0 \times T) + 3 \cos(5.21 + 20426.57 \times T) + 2 \cos(2.51 + 10213.29 \times T)$$

Cada termo da Latitude será então:

$$B_n = A \cos (B + C T)$$

STEP 4 – Calcular:

$$L = L_0 + L_1 + L_2 + L_3 + L_4 + L_5$$

$$B = B_0 + B_1 + B_2 + \dots + B_n$$

$$R = R_0 + R_1 + R_2 + \dots + R_n$$

Step 5 – Calcular pelo mesmo método  $L_0$ ,  $B_0$  e  $R_0$  – Coordenadas Eclípticas Heliocêntricas da Terra.

STEP 6 – Obter:

$$x = R \cos B \cos L - R_0 \cos B_0 \cos L_0$$

$$y = R \cos B \sin L - R_0 \cos B_0 \sin L_0$$

$$z = R \sin B - R_0 \sin B_0$$

STEP 7- Calcular a Longitude ( $\lambda$ ) e Latitude( $\beta$ ) Eclípticas Geocêntricas:

$$\tan \lambda = \frac{y}{x}$$

Procurar o quadrante apropriado para  $\lambda$  (ou usar uma função de “segundo” arco tangente)

$$\tan \beta = \frac{z}{\sqrt{x^2 + y^2}}$$

Em que  $n$  corresponde ao número de termos retidos e  $A$  corresponde ao coeficiente do termo retido mais pequeno, para cada planeta.

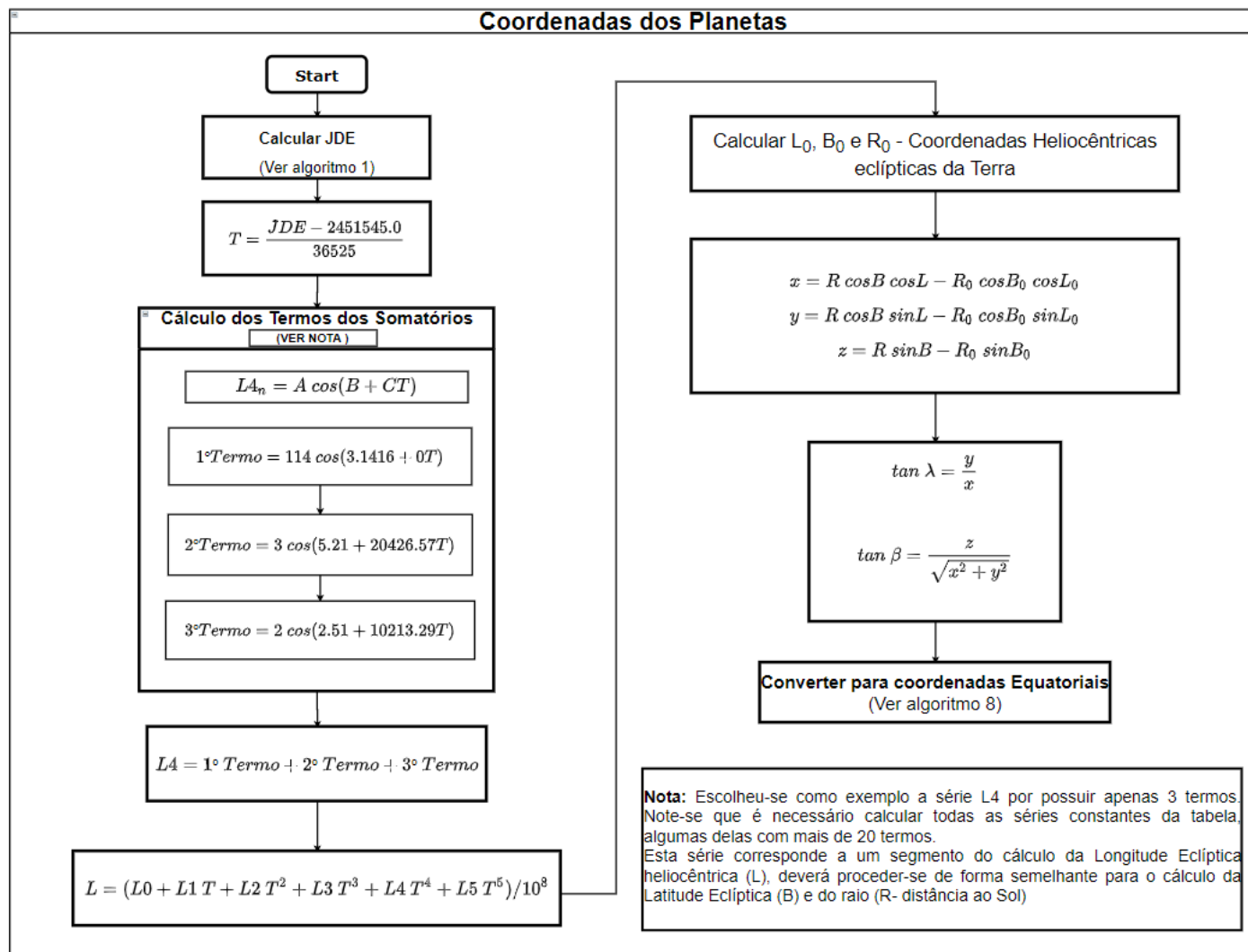


Figura 22 – Fluxograma do algoritmo de cálculo dos planetas

### Algoritmo 12 - Conversão de Coordenadas Eclípticas para Equatoriais

Dadas uma data e um par de coordenadas eclípticas: Longitude  $\lambda$  e Latitude  $\beta$  é possível converte-las para as coordenadas equatoriais equivalentes da seguinte forma:

STEP 1- Calcular o valor da obliquidade verdadeira para a data

- a) Encontrar o valor da obliquidade média  $\varepsilon_0$

$$\varepsilon_0 = 23^\circ 26' 21.448'' - 46.8150'' T - 0.00059'' T^2 + 0.001813'' T^3$$

- b) Calcular o valor da nutação em obliquidade e somar a  $\varepsilon_0$  para obter  $\varepsilon$ .

$$\varepsilon = \varepsilon_0 + \Delta\varepsilon$$

(Meeus, 1992)

STEP 2- Calcular os seguintes valores auxiliares:

$$x = \cos \lambda \cos \beta$$

$$y = \cos \varepsilon \sin \lambda$$

$$z = \sin \varepsilon \sin \lambda$$

STEP 3 - Calcular o valor auxiliar  $A$

$$A = \tan^{-1} y/x = \tan^{-1}(\cos \varepsilon \tan \lambda)$$

STEP 4-Determinar a Ascensão Reta  $\alpha$

- Se  $x < 0$  então  $a = A + 180^\circ$
- Se  $x > 0$  e  $y < 0$  então  $a = A + 360^\circ$
- Para outras condições de  $x$  e  $y$ ,  $a = A$

STEP 5- Obter a declinação a partir de:

$$\delta = \sin^{-1} z$$

(United-Kingdom-Hydrographic-Office, 2010)

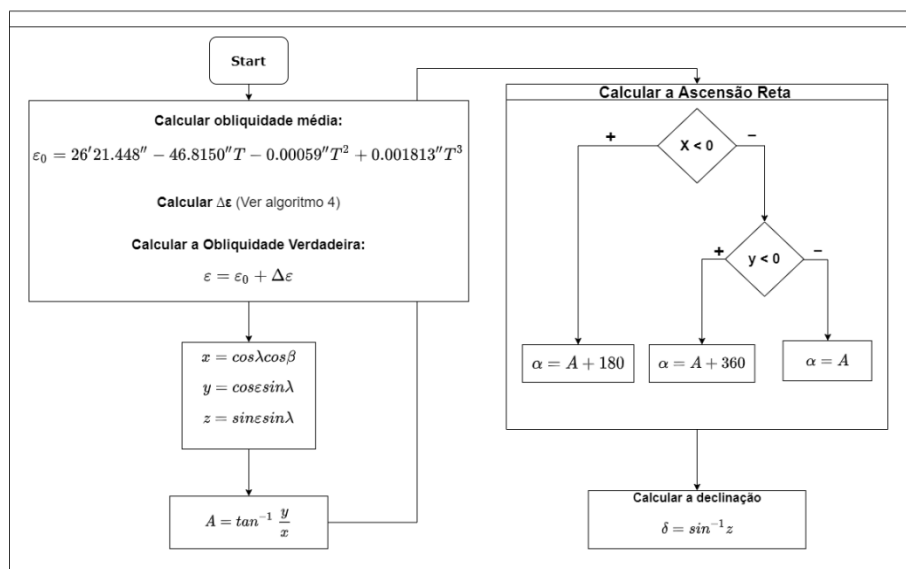


Figura 23 - Fluxograma do algoritmo de conversão de coordenadas eclípticas para equatoriais

### Algoritmo 13 - Cálculo de Nascimento, Ocaso e PM

O LHA correspondente à hora de nascimento ou ocaso de um corpo celeste é obtido segundo a fórmula:

$$\cos H_0 = -\tan \varphi \tan \delta$$

No entanto, a fórmula apresentada corresponde ao momento em que o centro do astro referido tem altura igual a 0. Usar-se-á então uma altitude standard correspondente ao momento verdadeiro do nascer e ocaso sendo que será:

$$h_0 = -0.5667, \text{ para estrelas e planetas}$$

$$h_0 = -0.8333, \text{ para o Sol}$$

Para a Lua o problema será mais complexo pois  $h_0$  não é constante, sendo então:

$$h_0 = -0.7275\pi - 0.5667, \text{ onde } \pi \text{ é a paralaxe horizontal da Lua.}$$

STEP 1 – Calcular o Tempo sideral aparente  $\theta_0$  às 0h UT do dia D pretendido no meridiano de Greenwich.

STEP 2- Calcular as ascensões retas  $\alpha$  e declinações  $\delta$  do astro nas datas D-1, D e D+1 às 0h TDT

$\alpha_1$  e  $\delta_1$  no dia D-1

$\alpha_2$  e  $\delta_2$  no dia D

$\alpha_3$  e  $\delta_3$  no dia D+1

STEP 3 – Calcular  $H_0$

$$\cos H_0 = \frac{\sin h_0 - \sin \varphi \sin \delta_2}{\cos \varphi \cos \delta_2}$$

Nota: Se o segundo membro não estiver entre -1 e 1 significa que o astro é circumpolar visível

Expressar  $H_0$  em graus.

STEP 4 – Calcular os seguintes instantes aproximados:

Para a Meridiana:

$$m0 = \frac{\alpha_2 + L - \theta_0}{360}$$

Para o Nascer:

$$m1 = m0 - \frac{H_0}{360}$$

Para o Ocaso:

$$m1 = m0 + \frac{H_0}{360}$$

Se algum destes valores não estiver entre 0 e 1 deverá adicionar-se ou subtrair 1 ao mesmo para que o valor se enquadre no intervalo referido.

STEP 5 – Calcular o tempo sideral em Greenwich correspondente aos instantes  $m_0$ ,  $m_1$ ,  $m_2$ .

$$\theta_0 = \theta_0 + 360.985647m$$

STEP 6 – Para  $n = m + \Delta T/86400$ , interpolar o valor de  $\alpha$  a partir de  $\alpha_1, \alpha_2, \alpha_3$ , e o valor de  $\delta$  a partir de  $\delta_1, \delta_2$  e  $\delta_3$

STEP 7– Encontrar o LHA do astro a partir de:

$$H = \theta_0 - L - \alpha$$

Calcular a altura do astro

$$h = \sin \varphi \sin \delta + \cos \varphi \cos \delta \cos H$$

STEP 8 – A correção para cada  $m$  será:

$$\text{Para a PM: } \Delta m = \frac{H}{360}$$

$$\text{Para nascimento/ocaso: } \Delta m = \frac{h-h_0}{360 \cos \delta \cos \varphi \sin H}$$

Para cada instante:  $m = m + \Delta m$ , o valor obtido deve ser convertido para horas multiplicando por 24. Se necessário repetir o processo de cálculo de  $\Delta m$  usando  $m$  como novo valor aproximado.

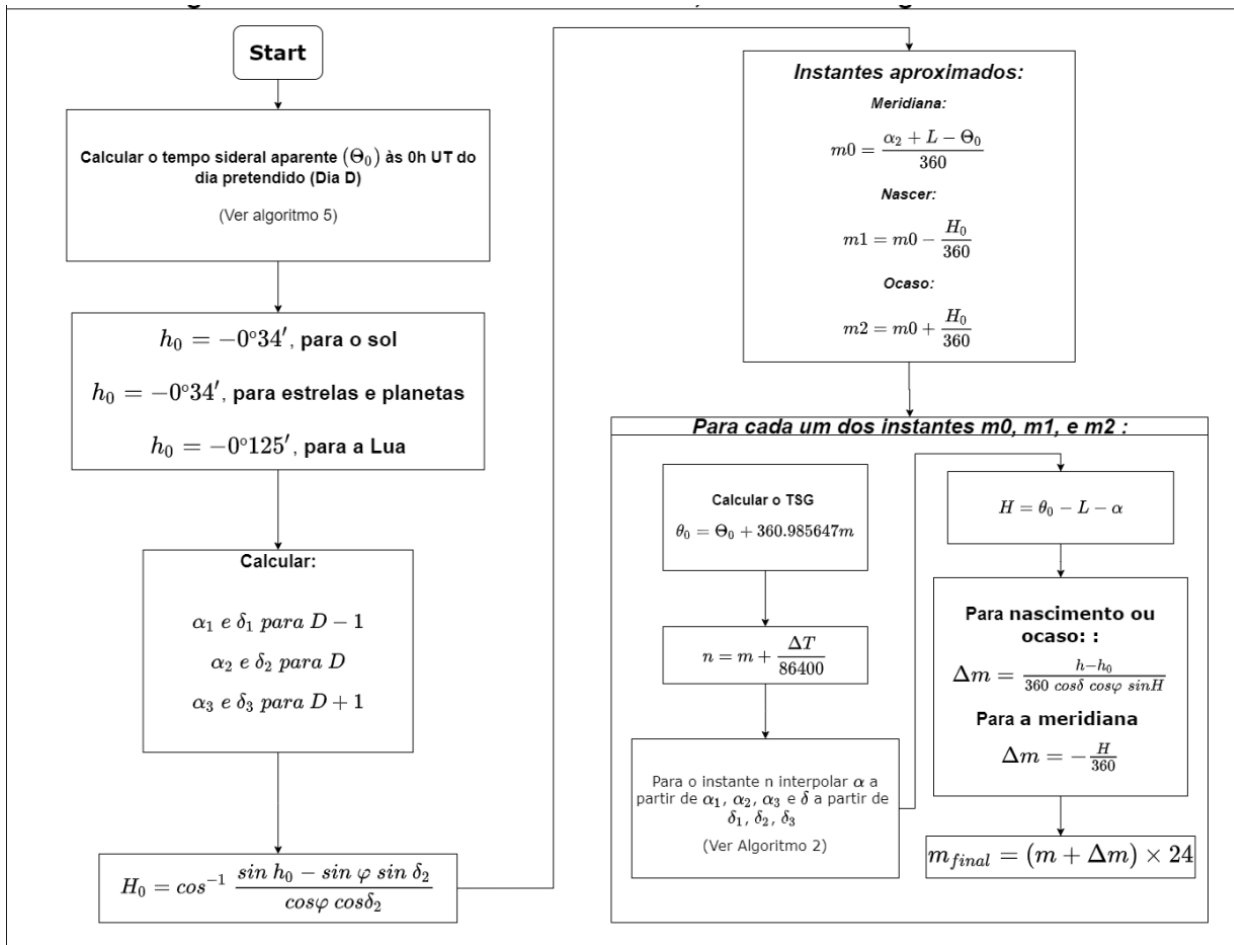


Figura 24 – Fluxograma do algoritmo de cálculo de Nascimento, Ocaso e Passagem Meridiana

### Algoritmo 14 - Cálculo das Horas dos Crepúsculos

O algoritmo para obtenção das horas dos crepúsculos é em tudo semelhante ao do nascer e ocaso do Sol segundo (Meeus, 1992), capítulo 15, com a exceção de que se procura encontrar o momento em que a altura do sol é  $-12^\circ$  e  $-6^\circ$  para os crepúsculos náutico e civil, respetivamente.

Ou seja  $h_0 = -12$  (nautico) e  $h_0 = -6$  (civil).

### **Algoritmo 15 - Cálculo de Horas Favoráveis para Observações ao Crepúsculo**

Este algoritmo é o procedimento leccionado na Escola Naval da Marinha de Guerra Portuguesa para cálculo das horas favoráveis de observação.

STEP 1- Calcular as horas dos nascer e ocaso do Sol para o dia pretendido bem como do crepúsculo civil.

STEP 2- Calcular a diferença de tempo  $\Delta t$  entre a hora do crepúsculo e a hora de nascer/ocaso. Aqui o processo será diferente para matutino/vespertino

AM:  $\Delta t = \text{hora do nascimento} - \text{hora do crepusculo matutino}$

PM:  $\Delta t = \text{hora do crepusculo vesp.} - \text{hora do ocaso}$

STEP 3- Calcular o início e fim dos períodos favoráveis.

$$\text{AM: } \begin{cases} \text{Início} = \text{hora do crepusculo} - \Delta t/2 \\ \text{Fim} = \text{hora do crepusculo} + \Delta t/2 \end{cases}$$

$$\text{PM: } \begin{cases} \text{Início} = \text{hora do crepusculo} - \Delta t/2 \\ \text{Fim} = \text{hora do crepusculo} + \Delta t/2 \end{cases}$$

### **Algoritmo 16 - Cálculo de Alturas e Azimutes**

Este algoritmo permite transformar coordenadas equatoriais em coordenadas horizontais para uma dada posição. Este é uma adaptação do algoritmo descrito em (Meeus, 1992), pp. 93.

$\theta_0$  - tempo sideral em Greenwich

LHA – ângulo horário no lugar

$\varphi$  - Latitude

L – Longitude

AR – Ascensão Reta

STEP 1 - Calcular o tempo sideral em Greenwich  $\theta_0$  para o instante pretendido

STEP 2 - Calcular:

$$LHA = \theta_0 - L - AR$$

(Retirar múltiplos de 360)

STEP 3 - Calcular a altura  $h$  e o valor intermédio  $Z$

$$h = \sin^{-1}(\sin \varphi \sin \delta + \cos \varphi \cos \delta \cos LHA)$$

$$Z = \cos^{-1} \left( \sin \delta - \frac{\sin \varphi \sin h}{\cos \varphi \cos h} \right)$$

STEP 4- Calcular o Azimute ( $Z_n$ )

Se:  $LHA > 180$ ,  $Z_n = Z$

Caso contrário,  $Z_n = 360 - Z$

### Algoritmo 17 - Cálculo de Sugestão de Observações

O algoritmo descrito abaixo procura efetuar uma busca seletiva balanceando alturas adequadas, magnitudes e distribuição dos azimutes das estrelas, obtendo um total de 7 estrelas bem distribuídas e o mais brilhantes possível, para o realizar de observações.

Uma vez calculados as alturas e azimutes de todos os astros visíveis (com altura superior a  $0^\circ$ ) para um dado instante e posição procede-se a uma selecção faseada que consiste em:

STEP 1- Seleccionar todos os planetas que tenham alturas entre  $10^\circ$  e  $70^\circ$  i.e: excluir os que não tenham alturas neste intervalo.

STEP 2 - Selecionar apenas as estrelas que tenham alturas entre os  $12^\circ$  e os  $59^\circ$  i.e: excluir as que não tenham alturas neste intervalo.

STEP 3- Dividir as estrelas em 3 grupos por azimutes:

$$\begin{cases} \text{Grupo 1} - \text{Estrelas com azimutes entre } 000 \text{ e } 160 \\ \text{Grupo 2} - \text{Estrelas com azimutes entre } 200 \text{ e } 360 \\ \text{Grupo 3} - \text{Estrelas com azimutes entre } 160 \text{ e } 200 \end{cases}$$

Dos grupos 1 e 2 serão seleccionadas 3 estrelas e do grupo 3 será seleccionada uma estrela (“estrela do meio”).

STEP 4-

- a) Encontrar as estrelas existentes entre 000 e 040. (no 1º grupo). Caso o sector esteja vazio, alargar o setor 10º sucessivamente até que se encontre pelo menos uma estrela no intervalo.
- b) Escolher a estrela mais brilhante do sector. Caso haja estrelas com a mesma magnitude, seleccionar a que tem azimute mais perto da fronteira inicial.

STEP 5- Proceder de forma semelhante ao STEP 4 para o grupo 3 (intervalo de 160 a 200). Encontrando a “estrela do meio” e respetivo azimute.

STEP 6- Escolher a primeira estrela do grupo 2 num procedimento semelhante ao STEP 4 mas usando como inicio do sector de busca: azimute da estrela do meio + 030 e fim do sector de busca: azimute da estrela do meio + 060.

STEP 7 –

- a) Definir novas fronteiras de procura para a segunda estrela dentro dos grupos 1 e 2:  
A fronteira inicial do setor deverá ser: azimute da primeira estrela +030 e a fronteira final deverá ser: azimute da primeira estrela + 060.
- b) Repetir o procedimento da alinea a) para encontrar a terceira estrela dos grupos 1 e 2.

### Capítulo 3. Descrição da Ferramenta

A obtenção de coordenadas celestes possíveis de serem utilizadas para a prática da Navegação astronómica é a principal função desta ferramenta operacional. Abaixo será apresentada e descrita a sua organização.

Para uma melhor compreensão das funcionalidades descritas é aconselhável a leitura deste capítulo acompanhada da experimentação prática do *software* disponível em anexo à dissertação. (o computador utilizado deverá dispor de sistema operativo Windows) :

Toda a construção deste programa foi realizada via *software* Matlab. No total o programa possui mais de 10.000 linhas de código que constituem funções complexas, como as de cálculos de coordenadas de planetas, que dentro delas englobam funções mais simples, como cálculo de conversões de unidades, modos de disposição, e funções próprias já existentes no *software* Matlab que devido à sua elevada quantidade e relativa simplicidade não serão aqui abordadas. i.e: não se considerou necessário ou relevante descrever como se convertem décimas de grau em minutos de arco, em Matlab, por exemplo.

#### 3.1 AstroNavTH2020

O AstroNavTH2020 consiste numa aplicação operacional que visa auxiliar o processo de Navegação astronómica. A existência destes programas é importante no exercer desta arte, pelo facilitar de cálculos que são normalmente trabalhosos de efectuar sem auxílio dos mesmos. Tem ainda uma vertente visual/didática que pode ser usada como auxílio no próprio ensino da navegação astronómica.

O programa está dividido em várias funcionalidades, disponíveis a partir de um Menu Inicial

As duas funcionalidades criadas no âmbito da presente dissertação de mestrado são a funcionalidade Almanaque e Planeamento de Observações

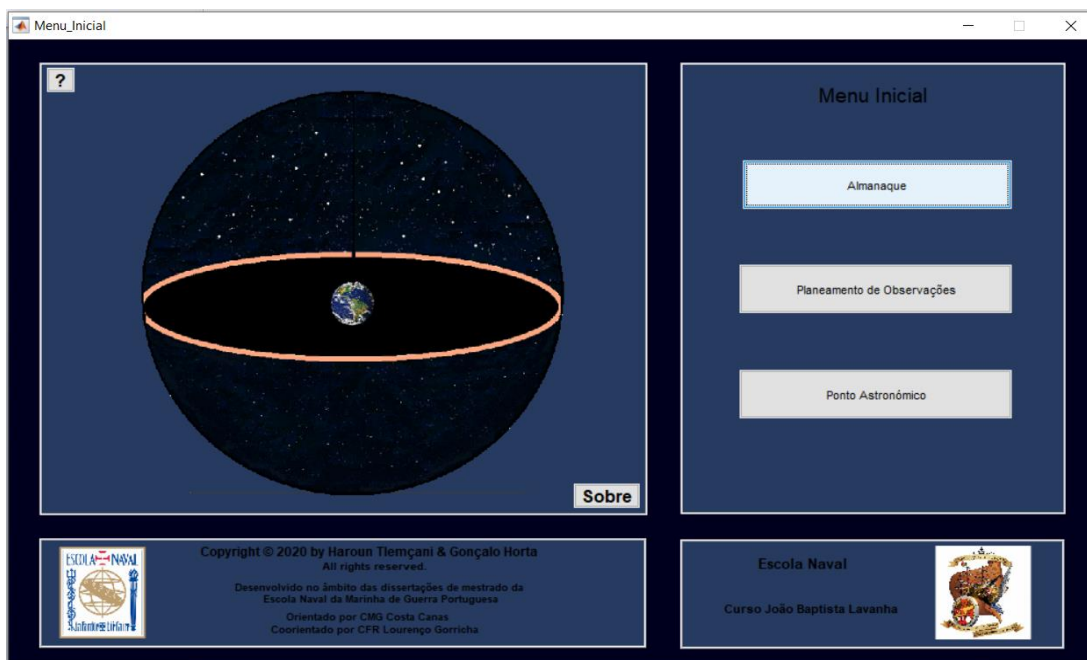


Figura 25 – Menu inicial do AstroNavTH2020

## Funcionalidade Almanaque

Esta funcionalidade pode ser acedida a partir do Menu Inicial abrindo automaticamente a GUI correspondente.

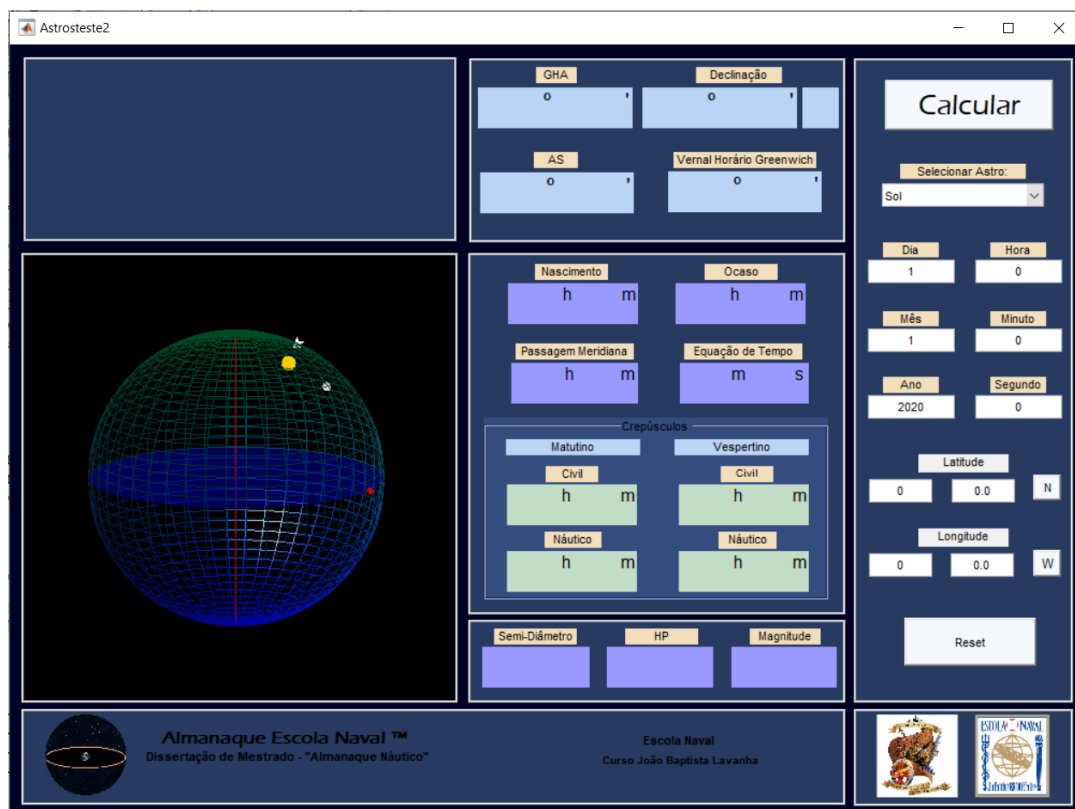


Figura 26 – Interface da funcionalidade Almanaque

Esta janela é onde são aplicados os principais algoritmos e visa fornecer as coordenadas GHA, e Declinação para os diferentes astros, para posterior aplicação no contexto da navegação astronómica. Desempenha ainda as funções de cálculo de horas de nascimentos e ocasos, bem como cálculo de: Crepúsculos, semi-diâmetros, HP's, passagem meridiana, equação de tempo, ângulo sideral, Vernal Hg e ainda a percentagem visível da Lua.

Ao utilizador basta seleccionar o astro pretendido e introduzir a data para a qual pretende os dados do mesmo.

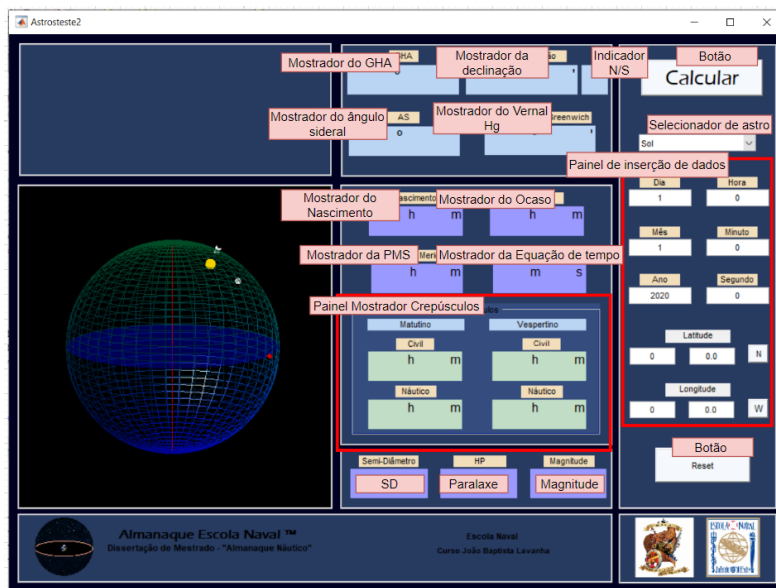


Figura 27 – Apresentação dos painéis, mostradores e botões da funcionalidade Almanaque

Uma vez nesta janela, é possível ao utilizador seleccionar um astro da seguinte lista:

Tabela 6 - Lista dos astros disponíveis no AstroNavTH2020

Sol	Acomar	Altair	Diphda	Markab	Rigil Kentaurus
Lua	Achernar	Ankaa	Dubhe	Menkar	Sabik
Vénus	Acrux	Antares	Elnath	Menkent	Schedar
Marte	Adhara	Arcturus	Eltanin	Miaplacidus	Shaula
Jupiter	Aldebaran	Atria	Enif	Mirfak	Sirius
Saturno	Alioth	Avior	Fomalhaut	Nunki	Spica
	Alkaid	Bellatrix	Gacrux	Peacock	Suhail
	Al Na'ir	Betelgeuse	Gienah	Pollux	Vega
	Alnilam	Canopus	Hadar	Procyon	Zubenelgenubi
	Alphard	Capella	Hamal	Rasalhague	Polaris
	Alphecca	Deneb	Kaus Australis	Regulus	S. Octantis
	Alpheratz	Denebola	Kochab	Rigel	

Esta encontra-se organizada dispondo primeiro Sol, Lua e planetas, e de seguida 57 estrelas por ordem alfabética, sendo que Polar e Octantis completam a lista das 59 no final.

Escolhido o astro pretendido, o utilizador pode selecionar uma data e hora para que lhe sejam então apresentados as coordenadas e dados desejados.

É ainda possível ao utilizador a inserção da sua posição que nesta funcionalidade alterará apenas as horas de nascimento e ocaso dos astros. É de destacar que estas horas serão sempre GMT.

### Funcionalidade Planeamento de Observações

Esta funcionalidade permite ao utilizador selecionar uma posição e uma data para as quais é possível:

- Calcular o período favorável para observações.
- Ver a disposição dos astros no Céu visível num gráfico 3D com possibilidade de rotação, bem como ver tabelados as respetivas coordenadas horizontais de cada astro.
- Ver uma sugestão de seleção de astros para observação baseada no algoritmo 17.
- Ver o movimento do Sol ao longo do dia acedendo aos seus azimutes e alturas na tabela disponibilizada.

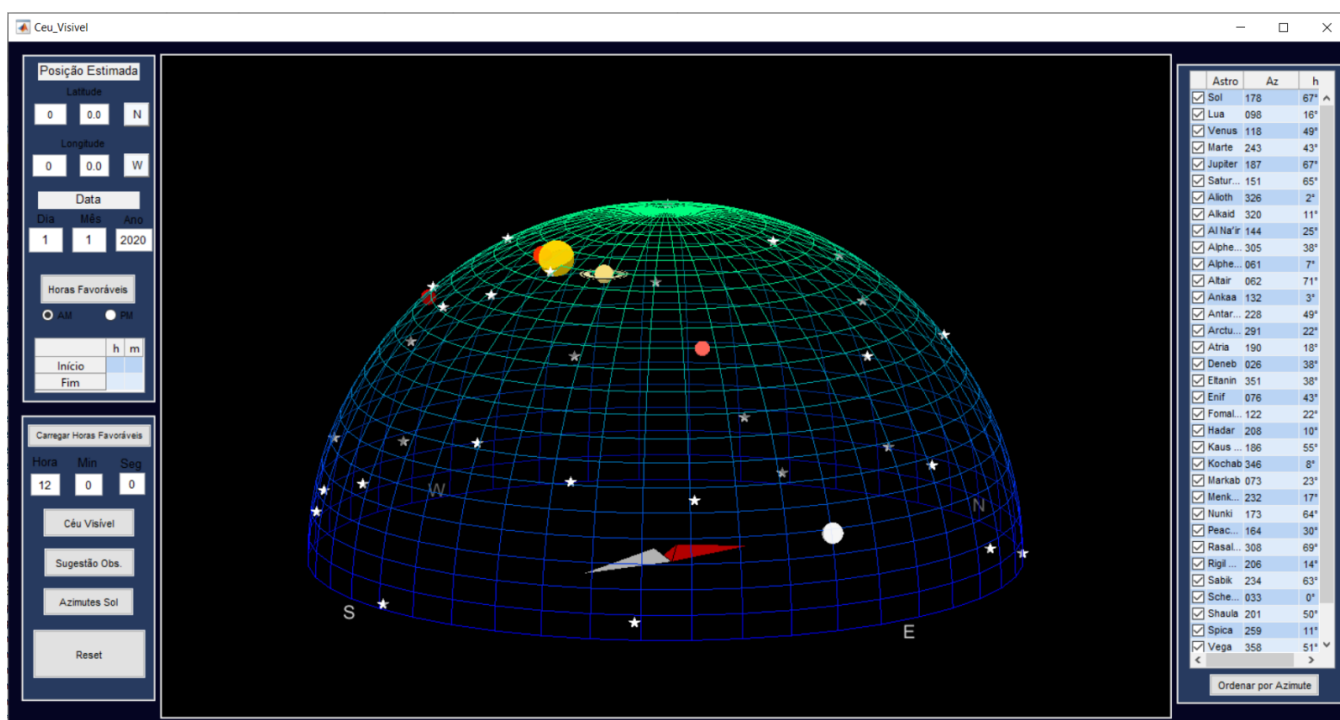


Figura 28 – Funcionalidade de Planeamento de observações- Botão “Céu Visível”

Esta interface tem um painel de inserção no canto superior esquerdo onde o utilizador deverá inserir a sua posição e data e pode, se quiser, seleccionar se pretende calcular as horas favoráveis para observações no período matutino ou no período vespertino. Estas serão disponibilizadas como hora de início e hora de fim na pequena tabela abaixo.

No painel em baixo deste é possível ao utilizador carregando no primeiro botão carregar as horas favoráveis calculadas no primeiro painel. As horas carregadas constituirão precisamente o instante médio entre as horas calculadas para início e fim do período favorável. O utilizador poderá também inserir uma qualquer hora à sua escolha e para esta data e hora poderá calcular através dos botões disponíveis:

A disposição de todos os astros visíveis no céu (acima da linha do horizonte), os astros sugeridos para observações, e no botão seguinte, a variação da posição do Sol no céu ao longo das horas do dia.

Por fim tem disponível um botão *Reset* que limpa todos os campos, gráfico e tabelas.

O gráfico tem os indicadores de azimute necessários para a sua correta interpretação, tais como marcação dos pontos cardeais, e os astros representados adequadamente: Sol a amarelo e maior que os planetas, a Lua a branco, Vénus a rosa, Marte a vermelho, Jupiter a Laranja e finalmente Saturno amarelo com os seus anéis desenhados. Todas as estrelas estão representadas de igual forma com o símbolo de uma estrela de 5 pontas a branco.

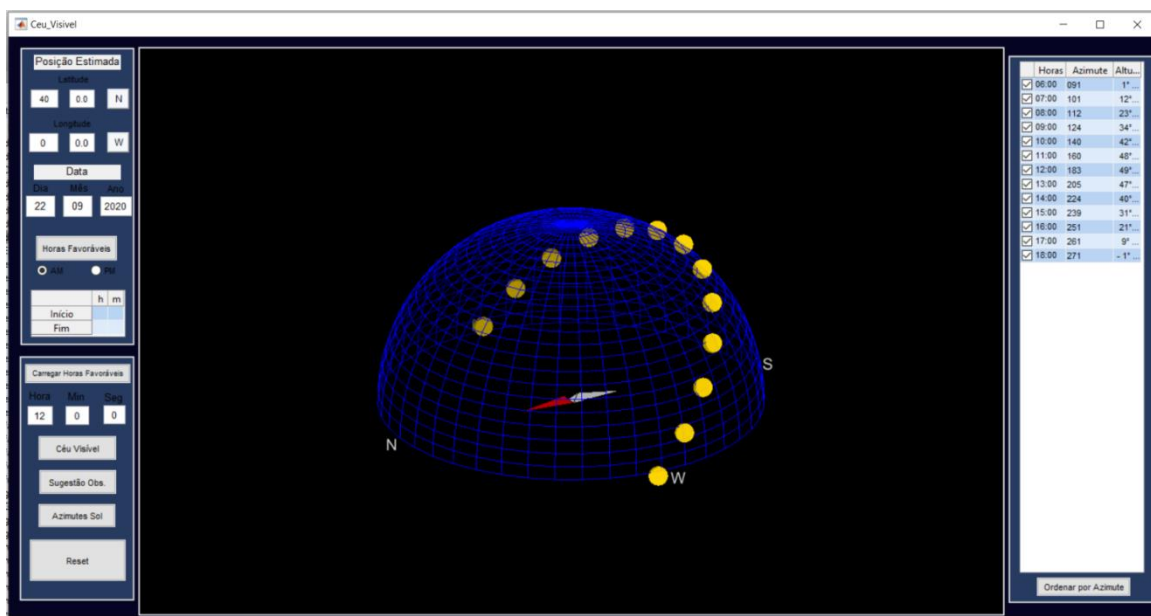


Figura 29 – Funcionalidade Planeamento de Observações – Botão “Azimutes do Sol”

O botão de Sugestão de Observações mostra-nos conforme o algoritmo 17, apenas 7 estrelas bem dispersas em azimute e com alturas adequadas. Para além destas exibe todos os planetas visíveis.

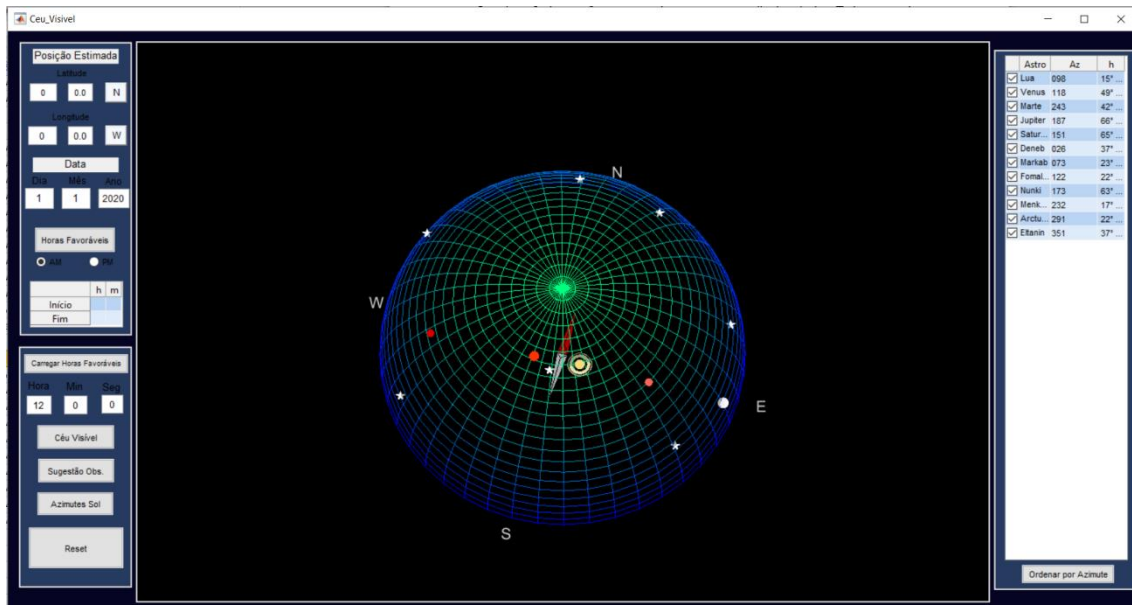


Figura 30 – Funcionalidade Planeamento de Observações- Botão “Sugestão de Observações”

## Capítulo 4. Discussão de Resultados

Idealmente, para testar a eficácia do programa, deveria ser realizado um estudo de comparação dos valores originados pelo programa com valores oficiais tabelados na versão oficial do Almanaque Náutico. Porém, tal estudo implicaria um infindável processo de digitação dos dados do almanaque, para que se conseguisse números suficientes para que o estudo fosse credível.

Assim, por forma a testar o programa em comparação com um dos programas referência utilizados na navegação astronómica (NavPac) construiu-se uma rotina de comparação de resultados para 33 datas (GDH's) distribuídas entre 2010 e 2020. O NavPac é um software disponibilizado pelo UKHO, a mesma entidade que publica as versões anuais do almanaque náutico usadas na Marinha de Guerra Portuguesa. Em relação aos dados do próprio almanaque náutico, um estudo simples revela que os dados do NavPac não divergem significativamente dos mesmos, sendo a maior parte dos desvios originados por aproximações e arredondamentos finais sem relevância no contexto da navegação astronómica.

Procurou-se que estas datas fossem uniformemente distribuídas ao longo dos vários meses do ano, dias do mês e horas do dia, para que todos os factores de influência estivessem envolvidos no estudo.

A tabela seguinte apresenta os maiores desvios (entre o AstroNavTH2020 e o NavPac) de GHA e Dec. encontrados no conjunto de 33 GDH's distintos.

Seguidamente apresentar-se-á um estudo pormenorizado para o desvio de alguns astros ao longo destes 33 GDH.

Todos os cálculos foram realizados com recurso ao programa Matlab.

Introduziram-se as diferentes datas no programa NavPac e copiaram-se os dados obtidos para documentos de texto que o Matlab consegue ler.

Após elaboradas as rotinas geradoras dos mesmos dados (para as mesmas datas e mesmos astros) usando os algoritmos mencionados acima, procedeu-se à elaboração de rotinas de comparação.

Estas rotinas calculam o módulo da diferença entre GHA's e declinações obtidos pelas duas fontes e organizam os dados de desvios obtidos por astro em tabelas que seguidamente se dispõem.

Nas tabelas que se seguem apresentar-se-ão os desvios, em minutos do grau, que cada astro revelou em relação ao NavPac ao longo dos 33 GDH's escolhidos.

As primeiras correspondem ao Sol usando os algoritmos 6 e 7:

#### 4.1 Sol

Tabela 8 – Resultados da comparação para o Sol com o Algoritmo 6

Data	Desvio_GHA	Desvio_Dec
'01/01/2010 - 12:00h'	0.3	0
'02/06/2010 - 18:00h'	0.2	0
'03/12/2010 - 20:51h'	0.2	0
'04/03/2011 - 01:30h'	0.1	0.1
'05/08/2011 - 11:00h'	0	0.1
'06/11/2011 - 22:30h'	0.1	0.1
'07/02/2012 - 02:00h'	0.3	0
'08/05/2012 - 14:30h'	0.3	0.1
'09/09/2012 - 21:00h'	0.2	0
'10/04/2013 - 03:00h'	0.4	0.2
'11/07/2013 - 10:30h'	0.3	0
'12/10/2013 - 20:00h'	0	0
'13/01/2014 - 02:00h'	0.2	0.1
'14/06/2014 - 07:00h'	0.3	0
'15/12/2014 - 21:00h'	0.4	0
'16/03/2015 - 09:00h'	0.3	0.1
'17/08/2015 - 17:00h'	0.1	0.1
'18/11/2015 - 22:00h'	0	0.1
'19/02/2016 - 05:00h'	0	0
'20/05/2016 - 16:00h'	0.2	0
'21/09/2016 - 21:00h'	0.4	0.2
'22/04/2017 - 08:00h'	0.4	0.1
'23/07/2017 - 19:00h'	0.3	0
'24/10/2017 - 23:00h'	0.1	0
'25/01/2018 - 00:00h'	0.1	0
'26/06/2018 - 12:30h'	0.2	0
'27/12/2018 - 22:00h'	0.1	0
'28/03/2019 - 02:00h'	0	0
'29/08/2019 - 07:00h'	0.3	0.1
'30/11/2019 - 18:00h'	0.3	0.1
'29/02/2020 - 00:00h'	0.2	0.1
'31/05/2020 - 12:00h'	0.2	0
'02/09/2020 - 23:00h'	0.2	0

Tabela 8 - Resultados da comparação para o Sol com o Algoritmo 7

Data	Desvio_GHA	Desvio_Dec
'01/01/2010 - 12:00h'	0	0
'02/06/2010 - 18:00h'	0	0
'03/12/2010 - 20:51h'	0	0
'04/03/2011 - 01:30h'	0.1	0
'05/08/2011 - 11:00h'	0	0
'06/11/2011 - 22:30h'	0	0
'07/02/2012 - 02:00h'	0	0
'08/05/2012 - 14:30h'	0	0
'09/09/2012 - 21:00h'	0	0
'10/04/2013 - 03:00h'	0.1	0
'11/07/2013 - 10:30h'	0	0
'12/10/2013 - 20:00h'	0	0
'13/01/2014 - 02:00h'	0	0
'14/06/2014 - 07:00h'	0	0
'15/12/2014 - 21:00h'	0	0
'16/03/2015 - 09:00h'	0	0
'17/08/2015 - 17:00h'	0	0
'18/11/2015 - 22:00h'	0	0
'19/02/2016 - 05:00h'	0	0
'20/05/2016 - 16:00h'	0	0
'21/09/2016 - 21:00h'	0	0
'22/04/2017 - 08:00h'	0	0
'23/07/2017 - 19:00h'	0	0
'24/10/2017 - 23:00h'	0	0
'25/01/2018 - 00:00h'	0	0
'26/06/2018 - 12:30h'	0	0
'27/12/2018 - 22:00h'	0	0
'28/03/2019 - 02:00h'	0	0
'29/08/2019 - 07:00h'	0.1	0
'30/11/2019 - 18:00h'	0	0
'29/02/2020 - 00:00h'	0	0
'31/05/2020 - 12:00h'	0	0
'02/09/2020 - 23:00h'	0	0

Verifica-se que os valores originados seguindo o algoritmo de Meeus coincidem mais com os do NavPac que os originados seguindo o algoritmo da nota técnica.

## 4.2 Lua

Procedeu-se de forma semelhante com a Lua:

Tabela 9 – Resultados da comparação para a Lua

Data	Desvio_GHA	Desvio_Dec
'01/01/2010 - 12:00h'	0	0
'02/06/2010 - 18:00h'	0.1	0.1
'03/12/2010 - 20:51h'	0	0
'04/03/2011 - 01:30h'	0	0
'05/08/2011 - 11:00h'	0.1	0.1
'06/11/2011 - 22:30h'	0	0
'07/02/2012 - 02:00h'	0.2	0.1
'08/05/2012 - 14:30h'	0.1	0.1
'09/09/2012 - 21:00h'	0	0
'10/04/2013 - 03:00h'	0	0.1
'11/07/2013 - 10:30h'	0.1	0.1
'12/10/2013 - 20:00h'	0	0.1
'13/01/2014 - 02:00h'	0	0
'14/06/2014 - 07:00h'	0	0.1
'15/12/2014 - 21:00h'	0	0.1
'16/03/2015 - 09:00h'	0.2	0.1
'17/08/2015 - 17:00h'	0	0
'18/11/2015 - 22:00h'	0.2	0.1
'19/02/2016 - 05:00h'	0.1	0.1
'20/05/2016 - 16:00h'	0	0
'21/09/2016 - 21:00h'	0.1	0.1
'22/04/2017 - 08:00h'	0.1	0.1
'23/07/2017 - 19:00h'	0	0.1
'24/10/2017 - 23:00h'	0.1	0
'25/01/2018 - 00:00h'	0.1	0
'26/06/2018 - 12:30h'	0	0
'27/12/2018 - 22:00h'	0	0
'28/03/2019 - 02:00h'	0	0
'29/08/2019 - 07:00h'	0	0
'30/11/2019 - 18:00h'	0.1	0.1
'29/02/2020 - 00:00h'	0.1	0
'31/05/2020 - 12:00h'	0.1	0.1
'02/09/2020 - 23:00h'	0	0.1

No caso da Lua os maiores desvios encontrados foram de 0.2' em GHA e 0.1' em declinação. Investigadas estas datas no *software* nav32 este apresenta um valor superior em 0.5' em relação ao NavPac e 0.7' superior ao AstroNavTH2020.

### 4.3 Vénus

Tabela 10 -- Resultados da comparação para Vénus

Data	Desvio_GHA	Desvio_Dec
'01/01/2010 - 12:00h'	0	0
'02/06/2010 - 18:00h'	0	0
'03/12/2010 - 20:51h'	0.3	0.1
'04/03/2011 - 01:30h'	0	0
'05/08/2011 - 11:00h'	0	0
'06/11/2011 - 22:30h'	0	0
'07/02/2012 - 02:00h'	0.1	0
'08/05/2012 - 14:30h'	0.1	0.1
'09/09/2012 - 21:00h'	0.1	0.1
'10/04/2013 - 03:00h'	0	0
'11/07/2013 - 10:30h'	0.1	0
'12/10/2013 - 20:00h'	0.1	0
<'13/01/2014 - 02:00h'	0.6	0.2 >
'14/06/2014 - 07:00h'	0.1	0
'15/12/2014 - 21:00h'	0	0
'16/03/2015 - 09:00h'	0	0
<'17/08/2015 - 17:00h'	0.8	0.2 >
'18/11/2015 - 22:00h'	0.1	0
'19/02/2016 - 05:00h'	0	0
'20/05/2016 - 16:00h'	0	0
'21/09/2016 - 21:00h'	0.1	0
'22/04/2017 - 08:00h'	0.2	0.1
'23/07/2017 - 19:00h'	0	0
'24/10/2017 - 23:00h'	0	0
'25/01/2018 - 00:00h'	0	0
'26/06/2018 - 12:30h'	0	0
'27/12/2018 - 22:00h'	0.1	0
'28/03/2019 - 02:00h'	0	0
'29/08/2019 - 07:00h'	0	0
'30/11/2019 - 18:00h'	0	0
'29/02/2020 - 00:00h'	0.1	0
'31/05/2020 - 12:00h'	0.6	0.2
'02/09/2020 - 23:00h'	0	0

Deu-se especial atenção às duas datas assinaladas na tabela para Vénus onde foram obtidos dois dos maiores desvios (0.6' e 0.8' de GHA e 0.2' de declinação). Deve destacar-se o facto de que em comparação com o *software* Nav32, para os mesmos GDH's, os resultados

obtidos pela ferramenta desenvolvida são coincidentes (desvio de 0.0'), sendo o NavPac que apresenta valores diferentes dos restantes dois.

#### 4.4 Marte

Tabela 11 – Resultados da comparação para Marte

Data	Desvio_GHA	Desvio_Dec
'01/01/2010 - 12:00h'	0	0
'02/06/2010 - 18:00h'	0	0
'03/12/2010 - 20:51h'	0	0
'04/03/2011 - 01:30h'	0	0
'05/08/2011 - 11:00h'	0	0
'06/11/2011 - 22:30h'	0	0
'07/02/2012 - 02:00h'	0	0
'08/05/2012 - 14:30h'	0	0.1
'09/09/2012 - 21:00h'	0	0
'10/04/2013 - 03:00h'	0	0
'11/07/2013 - 10:30h'	0	0
'12/10/2013 - 20:00h'	0	0
'13/01/2014 - 02:00h'	0	0
'14/06/2014 - 07:00h'	0	0
'15/12/2014 - 21:00h'	0	0
'16/03/2015 - 09:00h'	0	0
'17/08/2015 - 17:00h'	0	0
'18/11/2015 - 22:00h'	0	0
'19/02/2016 - 05:00h'	0.1	0
'20/05/2016 - 16:00h'	0	0.1
'21/09/2016 - 21:00h'	0.1	0
'22/04/2017 - 08:00h'	0	0
'23/07/2017 - 19:00h'	0	0
'24/10/2017 - 23:00h'	0	0
'25/01/2018 - 00:00h'	0.1	0
'26/06/2018 - 12:30h'	0	0
'27/12/2018 - 22:00h'	0	0
'28/03/2019 - 02:00h'	0	0
'29/08/2019 - 07:00h'	0	0
'30/11/2019 - 18:00h'	0	0
'29/02/2020 - 00:00h'	0	0
'31/05/2020 - 12:00h'	0	0
'02/09/2020 - 23:00h'	0	0

No caso de Marte os desvios são quase desprezáveis. Analisaram-se as datas em que houve desvios de 0.1' usando o nav32 como referência extra. Este coincidiu na maioria das vezes com o AstroNavTH2020.

## 4.5 Jupiter

Tabela 12 – Resultados da comparação para Jupiter

Data	Desvio_GHA	Desvio_Dec
'01/01/2010 - 12:00h'	0	0.1
'02/06/2010 - 18:00h'	0	0
'03/12/2010 - 20:51h'	0	0
'04/03/2011 - 01:30h'	0.1	0
'05/08/2011 - 11:00h'	0	0
'06/11/2011 - 22:30h'	0	0
'07/02/2012 - 02:00h'	0	0
'08/05/2012 - 14:30h'	0	0
'09/09/2012 - 21:00h'	0	0
'10/04/2013 - 03:00h'	0	0
'11/07/2013 - 10:30h'	0	0
'12/10/2013 - 20:00h'	0	0
'13/01/2014 - 02:00h'	0	0
'14/06/2014 - 07:00h'	0	0
'15/12/2014 - 21:00h'	0.1	0
'16/03/2015 - 09:00h'	0	0
'17/08/2015 - 17:00h'	0.1	0
'18/11/2015 - 22:00h'	0	0
'19/02/2016 - 05:00h'	0	0
'20/05/2016 - 16:00h'	0.1	0
'21/09/2016 - 21:00h'	0	0
'22/04/2017 - 08:00h'	0	0
'23/07/2017 - 19:00h'	0	0
'24/10/2017 - 23:00h'	0.1	0
'25/01/2018 - 00:00h'	0	0
'26/06/2018 - 12:30h'	0	0
'27/12/2018 - 22:00h'	0	0
'28/03/2019 - 02:00h'	0	0
'29/08/2019 - 07:00h'	0	0
'30/11/2019 - 18:00h'	0	0
'29/02/2020 - 00:00h'	0	0.1
'31/05/2020 - 12:00h'	0	0
'02/09/2020 - 23:00h'	0	0

Jupiter e Saturno, são dos planetas com trajectórias mais lentas nas respetivas órbitas em torno do Sol. As fórmulas usadas para o cálculo das suas coordenadas tornam-se assim

muito mais simples. (Meeus, 1992). Os cálculos das coordenadas destes planetas são assim também mais precisos e exatos. Os desvios de 0.1' podem ser considerados desprezáveis. Novamente o nav32 coincidiu maioritariamente com o AstroNavTH2020.

#### 4.6 Saturno

Saturno praticamente não apresentou desvios entre os dois programas excepto um desvio de 0.1' na declinação numa das datas. Mais uma vez o nav32 coincidiu exatamente com o

Tabela 13 - Resultados da comparação para Saturno

Data	Desvio_GHA	Desvio_Dec
'01/01/2010 - 12:00h'	0	0
'02/06/2010 - 18:00h'	0	0
'03/12/2010 - 20:51h'	0	0
'04/03/2011 - 01:30h'	0	0
'05/08/2011 - 11:00h'	0	0
'06/11/2011 - 22:30h'	0	0
'07/02/2012 - 02:00h'	0	0
'08/05/2012 - 14:30h'	0	0.1
'09/09/2012 - 21:00h'	0	0
'10/04/2013 - 03:00h'	0	0
'11/07/2013 - 10:30h'	0	0
'12/10/2013 - 20:00h'	0	0
'13/01/2014 - 02:00h'	0	0
'14/06/2014 - 07:00h'	0	0
'15/12/2014 - 21:00h'	0	0
'16/03/2015 - 09:00h'	0	0
'17/08/2015 - 17:00h'	0	0
'18/11/2015 - 22:00h'	0	0
'19/02/2016 - 05:00h'	0	0
'20/05/2016 - 16:00h'	0	0
'21/09/2016 - 21:00h'	0	0
'22/04/2017 - 08:00h'	0	0
'23/07/2017 - 19:00h'	0	0
'24/10/2017 - 23:00h'	0	0
'25/01/2018 - 00:00h'	0	0
'26/06/2018 - 12:30h'	0	0
'27/12/2018 - 22:00h'	0	0
'28/03/2019 - 02:00h'	0	0
'29/08/2019 - 07:00h'	0	0
'30/11/2019 - 18:00h'	0	0
'29/02/2020 - 00:00h'	0	0
'31/05/2020 - 12:00h'	0	0
'02/09/2020 - 23:00h'	0	0

AstroNavTH2020 neste valor de declinação.

#### 4.7 Polar

Tabela 14 - Resultados da comparação para a Polar

Data	Desvio_GHA	Desvio_Dec
'01/01/2010 - 12:00h'	0.5	0
'02/06/2010 - 18:00h'	0.5	0
'03/12/2010 - 20:51h'	1.6	0.1
'04/03/2011 - 01:30h'	1.4	0
'05/08/2011 - 11:00h'	0.1	0
'06/11/2011 - 22:30h'	0.2	0
'07/02/2012 - 02:00h'	0.4	0
'08/05/2012 - 14:30h'	0.8	0
'09/09/2012 - 21:00h'	1.2	0
'10/04/2013 - 03:00h'	1.3	0
'11/07/2013 - 10:30h'	1.4	0
'12/10/2013 - 20:00h'	1.7	0
'13/01/2014 - 02:00h'	1.8	0
'14/06/2014 - 07:00h'	0.7	0.1
'15/12/2014 - 21:00h'	1.1	0
'16/03/2015 - 09:00h'	0.2	0
'17/08/2015 - 17:00h'	1.3	0
'18/11/2015 - 22:00h'	0.8	0
'19/02/2016 - 05:00h'	0.1	0
'20/05/2016 - 16:00h'	1	0
'21/09/2016 - 21:00h'	0.6	0.1
'22/04/2017 - 08:00h'	0.6	0
'23/07/2017 - 19:00h'	0.9	0
'24/10/2017 - 23:00h'	0.3	0
'25/01/2018 - 00:00h'	0.1	0
'26/06/2018 - 12:30h'	0.3	0
'27/12/2018 - 22:00h'	0.1	0
'28/03/2019 - 02:00h'	0.5	0
'29/08/2019 - 07:00h'	0.1	0
'30/11/2019 - 18:00h'	0.3	0
'29/02/2020 - 00:00h'	0.3	0
'31/05/2020 - 12:00h'	0.2	0
'02/09/2020 - 23:00h'	0.4	0

As estrelas Polar e S.Octantis são os astros onde se encontrou mais desvios na comparação dos dois programas. Segundo (Meeus, 1992), é mais difícil calcular com exactidão as coordenadas de estrelas que se encontrem mais próximas dos polos celestes. Destaca-se

que as declinações das estrelas variam muito pouco, pois encontram-se muito longe da Terra. Para as estrelas Polar e S. Octantis os programas divergem bastante e divergem ambos ainda mais (por vezes cerca de 2° ou 3°) do programa nav32. Ou seja, comparativamente com os desvios em relação ao nav32, os resultados do AstroNavTH2020 em relação ao NavPac são praticamente coincidentes.

#### 4.8 S.Octantis

Tabela 15 - Resultados da comparação para S.Octantis

Data	Desvio_GHA	Desvio_Dec
'01/01/2010 - 12:00h'	0.2	0
'02/06/2010 - 18:00h'	1.1	0
'03/12/2010 - 20:51h'	1.3	0
'04/03/2011 - 01:30h'	0.2	0
'05/08/2011 - 11:00h'	0.2	0
'06/11/2011 - 22:30h'	0.1	0
'07/02/2012 - 02:00h'	0.2	0
'08/05/2012 - 14:30h'	0	0
'09/09/2012 - 21:00h'	0	0
'10/04/2013 - 03:00h'	0.3	0
'11/07/2013 - 10:30h'	0.3	0
'12/10/2013 - 20:00h'	0.3	0
'13/01/2014 - 02:00h'	0.5	0
'14/06/2014 - 07:00h'	0.5	0
'15/12/2014 - 21:00h'	0.4	0
'16/03/2015 - 09:00h'	0.3	0
'17/08/2015 - 17:00h'	0.1	0
'18/11/2015 - 22:00h'	0.2	0
'19/02/2016 - 05:00h'	0.7	0
'20/05/2016 - 16:00h'	0.8	0
'21/09/2016 - 21:00h'	0.1	0
'22/04/2017 - 08:00h'	0.4	0
'23/07/2017 - 19:00h'	0.4	0
'24/10/2017 - 23:00h'	0.9	0
'25/01/2018 - 00:00h'	1.2	0
'26/06/2018 - 12:30h'	0.8	0
'27/12/2018 - 22:00h'	1.1	0
'28/03/2019 - 02:00h'	0.8	0.1
'29/08/2019 - 07:00h'	0.4	0
'30/11/2019 - 18:00h'	0.4	0
'29/02/2020 - 00:00h'	0.1	0
'31/05/2020 - 12:00h'	0.2	0
'02/09/2020 - 23:00h'	0.5	0

#### 4.9 Outras estrelas

Para que se tenha noção da possibilidade de erro devido a declinações elevadas, escolheram-se para estudo 3 estrelas de declinações elevadas (Atria, Miaplacidus, Kochab) e 3 estrelas de declinações normalmente mais próximas do equador ( Alnilam, Bellatrix e Procyon).

##### Atria

A primeira estrela de declinação elevada mostrou alguma frequência de pequenos desvios maioritariamente desprezáveis.

Tabela 16 Resultados da comparação para a Atria

Data	Desvio_GHA	Desvio_Dec
'01/01/2010 - 12:00h'	0.1	0
'02/06/2010 - 18:00h'	0.1	0
'03/12/2010 - 20:51h'	0	0
'04/03/2011 - 01:30h'	0	0
'05/08/2011 - 11:00h'	0	0
'06/11/2011 - 22:30h'	0.1	0
'07/02/2012 - 02:00h'	0.1	0
'08/05/2012 - 14:30h'	0	0
'09/09/2012 - 21:00h'	0.1	0
'10/04/2013 - 03:00h'	0	0
'11/07/2013 - 10:30h'	0.1	0
'12/10/2013 - 20:00h'	0.1	0.1
'13/01/2014 - 02:00h'	0.1	0
'14/06/2014 - 07:00h'	0.1	0
'15/12/2014 - 21:00h'	0.1	0
'16/03/2015 - 09:00h'	0.1	0
'17/08/2015 - 17:00h'	0	0
'18/11/2015 - 22:00h'	0	0
'19/02/2016 - 05:00h'	0.1	0
'20/05/2016 - 16:00h'	0.1	0
'21/09/2016 - 21:00h'	0.1	0
'22/04/2017 - 08:00h'	0.1	0
'23/07/2017 - 19:00h'	0	0
'24/10/2017 - 23:00h'	0.1	0
'25/01/2018 - 00:00h'	0	0
'26/06/2018 - 12:30h'	0.1	0
'27/12/2018 - 22:00h'	0	0
'28/03/2019 - 02:00h'	0.1	0
'29/08/2019 - 07:00h'	0.1	0
'30/11/2019 - 18:00h'	0.1	0
'29/02/2020 - 00:00h'	0	0
'31/05/2020 - 12:00h'	0	0
'02/09/2020 - 23:00h'	0	0

## Miaplacidus

Tabela 17 - Resultados da comparação para a Miaplacidus

Data	Desvio_GHA	Desvio_Dec
'01/01/2010 - 12:00h'	0.1	0
'02/06/2010 - 18:00h'	0.1	0
'03/12/2010 - 20:51h'	0.1	0
'04/03/2011 - 01:30h'	0	0.1
'05/08/2011 - 11:00h'	0.1	0
'06/11/2011 - 22:30h'	0	0
'07/02/2012 - 02:00h'	0.1	0
'08/05/2012 - 14:30h'	0	0.1
'09/09/2012 - 21:00h'	0	0
'10/04/2013 - 03:00h'	0.1	0.1
'11/07/2013 - 10:30h'	0	0.1
'12/10/2013 - 20:00h'	0	0.1
'13/01/2014 - 02:00h'	0.1	0
'14/06/2014 - 07:00h'	0.1	0
'15/12/2014 - 21:00h'	0.1	0
'16/03/2015 - 09:00h'	0.1	0
'17/08/2015 - 17:00h'	0.1	0.1
'18/11/2015 - 22:00h'	0	0.1
'19/02/2016 - 05:00h'	0	0
'20/05/2016 - 16:00h'	0	0
'21/09/2016 - 21:00h'	0.1	0
'22/04/2017 - 08:00h'	0.1	0
'23/07/2017 - 19:00h'	0.1	0
'24/10/2017 - 23:00h'	0.1	0
'25/01/2018 - 00:00h'	0.1	0
'26/06/2018 - 12:30h'	0.2	0
'27/12/2018 - 22:00h'	0.1	0
'28/03/2019 - 02:00h'	0.1	0
'29/08/2019 - 07:00h'	0.1	0
'30/11/2019 - 18:00h'	0	0
'29/02/2020 - 00:00h'	0.1	0
'31/05/2020 - 12:00h'	0	0
'02/09/2020 - 23:00h'	0	0

O mesmo sucedeu com a Miaplacidus

## Kochab

Tabela 18 - Resultados da comparação para a Kochab

Data	Desvio_GHA	Desvio_Dec
'01/01/2010 - 12:00h'	0	0
'02/06/2010 - 18:00h'	0	0
'03/12/2010 - 20:51h'	0	0
'04/03/2011 - 01:30h'	0.1	0
'05/08/2011 - 11:00h'	0	0
'06/11/2011 - 22:30h'	0	0
'07/02/2012 - 02:00h'	0	0
'08/05/2012 - 14:30h'	0	0
'09/09/2012 - 21:00h'	0	0
'10/04/2013 - 03:00h'	0.1	0
'11/07/2013 - 10:30h'	0.1	0
'12/10/2013 - 20:00h'	0	0
'13/01/2014 - 02:00h'	0.1	0
'14/06/2014 - 07:00h'	0	0.1
'15/12/2014 - 21:00h'	0	0
'16/03/2015 - 09:00h'	0	0
'17/08/2015 - 17:00h'	0.1	0
'18/11/2015 - 22:00h'	0.1	0
'19/02/2016 - 05:00h'	0	0
'20/05/2016 - 16:00h'	0	0
'21/09/2016 - 21:00h'	0	0
'22/04/2017 - 08:00h'	0	0
'23/07/2017 - 19:00h'	0	0
'24/10/2017 - 23:00h'	0	0.1
'25/01/2018 - 00:00h'	0	0
'26/06/2018 - 12:30h'	0	0
'27/12/2018 - 22:00h'	0	0
'28/03/2019 - 02:00h'	0	0
'29/08/2019 - 07:00h'	0	0
'30/11/2019 - 18:00h'	0	0
'29/02/2020 - 00:00h'	0	0
'31/05/2020 - 12:00h'	0	0.1
'02/09/2020 - 23:00h'	0.1	0.1

Kochab repetiu o padrão embora com menos frequência.

## Alnilam

Tabela 19 - Resultados da comparação para a Alnilam

Data	Desvio_GHA	Desvio_Dec
'01/01/2010 - 12:00h'	0	0
'02/06/2010 - 18:00h'	0	0.1
'03/12/2010 - 20:51h'	0	0
'04/03/2011 - 01:30h'	0.1	0
'05/08/2011 - 11:00h'	0	0
'06/11/2011 - 22:30h'	0	0
'07/02/2012 - 02:00h'	0	0
'08/05/2012 - 14:30h'	0	0
'09/09/2012 - 21:00h'	0	0
'10/04/2013 - 03:00h'	0	0
'11/07/2013 - 10:30h'	0	0
'12/10/2013 - 20:00h'	0	0
'13/01/2014 - 02:00h'	0	0
'14/06/2014 - 07:00h'	0	0
'15/12/2014 - 21:00h'	0	0
'16/03/2015 - 09:00h'	0	0
'17/08/2015 - 17:00h'	0	0
'18/11/2015 - 22:00h'	0	0
'19/02/2016 - 05:00h'	0	0.1
'20/05/2016 - 16:00h'	0	0
'21/09/2016 - 21:00h'	0	0
'22/04/2017 - 08:00h'	0	0
'23/07/2017 - 19:00h'	0	0
'24/10/2017 - 23:00h'	0	0
'25/01/2018 - 00:00h'	0	0
'26/06/2018 - 12:30h'	0	0
'27/12/2018 - 22:00h'	0	0
'28/03/2019 - 02:00h'	0	0
'29/08/2019 - 07:00h'	0	0
'30/11/2019 - 18:00h'	0	0
'29/02/2020 - 00:00h'	0	0
'31/05/2020 - 12:00h'	0	0
'02/09/2020 - 23:00h'	0	0

Alnilam é a primeira das três estrelas de declinações mais próximas do Equador, revelando quase inexistência de desvios.

## Bellatrix

Tabela 20- Resultados da comparação para a Bellatrix

Data	Desvio_GHA	Desvio_Dec
'01/01/2010 - 12:00h'	0	0
'02/06/2010 - 18:00h'	0	0
'03/12/2010 - 20:51h'	0	0
'04/03/2011 - 01:30h'	0	0.1
'05/08/2011 - 11:00h'	0	0
'06/11/2011 - 22:30h'	0	0
'07/02/2012 - 02:00h'	0	0
'08/05/2012 - 14:30h'	0	0
'09/09/2012 - 21:00h'	0	0.1
'10/04/2013 - 03:00h'	0	0
'11/07/2013 - 10:30h'	0	0
'12/10/2013 - 20:00h'	0	0
'13/01/2014 - 02:00h'	0	0.1
'14/06/2014 - 07:00h'	0	0.1
'15/12/2014 - 21:00h'	0	0
'16/03/2015 - 09:00h'	0	0
'17/08/2015 - 17:00h'	0	0
'18/11/2015 - 22:00h'	0	0
'19/02/2016 - 05:00h'	0	0
'20/05/2016 - 16:00h'	0	0
'21/09/2016 - 21:00h'	0	0
'22/04/2017 - 08:00h'	0	0
'23/07/2017 - 19:00h'	0	0
'24/10/2017 - 23:00h'	0	0
'25/01/2018 - 00:00h'	0	0
'26/06/2018 - 12:30h'	0	0
'27/12/2018 - 22:00h'	0	0.1
'28/03/2019 - 02:00h'	0	0
'29/08/2019 - 07:00h'	0	0
'30/11/2019 - 18:00h'	0	0
'29/02/2020 - 00:00h'	0	0
'31/05/2020 - 12:00h'	0	0
'02/09/2020 - 23:00h'	0	0

O mesmo sucedeu com Bellatrix.

## Procyon

Tabela 21- Resultados da comparação para a Procyon

Data	Desvio_GHA	Desvio_Dec
'01/01/2010 - 12:00h'	0	0
'02/06/2010 - 18:00h'	0.1	0
'03/12/2010 - 20:51h'	0	0
'04/03/2011 - 01:30h'	0	0
'05/08/2011 - 11:00h'	0	0
'06/11/2011 - 22:30h'	0	0
'07/02/2012 - 02:00h'	0	0
'08/05/2012 - 14:30h'	0	0
'09/09/2012 - 21:00h'	0	0
'10/04/2013 - 03:00h'	0	0
'11/07/2013 - 10:30h'	0	0
'12/10/2013 - 20:00h'	0	0
'13/01/2014 - 02:00h'	0	0
'14/06/2014 - 07:00h'	0	0
'15/12/2014 - 21:00h'	0	0
'16/03/2015 - 09:00h'	0	0
'17/08/2015 - 17:00h'	0	0
'18/11/2015 - 22:00h'	0	0
'19/02/2016 - 05:00h'	0	0
'20/05/2016 - 16:00h'	0.1	0
'21/09/2016 - 21:00h'	0	0
'22/04/2017 - 08:00h'	0.1	0
'23/07/2017 - 19:00h'	0	0
'24/10/2017 - 23:00h'	0	0
'25/01/2018 - 00:00h'	0	0
'26/06/2018 - 12:30h'	0	0
'27/12/2018 - 22:00h'	0	0
'28/03/2019 - 02:00h'	0	0
'29/08/2019 - 07:00h'	0	0
'30/11/2019 - 18:00h'	0	0
'29/02/2020 - 00:00h'	0	0
'31/05/2020 - 12:00h'	0	0
'02/09/2020 - 23:00h'	0	0

Procyon apresentou resultados semelhantes.

Comprova-se de facto, um ligeiro aumento dos desvios quando se determinam as coordenadas de estrelas de declinações elevadas, apesar de estes pequenos desvios serem maioritariamente desprezáveis.

#### 4.10 Estudo Geral

Foram ainda realizadas duas tabelas em que se apresenta o astro que apresentou maior desvio para cada GDH. Na primeira tabela todos os 65 astros estão contemplados no algoritmo:

Tabela 22 – Astros com maiores desvios em cada data

Data	Maior_Desvio_GHA	Astro_GHA	Maior_Desvio_Dec	Astro_Dec
'01/01/2010 - 12:00h'	0.5	'Polar'	0.1	'Achernar'
'02/06/2010 - 18:00h'	1.1	'Octantis'	0.1	'Alnilam'
'03/12/2010 - 20:51h'	1.6	'Polar'	0.1	'Acamar'
'04/03/2011 - 01:30h'	1.4	'Polar'	0.1	'Adhara'
'05/08/2011 - 11:00h'	0.2	'Octantis'	0.1	'Rigil Kentaurus'
'06/11/2011 - 22:30h'	0.2	'Polar'	0.1	'Acamar'
'07/02/2012 - 02:00h'	0.4	'Polar'	0.1	'Lua'
'08/05/2012 - 14:30h'	0.8	'Polar'	0.1	'Denebola'
'09/09/2012 - 21:00h'	1.2	'Polar'	0.1	'Bellatrix'
'10/04/2013 - 03:00h'	1.3	'Polar'	0.2	'Sol'
'11/07/2013 - 10:30h'	1.4	'Polar'	0.1	'Adhara'
'12/10/2013 - 20:00h'	1.7	'Polar'	0.1	'Altair'
'13/01/2014 - 02:00h'	1.8	'Polar'	0.2	'Venus'
'14/06/2014 - 07:00h'	0.7	'Polar'	0.1	'Alphecca'
'15/12/2014 - 21:00h'	1.1	'Polar'	0.1	'Canopus'
'16/03/2015 - 09:00h'	0.3	'Octantis'	0.1	'Ankaa'
'17/08/2015 - 17:00h'	1.3	'Polar'	0.2	'Venus'
'18/11/2015 - 22:00h'	0.8	'Polar'	0.1	'Adhara'
'19/02/2016 - 05:00h'	0.7	'Octantis'	0.1	'Adhara'
'20/05/2016 - 16:00h'	1	'Polar'	0.1	'Aldebaran'
'21/09/2016 - 21:00h'	0.6	'Polar'	0.2	'Sol'
'22/04/2017 - 08:00h'	0.6	'Polar'	0.1	'Mirfak'
'23/07/2017 - 19:00h'	0.9	'Polar'	0.1	'Aldebaran'
'24/10/2017 - 23:00h'	0.9	'Octantis'	0.1	'Acamar'
'25/01/2018 - 00:00h'	1.2	'Octantis'	0.1	'Achernar'
'26/06/2018 - 12:30h'	0.8	'Octantis'	0.1	'Aldebaran'
'27/12/2018 - 22:00h'	1.1	'Octantis'	0.1	'Acamar'
'28/03/2019 - 02:00h'	0.8	'Octantis'	0.1	'Acamar'
'29/08/2019 - 07:00h'	0.4	'Octantis'	0.1	'Acrux'
'30/11/2019 - 18:00h'	0.4	'Octantis'	0.1	'Acrux'
'29/02/2020 - 00:00h'	0.3	'Polar'	0.1	'Betelgeuse'
'31/05/2020 - 12:00h'	0.6	'Venus'	0.2	'Venus'
'02/09/2020 - 23:00h'	0.5	'Octantis'	0.1	'Aldebaran'

Verifica-se a frequente incidência das estrelas polares e e S.Octantis (estrelas com as declinações mais elevadas. Como tal, realizou-se uma segunda tabela da qual se retiraram estas estrelas, obtendo assim os valores máximos de desvios de entre os restantes astros.

Na segunda tabela excluíram-se as estrelas Polar e S.Octantis:

Tabela 23 – Astros (excepto Polar e S.Octantis) com maiores desvios em cada data

Data	Maior_Desvio_GHA	Astro_GHA	Maior_Desvio_Dec	Astro_Dec
'01/01/2010 - 12:00h'	0.3	'Sol'	0.1	'Achernar'
'02/06/2010 - 18:00h'	0.2	'Sol'	0.1	'Alnilam'
'03/12/2010 - 20:51h'	0.3	'Venus'	0.1	'Acamar'
'04/03/2011 - 01:30h'	0.1	'Alioth'	0.1	'Adhara'
'05/08/2011 - 11:00h'	0.1	'Acrux'	0.1	'Rigil Kentaurus'
'06/11/2011 - 22:30h'	0.1	'Adhara'	0.1	'Acamar'
'07/02/2012 - 02:00h'	0.3	'Sol'	0.1	'Lua'
'08/05/2012 - 14:30h'	0.3	'Sol'	0.1	'Denebola'
'09/09/2012 - 21:00h'	0.2	'Sol'	0.1	'Bellatrix'
'10/04/2013 - 03:00h'	0.4	'Sol'	0.2	'Sol'
'11/07/2013 - 10:30h'	0.3	'Sol'	0.1	'Adhara'
'12/10/2013 - 20:00h'	0.1	'Achernar'	0.1	'Altair'
'13/01/2014 - 02:00h'	0.6	'Venus'	0.2	'Venus'
'14/06/2014 - 07:00h'	0.3	'Sol'	0.1	'Alphecca'
'15/12/2014 - 21:00h'	0.4	'Sol'	0.1	'Canopus'
'16/03/2015 - 09:00h'	0.3	'Sol'	0.1	'Ankaa'
'17/08/2015 - 17:00h'	0.8	'Venus'	0.2	'Venus'
'18/11/2015 - 22:00h'	0.2	'Lua'	0.1	'Adhara'
'19/02/2016 - 05:00h'	0.1	'Atria'	0.1	'Adhara'
'20/05/2016 - 16:00h'	0.2	'Sol'	0.1	'Aldebaran'
'21/09/2016 - 21:00h'	0.4	'Sol'	0.2	'Sol'
'22/04/2017 - 08:00h'	0.4	'Sol'	0.1	'Mirfak'
'23/07/2017 - 19:00h'	0.3	'Sol'	0.1	'Aldebaran'
'24/10/2017 - 23:00h'	0.1	'Alphecca'	0.1	'Acamar'
'25/01/2018 - 00:00h'	0.1	'Avior'	0.1	'Achernar'
'26/06/2018 - 12:30h'	0.2	'Miaplacidus'	0.1	'Aldebaran'
'27/12/2018 - 22:00h'	0.1	'Avior'	0.1	'Acamar'
'28/03/2019 - 02:00h'	0.1	'Adhara'	0.1	'Acamar'
'29/08/2019 - 07:00h'	0.3	'Sol'	0.1	'Acrux'
'30/11/2019 - 18:00h'	0.3	'Sol'	0.1	'Acrux'
'29/02/2020 - 00:00h'	0.2	'Sol'	0.1	'Betelgeuse'
'31/05/2020 - 12:00h'	0.6	'Venus'	0.2	'Venus'
'02/09/2020 - 23:00h'	0.2	'Sol'	0.1	'Aldebaran'

Esta foi elaborada usando o algoritmo 6 para cálculo das coordenadas solares.

É, agora, verificável que, usando o algoritmo 6, embora não sejam desvios significativos, o Sol é um dos astros que mais vezes teve o maior desvio da data.

A mesma tabela usando o algoritmo 7 para cálculo das coordenadas do Sol:

Tabela 24 – Astros (excepto Polar e S.Octantis) com maiores desvios em cada data usando o algoritmo 7 para o Sol

Data	Maior_Desvio_GHA	Astro_GHA	Maior_Desvio_Dec	Astro_Dec
'01/01/2010 - 12:00h'	0.1	'Achernar'	0.1	'Achernar'
'02/06/2010 - 18:00h'	0.1	'Acrux'	0.1	'Alnilam'
'03/12/2010 - 20:51h'	0.3	'Venus'	0.1	'Acamar'
'04/03/2011 - 01:30h'	0.1	'Alioth'	0.1	'Adhara'
'05/08/2011 - 11:00h'	0.1	'Acrux'	0.1	'Rigel Kentaurus'
'06/11/2011 - 22:30h'	0.1	'Adhara'	0.1	'Acamar'
'07/02/2012 - 02:00h'	0.2	'Lua'	0.1	'Lua'
'08/05/2012 - 14:30h'	0.1	'Avior'	0.1	'Denebola'
'09/09/2012 - 21:00h'	0.1	'Achernar'	0.1	'Bellatrix'
'10/04/2013 - 03:00h'	0.1	'Acrux'	0.1	'Eltanin'
'11/07/2013 - 10:30h'	0.1	'Atria'	0.1	'Adhara'
'12/10/2013 - 20:00h'	0.1	'Achernar'	0.1	'Altair'
'13/01/2014 - 02:00h'	0.6	'Venus'	0.2	'Venus'
'14/06/2014 - 07:00h'	0.1	'Acamar'	0.1	'Alphecca'
'15/12/2014 - 21:00h'	0.1	'Atria'	0.1	'Canopus'
'16/03/2015 - 09:00h'	0.2	'Lua'	0.1	'Ankaa'
'17/08/2015 - 17:00h'	0.8	'Venus'	0.2	'Venus'
'18/11/2015 - 22:00h'	0.2	'Lua'	0.1	'Adhara'
'19/02/2016 - 05:00h'	0.1	'Atria'	0.1	'Adhara'
'20/05/2016 - 16:00h'	0.1	'Alioth'	0.1	'Aldebaran'
'21/09/2016 - 21:00h'	0.1	'Alphecca'	0.1	'Achernar'
'22/04/2017 - 08:00h'	0.2	'Venus'	0.1	'Mirfak'
'23/07/2017 - 19:00h'	0.1	'Achernar'	0.1	'Aldebaran'
'24/10/2017 - 23:00h'	0.1	'Alphecca'	0.1	'Acamar'
'25/01/2018 - 00:00h'	0.1	'Avior'	0.1	'Achernar'
'26/06/2018 - 12:30h'	0.2	'Miaplacidus'	0.1	'Aldebaran'
'27/12/2018 - 22:00h'	0.1	'Avior'	0.1	'Acamar'
'28/03/2019 - 02:00h'	0.1	'Adhara'	0.1	'Acamar'
'29/08/2019 - 07:00h'	0.1	'Achernar'	0.1	'Acrux'
'30/11/2019 - 18:00h'	0.1	'Achernar'	0.1	'Acrux'
'29/02/2020 - 00:00h'	0.1	'Alkaid'	0.1	'Betelgeuse'
'31/05/2020 - 12:00h'	0.6	'Venus'	0.2	'Venus'
'02/09/2020 - 23:00h'	0.1	'Achernar'	0.1	'Aldebaran'

Findo este estudo, conclui-se que excluindo as estrelas Polar e S. Octantis, os maiores erros encontrados análise das 33 datas foram os de vénus planeta para o qual, como referido, os resultados do AstroNavTH2020 acabam por coincidir com os dados do nav32.

É observável que, na maioria dos casos, quando houve discórdia entre os programas AstroNavTH2020 e NavPac, o nav32: ou coincidiu com o AstroNavTH2020 (caso de Vénus), ou afastou-se significativamente dos dois restantes. (caso das estrelas Polar e S. Octantis).

## Conclusões

O trabalho de desenvolvimento duma ferramenta para cálculo de coordenadas celestes exige conhecimento teórico da área de astronomia, bem como uma constante visão sobre o objetivo final, para que esta seja de facto útil ao propósito a que se propõe. De facto, há que conhecer também o trabalho exigido na prática da navegação astronómica para que se consiga criar algo realmente útil de ponto de vista operacional. É desta prespetiva que se tentou a maximização da qualidade do *software* desenvolvido, pois, de facto, para efeitos de navegação astronómica, o rigor de cálculos para além do necessário deixa de ser o principal foco, em detrimento da necessidade de primar pela utilidade e facilidade. Tendo esta ideia em consideração, o rigor de cálculo necessário foi de facto uma prioridade que, quando garantida, abriu a possibilidade para um processo de melhoria constante do programa para o tornar mais eficiente, simples, de fácil utilização.

De facto, em relação aos programas de referência, este *software* não fica atrás em termos de exactidão. Os algoritmos utilizados garantem, na sua generalidade, uma exactidão mais que suficiente para o efeito da navegação astronómica.

As maiores limitações foram encontradas ao nível da programação, no que diz respeito à conciliação do que está descrito nos algoritmos apresentados com o que se introduz de facto no Matlab. Esta dificuldade foi sentida principalmente quando se programaram algoritmos com aplicabilidade para todos os astros, como é o caso dos nascimentos e ocasos, por exemplo. De facto, quando o algoritmo é o mesmo, e para um primeiro astro obtemos resultados corretos, e para um segundo obtemos resultados completamente errados, leva-nos a crer que a fonte do problema tem que ser os algoritmos próprios do cálculo das coordenadas do segundo astro. Parece lógico, e é precisamente por isso, que se considerou esta como a maior limitação encontrada. Quando a nossa intuição está errada. No caso dos nascimentos e ocasos, era mesmo este algoritmo (dos nascimentos e ocasos) que estava, de certa forma, mal traduzido em Matlab. Isto pois havia um caso especial que poderia ter acontecido com qualquer astro, mas é mais frequente com a Lua devido às maiores variações diárias da sua ascensão reta.

Conforme descrito em (Meeus, 1992), ao programar algoritmos astronómicos em qualquer linguagem, deve ter-se em atenção a continuidade da contagem dos ângulos. O problema descrito acima acontecia pois a função de cálculo de nascimentos e ocasos, que usa o

algoritmo 3 de interpolação, estava a registar diferenças enormes entre valores da ascensão reta da Lua que não deveriam variar assim tanto com o intervalo de um dia. Foi apenas quando se tomou consciência desta estranha variação que se percebeu onde o programa estava a errar. Para que se perceba será dado um exemplo concreto:

Supondo que num dia a Ascensão reta da Lua é de  $358^\circ$  e no dia seguinte é de  $7^\circ$ . A diferença que o programa deve considerar é de  $9^\circ$  e nunca de  $351^\circ$ . O erro não acontecia com o Sol simplesmente porque o algoritmo, por sorte, ainda não tinha sido testado num dia que a ascensão reta reiniciasse a sua contagem.

Na introdução da presente dissertação, mencionou-se o facto de que a navegação astronómica é frequentemente olhada no presente como uma mera redundância. Uma arte em desuso.

No realizar do trabalho, no entanto, percebeu-se que a automatização do processo da navegação astronómica está longe de ser algo obsoleto. Há de facto ainda pesquisa recente sobre o tema. São criados algoritmos de inteligência artificial aplicada à navegação astronómica e isso leva-nos a crer que de facto poderá haver espaço para investimentos futuros nesta área do saber.

Algumas limitações presentes desta arte são por exemplo a dificuldade da sua realização durante a noite devido à possível incapacidade de reconhecimento da linha do horizonte.

Sabemos que já existe tecnologia, como é o caso do sextante de bolha inventado pelo Almirante português Gago Coutinho, com esta finalidade, mas não será esta limitação contornável perante o paradeiro da tecnologia actual de processamento de imagem?

Não será possível a criação de um sistema de navegação em *real-time* independente de terceiros, com custos reduzidos e com vantagem táctica pela não existência de emissões?

De que forma o aumento do número de linhas de posição utilizadas pode diminuir a necessidade de rigor na medição de uma altura de um dado astro?

Estas são algumas perguntas que abrem possibilidades de investigação futura, nesta área que, como aqui evidenciado, de obsoleto, tem muito pouco.

## Referências Bibliográficas

- Brown, D. R. (2002). Babylonian Observations. *Highlights of Astronomy*.  
<https://doi.org/10.1017/s1539299600013630>
- Evans, J., & Jones, A. (2000). The History and Practice of Ancient Astronomy. *American Journal of Physics*.  
<https://doi.org/10.1119/1.19412>
- Fraknoi, A., Morrison, D., & Wolff, S. C. (2016). Astronomy. In *University Physics* (Vol. 1). Retrieved from <https://openstax.org/details/books/astronomy#details>
- Gameiro, E. da S. (1964). *Astronomia Náutica*.
- Hofmann-Wellenhof, B., Legat, K., & Wieser, M. (2003). *Navigation - Principles of Positioning and Guidance*.
- Meeus, J. (1992). Astronomical algorithms. *Choice Reviews Online*.  
<https://doi.org/10.5860/choice.30-0269>
- Reis, A. E. dos. (1997). *Medir Estrelas*.
- Rudgley, R. (1999). The lost civilizations of the Stone Age. *Choice Reviews Online*, 36(10), 36-5764-36-5764. <https://doi.org/10.5860/choice.36-5764>
- Stephenson, F. R., & Morrison, L. V. (1995). Long-term fluctuations in the Earth's rotation: 700 BC to AD 1990. *Philosophical Transactions of the Royal Society of London. Series A: Physical and Engineering Sciences*, 351(1695), 165-202.  
<https://doi.org/10.1098/rsta.1995.0028>
- Umland, H. (2001). *A Short Guide to Celestial Navigation*.
- United-Kingdom-Hydrographic-Office. (2010). Nautical Almanac Technical Note. *Observatory*, 44(74).
- Vincent, F. (1976). Positional Astronomy. *Physics Today*, 29(1), 73-75.  
<https://doi.org/10.1063/1.3023269>



## Apêndice A – Código do Programa

No presente apêndice apresenta-se o código do programa desenvolvido. Optou-se por dispor em primeiro lugar o código do Menu Inicial e das janelas que este abre que dizem respeito às duas funcionalidades: Almanaque e Planeamento de observações. De seguida apresentam-se as funções principais de cálculo, as funções auxiliares, as funções de construção dos gráficos e, finalmente, as funções de comparação com o NavPac. No final desse último conjunto de funções, apresentam-se os ficheiros de texto que contêm os dados copiados do programa NavPac em formato .txt e processados no Matlab com recurso à função `navpac`. É importante destacar que para correr as rotinas de cálculo dos desvios, estes ficheiros têm de estar presentes na “*current folder*” no Matlab.

### Menu Inicial

```
function varargout = Menu_Inicial(varargin)

gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                  'gui_Singleton',   gui_Singleton, ...
                  'gui_OpeningFcn', @Menu_Inicial_OpeningFcn, ...
                  'gui_OutputFcn',  @Menu_Inicial_OutputFcn, ...
                  'gui_LayoutFcn',  [], ...
                  'gui_Callback',    []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end

function Menu_Inicial_OpeningFcn(hObject, eventdata, handles, varargin)

handles.output = hObject;
%Simbolo Logo
axes(handles.axes1)
matlabImage = imread('iconnovo.png');
image(matlabImage)
axis off
axis image

%Simbolo Curso
axes(handles.axes5)
```

```

matlabImage = imread('simbcurso.bmp');
image(matlabImage)
axis off
axis image

%Simbolo Escola
axes(handles.axes4)
matlabImage = imread('simbesc.bmp');
image(matlabImage)
axis off
axis image
guidata(hObject, handles);

function varargout = Menu_Inicial_OutputFcn(hObject, eventdata, handles)

varargout{1} = handles.output;

function pushbutton1_Callback(hObject, eventdata, handles)
Astrosteste2

function pushbutton2_Callback(hObject, eventdata, handles)
Ceu_Visivel
function pushbutton3_Callback(hObject, eventdata, handles)
message='Ferramenta temporariamente indisponível';

myicon = imread('iconnovo.png');
h=msgbox(message, 'Sobre', 'custom', myicon);
function pushbutton4_Callback(hObject, eventdata, handles)
message='Neste campo será disponibilizado, quando possível, um manual de
utilização do software';

myicon = imread('iconnovo.png');
h=msgbox(message, 'Sobre', 'custom', myicon);

function pushbutton5_Callback(hObject, eventdata, handles)
message=['Este programa foi desenvolvido no ano letivo 2019/2020 no âmbito
'...
'das dissertações de mestrado dos ASPOF Fernandes Tlemçani e ASPOF Rebelo
Horta '...
'sob orientação de CMG Costa Canas, a quem se agradece pela '...
'disponibilidade e auxílio, prestados ao longo de todo o projecto, que
foram '...
'fundamentais para a sua realização.'];

myicon = imread('iconnovo.png');
h=msgbox(message, 'Sobre', 'custom', myicon);

```

## Funcionalidade Almanaque

```
function varargout = Astrosteste2(varargin)
```

```

gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                  'gui_Singleton',  gui_Singleton, ...
                  'gui_OpeningFcn', @Astroteste2_OpeningFcn, ...
                  'gui_OutputFcn',  @Astroteste2_OutputFcn, ...
                  'gui_LayoutFcn',  [], ...
                  'gui_Callback',    []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargin
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end

% --- Executes just before Astroteste2 is made visible.
function Astroteste2_OpeningFcn(hObject, eventdata, handles, varargin)

handles.output = hObject;
%Simbolo Logo
axes(handles.axes3)
matlabImage = imread('iconnovo.png');
image(matlabImage)
axis off
axis image

%Simbolo Curso
axes(handles.axes4)
matlabImage = imread('simbcurso.bmp');
image(matlabImage)
axis off
axis image

%Simbolo Escola
axes(handles.axes5)
matlabImage = imread('simbesc.bmp');
image(matlabImage)
axis off
axis image

%Esfera grande
axes(handles.axes7)
% har9
%
grid off
set(gca, 'visible', 'off')
set(gca, 'xtick', [])
set(gca, 'ytick', [])
set(gca, 'ztick', [])
rotate3d(handles.axes7)
hManager = uigetmodemanager();

```

```

hManager.CurrentMode.ModeStateData.textState = 0;

view(45,12)
%Grafico da Lua
axes(handles.axes9)
grid off
set(gca,'xtick',[])
set(gca,'ytick',[])
% Update handles structure
guidata(hObject, handles);

% --- Outputs from this function are returned to the command line.
function varargout = Astrosteste2_OutputFcn(hObject, eventdata, handles)

varargout{1} = handles.output;

% --- Executes on button press in pushbutton1.
function pushbutton1_Callback(hObject, eventdata, handles)

%=====OBTENÇÃO DE DADOS INTRODUIZIDOS=====

%Transformar as variáveis de entrada formato string em numeros acedendo a
%partir dos handles

dia_1=get(handles.edit2,'string');
dia_1=str2double(dia_1);

mes_1=get(handles.edit3,'string');
mes_1=str2double(mes_1);

ano_1=get(handles.edit4,'string');
ano_1=str2double(ano_1);

hora_hora=get(handles.edit5,'string');%horas inteiras introduzidas
hora_hora=str2double(hora_hora);

minuto_1=get(handles.edit8,'string');
minuto_1=str2double(minuto_1);           %minutos inteiros introduzidos
minuto_h=minuto_1/60;                   %obter os minutos em horas

segundo_1=get(handles.edit9,'string');
segundo_1=str2double(segundo_1);       %segundos inteiros introduzidos
segundo_h=segundo_1/3600;              %obter os segundos em horas

hora_1=hora_hora+minuto_h+segundo_h;

%Obter Latitudes
latitudegraus=get(handles.latitudegraus,'string');
latitudegraus=str2double(latitudegraus);

latitudeminutos=get(handles.latitudeminutos,'string');
latitudeminutos=str2double(latitudeminutos);

latitudesinal=get(handles.latsinal,'string');

```

```

j=strfind(latitudesinal,'S');

latitudedecimal=latitudegraus+latitudeminutos/60;    %definir a latitude em
graus em decimal
if j==1 ,latitudedecimal=-latitudedecimal;
end

%Obter Longitudes
longitudegraus=get(handles.longitudegraus,'string');
longitudegraus=str2double(longitudegraus);
longitudeminutos=get(handles.longitudeminutos,'string');
longitudeminutos=str2double(longitudeminutos);

longsinal=get(handles.longsinal,'string');
f=strfind(longsinal,'E');

longitudedecimal=longitudegraus+longitudeminutos/60;
if f==1 ,longitudedecimal=-longitudedecimal;
end

%=====CÁLCULO APLICADO AO ASTRO SELECCIONADO=====

%definir o astro em questão aplicando a função efemerides(=====ALTERAÇÃO:
APLICANDO A FUNÇÃO SOL=====) para o Sol e
%aparente para as restantes. O valor a corresponde ao numero do astro pela
%ordem em que aparece na lista. quando puseres lua e planetas poe nos a
%frente na lista

items = get(handles.popupmenu2,'String');
a = get(handles.popupmenu2,'Value');
set(handles.popupmenu2,'String',items);

switch a
    case 1%caso o astro escolhido seja o sol
        coordenadas=Sol(ano_1,mes_1,dia_1,hora_hora,minuto_1,segundo_1);
        %Buscar o nascer e ocaso do Sol

nasceroocaso=nascimentooeocaso('Sol',latitudedecimal,longitudedecimal,dia_1,mes_
1,ano_1);
        %Eq Tempo
        eq=coordenadas(5);
        eq_min=floor(eq);
        eq_seg=(eq-eq_min)*60;
        eq_seg=round(eq_seg);
        eq_seg=sprintf('%02d',eq_seg);
        eq_min=sprintf( '%02d', eq_min );

        %SD
        sd=coordenadas(4)*60;
        sd=round(sd,2);
        paralaxe=0.15;
        mag=-26.74;

    case 2        %LUA

```

```

coordenadas=Lua(ano_1,mes_1,dia_1,hora_hora,minuto_1,segundo_1);
coordenadas_seg=Lua(ano_1,mes_1,dia_1+1,hora_hora,minuto_1,segundo_1);

nascero caso=nascimento e ocaso('Lua',latitude decimal, longitude decimal,dia_1,mes_1,ano_1);
    %Paralaxe e SD
    paralaxe=coordenadas(3)*60;
    sd=paralaxe*0.2725;
    paralaxe=sprintf('%04.1f\n',paralaxe);
    sd=round(sd,2);

    frac_lua=coordenadas(4);
    fases_graf

%           %Magnitude
%           mag=-7.0;
case 3      %VENUS
    coordenadas=Venus(ano_1,mes_1,dia_1,hora_hora,minuto_1,segundo_1);

nascero caso=nascimento e ocaso('Venus',latitude decimal, longitude decimal,dia_1,mes_1,ano_1);
    mag=-4.9;
    paralaxe=coordenadas(4);
    paralaxe=sprintf('%03.1f\n',paralaxe);
    sd=coordenadas(3);
case 4      %MARTE
    coordenadas=Marte(ano_1,mes_1,dia_1,hora_hora,minuto_1,segundo_1);

nascero caso=nascimento e ocaso('Marte',latitude decimal, longitude decimal,dia_1,mes_1,ano_1);
    mag=0.3;
    paralaxe=coordenadas(4);
    paralaxe=sprintf('%03.1f\n',paralaxe);
    sd=coordenadas(3);
case 5      %JUPITER
    coordenadas=Jupiter(ano_1,mes_1,dia_1,hora_hora,minuto_1,segundo_1);

nascero caso=nascimento e ocaso('Jupiter',latitude decimal, longitude decimal,dia_1,mes_1,ano_1);
    mag=2.9;
case 6      %SATURNO
    coordenadas=Saturno(ano_1,mes_1,dia_1,hora_hora,minuto_1,segundo_1);

nascero caso=nascimento e ocaso('Saturno',latitude decimal, longitude decimal,dia_1,mes_1,ano_1);
    mag=0.5;
    otherwise
        coordenadas=aparente(a-6,dia_1,mes_1,ano_1,hora_1);
        nascero caso=nascimento e ocaso(a-6,latitude decimal, longitude decimal,dia_1,mes_1,ano_1);
        mag=magnitude(a-6);
end

%===== DISPLAYS =====%

```

```

    GHA=wrapTo360 (TSG (ano_1,mes_1,dia_1,hora_hora,minuto_1,segundo_1)-
coordenadas (1));
    AR=coordenadas (1);
    valorgha=sexanav (GHA);%conversao do GHA para sexagesimal
    GHA_graus=valorgha (1);%graus
    GHA_graus=sprintf ('%03d',GHA_graus);
    GHA_min=valorgha (2);%minutos
    GHA_min=sprintf ('%04.1f\n',GHA_min);

    set (handles.text6,'String',GHA_graus)%exibir graus
    set (handles.text21,'String',GHA_min)%exibir minutos

    %Display da Declinação
    declinacao=coordenadas (2);
    valordec=sexanav (declinacao);

    dec_graus=abs (valordec (1));%graus
    dec_graus=sprintf ('%02d',dec_graus);
    dec_min=abs (valordec (2));%minutos
    dec_min=sprintf ('%04.1f\n',dec_min);

    set (handles.text23,'String',dec_graus)%exibir graus
    set (handles.text24,'String',dec_min)%exibir minutos

    if declinacao<0%caso a declinação seja Sul
        set (handles.text18,'String','S'); %inserir Sul

    else%caso a declinação seja Norte
        set (handles.text18,'String','N')%inserir Norte
    end

    %display do angulo sideral

    angulosideral=360-AR;
    valoras=sexanav (angulosideral);
    as_graus=abs (valoras (1));%graus
    as_graus=sprintf ('%03d',as_graus);
    as_min=abs (valoras (2));%minutos
    as_min=sprintf ('%04.1f\n',as_min);

    set (handles.text26,'String',as_graus)%exibir graus
    set (handles.text27,'String',as_min)

    %Display SD
    if a==1||a==2 %||a==3||a==4

    sd=sprintf ('%1.1f',sd);
    set (handles.text14,'String',sd)%Display SD
    else
        set (handles.text14,'String','-')%Display SD
    end

    %display HP
    if a==2||a==1||a==3||a==4
        set (handles.text188,'String',paralaxe)
    else

```

```

        set(handles.text188,'String','-')
    end

    %Display da equação de tempo
    if a==1
        set(handles.text31,'String',eq_min);
        set(handles.text40,'String',eq_seg);
    else
        set(handles.text31,'String','-');
        set(handles.text40,'String','-');
    end
    %Display do Vernal Horário Greenwich (Calculado a partir da fórmula
    %simplificada do Meeus pela função TSG)

    vernalhg=TSG(ano_1,mes_1,dia_1,hora_hora,minuto_1,segundo_1);
    valor=sexnav(veralhg);
    vernalhg_graus=valor(1);
    vernalhg_graus=sprintf('%03d',veralhg_graus);
    vernalhg_min=valor(2);
    vernalhg_min=sprintf('%04.1f\n',veralhg_min);

    set(handles.text37,'String',veralhg_graus)
    set(handles.text38,'String',veralhg_min)

    %NASCER E OCASO DO SOL

    %Display do nascer
    nascer=hours(nascerocaso(1)) ;
    nascer.Format='hh:mm:ss';
    nascer=round(nascer,'minutes');
    nascer=datevec(nascer);
    nascerhora=nascer(4);
    nascerhora=sprintf( '%02d', nascerhora );
    nascermin=nascer(5);
    nascermin=sprintf( '%02d', nascermin );
    set(handles.text57,'String',nascerhora);
    set(handles.text58,'String',nascermin);

    %Display do ocaso do Sol
    ocaso=hours(nascerocaso(2));
    ocaso.Format='hh:mm:ss';
    ocaso=round(ocaso,'minutes');
    ocaso=datevec(ocaso);
    ocasohora=ocaso(4);
    ocasohora=sprintf( '%02d', ocasohora );
    ocasomin=ocaso(5);
    ocasomin=sprintf( '%02d', ocasomin );

    set(handles.text61,'String',ocasohora);
    set(handles.text62,'String',ocasomin);

    %Display da Hora da Meridiana

    meridianaemhoras=nascerocaso(3);
    meridiana=sexnav(meridianaemhoras);

```

```

meridiana_hora=meridiana(1);
meridiana_hora=sprintf( '%02d', meridiana_hora );
meridiana_min=round(meridiana(2));
meridiana_min=sprintf( '%02d', meridiana_min );

set(handles.text48,'String',meridiana_hora);
set(handles.text50,'String',meridiana_min);

%Display dos crepusculos
crep =
crepusculos(latitudedecimal,longitudedecimal,dia_1,mes_1,ano_1);

%Matutino Civil
matutinocivilhora=crep(1,1);
matutinocivilhora=sprintf( '%02d', matutinocivilhora );
matutinocivilmin=crep(1,2);
matutinocivilmin=sprintf( '%02d', matutinocivilmin );

set(handles.matutinocivilhora,'String',matutinocivilhora);
set(handles.matutinocivilmin,'String',matutinocivilmin);

%Matutino Nautico
matutinonauthora=crep(2,1);
matutinonauthora=sprintf( '%02d', matutinonauthora );
matutinonautmin=crep(2,2);
matutinonautmin=sprintf( '%02d', matutinonautmin );

set(handles.matutinonauthora,'String',matutinonauthora);
set(handles.matutinonautmin,'String',matutinonautmin);

%Vespertino Civil
vespcivilhora=crep(3,1);
vespcivilhora=sprintf( '%02d', vespcivilhora );
vespcivilmin=crep(3,2);
vespcivilmin=sprintf( '%02d', vespcivilmin );

set(handles.vespcivilhora,'String',vespcivilhora);
set(handles.vespcivilmin,'String',vespcivilmin);

%Vespertino Nautico
vespnauthora=crep(4,1);
vespnauthora=sprintf( '%02d', vespnauthora );
vespnautmin=crep(4,2);
vespnautmin=sprintf( '%02d', vespnautmin );

set(handles.vespnauthora,'String',vespnauthora);
set(handles.vespnautmin,'String',vespnautmin);

%Magnitude
if a==2
    set(handles.text191,'String','-');
else
mag=sprintf( '%0.1f', mag );
set(handles.text191,'String',mag);
end

```

```

    %Percentagem de iluminação da Lua
    if a==2
    perc=frac_lua*100;
    perc=round(perc);
    perc=sprintf( '%2d%%', perc );
    set(handles.text193,'String',perc);
    end

function edit2_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit3_Callback(hObject, eventdata, handles)

function edit3_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit4_Callback(hObject, eventdata, handles)

function edit4_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit5_Callback(hObject, eventdata, handles)

function edit5_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit6_Callback(hObject, eventdata, handles)

function edit6_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit7_Callback(hObject, eventdata, handles)

function edit7_CreateFcn(hObject, eventdata, handles)

```

```
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function popupmenu2_Callback(hObject, eventdata, handles)

function popupmenu2_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit8_Callback(hObject, eventdata, handles)

function edit8_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit9_Callback(hObject, eventdata, handles)

function edit9_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function latitudegraus_Callback(hObject, eventdata, handles)

function latitudegraus_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function latitudeminutos_Callback(hObject, eventdata, handles)

function latitudeminutos_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit2_Callback(hObject, eventdata, handles)

function longitudegraus_Callback(hObject, eventdata, handles)
```

```

function longitudegraus_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function longitudeminutos_Callback(hObject, eventdata, handles)

function longitudeminutos_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

%=====BOTÃO DE RESET=====

function pushbutton3_Callback(hObject, eventdata, handles)

set(handles.text6,'String',' ')%exibir graus
set(handles.text21,'String',' ')%exibir minutos
set(handles.text23,'String',' ')%exibir graus
set(handles.text24,'String',' ')
set(handles.text26,'String',' ')%exibir graus
set(handles.text27,'String',' ')
set(handles.text14,'String',' ');
set(handles.text31,'String',' ');
set(handles.text40,'String',' ');
set(handles.text37,'String',' ');
set(handles.text38,'String',' ');
set(handles.text48,'String',' ');
set(handles.text50,'String',' ');
set(handles.text57,'String',' ');
set(handles.text58,'String',' ');
set(handles.text61,'String',' ');
set(handles.text62,'String',' ');
set(handles.matutinocivilhora,'String',' ');
set(handles.matutinocivilmin,'String',' ');
set(handles.vespcivilhora,'String',' ');
set(handles.matutinonauthora,'String',' ');
set(handles.matutinonautmin,'String',' ');
set(handles.vespcivilmin,'String',' ');
set(handles.vespnauthora,'String',' ');
set(handles.vespnautmin,'String',' ');
set(handles.text18,'String',' ');
set(handles.text188,'String',' ');
set(handles.text193,'String',' ');

axes(handles.axes7)
cla
axes(handles.axes9)
cla

```

```

function axes3_CreateFcn(hObject, eventdata, handles)

function axes4_CreateFcn(hObject, eventdata, handles)

function axes5_CreateFcn(hObject, eventdata, handles)

function edit15_Callback(hObject, eventdata, handles)

function edit15_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function longsinal_Callback(hObject, eventdata, handles)
longsinal=get(handles.longsinal,'string');

k=strfind(longsinal,'E');
if k==1
    set(handles.longsinal,'String','W');
else
set(handles.longsinal,'String','E');
end

function longsinal_CreateFcn(hObject, eventdata, handles)

%
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function latsinal_Callback(hObject, eventdata, handles)

latitudesinal=get(handles.latsinal,'string');

j=strfind(latitudesinal,'S');
if j==1
    set(handles.latsinal,'String','N');
else
set(handles.latsinal,'String','S');
end

function latsinal_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end
function pushbutton1_CreateFcn(hObject, eventdata, handles)

function Untitled_1_Callback(hObject, eventdata, handles)

function pushbutton6_Callback(hObject, eventdata, handles)

```

```
function pushbutton7_Callback(hObject, eventdata, handles)
```

```
% -----
```

## Funcionalidade Planejamento de Observações

```
function varargout = Ceu_Visivel(varargin)

gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                  'gui_Singleton',  gui_Singleton, ...
                  'gui_OpeningFcn', @Ceu_Visivel_OpeningFcn, ...
                  'gui_OutputFcn',  @Ceu_Visivel_OutputFcn, ...
                  'gui_LayoutFcn',  [], ...
                  'gui_Callback',   []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargin
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end

function Ceu_Visivel_OpeningFcn(hObject, eventdata, handles, varargin)
set(handles.uitable6, 'Data', cell(65,4));
set(handles.uitable7, 'Data', cell(2,2));
handles.output = hObject;

axes(handles.axes2)
grafico

function varargout = Ceu_Visivel_OutputFcn(hObject, eventdata, handles)
handles.output = hObject;
varargout{1} = handles.output;

%=====BOTÃO HORAS FAVORAVEIS=====%
function pushbutton22_Callback(hObject, eventdata, handles)
sacar_var;

horas_fav=favoraveis(latitudedecimal,longitudedecimal,dia,mes,ano);
horas_fav_h=num2str(horas_fav(:,1),'%02g');
horas_fav_m=num2str(horas_fav(:,2),'%02g');

horas_fav_h=cellstr(horas_fav_h);
horas_fav_m=cellstr(horas_fav_m);

if get(handles.radiobutton1, 'Value') ==1
```

```

yourdata =[horas_fav_h(1) horas_fav_m(1)
           horas_fav_h(2) horas_fav_m(2)];
elseif get(handles.radiobutton2,'Value') ==1
    yourdata =[horas_fav_h(3) horas_fav_m(3)
              horas_fav_h(4) horas_fav_m(4)];
end

set(handles.uitable7, 'Data', cell(size(get(handles.uitable7,'Data'))));

set(handles.uitable7,'Data',yourdata);

%=====BOTÃO CEU VISIVEL=====

function pushbutton13_Callback(hObject, eventdata, handles)
axes (handles.axes2)
cla
grafico
sacar_var;

names={'Sol';'Lua';'Venus';'Marte';'Jupiter';'Saturno';'Acomar';'Achernar';'Ac
rux';'Adhara';'Aldebaran';'Alioth';'Alkaid';'Al
Na'ir';'Alnilam';'Alphard';'Alphecca';'Alpheratz';'Altair';'Ankaa';'Antares';'
Arcturus';'Atria';'Arior';'Bellatrix';'Betelgeuse';'Canopus';'Capella';'Deneb'
;'Denebola';'Diphda';'Dubhe';'Elnath';'Eltanin';'Enif';'Fomalhaut';'Gacrux';'G
ienah';'Hadar';'Hamal';'Kaus
Australis';'Kochab';'Markab';'Menkar';'Menkent';'Miaplacidus';'Mirfak';'Nunki'
;'Peacock';'Pollux';'Procyon';'Rasalhague';'Regulus';'Rigel';'Rigil
Kentaurus';'Sabik';'Schedar';'Shaula';'Sirius';'Spica';'Suhail';'Vega';'Zubene
lgenubi';'Polaris';'Octantis'};

C=tabela(ano,mes,dia,hora,minuto,segundo,latitudedecimal,longitudedecimal);

alturas=(C(:,1));
azimutes=(C(:,2));

%Na tabela aparecerão apenas os astros visiveis
controlo=zeros(65,1);
for k=1:1:65
    if C(k,1) > 0
        controlo(k,1)=1;
    end
end
names(controlo == 0) = [];
alturas(controlo == 0) = [];
azimutes(controlo == 0) = [];

checkbox1=cell([numel(alturas) 1]);
for k=1:1:numel(alturas)
checkbox1{k,1}=true;
end

%passagem dos elementos da tabela para cell
% alturas = angle2str(alturas,'none','degrees2dm',-1);

```

```

alturas = angle2str(alturas, 'none', 'degrees', 0);

alturas=cellstr(alturas);
azimutes=form_az(azimutes);
azimutes=cellstr(azimutes);

yourdata =[checkboxx1 names azimutes alturas ];
cnames = {'', 'Astro', 'Az', 'h'};
columneditable = [true false false false] ;
set(handles.uitable6, 'Data', cell(size(get(handles.uitable6, 'Data'))));

set(handles.uitable6, 'Data', yourdata, 'ColumnEditable',
columneditable, 'ColumnName', cnames);

alturas=(C(:,1));
azimutes=(C(:,2));

% yourdata = yourdata(sortOrder, :);
% set(handles.uitable6, 'Data', yourdata, 'ColumnEditable',
columneditable, 'ColumnName', cnames);
def_graf

controlo=zeros(65,1);
for k=1:1:65
    if C(k,1) > 0
        controlo(k,1)=1;
    end
end

alturasol=alturas(1);
azimutesol=azimutes(1);

alturalua=alturas(2);
azimutelua=azimutes(2);

alturavenus=alturas(3);
azimutevenus=azimutes(3);

alturamarte=alturas(4);
azimutemarte=azimutes(4);

alturajupiter=alturas(5);
azimutejupiter=azimutes(5);

alturasaturno=alturas(6);
azimutesaturno=azimutes(6);

%SOL
if controlo(1,1)==1
axes(handles.axes2)
sol_grafico(alturasol, azimutesol, theta, phi)
end
%LUA

```

```

if controlo(2,1)==1
axes(handles.axes2)
lua_grafico(alturalua,azimutelua,theta,phi)
end
%VENUS
if controlo(3,1)==1
axes(handles.axes2)
venus_grafico(alturavenus,azimutevenus,theta,phi)
end
% MARTE
if controlo(4,1)==1
axes(handles.axes2)
marte_grafico(alturamarate,azimutemarte,theta,phi)
end
%JUPITER
if controlo(5,1)==1
axes(handles.axes2)
jupiter_grafico(alturajupiter,azimutejupiter,theta,phi)
end
%SATURNO
if controlo(6,1)==1
axes(handles.axes2)
saturno_grafico(alturasaturno,azimutesaturno,theta,phi)
end

axes(handles.axes2)
for a=7:1:65
if controlo(a,1)==1

    estrela_grafico(alturas(a),azimutes(a))
end
end

function editlongg_Callback(hObject, eventdata, handles)
function edit29_Callback(hObject, eventdata, handles)
function edit30_Callback(hObject, eventdata, handles)
function editlongg_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function editlatg_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end
function editlatg_Callback(hObject, eventdata, handles)

function editlatm_CreateFcn(hObject, eventdata, handles)

```

```

if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end
function editlatm_Callback(hObject, eventdata, handles)
function editlongm_Callback(hObject, eventdata, handles)

function editlongm_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function pushnorte_Callback(hObject, eventdata, handles)

function pushoeste_Callback(hObject, eventdata, handles)

function editdia_Callback(hObject, eventdata, handles)

function editdia_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function editmes_Callback(hObject, eventdata, handles)

function editmes_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function editano_Callback(hObject, eventdata, handles)

function editano_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end
function edit31_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end
function edit30_CreateFcn(hObject, eventdata, handles)

```

```

if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit31_Callback(hObject, eventdata, handles)

function pushbutton15_CreateFcn(hObject, eventdata, handles)

function edit29_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function uitable6_CellSelectionCallback(hObject, eventdata, handles)

function latsinal_Callback(hObject, eventdata, handles)

latitudesinal=get(handles.latsinal,'string');

j=strfind(latitudesinal,'S');
if j==1
    set(handles.latsinal,'String','N');
else
set(handles.latsinal,'String','S');
end

function latsinal_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function longsinal_Callback(hObject, eventdata, handles)
longsinal=get(handles.longsinal,'string');

k=strfind(longsinal,'E');
if k==1
    set(handles.longsinal,'String','W');
else
set(handles.longsinal,'String','E');
end

function longsinal_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

%=====BOTÃO RESET=====
function Resetbutton_CreateFcn(hObject, eventdata, handles)

```

```

function Resetbutton_Callback(hObject, eventdata, handles)
axes(handles.axes2)
cla
grafico
yourdata =[];

% set(handles.uitable7, 'Data', cell(size(get(handles.uitable7,'Data'))));

set(handles.uitable7,'Data',yourdata);
set(handles.uitable6,'Data',yourdata);

%===== BOTÃO CARREGAR HORAS =====%
function pushbutton14_CreateFcn(hObject, eventdata, handles)
function pushbutton14_Callback(hObject, eventdata, handles)
tableData = get(handles.uitable7, 'data');
h_inicio=tableData(1);
m_inicio=tableData(3);

h_fim=tableData(2);
m_fim=tableData(4);

hora_inicio=str2double(cell2mat(h_inicio))+str2double(cell2mat(m_inicio))/60;
hora_fim=str2double(cell2mat(h_fim))+str2double(cell2mat(m_fim))/60;

inst_fav=horassexa((hora_inicio+hora_fim)/2);
set(handles.edit29,'String', num2str(inst_fav(1)));
set(handles.edit30,'String', num2str(inst_fav(2)));
set(handles.edit31,'String', num2str(inst_fav(3)));

%===== BOTÕES DE SELECÇÃO AM/PM=====%

function radiobutton1_Callback(hObject, eventdata, handles)

function radiobutton2_Callback(hObject, eventdata, handles)

function uitable7_CreateFcn(hObject, eventdata, handles)
set(hObject, 'Data', cell(2));

function uitable6_CreateFcn(hObject, eventdata, handles)

%===== BOTÃO SUGESTÃO =====%

function pushbutton15_Callback(hObject, eventdata, handles)
axes(handles.axes2)
cla
grafico
sacar_var;
names={'Sol';'Lua';'Venus';'Marte';'Jupiter';'Saturno';'Acomar';'Achernar';'Ac
rux';'Adhara';'Aldebaran';'Alioth';'Alkaid';'Al
Na'ir';'Alnilam';'Alphard';'Alphecca';'Alpheratz';'Altair';'Ankaa';'Antares';'

```

```

Arcturus'; 'Atria'; 'Arior'; 'Bellatrix'; 'Betelgeuse'; 'Canopus'; 'Capella'; 'Deneb'
; 'Denebola'; 'Diphda'; 'Dubhe'; 'Elnath'; 'Eltanin'; 'Enif'; 'Fomalhaut'; 'Gacrux'; 'G
ienah'; 'Hadar'; 'Hamal'; 'Kaus
Australis'; 'Kochab'; 'Markab'; 'Menkar'; 'Menkent'; 'Miaplacidus'; 'Mirfak'; 'Nunki'
; 'Peacock'; 'Pollux'; 'Procyon'; 'Rasalhague'; 'Regulus'; 'Rigel'; 'Rigil
Kentaurus'; 'Sabik'; 'Schedar'; 'Shaula'; 'Sirius'; 'Spica'; 'Suhail'; 'Vega'; 'Zubene
lgenubi'; 'Polaris'; 'Octantis'};

C=tabela(ano,mes,dia,hora,minuto,segundo,latitudedecimal,longitudedecimal);

alturas=(C(:,1));
azimutes=(C(:,2));

controlo=zeros(65,1);
for k=1:1:65
    if C(k,1) > 10 && C(k,1) < 70
        controlo(k,1)=1;
    end
end
names(controlo == 0) = [];
alturas(controlo == 0) = [];
azimutes(controlo == 0) = [];
def_graf;
mar=strcmp('Marte',names);
ven=strcmp('Venus',names);
jup=strcmp('Jupiter',names);
sat=strcmp('Saturno',names);
lua=strcmp('Lua',names);

I_lua=find(lua);
I_mar=find(mar);
I_ven=find(ven);
I_jup=find(jup);
I_sat=find(sat);

alturastab=[];
azimutestab=[];
namestab=[];

if ~isempty(I_lua)
axes(handles.axes2)
lua_grafico(alturas(I_lua),azimutes(I_lua),theta,phi)
alturastab=[alturastab;alturas(I_lua)];
azimutestab=[azimutestab;azimutes(I_lua)];
namestab=[namestab;names(I_lua)];
end

if ~isempty(I_ven)
axes(handles.axes2)
venus_grafico(alturas(I_ven),azimutes(I_ven),theta,phi)
alturastab=[alturastab;alturas(I_ven)];
azimutestab=[azimutestab;azimutes(I_ven)];
namestab=[namestab;names(I_ven)];
end

```

```

if ~isempty(I_mar)
axes(handles.axes2)
marte_grafico(alturas(I_mar),azimutes(I_mar),theta,phi)
alturastab=[alturastab;alturas(I_mar)];
azimutestab=[azimutestab;azimutes(I_mar)];
namestab=[namestab;names(I_mar)];
end

if ~isempty(I_jup)
axes(handles.axes2)
jupiter_grafico(alturas(I_jup),azimutes(I_jup),theta,phi)
alturastab=[alturastab;alturas(I_jup)];
azimutestab=[azimutestab;azimutes(I_jup)];
namestab=[namestab;names(I_jup)];
end

if ~isempty(I_sat)
axes(handles.axes2)
saturno_grafico(alturas(I_sat),azimutes(I_sat),theta,phi)
alturastab=[alturastab;alturas(I_sat)];
azimutestab=[azimutestab;azimutes(I_sat)];
namestab=[namestab;names(I_sat)];
end

%Escolha das Estrelas
escolha_estrelas

alturastab=[alturastab;alturas(estrela_1);alturas(estrela_2);alturas(estrela_3)
);alturas(estrela_4);alturas(estrela_5);alturas(estrela_6);alturas(estrela_7);
];
azimutestab=[azimutestab;azimutes(estrela_1);azimutes(estrela_2);azimutes(estr
ela_3);azimutes(estrela_4);azimutes(estrela_5);azimutes(estrela_6);azimutes(es
trela_7);];
namestab=[namestab;names(estrela_1);names(estrela_2);names(estrela_3);names(es
trela_4);names(estrela_5);names(estrela_6);names(estrela_7);] ;

checkboxx1=cell([numel(alturastab) 1]);
for k=1:1:numel(alturastab)
checkboxx1{k,1}=true;
end

alturastab = angle2str(alturastab,'none','degrees2dm',-1);

alturastab=cellstr(alturastab);
azimutestab=form_az(azimutestab);
azimutestab=cellstr(azimutestab);

yourdata =[checkboxx1 namestab azimutestab alturastab ];

columneditable = [true false false false] ;
cnames = {'','Astro','Az','h'};

set(handles.uitable6, 'Data', cell(size(get(handles.uitable6,'Data'))));

```

```

set(handles.uitable6,'Data',yourdata,'ColumnEditable',
columneditable,'ColumnName',cnames);

%===== BOTÃO AZIMUTES DO SOL =====%

function pushbutton27_Callback(hObject, eventdata, handles)
axes(handles.axes2)
cla
grafico
def_graf;
sacar_var
% dados=zeros(23,6);
AR=zeros(23,1);
dec=zeros(23,1);
az_sol=zeros(23,1);
h_sol=zeros(23,1);

nascer_ocaso=nascimentoeocaso('Sol',latitudedecimal,longitudedecimal,dia,mes,ano);
nascer=round(nascer_ocaso(1));
ocaso=round(nascer_ocaso(2));

for h=nascer:1:ocaso
%   horastab=[horastab;h];
%   dados=efemerides(dia,mes,ano,h,0,0);
    dados=Sol(ano,mes,dia,h,0,0);

    AR(h)=dados(1);
    dec(h)=dados(2);
dados_hori=horizontais(ano,mes,dia,h,0,0,latitudedecimal,longitudedecimal,AR(h),dec(h));
az_sol(h)=dados_hori(2);
h_sol(h)=dados_hori(1);
sol_grafico(h_sol(h),az_sol(h),theta,phi);
end
view(50,10)

az_sol(az_sol == 0) = [];
h_sol(h_sol==0)= [] ;
horastab=nascer:1:ocaso;
horastab=(horastab)';
checkbox1=cell(numel(h_sol),1);
for k=1:1:numel(h_sol)
checkbox1{k,1}=true;
end
h_sol = angle2str(h_sol,'none','degrees2dm',-1);
% h_sol=num2str(h_sol);
h_sol=cellstr(h_sol);
az_sol=form_az(round(az_sol));
az_sol=cellstr(az_sol);

horastab=num2str(horastab(:,1),'%02g:00');

horastab=cellstr(horastab);

```

```

yourdata =[checkboxx1 horastab az_sol h_sol ];

columneditable = [true false false false] ;

cnames = {'','Horas','Azimute','Altura'};

set(handles.uitable6, 'Data', cell(size(get(handles.uitable6,'Data'))));

set(handles.uitable6,'Data',yourdata,'ColumnEditable',
columneditable,'ColumnName',cnames);

% --- Executes on button press in pushbutton28.
function pushbutton28_Callback(hObject, eventdata, handles)

tableData = get(handles.uitable6, 'data');
checkboxx1=tableData(:,1);
names=tableData(:,2);
alturas=tableData(:,4);
azimutes=tableData(:,3);
azimutes=str2double(azimutes);

[sortedValues, sortOrder] = sort(azimutes);
azimutes=form_az(round(azimutes));
azimutes=cellstr(azimutes);

alturas = alturas(sortOrder, :);
azimutes=azimutes(sortOrder, :);
names=names(sortOrder, :);

yourdata =[checkboxx1 names azimutes alturas ];
columneditable = [true false false false] ;
cnames = {'','Astro','Az','h'};
set(handles.uitable6,'Data',yourdata,'ColumnEditable',
columneditable,'ColumnName',cnames);

```

## Terra

```

function dadosterra =Terra(ano,mes,dia,hora,minuto,segundo)

% Cálculo de tau
if floor(hora)==hora
else
    min=(hora-floor(hora))*60;
    minuto=floor(min);
    segundo=round((min-minuto)*60);
    hora=floor(hora);
end

```

```

jd=diajuliano(ano,mes,dia);
deltaT=dT(jd,ano)*3600*24*36525;
t = datetime(ano,mes,dia,hora,minuto,segundo);
TD = t + seconds(deltaT);
JDE=juliandate(TD);

tau=(JDE-2451545)/365250;

Matrizes_Terra;

%Cálculo de L0
mL0=MatrizL0;
for n = 1:1:64

L0_termoN(n)=mL0(n,1)*cos(mL0(n,2)+mL0(n,3)*tau);

end

L0=sum(L0_termoN);

%Cálculo de L1
mL1=MatrizL1;
for n = 1:1:34

L1_termoN(n)=mL1(n,1)*cos(mL1(n,2)+mL1(n,3)*tau);

end

L1=sum(L1_termoN);

%Cálculo de L2
mL2=MatrizL2;
for n = 1:1:20

L2_termoN(n)=mL2(n,1)*cos(mL2(n,2)+mL2(n,3)*tau);

end

L2=sum(L2_termoN);

%Cálculo de L3

```

```

mL3=MatrizL3;
for n = 1:1:7

L3_termoN(n)=mL3(n,1)*cos(mL3(n,2)+mL3(n,3)*tau);

end

L3=sum(L3_termoN);

%Cálculo de L4
mL4=MatrizL4;
for n = 1:1:3

L4_termoN(n)=mL4(n,1)*cos(mL4(n,2)+mL4(n,3)*tau);

end

L4=sum(L4_termoN);

%Cálculo de L5

mL5=MatrizL5;
for n = 1:1:1

L5_termoN(n)=mL5(n,1)*cos(mL5(n,2)+mL5(n,3)*tau);

end

L5=sum(L5_termoN);

%Cálculo de L

L=(L0+L1*tau+L2*tau^2+L3*tau^3+L4*tau^4+L5*tau^5)/10^8;
Lgraus=radtodeg(L);
L=wrapTo360(Lgraus);

%LATITUDE

mB0=MatrizB0;
for n = 1:1:5

B0_termoN(n)=mB0(n,1)*cos(mB0(n,2)+mB0(n,3)*tau);

end

```

```

B0=sum(B0_termoN);

%Cálculo de B1

mB1=MatrizB1;
for n = 1:1:2

B1_termoN(n)=mB1(n,1)*cos(mB1(n,2)+mB1(n,3)*tau);

end

B1=sum(B1_termoN);

%Cálculo de B
B=(B0+B1*tau)/10^8;
B=radtodeg(B);

%DISTÂNCIA AO SOL

mR0=MatrizR0;
for n = 1:1:40

R0_termoN(n)=mR0(n,1)*cos(mR0(n,2)+mR0(n,3)*tau);

end

R0=sum(R0_termoN);

mR1=MatrizR1;
for n = 1:1:10

R1_termoN(n)=mR1(n,1)*cos(mR1(n,2)+mR1(n,3)*tau);

end

R1=sum(R1_termoN);

%Cálculo de R2
mR2=MatrizR2;
for n = 1:1:6

R2_termoN(n)=mR2(n,1)*cos(mR2(n,2)+mR2(n,3)*tau);

end

```

```

R2=sum(R2_termoN);

%Cálculo de R3

mR3=MatrizR3;
for n = 1:1:2

R3_termoN(n)=mR3(n,1)*cos(mR3(n,2)+mR3(n,3)*tau);

end

R3=sum(R3_termoN);

%Cálculo de R4
mR4=MatrizR4;
for n = 1:1:1

R4_termoN(n)=mR4(n,1)*cos(mR4(n,2)+mR4(n,3)*tau);

end

R4=sum(R4_termoN);

%Cálculo de R

R=(R0+R1*tau+R2*tau^2+R3*tau^3+R4*tau^4)/10^8;

dadosterra(1)=L;
dadosterra(2)=B;
dadosterra(3)=R;

```

## Sol – Algoritmo 6

```

function coordenadas=efemerides(dia,mes,ano,hora,minuto,segundo)
%Esta função calcula a ascensao reta aparente, declinação,
%semidiametro do sol com uma precisão de 0'.6 no periodo de 2000AC a 2200DC

%STEP 1 - dia juliano da data inserida

hora1=hora+minuto/60+segundo/3600;
jd=diajuliano(dia,mes,ano);
t=t2000(jd,hora1);
%t será o intervalo de 01/01/2000 as 12UTC em séculos julianos de 36525

```

```

%dias

%STEP 2- calcular o intervalo de tempo correspondente numa escala de tempo
%uniforme [This is based on formulae by Stephenson and Morrison in Phil.
Trans. R. Soc. Lond. A 313,
%47-70, (1984), "Long-term changes in the rotation of the Earth: 700 BC to AD
1980". Current
%values are published in section K of The Astronomical Almanac]

deltaT=dT(t,ano); %dynamic time ver artigo

T=t+deltaT; %QUANDO ACABAR DEBUG TIRA AQUI %

%STEP 3- calcular argumentos do sol

LAMBDAm=wrapTo360(280.46645 + wrapTo360(36000.76975*T) +
wrapTo360(0.0003132*(T^2))); %retira multiplos de 360

G=wrapTo360(357.529 + 35999.05029*T);

Eo=23.4393 - 0.01301*T - 0.0000001*(T^2) + 0.0000006*(T^3);

C=(1.91467 - 0.00482 * T - 0.000015* T^2) * sind(G) + (0.01999) * sind (2*G);

%STEP 4

OMEGA = wrapTo360(125.045 - 1934.136 * T);%acrescentei o wrap que nao
%pus no fluxograma

%=====
%deltas=nutacaoobliq(dia,mes,ano,hora,minuto,segundo);
%deltaL=deltas(1);
%deltaE=deltas(2)

%=====
%Em alternativa calcular o DeltaE como deve de ser e usar
E=Eo+0.0026*cosd(OMEGA);

LAMBDA=wrapTo360(LAMBDAm + C - 0.00569 - 0.00478 *sind(OMEGA));

%STEP 5 a)

x=cosd(LAMBDA);

y=cosd(E) *sind(LAMBDA);

z=sind(E) * sind(LAMBDA);

A = atand(y/x); %em alternativa usar A=tan?1(cos ? tan ?)

if x<0, alfa= A+180;
elseif(x>0 && y<0), alfa=A+360;

```

```

else alfa=A;
end
A=A;
%STEP 5 b)
dec=asind(z);
SD=0.2666/(1-0.017*cosd(G));

%STEP 6

%definir GHA
%GHA=wrapTo360(TSG(ano,mes,dia,hora,minuto,segundo)-alfa);
GHA = wrapTo360(100.4606 + 36000.76998 * t + 0.000387 * (t^2) + 15*hora1 -
0.0048 *sind(OMEGA) *cosd( Eo) - alfa);

%Determinação da equação de tempo
eq1 = GHA -(15*hora1 - 180);
if eq1 > 10 , eq1 = eq1 - 360;
end
eq=abs(eq1*24/360)*60;

%Definir o vetor tt constituído pelo intercalo entre hora-h e hora+h onde
%se vai procurar o zero do GHA que corresponde ao momento da passagem
meridiana

if hora<11.5
    tt=linspace(hora,hora+12.499999,1000);
elseif hora>12.5
    tt=linspace(hora-12.499999,hora,1000);
else
    tt=linspace(hora-1,hora+1,1000);
end
%Definir vetor que vai dispor todos os valores de GHA para o intervalo
%definido no vetor tt

%calculo da hora da meridiana
GHA_tt=wrapTo360(100.4606+wrapTo360(36000*t)+0.76998*t+0.000387*t^2+15*tt-
0.0048*sind(OMEGA)*cosd(Eo)-alfa);

[mm, id]=min(GHA_tt);
meridiana=tt(id);

%cálculo da hora do nascer do Sol
Jtransit=2451545 + 0.0053*sind(G)-0.0069*sind(2*LAMBDA);
meridianahhmmss=horassexa(meridiana);
jt=juliandate(ano,mes,dia,meridianahhmmss(1),meridianahhmmss(2),meridianahhmmss(3));

%definição das variáveis de saída
coordenadas(1)=alfa;
coordenadas(2)=dec;
coordenadas(3)=GHA;
coordenadas(4)=SD;

```

```
coordenadas(5)=eq;
coordenadas(6)=meridiana;
end
```

## Sol – Algoritmo 7

```
function dados_sol=Sol(ano, mes, dia, hora, minuto, segundo)
horal=hora+minuto/60+segundo/3600;
jd=diajuliano(dia,mes,ano);
t=t2000(jd,horal);

deltaT=dT(t,ano); %dynamic time ver artigo

T=t+deltaT;

dados_terra=Terra(ano, mes, dia, hora, minuto, segundo);
L=dados_terra(1);
B=dados_terra(2);
R=dados_terra(3);
geo_long=L+180;
BETA=-B;

nut=nutacaoobliq(T);

deltaL=nut(1);
% deltaE=nut(2);
% obliq_media=obliqmedia(T);

geo_long=geo_long+deltaL;
geo_long=geo_long -0.005691611111111111/R;

Equatoriais=Eclip2Eq(geo_long,BETA,T);
AR=Equatoriais(1);
Dec=Equatoriais(2);
GHA=wrapTo360(TSG(ano,mes,dia,hora,minuto,segundo)-AR);

%Eq. Tempo
eq1 = GHA -(15*horal - 180);
if eq1 > 10 , eq1 = eq1 - 360;
end
eq=abs(eq1*24/360)*60;
%SD
G=wrapTo360(357.529 + 35999.05029*T);
SD=0.2666/(1-0.017*cosd(G));

dados_sol(1)=AR;
dados_sol(2)=Dec;
dados_sol(3)=GHA;
dados_sol(4)=SD;
dados_sol(5)=eq;
```

```

dados_sol (6)=geo_long;
dados_sol (7)=R;

end

```

## Lua

```

function dadosdalu=Lua(ano,mes,dia,hora,minuto,segundo)

T=Tuniforme(ano,mes,dia,hora,minuto,segundo);

%Moon's mean longitude
Llinha=wrapTo360(218.3164477 + 481267.88123421 * T) - 0.0015786 * T^2 +
T^3/538841 - T^4/65194000;

%Mean elongation of the moon
D=wrapTo360(297.8501921 + 445267.1114034 * T - 0.0018819*(T^2)+(T^3)/545868 -
T^4/113065000);

%Mean anomaly of the Sun
M=wrapTo360(357.5291092 + 35999.0502909 * T - 0.0001536 * T^2 - T^3/24490000);

%Mean anomaly of the Moon
Mlinha=wrapTo360(134.9633964+ + 477198.8675055 * T + 0.0087414 * T^2 +
T^3/69699) - T^4/14712000;

%Moon's argument of latitude
F=wrapTo360(93.2720950 + 483202.0175233 * T - 0.0036539*T^2 - T^3/3526000) -
T^4/863310000;

%Longitude do nó ascendente
% OMEGA=wrapTo360(125.04452 - 1934.136261*T + 0.0020708*T^2 + T^3/450000);

A1 = wrapTo360(119.75 + 131.849 * T);
A2 = wrapTo360(53.09 + 479264.290 * T);
A3 = wrapTo360(313.45 + 481266.484 * T);

E = 1-0.002516 * T -0.0000074 * T^2;

Matrizes_lua;

%Para a Longitude
argumento=zeros(60,1);
Eln=zeros(60,1);
for n = 1:1:60

an1 = LR_MatrizDMMF(n,1);
an2 = LR_MatrizDMMF(n,2);
an3 = LR_MatrizDMMF(n,3);
an4 = LR_MatrizDMMF(n,4);

```

```

argumento(n)=an1*D + an2*M + an3*Mlinha + an4*F;

Eln(n)=C_L(n) * sind(argumento(n));

if an2==1 || an2==-1
Eln(n)=Eln(n)*E;
elseif an2==-2 || an2==2

    Eln(n)= Eln(n) * E^2;
else
    Eln(n)=Eln(n);
end
end

El=sum(Eln);

%Para a Latitude

Ebn=zeros(60,1);
for n = 1:1:60

an1 = lat_MatrizDMMF(n,1);
an2 = lat_MatrizDMMF(n,2);
an3 = lat_MatrizDMMF(n,3) ;
an4 = lat_MatrizDMMF(n,4);

argumento(n)=an1*D + an2*M + an3*Mlinha + an4*F;

Ebn(n)=C_b(n) * sind(argumento(n));

if an2==1 || an2==-1
Ebn(n)=Ebn(n)*E;
elseif an2==-2 || an2==2

    Ebn(n)= Ebn(n) * E^2;
else
    Ebn(n)=Ebn(n);
end

end

Eb=sum(Ebn);

```

```

%Para a distancia
Ern=zeros(60,1);
for n = 1:1:60

an1 = LR_MatrizDMMF(n,1);
an2 = LR_MatrizDMMF(n,2);
an3 = LR_MatrizDMMF(n,3) ;
an4 = LR_MatrizDMMF(n,4);

argumento(n)=an1*D + an2*M + an3*Mlinha + an4*F;

Ern(n)=C_r(n) * cosd(argumento(n));

if an2==1||an2==-1
Ern(n)=Ern(n)*E;
elseif an2==-2||an2==2

    Ern(n)= Ern(n)* E^2;
else
    Ern(n)=Ern(n);
end

end

Er=sum(Ern);
%Definir os aditivos

aditivoE1=3958*sind(A1)+1962*sind(Llinha-F)+318*sind(A2);

aditivoEb= -2235*sind(Llinha)+382*sind(A3) + 175*sind(A1-F) +
175*sind(A1+F)+127*sind(Llinha-Mlinha)-115*sind(Llinha + Mlinha);

%Correção com os aditivos

El=El+aditivoE1;
Eb=Eb+aditivoEb;

%Longitude, latitude e distancia da lua em km
LAMBDA=Llinha + El/1000000;

Beta=Eb/1000000;

delta=385000.56+Er/1000;

correcoes=nutacaoobliq(T);

nutacaolong=correcoes(1);

```

```

LAMBDAaparente=LAMBDA + nutacaolong;

saidas=Eclip2Eq(LAMBDAaparente, Beta,T);

AR=saidas(1);
dec=saidas(2);

HP=asind(6378.14/delta);

%CÁLCULO DA FRAÇÃO ILUMINADA
dados_sol=Sol(ano,mes,dia,hora,minuto,segundo);
AR_sol=dados_sol(1);
dec_sol=dados_sol(2);
R=dados_sol(7)*149598000;
% Encontrar o ângulo da fase i
ELONG=acosd(sind(dec_sol)*sind(dec)+cosd(dec_sol)*cosd(dec)*cosd(AR_sol-AR));
I_livro=atand((R*sind(ELONG))/(delta-R*cosd(ELONG)));

if I_livro<0
    I_livro=I_livro+180;
elseif I_livro>180
    I_livro=I_livro-180;
end

%Fracção iluminada k:
k=(1+ cosd (I_livro))/2;

dadosdalu(1)=AR;
dadosdalu (2) =dec;
dadosdalu (3)=HP;
dadosdalu(4)=k;
end

```

## Venus

```

function saidas=Venus(ano,mes,dia,hora,minuto,segundo)

%Esta função calcula as coordenadas equatoriais de Vénus para uma
%determinada data yy mm dd hh mm ss

%Para este cálculo o Tempo tau32 juliano será medido em milénios desde
%a época de referência J2000. A designação tau32 foi adoptada para que
%seja claramente a referência ao tau calculado no cap32 de Meeus
%Não confundir com o tau33 que é uma diferença de tempo entre
%o tempo que a Luz leva a chegar do planeta até à Terra, conforme
%descrito no cap33 de Meeus.

%Cálculo de tau32

jd=diajuliano(ano,mes,dia);

```

```

deltaT=dT(jd,ano)*3600*24*36525;
t = datetime(ano,mes,dia,hora,minuto,segundo);
TD = t + seconds(deltaT);
JDE=juliandate(TD);
tau32=(JDE-2451545)/365250;

%Cálculo das coordenadas heliocentricas de Venus

helio_venus=heliovenus(tau32);

L=helio_venus(1);
B=helio_venus(2);
R=helio_venus(3);

dadosterra=Terra(ano,mes,dia,hora,minuto,segundo);

L0=dadosterra(1);
B0=dadosterra(2);
R0=dadosterra(3);

x=R*cosd(B)*cosd(L)-R0*cosd(B0)*cosd(L0);
y=R*cosd(B)*sind(L)-R0*cosd(B0)*sind(L0);
z=R*sind(B)-R0*sind(B0);

%Distância à Terra
delta=sqrt(x^2+y^2+z^2);

%tau33 tempo que a luz demora a chegar do planeta à Terra
tau33= 0.0057755183*delta;

novoJDE=JDE-tau33;

novotau32=(novoJDE-2451545)/365250;

heliocorrigido=heliovenus(novotau32);

L=heliocorrigido(1);
B=heliocorrigido(2);
R=heliocorrigido(3);

x=R*cosd(B)*cosd(L)-R0*cosd(B0)*cosd(L0);
y=R*cosd(B)*sind(L)-R0*cosd(B0)*sind(L0);
z=R*sind(B)-R0*sind(B0);

LAMBDA=wrapTo360(atan2d(y,x));
BETA=atand(z/(sqrt(x^2+y^2)));

%Determinar as correções de aberração necessárias para a data
%novotau32 é multiplicado por 10 para que esteja em séculos (necessário
%para a entrada na função aberr)

corraberr=aberr(novotau32*10,LAMBDA,BETA);

```

```

deltaLAMBDA=corraberr(1);
deltaBETA=corraberr(2);

LAMBDA=LAMBDA+deltaLAMBDA;
BETA=BETA+deltaBETA;

corrnut=nutacaoobliq(novotau32*10);

LAMBDA=LAMBDA+corrnut(1);

equatoriais=Eclip2Eq(LAMBDA,BETA,novotau32*10);
sd=(0.002316666666666667/delta)*60;
hp=(0.002442777777777778/delta)*60;
saidas(1)=equatoriais(1);
saidas(2)=equatoriais(2);
saidas(3)=sd;
saidas(4)=hp;

```

## Marte

```

function saidas=Marte(ano,mes,dia,hora,minuto,segundo)

%Esta função calcula as coordenadas equatoriais de Marte para uma
%determinada data yy mm dd hh mm ss

%Para este cálculo o Tempo tau32 juliano será medido em milénios desde
%a época de referência J2000. A designação tau32 foi adoptada para que
%seja claramente a referência ao tau calculado no cap32 de Meeus
%Não confundir com o tau33 que é uma diferença de tempo entre
%o tempo que a Luz leva a chegar do planeta até à Terra, conforme
%descrito no cap33 de Meeus.

%Cálculo de tau32

jd=diajuliano(ano,mes,dia);
deltaT=dT(jd,ano)*3600*24*36525;
t = datetime(ano,mes,dia,hora,minuto,segundo);
TD = t + seconds(deltaT);
JDE=juliandate(TD);
tau32=(JDE-2451545)/365250;

%Cálculo das coordenadas heliocentricas de Marte

helio_marte=heliomarte(tau32);

L=helio_marte(1);
B=helio_marte(2);

```

```

R=helio_marte(3);

dadosterra=Terra(ano,mes,dia,hora,minuto,segundo);

L0=dadosterra(1);
B0=dadosterra(2);
R0=dadosterra(3);

x=R*cosd(B)*cosd(L)-R0*cosd(B0)*cosd(L0);
y=R*cosd(B)*sind(L)-R0*cosd(B0)*sind(L0);
z=R*sind(B)-R0*sind(B0);

%Distância à Terra
delta=sqrt(x^2+y^2+z^2);

%tau33 tempo que a luz demora a chegar do planeta à Terra
tau33= 0.0057755183*delta;

novoJDE=JDE-tau33;

novotau32=(novoJDE-2451545)/365250;

heliocorrigido=heliomarte(novotau32);

L=heliocorrigido(1);
B=heliocorrigido(2);
R=heliocorrigido(3);

x=R*cosd(B)*cosd(L)-R0*cosd(B0)*cosd(L0);
y=R*cosd(B)*sind(L)-R0*cosd(B0)*sind(L0);
z=R*sind(B)-R0*sind(B0);

LAMBDA=wrapTo360(atan2d(y,x));
BETA=atand(z/(sqrt(x^2+y^2)));

%Determinar as correções de aberração necessárias para a data
%novotau32 é multiplicado por 10 para que esteja em séculos (necessário
%para a entrada na função aberr)

corraberr=aberr(novotau32*10,LAMBDA,BETA);

deltaLAMBDA=corraberr(1);
deltaBETA=corraberr(2);

LAMBDA=LAMBDA+deltaLAMBDA;
BETA=BETA+deltaBETA;

corrnut=nutacaoobliq(novotau32*10);

LAMBDA=LAMBDA+corrnut(1);

```

```

equatoriais=Eclip2Eq(LAMBDA,BETA,novotau32*10);
sd=(0.0013/delta)*60;
hp=(0.002442777777777778/delta)*60;
saidas(1)=equatoriais(1);
saidas(2)=equatoriais(2);
saidas(3)=sd;
saidas(4)=hp;

```

## Jupiter

```

function equatoriais=Jupiter(ano,mes,dia,hora,minuto,segundo)

%Esta função calcula as coordenadas equatoriais de Jupiter para uma
%determinada data yy mm dd hh mm ss

%Para este cálculo o Tempo tau32 juliano será medido em milénios desde
%a época de referência J2000. A designação tau32 foi adoptada para que
%seja claramente a referência ao tau calculado no cap32 de Meeus
%Não confundir com o tau33 que é uma diferença de tempo entre
%o tempo que a Luz leva a chegar do planeta até à Terra, conforme
%descrito no cap33 de Meeus.

%Cálculo de tau32

jd=diajuliano(ano,mes,dia);
deltaT=dT(jd,ano)*3600*24*36525;
t = datetime(ano,mes,dia,hora,minuto,segundo);
TD = t + seconds(deltaT);
JDE=juliandate(TD);
tau32=(JDE-2451545)/365250;

%Cálculo das coordenadas heliocentricas de jupiter

helio_jupiter=heliojupiter(tau32);

L=helio_jupiter(1);
B=helio_jupiter(2);
R=helio_jupiter(3);

dadosterra=Terra(ano,mes,dia,hora,minuto,segundo);

L0=dadosterra(1);
B0=dadosterra(2);
R0=dadosterra(3);

x=R*cosd(B)*cosd(L)-R0*cosd(B0)*cosd(L0);
y=R*cosd(B)*sind(L)-R0*cosd(B0)*sind(L0);
z=R*sind(B)-R0*sind(B0);

```

```

%Distância à Terra
delta=sqrt(x^2+y^2+z^2);

%tau33 tempo que a luz demora a chegar do planeta à Terra
tau33= 0.0057755183*delta;

novoJDE=JDE-tau33;

novotau32=(novoJDE-2451545)/365250;

heliocorrigido=heliojupiter(novotau32);

L=heliocorrigido(1);
B=heliocorrigido(2);
R=heliocorrigido(3);

x=R*cosd(B)*cosd(L)-R0*cosd(B0)*cosd(L0);
y=R*cosd(B)*sind(L)-R0*cosd(B0)*sind(L0);
z=R*sind(B)-R0*sind(B0);

LAMBDA=wrapTo360(atan2d(y,x));
BETA=atand(z/(sqrt(x^2+y^2)));

%Determinar as correções de aberração necessárias para a data
%novotau32 é multiplicado por 10 para que esteja em séculos (necessário
%para a entrada na função aberr)

corraberr=aberr(novotau32*10,LAMBDA,BETA);

deltaLAMBDA=corraberr(1);
deltaBETA=corraberr(2);

LAMBDA=LAMBDA+deltaLAMBDA;
BETA=BETA+deltaBETA;

corrnut=nutacaoobliq(novotau32*10);

LAMBDA=LAMBDA+corrnut(1);

equatoriais=Eclip2Eq(LAMBDA,BETA,novotau32*10);

```

## Saturno

```

function equatoriais=Saturno(ano,mes,dia,hora,minuto,segundo)

%Esta função calcula as coordenadas equatoriais de Saturno para uma
%determinada data yy mm dd hh mm ss

```

```
%Para este cálculo o Tempo tau32 juliano será medido em milénios desde
%a época de referência J2000. A designação tau32 foi adoptada para que
%seja claramente a referência ao tau calculado no cap32 de Meeus
%Não confundir com o tau33 que é uma diferença de tempo entre
%o tempo que a Luz leva a chegar do planeta até à Terra, conforme
%descrito no cap33 de Meeus.
```

```
%Cálculo de tau32
```

```
jd=diajuliano(ano,mes,dia);
deltaT=dT(jd,ano)*3600*24*36525;
t = datetime(ano,mes,dia,hora,minuto,segundo);
TD = t + seconds(deltaT);
JDE=juliandate(TD);
tau32=(JDE-2451545)/365250;
```

```
%Cálculo das coordenadas heliocentricas de Saturno
```

```
helio_saturno=heliosaturno(tau32);
```

```
L=helio_saturno(1);
B=helio_saturno(2);
R=helio_saturno(3);
```

```
dadosterra=Terra(ano,mes,dia,hora,minuto,segundo);
```

```
L0=dadosterra(1);
B0=dadosterra(2);
R0=dadosterra(3);
```

```
x=R*cosd(B)*cosd(L)-R0*cosd(B0)*cosd(L0);
y=R*cosd(B)*sind(L)-R0*cosd(B0)*sind(L0);
z=R*sind(B)-R0*sind(B0);
```

```
%Distância à Terra
```

```
delta=sqrt(x^2+y^2+z^2);
```

```
%tau33 tempo que a luz demora a chegar do planeta à Terra
```

```
tau33= 0.0057755183*delta;
```

```
novoJDE=JDE-tau33;
```

```
novotau32=(novoJDE-2451545)/365250;
```

```
heliocorrigido=heliosaturno(novotau32);
```

```
L=heliocorrigido(1);
B=heliocorrigido(2);
R=heliocorrigido(3);
```

```
x=R*cosd(B)*cosd(L)-R0*cosd(B0)*cosd(L0);
y=R*cosd(B)*sind(L)-R0*cosd(B0)*sind(L0);
```

```

z=R*sind(B)-R0*sind (B0);

LAMBDA=wrapTo360 (atan2d (y,x));
BETA=atand (z/ (sqrt (x^2+y^2)));

%Determinar as correções de aberração necessárias para a data
%novotau32 é multiplicado por 10 para que esteja em séculos (necessário
%para a entrada na função aberr)

corraberr=aberr (novotau32*10,LAMBDA,BETA);

deltaLAMBDA=corraberr (1);
deltaBETA=corraberr (2);

LAMBDA=LAMBDA+deltaLAMBDA;
BETA=BETA+deltaBETA;

corrnut=nutacaoobliq (novotau32*10);

LAMBDA=LAMBDA+corrnut (1);

equatoriais=Eclip2Eq (LAMBDA,BETA,novotau32*10);

```

## $\Delta T$

```

function dT=dT(t,ano)
if -390<ano && ano<948

dT=(28.43 + 4.525*t + 1.404*t^2) * 10^-8;
elseif 2000<=ano && ano<2050
    a=ano-2000;
    x=(a*14)/50;
    dT=x+64;
    dT=dT/(3600*24*36525);

elseif 1950<ano && ano<2000
    switch ano
        case 1950, dT= 29.1 ;
        case 1951, dT=29.5 ;
        case 1952, dT= 30 ;
        case 1953, dT= 30.4;
        case 1954, dT=30.7 ;
        case 1955, dT= 31 ;
        case 1956, dT= 31.4;
        case 1957, dT= 31.8 ;
        case 1958, dT= 32.2 ;
        case 1959, dT= 32.8 ;

        case 1960, dT= 33.1 ;

```

```

case 1961, dT= 34 ;
case 1962, dT=34.5 ;
case 1963, dT=34.7 ;
case 1964, dT=35 ;
case 1965, dT=35.8 ;
case 1966, dT=36.5 ;
case 1967, dT= 37 ;
case 1968, dT= 38.3 ;
case 1969, dT= 39 ;

case 1970, dT= 40.2 ;
case 1971, dT= 41 ;
case 1972, dT= 42.2 ;
case 1973, dT= 43 ;
case 1974, dT= 44.5 ;
case 1975, dT= 45.5 ;
case 1976, dT= 46.5 ;
case 1977, dT= 47.5 ;
case 1978, dT= 48.5 ;
case 1979, dT= 49.5 ;

case 1980, dT= 50.5 ;
case 1981, dT= 51 ;
case 1982, dT= 52.2 ;
case 1983, dT= 53 ;
case 1984, dT= 53.8 ;
case 1985, dT= 54 ;
case 1986, dT= 54.9 ;
case 1987, dT= 55.4 ;
case 1988, dT= 55.8 ;
case 1989, dT= 56.4 ;

case 1990, dT= 56.9 ;
case 1991, dT= 57 ;
case 1992, dT= 58.3 ;
case 1993, dT= 59 ;
case 1994, dT= 60 ;
case 1995, dT= 60.8 ;
case 1996, dT= 61.6 ;
case 1997, dT= 62.5 ;
case 1998, dT= 63.0 ;
case 1999, dT= 63.5 ;
end
dT=dT/(3600*24*36525);
else
dT=0.808*(t - 2)^2 * 10^-8;
dT=dT/(3600*24*36525);
end

```

## Estrelas

```
function coordenadas=aparente(estrela,dia,mes,ano,hora)
```

```

%Esta função calcula a ascensão reta aparente, declinação,
%semidiâmetro do sol com uma precisão de 0'.6 no período de 2000AC a 2200DC
%Baseada na nota técnica do almanaque náutico do UKHO
%STEP 1 - dia juliano da data inserida
%recorri a esta função para que todos os resultados
%apresentados ao longo da função tivessem

%Entrada=input('Insira nome da estrela: ');

jd=diajuliano(dia,mes,ano);
d = jd + hora/24 - 2451545;
d=round(d,4);
t = d/36525;
t=round(t,9);
%neste caso T=t
T=t;
%STEP 2 a)- Conversão da tabela invocando os vetores criados em ficheiros à
%parte

%invocação dos vetores constituintes da tabela (cada vetor corresponde a
%uma coluna da tabela

%TABELA:
vector_estrelas;
vector_LAMBDA0;
vector_MIU;
vector_BETA0;
vector_MIU_linha;

%Se a variável estrela de entrada corresponder ao número da estrela na
%tabela o I é automaticamente igual a esse número
if isnumeric(estrela)==1
    I=estrela;
%Caso seja literal(string) a função strcmp procurará o nome da estrela e
%descobrirá o número correspondente
else
X=strcmp(estrela,vector_estrelas);
I=find(X);
end
%STEP 2 b)

LAMBDA0=vetor_LAMBDA0(I);
MIU=vetor_MIU(I);
BETA0=vetor_BETA0(I);
MIU_linha=vetor_MIU_linha(I);

LAMBDA1=LAMBDA0+MIU*T;
BETA1=BETA0+MIU_linha*T;
%STEP3

```

```

LAMBDAm=wrapTo360(280.460 + 36000.770*T );

LAMBDA2=LAMBDA1 - 0.0057* cosd(LAMBDA1 - LAMBDAm)/ cosd (BETA1);
BETA2 = BETA1 + 0.0057* sind(LAMBDA1 - LAMBDAm)* sind (BETA1);

%STEP 4
a = 1.39697 *T + 0.000309 * T^2;
b = 0.0131 *T - 0.00001 * T^2;
c = 5.1236 + 0.2416 *T;

BETA3 = BETA2 + b *sind(LAMBDA2 + c);
LAMBDA3 = LAMBDA2 + a - b * cosd(LAMBDA2 + c)* tand(BETA3);
%STEP 5
OMEGA = wrapTo360(125.045 - 1934.136 * T);
EPSILON0 = 23.4393 - 0.0130 *T;
EPSILON = EPSILON0 + 0.0026 * cosd(OMEGA);
LAMBDA = LAMBDA3 - 0.0048* sind(OMEGA);
BETA = BETA3;

%STEP 6 a)
x = cosd (BETA) *cosd(LAMBDA);
y = cosd(EPSILON)* cosd (BETA)* sind(LAMBDA) - sind(EPSILON)* sind (BETA);
z = sind (EPSILON)* cosd (BETA) *sind (LAMBDA) + cosd (EPSILON)* sind(BETA);

A = atand( y/x);
%STEP 6 b)
if x<0 ,alfa= A+180;
elseif(x>0 && y<0) ,alfa=A+360;
else alfa=A;
end

dec=asind(z);
%STEP 7
GHA=wrapTo360(100.4606+wrapTo360(36000*t)+0.76998*t+0.000387*t^2+15*hora-
0.0048*sind(OMEGA)*cosd(EPSILON)-alfa);
% GHA=wrapTo360(TSG(ano,mes,dia,hora,0,0)-alfa);

vernalHG=wrapTo360(100.4606+wrapTo360(36000*t)+0.76998*t+0.000387*t^2+15*hora-
0.0048*sind(OMEGA)*cosd(EPSILON));
coordenadas(1)=alfa;
coordenadas(2)=dec;
coordenadas(3)=GHA;
coordenadas(4)=vernalHG;
end

```

## Dia Juliano

```

function nu=TSG(ano,mes,dia,hora,minuto,segundo)

%Esta função descobre o Tempo sideral em Greenwich aparente
%para um dado T

```

```

%Baseado no livro do Meeus capitulo 12

%Haroun Tlemçani
%=====

%deltaT=dT(jd0,ano)*36525*24*3600
%dataTD=data+seconds(deltaT)

JD=juliandate(ano,mes,dia,hora,minuto,segundo);
T=(JD-2451545)/36525;
%T=round(T,8)
deltas=nutacaoobliq(T);
deltaL=deltas(1);
deltaE=deltas(2);
Emedia=obliqmedia(T);
Epsilon=Emedia+deltaE;

nuM=280.46061837+360.98564736629*(JD-2451545)+((0.000387933*T^2))-
((T^3)/38710000);
nuM=wrapTo360(nuM);
nu=wrapTo360((nuM)+deltaL*cosd(Epsilon));

end

```

## TSG

```

function nu=TSG(ano,mes,dia,hora,minuto,segundo)

%Esta função descobre o Tempo sideral em Greenwich aparente
%para um dado T

%Baseado no livro do Meeus capitulo 12

%Haroun Tlemçani
%=====

%deltaT=dT(jd0,ano)*36525*24*3600
%dataTD=data+seconds(deltaT)

JD=juliandate(ano,mes,dia,hora,minuto,segundo);
T=(JD-2451545)/36525;

```

```

%T=round(T,8)
deltas=nutacaoobliq(T);
deltaL=deltas(1);
deltaE=deltas(2);
Emedia=obliqmedia(T);
Epsilon=Emedia+deltaE;

nuM=280.46061837+360.98564736629*(JD-2451545)+((0.000387933*T^2))-
((T^3)/38710000);
nuM=wrapTo360(nuM);
nu=wrapTo360((nuM)+deltaL*cosd(Epsilon));

end

```

## Efeitos da Nutação

```

function deltas=nutacaoobliq(T)

%
% %Mean elongation of the moon from Sun
% D=wrapTo360(297.85036+445267.111480*T-0.0019142*(T^2)+(T^3)/189474);
%
% %Mean anomaly of the Sun
% M=wrapTo360(357.52772 + 35999.050340*T-0.0001603*T^2-T^3/300000);
%
% %Mean anomaly of the Moon
% Mlinha=wrapTo360(134.96298+ + 477198.867398*T + 0.0086972*T^2 + T^3/56250);
%
% %Moon's argument of latitude
% F=wrapTo360(93.27191+483202.017538*T - 0.0036825*T^2 + T^3/327270);

%Longitude do nó ascendente
OMEGA=wrapTo360(125.04452 - 1934.136261*T + 0.0020708*T^2 + T^3/450000);

%Mean Longitude of the Sun
L=wrapTo360(280.46665+36000.7698*T);
%Mean Longitude of the Moon
Llinha=wrapTo360(218.3165+481267.8813*T);

%Calcular Nutation in Longitude
deltaL=-4.77777777777778e-03*sind(OMEGA)+(-3.66666666666667e-04*sind(2*L))-
6.38888888888890e-05*sind(2*Llinha)+ 5.83333333333333e-05*sind(2*OMEGA);

%Calcular Nutation in obliquity
deltaEpsilon=2.55555555555555e-03*cosd(OMEGA)+1.58333333333333e-
04*cosd(2*L)+2.77777777777778e-05*cosd(2*Llinha)-2.5e-05*cosd(2*OMEGA);

```

```
%Deltas em graus
deltas (1)=deltaL;
deltas (2)=deltaEpsilon;
end
```

## Obliquidade Média

```
function E0=obliquedia(T)

E0=23.4392911111111110 - 0.013004166666667*T - 1.638888888888889e-07 *(T.^2) +
5.036111111111110e-07*(T.^3);
end
```

## Nascimento , Ocaso e PM

```
%Função de Cálculo do Nascimento e Ocaso de um astro

%Baseado no livro de Meeus capitulo 15

%Para efeito de elaboração do almanaque pode considerar-se a longitude = 0 na
fórmula do Meeus e a função será
%elaborada pela ordem dos passos descritos no livro. Os valores desejados
%serão obtidos por sucessivas iteradas.

%Os outputs serão por ordem: nascimento, ocaso, PMS.

%Haroun Tlemçani

%=====

%Serão necessários inicialmente os dados:

%Qual o astro?
%Latitude
%Longitude (O almanaque dá sempre para Greenwich ou seja 0)
%Data
%=====
function output=nascimentoeocaso(astro,latitude,longitude,dia,mes,ano)
%escrever as expressoes de AR e dec para o dia anterior e dia seguinte

datacorrente=datetime(ano,mes,dia);
dataanterior=datacorrente-1;
dataseguinte=datacorrente+1;

                                     %Passar as datas para formatos trabalháveis
a=datevec(dataanterior);
c=datevec(datacorrente);
s=datevec(dataseguinte);
```

```

%As funções AR0 e dec0 dão estas coordenadas para as 0h UT de um dado dia

alfa1=ascensaoreta0(astro,a(3),a(2),a(1));
alfa2=ascensaoreta0(astro,c(3),c(2),c(1)); %Definir as AR's para os 3 dias
alfa3=ascensaoreta0(astro,s(3),s(2),s(1));

dec1=declinacao0(astro,a(3),a(2),a(1));
dec2=declinacao0(astro,c(3),c(2),c(1)); %Definir as Dec's para os 3 dias
dec3=declinacao0(astro,s(3),s(2),s(1));

%listar o vetor dos astros
vetor_estrelas;

if isnumeric(astro)==1
    I=astro;
end
%encontrar o astro escolhido na lista
X=strcmp(astro,vetor_estrelas);
I=find(X);
if I==60 %0 sol é o astro 60 no meu vetor_estrelas
    h0=-0.8333;

elseif I==61 %caso seja a lua (astro 61) usaremos o valor medio h0
    =0.125

    h0=0.125;
else %para outras estrelas e planetas pode-se considerar
este valor de h0
    h0=-0.5667;
end

%Valor de referência para verificar se o astro é circumpolar

comp=abs(sind(latitude)*sind(dec2)/((cosd(latitude)*cosd(dec2))));

if comp>1
    disp('astro circumpolar');
    H0=wrapTo360(acosd((sind(h0)-
sind(latitude)*sind(dec2))/cosd(latitude)*cosd(dec2)));
else
    H0=wrapTo360(acosd((sind(h0)-
sind(latitude)*sind(dec2))/cosd(latitude)*cosd(dec2)));

end

nu=TSG(ano,mes,dia,0,0,0);

%Efetuar da primeira aproximação dos momentos

%momento aproximado da meridiana
m0=(alfa2+longitude-nu)/360 ;

```

```

if m0>1
    m0=m0-1;
elseif m0<0;
    m0=m0+1;
else m0=m0;
end

%momento aproximado do nascer
m1=m0-(H0)/360 ;
if m1>1
    m1=m1-1;
elseif m1<0
    m1=m1+1;
else
    m1=m1;
end

%momento aproximado do ocaso
m2=m0+(H0)/360 ;
if m2>1
    m2=m2-1;
elseif m2<0
    m2=m2+1;
else
    m2=m2;
end

%Sidereal Time at greenwich

nu00=nu+360.985647*m0 ;    %Para a hora da meridiana
nu01=nu+360.985647*m1 ;    %Para a hora do nascer
nu02=nu+360.985647*m2;    %Para a hora do ocaso

%Interpolar AR e dec para os momentos desejados:

alo=diajuliano(c(3),c(2),c(1)); %dia juliano da data corrente

%O momento desejado (agora chamado 'n') depois de somado o respectivo deltaT
%será n1 para o nascer e n2 para o ocaso
%Para o nascer usa-se m1 + deltaT correspondente da data corrente
n0=m0+(dT(alo,ano)*36525);
n1=m1+(dT(alo,ano)*36525) ;
%resultados certos em comparação com o livro e almanaque a expressão de deltaT
que tenho está feita para adicionar segundos a séculos penso eu.

%A função interpoladora recebendo também os valores dos dias anterior e
seguinte devolverá AR e dec para os momentos n.
alfanascer=interpolar(n1,alfa1,alfa2,alfa3);
decnascer=interpolar(n1,dec1,dec2,dec3);

%Para o ocaso usa-se n2 e m2
n2=m2+(dT(alo,ano))*36525 ;

```

```

alfaocaso=interpolar (n2, alfa1, alfa2, alfa3);
decocaso=interpolar (n2, dec1, dec2, dec3);

%Para a meridiana
alfameridiana=interpolar (n0, alfa1, alfa2, alfa3);

if I==61
    m0h=m0*24;
    dados=Lua (ano, mes, dia, m0h, 0, 0);
    h0=0.7275*dados (3)-0.5666666666666667;
else h0=h0;
end

%NASCER
Hnascer=nu01-longitude-alfanascer; %nu01 tempo sideral à hora do nascer
hnascer=asind (sind (latitude) *sind (decnascer)+cosd (latitude) *cosd (decnascer) *cosd (Hnascer));
deltam1=(hnascer-h0)/(360*cosd (decnascer) *cosd (latitude) *sind (Hnascer));
m1=(m1+deltam1);

%OCASO
Hocaso=nu02-longitude-alfaocaso; %nu02 tempo sideral à hora do ocaso
hocaso=asind (sind (latitude) *sind (decocaso)+cosd (latitude) *cosd (decocaso) *cosd (Hocaso));
deltam2=(hocaso-h0)/(360*cosd (decocaso) *cosd (latitude) *sind (Hocaso));
m2=(m2+deltam2);

%Meridiana
Hmeridiana=nu00-longitude-alfameridiana; %nu02 tempo sideral à hora do ocaso
Hmeridiana=wrapTo180 (Hmeridiana);
deltam0=(-(Hmeridiana/360));
m0=(m0+deltam0);

nu00=nu+360.985647*m0 ; %Para a hora da meridiana
nu01=nu+360.985647*m1 ; %Para a hora do nascer
nu02=nu+360.985647*m2; %Para a hora do ocaso

n0=m0+(dT (alo, ano) *36525);
n1=m1+(dT (alo, ano) *36525) ;
n2=m2+(dT (alo, ano) *36525) ;

alfanascer=interpolar (n1, alfa1, alfa2, alfa3);
decnascer=interpolar (n1, dec1, dec2, dec3);

```

```

alfaocaso=interpolar(n2,alfa1,alfa2,alfa3);
decocaso=interpolar(n2,dec1,dec2,dec3);

alfameridiana=interpolar(n0,alfa1,alfa2,alfa3);

%NASCER
Hnascer=nu01-longitude-alfanascer; %nu01 tempo sideral à hora do nascer
hnascer=asind(sind(latitude)*sind(decnascer)+cosd(latitude)*cosd(decnascer)*cosd(Hnascer));
deltam1=(hnascer-h0)/(360*cosd(decnascer)*cosd(latitude)*sind(Hnascer));
m1=(m1+deltam1);

%OCASO
Hocaso=nu02-longitude-alfaocaso; %nu02 tempo sideral à hora do ocaso
hocaso=asind(sind(latitude)*sind(decocaso)+cosd(latitude)*cosd(decocaso)*cosd(Hocaso));
deltam2=(hocaso-h0)/(360*cosd(decocaso)*cosd(latitude)*sind(Hocaso));
m2=(m2+deltam2);

%Meridiana
Hmeridiana=nu00-longitude-alfameridiana; %nu02 tempo sideral à hora do ocaso
Hmeridiana=wrapTo180(Hmeridiana);
deltam0=(-(Hmeridiana/360));
m0=(m0+deltam0);

m0=m0*24;
m1=m1*24;
m2=m2*24;

output(1)=m1;
output(2)=m2;
output(3)=m0;
end

```

## Crepúsculos

```

%Esta função dá o crepúsculo náutico e civil para uma posição e data.
%Esta foi elaborada com base no livro de Meeus, sendo praticamente igual
%à função de nascimento e ocaso substituindo, porém, o valor da
%altura procurada do Sol(-0.8333)° por -12° para o crepusculo nautico e -6°
para o
%civil.

%Haroun Tlemçani
%=====

```

```

function [crep]=crepusculos(latitude,longitude,dia,mes,ano)

datacorrente=datetime(ano,mes,dia);
dataanterior=datacorrente-1;
dataseguinte=datacorrente+1;

                                     %Passar as datas para formatos trabalháveis
a=datevec(dataanterior);
c=datevec(datacorrente);
s=datevec(dataseguinte);

alfa1=ascensaoreta0('Sol',a(3),a(2),a(1));
alfa2=ascensaoreta0('Sol',c(3),c(2),c(1)); %Definir as AR's para os 3 dias
alfa3=ascensaoreta0('Sol',s(3),s(2),s(1));

dec1=declinacao0('Sol',a(3),a(2),a(1));
dec2=declinacao0('Sol',c(3),c(2),c(1)); %Definir as Dec's para os 3 dias
dec3=declinacao0('Sol',s(3),s(2),s(1));

hnautico=-12; %Definição das alturas procuradas
hcivil=-6;

nu=TSG(ano,mes,dia,0,0,0); %Definição do Vernal horário
Greenwich às 0 UT

comp=abs(sind(latitude)*sind(dec2)/((cosd(latitude)*cosd(dec2))));

if comp>1
    disp('O Sol é circumpolar para esta posição');
else
    H0civil=wrapTo360(acosd((sind(hcivil)-
sind(latitude)*sind(dec2))/cosd(latitude)*cosd(dec2)));

    H0nautico=wrapTo360(acosd((sind(hnautico)-
sind(latitude)*sind(dec2))/cosd(latitude)*cosd(dec2)));
end
%ir buscar os valores de nascimento e ocaso do sol para proceder à correção
%para as horas dos crepusculos

valores=nascimentoeocaso('Sol',latitude,longitude,dia,mes,ano);
m1= valores(1)/24;
m2=valores(2)/24;

%Determinar o vernal GHA:

nu01=nu+360.985647*m1 ; %Para a hora do nascer
nu02=nu+360.985647*m2; %Para a hora do ocaso

alo=diajuliano(c(3),c(2),c(1)); %dia juliano da data corrente

%O momento desejado (agora chamado 'n') depois de somado o respetivo deltaT
%será n1 para o matutino e n2 para o vespertino

```

```

%Para o nascer usa-se m1 + deltaT correspondente da data corrente

n1=m1+(dT(alo,ano)*36525);
n2=m2+(dT(alo,ano)*36525);
%Crepúsculos Civil e Náuticos Matutinos

alfamatutino=interpolar(n1,alfa1,alfa2,alfa3);
decmatutino=interpolar(n1,dec1,dec2,dec3);

Hmatutino=nu01-longitude-alfamatutino; %nu01 tempo sideral à hora do nascer
hmatutino=asind(sind(latitude)*sind(decmatutino)+cosd(latitude)*cosd(decmatuti
no)*cosd(Hmatutino));

deltamlcivil=(hmatutino-
hcivil)/(360*cosd(decmatutino)*cosd(latitude)*sind(Hmatutino));
m1civil=(m1+deltamlcivil)*24;

deltamlnautico=(hmatutino-
hnautico)/(360*cosd(decmatutino)*cosd(latitude)*sind(Hmatutino));
m1nautico=(m1+deltamlnautico)*24;

%Crepúsculos Civil e Náuticos Vespertinos

alfavespertino=interpolar(n2,alfa1,alfa2,alfa3);
decvespertino=interpolar(n2,dec1,dec2,dec3);

Hvespertino=nu02-longitude-alfavespertino; %nu01 tempo sideral à hora do
nascerc
hvespertino=asind(sind(latitude)*sind(decvespertino)+cosd(latitude)*cosd(decve
spertino)*cosd(Hmatutino));

deltam2civil=(hvespertino-
hcivil)/(360*cosd(decvespertino)*cosd(latitude)*sind(Hvespertino));
m2civil=(m2+deltam2civil)*24;

deltam2nautico=(hvespertino-
hnautico)/(360*cosd(decvespertino)*cosd(latitude)*sind(Hvespertino));
m2nautico=(m2+deltam2nautico)*24;

m1civil=hours(m1civil) ;
m1civil.Format='hh:mm:ss';
m1civil=round(m1civil,'minutes');
m1civil=datevec(m1civil);

m1nautico=hours(m1nautico) ;
m1nautico.Format='hh:mm:ss';
m1nautico=round(m1nautico,'minutes');
m1nautico=datevec(m1nautico);

m2civil=hours(m2civil) ;
m2civil.Format='hh:mm:ss';
m2civil=round(m2civil,'minutes');
m2civil=datevec(m2civil);

```

```

m2nautico=hours(m2nautico) ;
m2nautico.Format='hh:mm:ss';
m2nautico=round(m2nautico,'minutes');
m2nautico=datevec(m2nautico);

%crep (1,1)=m1civil
%crep (1,2)=m1nautico
%crep (2,1)=m2civil
%crep (2,2)=m2nautico
crep=[m1civil; m1nautico; m2civil; m2nautico];
%Limpeza Crep
crep=crep(1:4,4:5);
end

```

## Efeito da Aberração

```

function deltas=aberr(T,LAMBDA,BETA)
%Esta função dá o efeito da aberração anual nas longitudes
%e latitudes eclípticas de um determinado astro de acordo com
%Meeus cap23

%Constante da aberração em graus (20.49552/3600)
k=20.49552;
%Longitude verdadeira do sol para a data
STL=soltrueL(T);

%Excentricidade da órbita da Terra
e=0.016708634-0.000042037*T-0.0000001267*T^2;

%Longitude do periélio
Lperi=102.93735+1.71946*T+0.00046*T^2;

deltaLAMBDA=(-k*cosd(STL-LAMBDA)+e*k*cosd(Lperi-LAMBDA))/cosd(BETA);
deltaLAMBDA=deltaLAMBDA/3600;

deltaBETA=-k*sind(BETA)*(sind(STL-LAMBDA))-e*sind(Lperi-LAMBDA);
deltaBETA=deltaBETA/3600;

deltas(1)=deltaLAMBDA;
deltas(2)=deltaBETA;
end

```

## Eclípticas para Equatoriais

```

function posicao =Eclip2Eq(LAMBDA, BETA,T)

```

```

E0=obliquedia(T);
deltas=nutacaoobliq(T);

EPSILON=E0+deltas(2);

x = cosd (BETA) *cosd(LAMBDA);
y = cosd(EPSILON)* cosd (BETA)* sind(LAMBDA) - sind(EPSILON)* sind (BETA);
z = sind (EPSILON)* cosd (BETA) *sind (LAMBDA) + cosd (EPSILON)* sind(BETA);

A = atand( y/x);
%STEP 6 b)
if x<0 ,alfa= A+180;
elseif(x>0 && y<0) ,alfa=A+360;
else alfa=A;
end

dec=asind(z);
posicao(1)=alfa;
posicao(2)= dec;
end

```

## Horizontais

```

function horizontais = horizontais(ano,mes,dia,hora,minuto,segundo,latitude,
longitude, AR, dec )

nu0=TSG(ano,mes,dia,hora,minuto,segundo);
H=nu0-longitude-AR;
H=wrapTo360(H);

h=asind(sind(latitude)*sind(dec)+cosd(latitude)*cosd(dec)*cosd(H));

Z = acosd((sind(dec) - sind(latitude) * sind(h)) / (cosd(latitude) *
cosd(h)));

    if H>180
        Zn=Z;
    else
        Zn=360-Z;
    end

horizontais (1)=h;
horizontais (2)=Zn;
end

```

## Interpolação

```
function y=interpolar(n,y1,y2,y3)
a=y2-y1;
b=y3-y2;

if abs(a)>180
    y1=y1-360;
    a=y2-y1;
else a=a;
end

if abs(b)>180
    y3=y3+360;
    b=y3-y2;
else b=b;
end
c=b-a;

y=y2+(n/2)*(a+b+n*c);

end
```

## Magnitudes

```
function mag=magnitude(estrela)
vector_estrelas
if isnumeric(estrela)==1
    I=estrela;
%Caso seja literal(string) a função strcmp procurará o nome da estrela e
%descobrirá o número correspondente
else
X=strcmp(estrela,vector_estrelas);
I=find(X);
end
mags=[3.1
0.6
1.1
1.6
1.1
1.7
1.9
2.2
1.8
2.2
2.3
2.2
0.9
2.4
```

```
1.2
0.2
1.7
1.7
0.1
-0.9
0.2
1.3
2.2
2.2
2
1.8
2.4
2.5
1.3
1.6
2.8
0.9
2.2
2
2.2
2.6
2.8
2.3
1.8
1.9
2.1
2.1
1.2
0.5
2.1
1.3
0.3
0.1
2.6
2.5
1.7
-1.6
1.2
2.2
0.1
2.9
2
5.5
-26.5
-7.0
-4.9
0.3
-2.9
0.5];
```

```
mag=mags (I) ;
end
```

## Heliocêntricas de Venus

```

function helio=heliovenus(tau32)
Matrizes_Venus;

%Cálculo de L0
mL0=MatrizL0;
L0_termoN=zeros(24,1);
for n = 1:1:24

L0_termoN(n)=mL0(n,1)*cos(mL0(n,2)+mL0(n,3)*tau32);

end

L0=sum(L0_termoN);

%Cálculo de L1
mL1=MatrizL1;
L1_termoN=zeros(12,1);
for n = 1:1:12

L1_termoN(n)=mL1(n,1)*cos(mL1(n,2)+mL1(n,3)*tau32);

end

L1=sum(L1_termoN);

%Cálculo de L2
mL2=MatrizL2;
L2_termoN=zeros(8,1);
for n = 1:1:8

L2_termoN(n)=mL2(n,1)*cos(mL2(n,2)+mL2(n,3)*tau32);

end

L2=sum(L2_termoN);

%Cálculo de L3
mL3=MatrizL3;

```

```

L3_termoN=zeros(3,1);
for n = 1:1:3

L3_termoN(n)=mL3(n,1)*cos(mL3(n,2)+mL3(n,3)*tau32);

end

L3=sum(L3_termoN);

%Cálculo de L4
mL4=MatrizL4;
L4_termoN=zeros(3,1);
for n = 1:1:3

L4_termoN(n)=mL4(n,1)*cos(mL4(n,2)+mL4(n,3)*tau32);

end

L4=sum(L4_termoN);

%Cálculo de L5

mL5=MatrizL5;

for n = 1:1:1

L5_termoN(n)=mL5(n,1)*cos(mL5(n,2)+mL5(n,3)*tau32);

end

L5=sum(L5_termoN);

%Cálculo de L

L=(L0+L1*tau32+L2*tau32^2+L3*tau32^3+L4*tau32^4+L5*tau32^5)/10^8;
Lgraus=radtodeg(L);
L=wrapTo360(Lgraus);

%LATITUDE

mB0=MatrizB0;
B0_termoN=zeros(9,1);
for n = 1:1:9

B0_termoN(n)=mB0(n,1)*cos(mB0(n,2)+mB0(n,3)*tau32);

end

```

```
B0=sum(B0_termoN);

%Cálculo de B1

mB1=MatrizB1;
B1_termoN=zeros(4,1);
for n = 1:1:4

B1_termoN(n)=mB1(n,1)*cos(mB1(n,2)+mB1(n,3)*tau32);

end

B1=sum(B1_termoN);

%Cálculo de B2

mB2=MatrizB2;
B2_termoN=zeros(4,1);
for n = 1:1:4

B2_termoN(n)=mB2(n,1)*cos(mB2(n,2)+mB2(n,3)*tau32);

end

B2=sum(B2_termoN);

%Cálculo de B3

mB3=MatrizB3;
B3_termoN=zeros(4,1);
for n = 1:1:4

B3_termoN(n)=mB3(n,1)*cos(mB3(n,2)+mB3(n,3)*tau32);

end

B3=sum(B3_termoN);

%Cálculo de B4

mB4=MatrizB4;
for n = 1:1:1
```

```

B4_termoN(n)=mB4(n,1)*cos(mB4(n,2)+mB4(n,3)*tau32);

end

B4=sum(B4_termoN);

%Cálculo de B

B=(B0+B1*tau32+B2*tau32^2+B3*tau32^3+B4*tau32^4)/10^8;
B=radtodeg(B);

%Distância ao Sol

mR0=MatrizR0;

R0_termoN=zeros(12,1);
for n = 1:1:12

R0_termoN(n)=mR0(n,1)*cos(mR0(n,2)+mR0(n,3)*tau32);

end

R0=sum(R0_termoN);

%Cálculo de R1

mR1=MatrizR1;

R1_termoN=zeros(3,1);
for n = 1:1:3

R1_termoN(n)=mR1(n,1)*cos(mR1(n,2)+mR1(n,3)*tau32);

end

R1=sum(R1_termoN);

%Cálculo de R2

mR2=MatrizR2;

```

```

R2_termoN=zeros(3,1);
for n = 1:1:3

R2_termoN(n)=mR2(n,1)*cos(mR2(n,2)+mR2(n,3)*tau32);

end

R2=sum(R2_termoN);

%Cálculo de R3

mR3=MatrizR3;
for n = 1:1:1

R3_termoN(n)=mR3(n,1)*cos(mR3(n,2)+mR3(n,3)*tau32);

end

R3=sum(R3_termoN);

%Cálculo de R4

mR4=MatrizR4;
for n = 1:1:1

R4_termoN(n)=mR4(n,1)*cos(mR4(n,2)+mR4(n,3)*tau32);

end

R4=sum(R4_termoN);

%Cálculo de R

R=(R0+R1*tau32+R2*tau32^2+R3*tau32^3+R4*tau32^4)/10^8;

helio(1)=L;
helio(2)=B;
helio(3)=R;
end

```

## Heliocêntricas de Marte

```
function helio=heliomarte(tau32)
Matrizes_Marte;

%Cálculo de L0
mL0=MatrizL0;
L0_termoN=zeros(69,1);
for n = 1:1:69

L0_termoN(n)=mL0(n,1)*cos(mL0(n,2)+mL0(n,3)*tau32);

end

L0=sum(L0_termoN);

%Cálculo de L1
mL1=MatrizL1;
L1_termoN=zeros(46,1);
for n = 1:1:46

L1_termoN(n)=mL1(n,1)*cos(mL1(n,2)+mL1(n,3)*tau32);

end

L1=sum(L1_termoN);

%Cálculo de L2
mL2=MatrizL2;
L2_termoN=zeros(33,1);
for n = 1:1:33

L2_termoN(n)=mL2(n,1)*cos(mL2(n,2)+mL2(n,3)*tau32);

end

L2=sum(L2_termoN);

%Cálculo de L3
mL3=MatrizL3;
```

```

L3_termoN=zeros(12,1);
for n = 1:1:12

L3_termoN(n)=mL3(n,1)*cos(mL3(n,2)+mL3(n,3)*tau32);

end

L3=sum(L3_termoN);

%Cálculo de L4
mL4=MatrizL4;
L4_termoN=zeros(8,1);
for n = 1:1:8

L4_termoN(n)=mL4(n,1)*cos(mL4(n,2)+mL4(n,3)*tau32);

end

L4=sum(L4_termoN);

%Cálculo de L5

mL5=MatrizL5;
L5_termoN=zeros(2,1);
for n = 1:1:2

L5_termoN(n)=mL5(n,1)*cos(mL5(n,2)+mL5(n,3)*tau32);

end

L5=sum(L5_termoN);

%Cálculo de L

L=(L0+L1*tau32+L2*tau32^2+L3*tau32^3+L4*tau32^4+L5*tau32^5)/10^8;
Lgraus=radtodeg(L);
L=wrapTo360(Lgraus);

%LATITUDE

mB0=MatrizB0;
B0_termoN=zeros(16,1);
for n = 1:1:16

B0_termoN(n)=mB0(n,1)*cos(mB0(n,2)+mB0(n,3)*tau32);

end

```

```

B0=sum(B0_termoN);

%Cálculo de B1

mB1=MatrizB1;
B1_termoN=zeros(9,1);
for n = 1:1:9

B1_termoN(n)=mB1(n,1)*cos(mB1(n,2)+mB1(n,3)*tau32);

end

B1=sum(B1_termoN);

%Cálculo de B2

mB2=MatrizB2;
B2_termoN=zeros(7,1);
for n = 1:1:7

B2_termoN(n)=mB2(n,1)*cos(mB2(n,2)+mB2(n,3)*tau32);

end

B2=sum(B2_termoN);

%Cálculo de B3

mB3=MatrizB3;
B3_termoN=zeros(4,1);
for n = 1:1:4

B3_termoN(n)=mB3(n,1)*cos(mB3(n,2)+mB3(n,3)*tau32);

end

B3=sum(B3_termoN);

%Cálculo de B4

mB4=MatrizB4;
B4_termoN=zeros(3,1);
for n = 1:1:3

```

```

B4_termoN(n)=mB4(n,1)*cos(mB4(n,2)+mB4(n,3)*tau32);

end

B4=sum(B4_termoN);

%Cálculo de B

B=(B0+B1*tau32+B2*tau32^2+B3*tau32^3+B4*tau32^4)/10^8;
B=radtodeg(B);

%Distância ao Sol

mR0=MatrizR0;

R0_termoN=zeros(45,1);
for n = 1:1:45

R0_termoN(n)=mR0(n,1)*cos(mR0(n,2)+mR0(n,3)*tau32);

end

R0=sum(R0_termoN);

%Cálculo de R1

mR1=MatrizR1;

R1_termoN=zeros(27,1);
for n = 1:1:27

R1_termoN(n)=mR1(n,1)*cos(mR1(n,2)+mR1(n,3)*tau32);

end

R1=sum(R1_termoN);

%Cálculo de R2

```

```

mR2=MatrizR2;

R2_termoN=zeros(11,1);
for n = 1:1:11

R2_termoN(n)=mR2(n,1)*cos(mR2(n,2)+mR2(n,3)*tau32);

end

R2=sum(R2_termoN);

%Cálculo de R3

mR3=MatrizR3;
R3_termoN=zeros(6,1);
for n = 1:1:6

R3_termoN(n)=mR3(n,1)*cos(mR3(n,2)+mR3(n,3)*tau32);

end

R3=sum(R3_termoN);

%Cálculo de R4

mR4=MatrizR4;
R4_termoN=zeros(4,1);
for n = 1:1:4

R4_termoN(n)=mR4(n,1)*cos(mR4(n,2)+mR4(n,3)*tau32);

end

R4=sum(R4_termoN);

%Cálculo de R

R=(R0+R1*tau32+R2*tau32^2+R3*tau32^3+R4*tau32^4)/10^8;

helio(1)=L;
helio(2)=B;
helio(3)=R;
end

```

## Heliocêntricas de Jupiter

```

function helio=heliojupiter(tau32)
Matrizes_Jupiter;

%Cálculo de L0
mL0=MatrizL0;
L0_termoN=zeros(64,1);
for n = 1:1:64

L0_termoN(n)=mL0(n,1)*cos(mL0(n,2)+mL0(n,3)*tau32);

end

L0=sum(L0_termoN);

%Cálculo de L1
mL1=MatrizL1;
L1_termoN=zeros(61,1);
for n = 1:1:61

L1_termoN(n)=mL1(n,1)*cos(mL1(n,2)+mL1(n,3)*tau32);

end

L1=sum(L1_termoN);

%Cálculo de L2
mL2=MatrizL2;
L2_termoN=zeros(57,1);
for n = 1:1:57

L2_termoN(n)=mL2(n,1)*cos(mL2(n,2)+mL2(n,3)*tau32);

end

L2=sum(L2_termoN);

%Cálculo de L3
mL3=MatrizL3;

```

```

L3_termoN=zeros(39,1);
for n = 1:1:39

L3_termoN(n)=mL3(n,1)*cos(mL3(n,2)+mL3(n,3)*tau32);

end

L3=sum(L3_termoN);

%Cálculo de L4
mL4=MatrizL4;
L4_termoN=zeros(19,1);
for n = 1:1:19

L4_termoN(n)=mL4(n,1)*cos(mL4(n,2)+mL4(n,3)*tau32);

end

L4=sum(L4_termoN);

%Cálculo de L5

mL5=MatrizL5;
L5_termoN=zeros(5,1);
for n = 1:1:5

L5_termoN(n)=mL5(n,1)*cos(mL5(n,2)+mL5(n,3)*tau32);

end

L5=sum(L5_termoN);

%Cálculo de L

L=(L0+L1*tau32+L2*tau32^2+L3*tau32^3+L4*tau32^4+L5*tau32^5)/10^8;
Lgraus=radtodeg(L);
L=wrapTo360(Lgraus);

%LATITUDE

mB0=MatrizB0;
B0_termoN=zeros(26,1);
for n = 1:1:26

B0_termoN(n)=mB0(n,1)*cos(mB0(n,2)+mB0(n,3)*tau32);

end

```

```

B0=sum(B0_termoN);

%Cálculo de B1

mB1=MatrizB1;
B1_termoN=zeros(22,1);
for n = 1:1:22

B1_termoN(n)=mB1(n,1)*cos(mB1(n,2)+mB1(n,3)*tau32);

end

B1=sum(B1_termoN);

%Cálculo de B2

mB2=MatrizB2;
B2_termoN=zeros(14,1);
for n = 1:1:14

B2_termoN(n)=mB2(n,1)*cos(mB2(n,2)+mB2(n,3)*tau32);

end

B2=sum(B2_termoN);

%Cálculo de B3

mB3=MatrizB3;
B3_termoN=zeros(9,1);
for n = 1:1:9

B3_termoN(n)=mB3(n,1)*cos(mB3(n,2)+mB3(n,3)*tau32);

end

B3=sum(B3_termoN);

%Cálculo de B4

mB4=MatrizB4;
B4_termoN=zeros(6,1);
for n = 1:1:6

```

```

B4_termoN(n)=mB4(n,1)*cos(mB4(n,2)+mB4(n,3)*tau32);

end

B4=sum(B4_termoN);

%Cálculo de B5
mB5=MatrizB5;
B5_termoN=zeros(1,1);
for n = 1:1:1

B5_termoN(n)=mB5(n,1)*cos(mB5(n,2)+mB5(n,3)*tau32);

end

B5=sum(B5_termoN);

%Cálculo de B
B=(B0+B1*tau32+B2*tau32^2+B3*tau32^3+B4*tau32^4+B5*tau32^5)/10^8;
B=radtodeg(B);

%Distância ao Sol

mR0=MatrizR0;

R0_termoN=zeros(46,1);
for n = 1:1:46

R0_termoN(n)=mR0(n,1)*cos(mR0(n,2)+mR0(n,3)*tau32);

end

R0=sum(R0_termoN);

%Cálculo de R1

mR1=MatrizR1;

R1_termoN=zeros(43,1);
for n = 1:1:43

```

```

R1_termoN(n)=mR1(n,1)*cos(mR1(n,2)+mR1(n,3)*tau32);

end

R1=sum(R1_termoN);

%Cálculo de R2
mR2=MatrizR2;

R2_termoN=zeros(36,1);
for n = 1:1:36

R2_termoN(n)=mR2(n,1)*cos(mR2(n,2)+mR2(n,3)*tau32);

end

R2=sum(R2_termoN);

%Cálculo de R3

mR3=MatrizR3;
R3_termoN=zeros(28,1);
for n = 1:1:28

R3_termoN(n)=mR3(n,1)*cos(mR3(n,2)+mR3(n,3)*tau32);

end

R3=sum(R3_termoN);

%Cálculo de R4
mR4=MatrizR4;
R4_termoN=zeros(15,1);
for n = 1:1:15

R4_termoN(n)=mR4(n,1)*cos(mR4(n,2)+mR4(n,3)*tau32);

end

R4=sum(R4_termoN);

%Cálculo de R5
mR5=MatrizR5;

```

```

R5_termoN=zeros(7,1);
for n = 1:1:7

R5_termoN(n)=mR5(n,1)*cos(mR5(n,2)+mR5(n,3)*tau32);

end

R5=sum(R5_termoN);
%Cálculo de R

R=(R0+R1*tau32+R2*tau32^2+R3*tau32^3+R4*tau32^4+R5*tau32^5)/10^8;

helio(1)=L;
helio(2)=B;
helio(3)=R;
end

```

## Heliocêntricas de Saturno

```

function helio=heliosaturno(tau32)
Matrizes_Saturno;

%Cálculo de L0
mL0=MatrizL0;
L0_termoN=zeros(63,1);
for n = 1:1:63

L0_termoN(n)=mL0(n,1)*cos(mL0(n,2)+mL0(n,3)*tau32);

end

L0=sum(L0_termoN);

%Cálculo de L1

mL1=MatrizL1;
L1_termoN=zeros(79,1);
for n = 1:1:79

L1_termoN(n)=mL1(n,1)*cos(mL1(n,2)+mL1(n,3)*tau32);

end

L1=sum(L1_termoN);

```

```

%Cálculo de L2
mL2=MatrizL2;
L2_termoN=zeros(63,1);
for n = 1:1:63

L2_termoN(n)=mL2(n,1)*cos(mL2(n,2)+mL2(n,3)*tau32);

end

L2=sum(L2_termoN);

%Cálculo de L3

mL3=MatrizL3;
L3_termoN=zeros(48,1);
for n = 1:1:48

L3_termoN(n)=mL3(n,1)*cos(mL3(n,2)+mL3(n,3)*tau32);

end

L3=sum(L3_termoN);

%Cálculo de L4
mL4=MatrizL4;
L4_termoN=zeros(27,1);
for n = 1:1:27

L4_termoN(n)=mL4(n,1)*cos(mL4(n,2)+mL4(n,3)*tau32);

end

L4=sum(L4_termoN);

%Cálculo de L5

mL5=MatrizL5;
L5_termoN=zeros(12,1);
for n = 1:1:12

L5_termoN(n)=mL5(n,1)*cos(mL5(n,2)+mL5(n,3)*tau32);

```

```

end

L5=sum(L5_termoN);

%Cálculo de L

L=(L0+L1*tau32+L2*tau32^2+L3*tau32^3+L4*tau32^4+L5*tau32^5)/10^8;
Lgraus=radtodeg(L);
L=wrapTo360(Lgraus);

%LATITUDE

mB0=MatrizB0;
B0_termoN=zeros(234,1);
for n = 1:1:34

B0_termoN(n)=mB0(n,1)*cos(mB0(n,2)+mB0(n,3)*tau32);

end

B0=sum(B0_termoN);

%Cálculo de B1

mB1=MatrizB1;
B1_termoN=zeros(32,1);
for n = 1:1:32

B1_termoN(n)=mB1(n,1)*cos(mB1(n,2)+mB1(n,3)*tau32);

end

B1=sum(B1_termoN);

%Cálculo de B2

mB2=MatrizB2;
B2_termoN=zeros(29,1);
for n = 1:1:29

B2_termoN(n)=mB2(n,1)*cos(mB2(n,2)+mB2(n,3)*tau32);

end

B2=sum(B2_termoN);

```

```

%Cálculo de B3

mB3=MatrizB3;
B3_termoN=zeros(21,1);
for n = 1:1:21

B3_termoN(n)=mB3(n,1)*cos(mB3(n,2)+mB3(n,3)*tau32);

end

B3=sum(B3_termoN);

%Cálculo de B4

mB4=MatrizB4;
B4_termoN=zeros(12,1);
for n = 1:1:12

B4_termoN(n)=mB4(n,1)*cos(mB4(n,2)+mB4(n,3)*tau32);

end

B4=sum(B4_termoN);

%Cálculo de B5

mB5=MatrizB5;
B5_termoN=zeros(2,1);
for n = 1:1:2

B5_termoN(n)=mB5(n,1)*cos(mB5(n,2)+mB5(n,3)*tau32);

end

B5=sum(B5_termoN);

%Cálculo de B

B=(B0+B1*tau32+B2*tau32^2+B3*tau32^3+B4*tau32^4+B5*tau32^5)/10^8;
B=radtodeg(B);

%Distância ao Sol

```

```

mR0=MatrizR0;

R0_termoN=zeros(44,1);
for n = 1:1:44

R0_termoN(n)=mR0(n,1)*cos(mR0(n,2)+mR0(n,3)*tau32);

end

R0=sum(R0_termoN);

%Cálculo de R1

mR1=MatrizR1;

R1_termoN=zeros(38,1);
for n = 1:1:38

R1_termoN(n)=mR1(n,1)*cos(mR1(n,2)+mR1(n,3)*tau32);

end

R1=sum(R1_termoN);

%Cálculo de R2

mR2=MatrizR2;

R2_termoN=zeros(32,1);
for n = 1:1:32

R2_termoN(n)=mR2(n,1)*cos(mR2(n,2)+mR2(n,3)*tau32);

end

R2=sum(R2_termoN);

%Cálculo de R3

mR3=MatrizR3;
R3_termoN=zeros(28,1);
for n = 1:1:28

```

```

R3_termoN(n)=mR3(n,1)*cos(mR3(n,2)+mR3(n,3)*tau32);

end

R3=sum(R3_termoN);

%Cálculo de R4
mR4=MatrizR4;
R4_termoN=zeros(23,1);
for n = 1:1:23

R4_termoN(n)=mR4(n,1)*cos(mR4(n,2)+mR4(n,3)*tau32);

end

R4=sum(R4_termoN);

%Cálculo de R5
mR5=MatrizR5;
R5_termoN=zeros(18,1);
for n = 1:1:18

R5_termoN(n)=mR5(n,1)*cos(mR5(n,2)+mR5(n,3)*tau32);

end

R5=sum(R5_termoN);
%Cálculo de R
R=(R0+R1*tau32+R2*tau32^2+R3*tau32^3+R4*tau32^4+R5*tau32^5)/10^8;

helio(1)=L;
helio(2)=B;
helio(3)=R;
end

```

## Horas Favoráveis

```

function horas=favoraveis(latitude,longitude,dia,mes,ano)
nasc_ocas=nascimentooeocaso('Sol',latitude,longitude,dia,mes,ano);
horas=crepusculos(latitude,longitude,dia,mes,ano);

nascimento=nasc_ocas(1);
ocaso=nasc_ocas(2);

%Hora dos crepusculos em horas(necessário converter os minutos)
crepcivilmat=horas(1,1)+horas(1,2)/60;
crepcivilvesp=horas(3,1)+horas(3,2)/60;

```

```

deltat_m=nascimento-crepcivilmat;
deltat_t=crepcivilvesp-ocaso;

%Manhã
inicio_m=crepcivilmat-deltat_m/2;
fim_m=crepcivilmat+deltat_m/2;

%Tarde
inicio_t=crepcivilvesp-deltat_m/2;
fim_t=crepcivilvesp+deltat_t/2;

% horas=[inicio_m ,fim_m
%         inicio_t, fim_t];
inicio_m=hours(inicio_m) ;
inicio_m.Format='hh:mm:ss';
inicio_m=round(inicio_m,'minutes');
inicio_m=datevec(inicio_m);

fim_m=hours(fim_m) ;
fim_m.Format='hh:mm:ss';
fim_m=round(fim_m,'minutes');
fim_m=datevec(fim_m);

inicio_t=hours(inicio_t) ;
inicio_t.Format='hh:mm:ss';
inicio_t=round(inicio_t,'minutes');
inicio_t=datevec(inicio_t);

fim_t=hours(fim_t) ;
fim_t.Format='hh:mm:ss';
fim_t=round(fim_t,'minutes');
fim_t=datevec(fim_t);

horas=[inicio_m; fim_m
       inicio_t ;fim_t];
horas=horas(1:4,4:5);
end

```

## Funções Auxiliares

### angle2str

```

function str = angle2str(angin, signcode, units, digits)
%Esta função é uma adaptação da função Angl2str da MathWorks.

error(nargchk(1, 4, nargin, 'struct'))

```

```

switch(nargin)
    case 1
        signcode = 'none';
        units = 'degrees';
        digits = -2;
    case 2
        units = 'degrees';
        digits = -2;
    case 3
        % When we have three input arguments, the third one could be
        % either UNITS or DIGITS (N).
        if ischar(units)
            units = checkunits(units);
            digits = -2;
        else
            digits = units;
            units = 'degrees';
        end
    case 4
        units = checkunits(units);
end

angin = ignoreComplex(angin, mfilename, 'ANGLE');

str = doAngle2str(angin, signcode, units, digits);

%-----

function strout = doAngle2str(angin, signcode, units, digits)

% Ensure that inputs are a column vector
angin = angin(:);

switch units
    case 'degrees'
        middle = formatDegrees(angin, digits);

    case 'degrees2dm'
        middle = formatDM(angin, digits);

    case 'degrees2dms'
        middle = formatDMS(angin, digits);

    case 'radians'
        middle = formatRadians(angin, digits);
end

[prefix, suffix] = buildPrefixAndSuffix(angin, signcode);

strout = buildStringMatrix(prefix, middle, suffix);

%-----

function [prefix, suffix] = buildPrefixAndSuffix(angin, signcode)

```

```

% Compute the prefix and suffix matrices.
% Note that the * character forces a space in the output

switch lower(signcode)
  case 'ns'
    prefix = [];
    suffix = '**';
    suffix = suffix(ones(size(angin)),:);
    suffix(angin > 0, 2) = 'N';
    suffix(angin < 0, 2) = 'S';

  case 'ew'
    prefix = [];
    suffix = '**';
    suffix = suffix(ones(size(angin)),:);
    suffix(angin > 0, 2) = 'E';
    suffix(angin < 0, 2) = 'W';

  case 'pm'
    prefix = ' ';
    prefix = prefix(ones(size(angin)),:);
    prefix(angin > 0) = '+';
    prefix(angin < 0) = '-';
    suffix = [];

  case 'none'
    prefix = ' ';
    prefix = prefix(ones(size(angin)),:);
    prefix(angin < 0) = '-';
    suffix = [];

  otherwise
    eid = sprintf('map:%s:unknownFormatString', mfilename);
    error(eid, '%s', 'Unrecognized SIGNCODE string')

end

%-----
function strout = buildStringMatrix(prefix, middle, suffix)

strout = [prefix middle suffix];

% Right justify each row of the output matrix. This places
% all extra spaces in the leading position. Then strip these
% lead zeros. Left justifying and then a DEBLANK call will
% not ensure that all strings line up. LEADBLNK only strips
% leading blanks which appear in all rows of a string matrix,
% thereby not messing up any right justification of the string matrix.
strout = strjust(strout);
strout = leadblnk(strout, ' ');

% Replace the hold characters with a space
strout(strout == '**') = ' ';

% Pad matrix with a space at front and back to avoid touching the map frame

```

```

strout = [ char(32*ones(size(strout,1),1)) strout
char(32*ones(size(strout,1),1))];

%-----

function units = checkunits(units)

% All comparisons here are case-insensitive. Error on 'dms' or 'dm'
% because the old encodings are no longer supported. Then check for an
% exact match to 'degrees2dms' or 'degrees2dm'. Then check for a full
% or partial match to 'degrees' or 'radians'.

if strcmpi(units, 'dms')
    error('map:angl2str:obsoleteDMS', ...
        ['DMS angle encoding is obsolete.\n', ...
        'To format output in degrees-minutes-seconds, set UNITS to
        ''degrees2dms''.'])
elseif strcmpi(units, 'dm')
    error('map:angl2str:obsoleteDM', ...
        ['DM angle encoding is obsolete.\n', ...
        'To format output in degrees-minutes, set UNITS to ''degrees2dm''.'])
elseif any(strcmpi(units, {'degrees2dms','degrees2dm'}))
    units = lower(units);
else
    units = checkangleunits(units);
end

%-----

function middle = formatDMS(angin, digits)

% Tony's trick
onesize = ones(size(angin));

spacestr = '*';
spacestr = spacestr(onesize);

quotestr = ''';
quotestr = quotestr(onesize);

dblquotestr = '"';
dblquotestr = dblquotestr(onesize);

% Convert to matrix format. Work with only the absolute values
% since the sign is taken care of by the prefix string
dms = degrees2dms(abs(angin));

d = dms(:,1);
m = dms(:,2);
s = dms(:,3);

% Avoid the equivalent of "m = -0;" because it may cause an
% unnecessary '-' sign in the output of num2str.

m(m == 0) = abs(m(m == 0));

```

```

if digits >= 2
    % We are rounding seconds away completely;
    % make sure to go to the nearest minute.
    m = m + round(s/60);
    s = 0;
else
    s = roundn(s,digits);
end

% Degrees, minutes and seconds

d_str = num2str(d,'%4g');          % Convert degrees to a string
m_str = num2str(m,'%02g');        % Convert minutes to a string
s_str = num2str(s,formatstr(digits)); % Convert seconds to a padded string

% Construct the display string

middle = [leadblnk(d_str) degsymbol(size(angin)) ...
         spacestr m_str quotestr spacestr s_str dblquotestr];

%-----

function middle = formatDM(angin, digits)

% Tony's trick
onesize = ones(size(angin));

spacestr = '*';
spacestr = spacestr(onesize);

quotestr = '''';
quotestr = quotestr(onesize);

% Convert to matrix format. Work with only the absolute values
% since the sign is taken care of by the prefix string

dm = degrees2dm(abs(angin));
d = dm(:,1);
m = dm(:,2);

if digits >= 2
    % We are rounding minutes away completely;
    % make sure to go to the nearest degree.
    d = d + round(m/60);
    m = 0;
else
    m = roundn(m,digits);
end

d_str = num2str(d,'%4g');
m_str = num2str(m,formatstr(digits));

middle = [leadblnk(d_str) degsymbol(size(angin)) spacestr m_str quotestr];

```

```

%-----
function fmt = formatstr(digits)
% Construct the format string for converting seconds (or minutes)

rightdigits = abs(min(digits,0));
if rightdigits > 0
    totaldigits = 3 + rightdigits;
else
    totaldigits = 2 + rightdigits;
end

fmt = ['%0',num2str(totaldigits),'.',num2str(rightdigits),'f'];

%-----

function middle = formatDegrees(angin, digits)

angin = roundn(angin,digits);
formatstr = ['%20.',num2str(abs(min(digits,0))),'f'];
str = num2str(abs(angin),formatstr);
middle = [leadblnk(str) degsymbol(size(angin))];

%-----

function middle = formatRadians(angin, digits)

angin = roundn(angin,digits);
formatstr = ['%20.',num2str(abs(min(digits,0))),'f'];
str = num2str(abs(angin),formatstr);
unitsymbol = '*R';
unitsymbol = unitsymbol(ones(size(angin)),:);
middle = [leadblnk(str) unitsymbol];

%-----

function unitsymbol = degsymbol(sizein)

unitsymbol = char(176);
unitsymbol = unitsymbol(ones(sizein),:);

```

### ascensaoreta0

```

function AR=ascensaoreta0(astro, dia, mes, ano)
vector_estrelas;
X=strcmp(astro,vetor_estrelas);
I=find(X);

if I == 60
    dados = Sol(ano,mes,dia,0,0,0);

```

```

elseif I==61
    dados=Lua(ano,mes,dia,0,0,0);
elseif I==62
    dados=Venus(ano,mes,dia,0,0,0);
elseif I==63
    dados=Marte(ano,mes,dia,0,0,0);
elseif I==64
    dados=Jupiter(ano,mes,dia,0,0,0);
elseif I==65
    dados=Saturno(ano,mes,dia,0,0,0);
else
    dados=aparente(astro,dia,mes,ano,0);

%Subtileza: Se o astro não for
%encontrado significa que o valor que veio era numérico e corresponde
%ao a-6 da GUI que bate certo no vetor estrelas. Este pode ser
%diretamente usado na função aparente, pois esta aceita o valor
%numérico correspondente à estrela pretendida.
end
AR=dados(1);
end

```

## declinacao0

```

function dec=declinacao0(astro, dia, mes, ano)
%Esta Função necessária para o nasc/ocaso
%calcula a declinação para as 0h de um dado dia

vector_estrelas;
X=strcmp(astro, vetor_estrelas);
I=find(X);

if I == 60
    dados = Sol(ano,mes,dia,0,0,0);
elseif I==61
    dados=Lua(ano,mes,dia,0,0,0);
elseif I==62
    dados=Venus(ano,mes,dia,0,0,0);
elseif I==63
    dados=Marte(ano,mes,dia,0,0,0);
elseif I==64
    dados=Jupiter(ano,mes,dia,0,0,0);
elseif I==65
    dados=Saturno(ano,mes,dia,0,0,0);
else
    dados=aparente(astro,dia,mes,ano,0); %Subtileza: Se o astro não for
%encontrado significa que o valor que veio era numérico e corresponde
%ao a-6 da GUI que bate certo no vetor estrelas. Este pode ser
%diretamente usado na função aparente, pois esta aceita o valor
%numérico correspondente à estrela pretendida.
end

```

```
dec=dados(2);  
end
```

### deg2hours

```
function horas=deg2hours(graus)  
  
horas_h =graus*24/360;  
h_h=floor(horas_h);  
  
minutos_h=(horas_h - h_h)*60;  
minutos=floor(minutos_h);  
  
segundos=(minutos_h-minutos)*60;  
  
horas(1)=h_h;  
horas(2)=minutos;  
horas(3)=segundos;  
end
```

### Escolha de Estrelas

```
% ESCOLHA DAS ESTRELAS A DISPOR %  
  
%estrela 1  
estrelas_1=find(azimutes>0 & azimutes<40);  
if isempty(estrelas_1)  
    estrelas_1=find(azimutes>0 & azimutes<50);  
end  
if isempty(estrelas_1)  
    estrelas_1=find(azimutes>0 & azimutes<60);  
end  
  
mags=magnitude(estrelas_1);  
magmin=min(mags);  
maisbrilhante= mags==magmin;  
  
estrela_1=estrelas_1(maisbrilhante);  
estrela_1=estrela_1(1);  
%estrela 2  
Inicio_az_2= azimutes(estrela_1)+30;  
fim_az_2=Inicio_az_2+30;  
  
estrelas_2=find(azimutes>Inicio_az_2 & azimutes<fim_az_2);  
if isempty(estrelas_2)  
    estrelas_2=find(azimutes>Inicio_az_2 & azimutes<fim_az_2+10);
```

```

end
if isempty(estrelas_2)
    estrelas_2=find(azimutes>Inicio_az_2 & azimutes<fim_az_2+20);
end
mags=magnitude(estrelas_2);
magmin=min(mags);
maisbrilhante= mags==magmin;

estrela_2=estrelas_2(maisbrilhante);
estrela_2=estrela_2(1);
%estrela 3
Inicio_az_3= azimutes(estrela_2)+30;
fim_az_3=Inicio_az_3+30;

estrelas_3=find(azimutes>Inicio_az_3 & azimutes<fim_az_3);
if isempty(estrelas_3)
    estrelas_3=find(azimutes>Inicio_az_3 & azimutes<fim_az_3+10);
end
if isempty(estrelas_3)
    estrelas_3=find(azimutes>Inicio_az_3 & azimutes<fim_az_3+20);
end

mags=magnitude(estrelas_3);
magmin=min(mags);
maisbrilhante= mags==magmin;

estrela_3=estrelas_3(maisbrilhante);
estrela_3=estrela_3(1);
%estrela 4 (do meio)
Inicio_az_4= 160;
fim_az_4=200;

estrelas_4=find(azimutes>Inicio_az_4 & azimutes<fim_az_4);
if isempty(estrelas_4)
    estrelas_4=find(azimutes>Inicio_az_4 & azimutes<fim_az_4+10);
end
if isempty(estrelas_4)
    estrelas_4=find(azimutes>Inicio_az_4 & azimutes<fim_az_4+20);
end

mags=magnitude(estrelas_4);
magmin=min(mags);
maisbrilhante= mags==magmin;

estrela_4=estrelas_4(maisbrilhante);
estrela_4=estrela_4(1);
%estrela 5
Inicio_az_5= azimutes(estrela_4)+30;
fim_az_5=Inicio_az_5+30;

estrelas_5=find(azimutes>Inicio_az_5 & azimutes<fim_az_5);
if isempty(estrelas_5)
    estrelas_5=find(azimutes>Inicio_az_5 & azimutes<fim_az_5+10);
end
if isempty(estrelas_5)

```

```

    estrelas_5=find(azimutes>Inicio_az_5 & azimutes<fim_az_5+20);
end
mags=magnitude(estrelas_5);
magmin=min(mags);
maisbrilhante= mags==magmin;

estrela_5=estrelas_5(maisbrilhante);
estrela_5=estrela_5(1);
%estrela 6
Inicio_az_6= azimutes(estrela_5)+30;
fim_az_6=Inicio_az_6+30;

estrelas_6=find(azimutes>Inicio_az_6 & azimutes<fim_az_6);
if isempty(estrelas_6)
    estrelas_6=find(azimutes>Inicio_az_6 & azimutes<fim_az_6+10);
end
if isempty(estrelas_6)
    estrelas_6=find(azimutes>Inicio_az_6 & azimutes<fim_az_6+20);
end
mags=magnitude(estrelas_6);
magmin=min(mags);
maisbrilhante= mags==magmin;

estrela_6=estrelas_6(maisbrilhante);
estrela_6=estrela_6(1);

%estrela 7
Inicio_az_7= azimutes(estrela_6)+30;
fim_az_7=Inicio_az_7+30;

estrelas_7=find(azimutes>Inicio_az_7 & azimutes<fim_az_7);
if isempty(estrelas_7)
    estrelas_7=find(azimutes>Inicio_az_7 & azimutes<fim_az_7+10);
end
if isempty(estrelas_7)
    estrelas_7=find(azimutes>Inicio_az_7 & azimutes<fim_az_7+20);
end
mags=magnitude(estrelas_7);
magmin=min(mags);
maisbrilhante= mags==magmin;

estrela_7=estrelas_7(maisbrilhante);
estrela_7=estrela_7(1);

estrela_grafico(alturas(estrela_1),azimutes(estrela_1));
estrela_grafico(alturas(estrela_2),azimutes(estrela_2));
estrela_grafico(alturas(estrela_3),azimutes(estrela_3));
estrela_grafico(alturas(estrela_4),azimutes(estrela_4));
estrela_grafico(alturas(estrela_5),azimutes(estrela_5));
estrela_grafico(alturas(estrela_6),azimutes(estrela_6));
estrela_grafico(alturas(estrela_7),azimutes(estrela_7));

```

## Matrizes Lua

```
%Matrizes da Lua
%Neste ficheiro estão definidas as matrizes:
%LR_DMMF
```

```
LR_MatrizDMMF = [
    0  0  1  0
    2  0 -1  0
    2  0  0  0
    0  0  2  0
    0  1  0  0
    0  0  0  2
    2  0 -2  0
    2 -1 -1  0
    2  0  1  0
    2 -1  0  0
    0  1 -1  0
    1  0  0  0
    0  1  1  0
    2  0  0 -2
    0  0  1  2
    0  0  1 -2
    4  0 -1  0
    0  0  3  0
    4  0 -2  0
    2  1 -1  0
    2  1  0  0
    1  0 -1  0
    1  1  0  0
    2 -1  1  0
    2  0  2  0
    4  0  0  0
    2  0 -3  0
    0  1 -2  0
    2  0 -1  2
    2 -1 -2  0
    1  0  1  0
    2 -2  0  0 %muda pagina
    0  1  2  0
    0  2  0  0
    2 -2 -1  0
    2  0  1 -2
    2  0  0  2
    4 -1 -1  0
    0  0  2  2
    3  0 -1  0
    2  1  1  0
    4 -1 -2  0
    0  2 -1  0
    2  2 -1  0
```

```

2  1  -2  0
2  -1  0  -2
4  0  1  0
0  0  4  0
4  -1  0  0
1  0  -2  0
2  1  0  -2
0  0  2  -2
1  1  1  0
3  0  -2  0
4  0  -3  0
2  -1  2  0
0  2  1  0
1  1  -1  0
2  0  3  0
2  0  -1  -2];

```

```

lat_MatrizDMMF=[
0  0  0  1
0  0  1  1
0  0  1  -1
2  0  0  -1
2  0  -1  1
2  0  -1  -1
2  0  0  1
0  0  2  1
2  0  1  -1
0  0  2  -1
2  -1  0  -1
2  0  -2  -1
2  0  1  1
2  1  0  -1
2  -1  -1  1
2  -1  0  1
2  -1  -1  -1
0  1  -1  -1
4  0  -1  -1
0  1  0  1
0  0  0  3
0  1  -1  1
1  0  0  1
0  1  1  1
0  1  1  -1
0  1  0  -1
1  0  0  -1
0  0  3  1
4  0  0  -1
4  0  -1  1
0  0  1  -3
4  0  -2  1
2  0  0  -3
2  0  2  -1
2  -1  1  -1
2  0  -2  1
0  0  3  -1

```

%MUDA COLUNA

```
2 0 2 1
2 0 -3 -1
2 1 -1 1
2 1 0 1
4 0 0 1
2 -1 1 1
2 -2 0 -1
0 0 1 3
2 1 1 -1
1 1 0 -1
1 1 0 1
0 1 -2 -1
2 1 -1 -1
1 0 1 1
2 -1 -2 -1
0 1 2 1
4 0 -2 -1
4 -1 -1 -1
1 0 1 -1
4 0 1 -1
1 0 -1 -1
4 -1 0 -1
2 -2 0 1];
```

```
C_L = [
6288774
1274027
658314
213618
-185116
-114332
58793
57066
53322
45758
-40923
-34720
-30383
15327
-12528
10980
10675
10034
8548
-7888
-6766
-5163
4987
4036
3994
3861
3665
-2689
```

```
-2602
2390
-2348
2236      %muda pagina
-2120
-2069
2048
-1773
-1595
1215
-1110
-892
-810
759
-713
-700
691
596
549
537
520
-487
-399
-381
351
-340
330
327
-323
299
294
0];

C_r = [
-20905355
-3699111
-2955968
-569925
48888
-3149
246158
-152138
-170733
-204586
-129620
108743
104755
10321
0
79661
-34782
-23210
-21636
24208
30824
```

```
-8379
-16675
-12831
-10445
-11650
14403
-7003
0
10056
6322
-9884      %Muda coluna
5751
0
-4950
4130
0
-3958
0
3258
2616
-1897
-2117
2354
0
0
-1423
-1117
-1571
-1739
0
-4421
0
0
0
0
1165
0
0
8752];

C_b = [
5128122
280602
277693
173237
55413
46271
32573
17198
9266
8822
8216
4324
4200
```

-3359  
2463  
2211  
2065  
-1870  
1828  
-1794  
-1749  
-1565  
-1491  
-1475  
-1410  
-1344  
-1335  
1107  
1021  
833  
777  
671  
607  
596  
491  
-451  
439  
422  
421  
-366  
-351  
331  
315  
302  
-283  
-229  
223  
223  
-220  
-220  
-185  
181  
-177  
176  
166  
-164  
132  
-119  
115  
107];

%Muda Coluna

## Matrizes Venus

```

%Matrizes de Vénus

%Matriz L0
MatrizL0 = [317614667  0      0
            1353968   5.5931332 10213.2855462
            89892    5.30650   20426.57109
            5477     4.4163    7860.4194
            3456     2.6996    11790.6291
            2372     2.9938    3930.2097
            1664     4.2502    1577.3435
            1438     4.1575    9683.5946
            1317     5.1867    26.2983
            1201     6.1536    30639.8566
            769      0.816     9437.763
            761      1.950     529.691
            708      1.065     775.523
            585      3.998     191.448
            500      4.123     15720.839
            429      3.586     19367.189
            327      5.677     5507.553
            326      4.591     10404.734
            232      3.163     9153.904
            180      4.653     1109.379
            155      5.570     19651.048
            128      4.226     20.775
            128      0.962     5661.332
            106      1.537     801.821  ] ;

MatrizL1 = ...
[1021352943053.0  0      0
 95708           2.46424 10213.28555
14445            0.51625 20426.57109
213             1.795   30639.857
174             2.655   26.298
152             6.106   1577.344
82              5.70    191.45
70             2.68    9437.76
52             3.60    775.52
38             1.03    529.69
30             1.25    5507.55
25            6.11    10404.73  ];

MatrizL2= ...
[54127  0      0
 3891  0.3451 10213.2855
1338  2.0201 20426.5711
 24   2.05   26.30
 19   3.54   30639.86
 10   3.97   775.52
  7   1.52   1577.34
  6   1.00   191.45  ];
    
```

MatrizL3 = ...

```
[136 4.804 10213.286
 78 3.67 20426.57
 26 0 0 ];
```

MatrizL4 =...

```
[ 114 3.1416 0
 3 5.21 20426.57
 2 2.51 10213.29 ];
```

MatrizL5 =...

```
[1 3.14 0] ;
```

MatrizB0=...

```
[ 5923638 0.2670278 10213.2855462
40108 1.14737 20426.57109
32815 3.14159 0
1011 1.0895 30639.8566
149 6.254 18073.705
138 0.860 1577.344
130 3.672 9437.763
120 3.705 2352.866
108 4.539 22003.915 ];
```

MatrizB1=...

```
[ 513348 1.803643 10213.285546
4380 3.3862 20426.5711
199 0 0
197 2.530 30639.857 ];
```

MatrizB2=...

```
[22378 3.38509 10213.28555
282 0 0
173 5.256 20426.571
27 3.87 30639.86 ];
```

```
MatrizB3=...
[647  4.992  10213.286
 20   3.14   0
 6    0.77   20426.57
 3    5.44   30639.86  ];

MatrizB4=...
[14  0.32  10213.29 ] ;

MatrizR0=...
[72334821  0          0
 489824    4.021518  10213.285546
 1658      4.9021    20426.5711
 1632      2.8455    7860.4194
 1378      1.1285    11790.6291
 498       2.587     9683.595
 374       1.423     3930.210
 264       5.529     9437.763
 237       2.551     15720.839
 222       2.013     19367.189
 126       2.728     1577.344
 119       3.020     10404.734 ] ;

MatrizR1=...
[34551  0.89199  10213.28555
 234    1.772    20426.571
 234    3.142    0    ];

MatrizR2=...
[1407  5.0637  10213.2855
 16    5.47    20426.57
 13    0        0    ] ;

MatrizR3=...
[50  3.22  10213.29 ] ;

MatrizR4=...
[1  0.92  10213.29 ] ;
```

## Matrizes Marte

```
%Matrizes de Vénus
```

```
%Matriz L0
```

```
MatrizL0 = ...
```

```
[620347712    0          0
18656368     5.05037100 3340.61242670
1108217      5.4009984   6681.2248534
91798        5.75479     10021.83728
27745        5.97050     3.52312
12316        0.84956     2810.92146
10610        2.93959     2281.23050
8927         4.1570     0.0173
8716         6.1101     13362.4497
7775         3.3397     5621.8429
6798         0.3646     398.1490
4161         0.2281     2942.4634
3575         1.6619     2544.3144
3075         0.8570     191.4483
2938         6.0789     0.0673
2628         0.6481     3337.0893
2580         0.0300     3344.1355
2389         5.0390     796.2980
1799         0.6563     529.6910
1546         2.9158     1751.5395
1528         1.1498     6151.5339
1286         3.0680     2146.1654
1264         3.6228     5092.1520
1025         3.6933     8962.4553
892          0.183     16703.062
859          2.401     2914.014
833          4.495     3340.630
833          2.464     3340.595
749          3.822     155.420
724          0.675     3738.761
713          3.663     1059.382
655          0.489     3127.313
636          2.922     8432.764
553          4.475     1748.016
550          3.810     0.980
472          3.625     1194.447
426          0.554     6283.076
415          0.497     213.299
312          0.999     6677.702
307          0.381     6684.748
302          4.486     3532.061
299          2.783     6254.627
293          4.221     20.775
284          5.769     3149.164
281          5.882     1349.867
```

```

274      0.542      3340.545
274      0.134      3340.680
239      5.372      4136.910
236      5.755      3333.499
231      1.282      3870.303
221      3.505      382.897
204      2.821      1221.849
193      3.357      3.590
189      1.491      9492.146
179      1.006      951.718
174      2.414      553.569
172      0.439      5486.778
160      3.949      4562.461
144      1.419      135.065
140      3.326      2700.715
138      4.301      7.114
131      4.045      12303.068
128      2.208      1592.596
128      1.807      5088.629
117      3.128      7903.073
113      3.701      1589.073
110      1.052      242.729
105      0.785      8827.390
100      3.243      11773.377 ];

```

MatrizI1 = ...

```

[ 334085627474.0  0      0
1458227      3.6042605  3340.6124267
164901      3.926313   6681.224853
19963      4.26594   10021.83728
3452      4.7321   3.5231
2485      4.6128   13362.4497
842      4.459   2281.230
538      5.016   398.149
521      4.994   3344.136
433      2.561   191.448
430      5.316   155.420
382      3.539   796.298
314      4.963   16703.062
283      3.160   2544.314
206      4.569   2146.165
169      1.329   3337.089
158      4.185   1751.540
134      2.233   0.980
134      5.974   1748.016
118      6.024   6151.534
117      2.213   1059.382
114      2.129   1194.447
114      5.428   3738.761
91      1.10   1349.87
85      3.91   553.57
83      5.30   6684.75
81      4.43   529.69
80      2.25   8962.46
73      2.50   951.72

```

73	5.84	242.73
71	3.86	2914.01
68	5.02	382.90
65	1.02	3340.60
65	3.05	3340.63
62	4.15	3149.16
57	3.89	4136.91
48	4.87	213.30
48	1.18	3333.50
47	1.31	3185.19
41	0.71	1592.60
40	2.73	7.11
40	5.32	20043.67
33	5.41	6283.08
28	0.05	9492.15
27	3.89	1221.85
27	5.11	2700.72 ] ;

MatrizL2= ...

[58016	2.04979	3340.61243		
	54188	0	0	
	13908	2.45742	6681.22485	
	2465	2.8000	10021.8373	
	398	3.141	13362.450	
	222	3.194	3.523	
	121	0.543	155.420	
	62	3.49	16703.06	
	54	3.54	3344.14	
	34	6.00	2281.23	
	32	4.14	191.45	
	30	2.00	796.30	
	23	4.33	242.73	
	22	3.45	398.15	
	20	5.42	553.57	
	16	0.66	0.98	
	16	6.11	2146.17	
	16	1.22	1748.02	
	15	6.10	3185.19	
	14	4.02	951.72	
	14	2.62	1349.87	
	13	0.60	1194.45	
	12	3.86	6684.75	
	11	4.72	2544.31	
	10	0.25	382.90	
	9	0.68	1059.38	
	9	3.83	20043.67	
	9	3.88	3738.76	
	8	5.46	1751.54	
	7	2.58	3149.16	
	7	2.38	4136.91	
	6	5.48	1592.60	
	6	2.34	3097.88 ] ;	

MatrizL3 = ...

```
[ 1482  0.4443  3340.6124
      662  0.885  6681.225
      188  1.288  10021.837
      41   1.65  13362.45
      26   0     0
      23   2.05  155.42
      10   1.58  3.52
      8    2.00  16703.06
      5    2.82  242.73
      4    2.02  3344.14
      3    4.59  3185.19
      3    0.65  553.57  ];
```

```
MatrizL4 =...
  [ 114  3.1416  0
      29  5.64  6681.22
      24  5.14  3340.61
      11  6.03  10021.84
      3   0.13  13362.45
      3   3.56  155.42
      1   0.49  16703.06
      1   1.32  242.73  ];
```

```
MatrizL5 =...
  [ 1  3.14  0
      1  4.04  6681.22  ];
```

```
MatrizB0=...
  [ 3197135  3.7683204  3340.6124267
      298033  4.106170  6681.224853
      289105  0         0
      31366   4.44651  10021.83728
      3484    4.7881   13362.4497
      443     5.026   3344.136
      443     5.652   3337.089
      399     5.131   16703.062
      293     3.793   2281.230
      182     6.136   6151.534
      163     4.264   529.691
      160     2.232   1059.382
      149     2.165   5621.843
      143     1.182   3340.595
      143     3.213   3340.630
      139     2.418   8962.455  ];
```

```
MatrizB1=...
  [ 350069  5.368478  3340.612427
      14116  3.14159  0
```

```

9671 5.4788 6681.2249
1472 3.2021 10021.8373
426 3.408 13362.450
102 0.776 3337.089
79 3.72 16703.06
33 3.46 5621.84
26 2.48 2281.23 ];

```

MatrizB2=...

```

[ 16727 0.60221 3340.61243
4987 4.1416 0
302 3.559 6681.225
26 1.90 13362.45
21 0.92 10021.84
12 2.24 3337.09
8 2.25 16703.06 ] ;

```

MatrizB3=...

```

[ 607 1.981 3340.612
43 0 0
14 1.80 6681.22
3 3.45 10021.84 ] ;

```

MatrizB4=...

```

[ 13 0 0
11 3.46 3340.61
1 0.50 6681.22 ] ;

```

MatrizR0=...

```

[ 153033488 0 0
14184953 3.47971284 3340.61242670
660776 3.817834 6681.224853
46179 4.15595 10021.83728
8110 5.5596 2810.9215
7485 1.7724 5621.8429
5523 1.3644 2281.2305
3825 4.4941 13362.4497
2484 4.9255 2942.4634
2307 0.0908 2544.3144
1999 5.3606 3337.0893

```

1960	4.7425	3344.1355
1167	2.1126	5092.1520
1103	5.0091	398.1490
992	5.839	6151.534
899	4.408	529.691
807	2.102	1059.382
798	3.448	796.298
741	1.499	2146.165
726	1.245	8432.764
692	2.134	8962.455
633	0.894	3340.595
633	2.924	3340.630
630	1.287	1751.540
574	0.829	2914.014
526	5.383	3738.761
473	5.199	3127.313
348	4.832	16703.062
284	2.907	3532.061
280	5.257	6283.076
276	1.218	6254.627
275	2.908	1748.016
270	3.764	5884.927
239	2.037	1194.447
234	5.105	5486.778
228	3.255	6872.673
223	4.199	3149.164
219	5.583	191.448
208	5.255	3340.545
208	4.846	3340.680
186	5.699	6677.702
183	5.081	6684.748
179	4.184	3333.499
176	5.953	3870.303
164	3.799	4136.910 ] ;

MatrizR1=...

[1107433	2.0325052	3340.6124267
103176	2.370718	6681.224853
12877	0	0
10816	2.70888	10021.83728
1195	3.0470	13362.4497
439	2.888	2281.230
396	3.423	3344.136
183	1.584	2544.314
136	3.385	16703.062
128	6.043	3337.089
128	0.630	1059.382
127	1.954	796.298
118	2.998	2146.165
88	3.42	398.15
83	3.86	3738.76
76	4.45	6151.53
72	2.76	529.69

```

67      2.55      1751.54
66      4.41      1748.02
58      0.54      1194.45
54      0.68      8962.46
51      3.73      6684.75
49      5.73      3340.60
49      1.48      3340.63
48      2.58      3149.16
48      2.29      2914.01
39      2.32      4136.91 ] ;

```

MatrizR2=...

```

[44242  0.47931  3340.61243
      8138  0.8700  6681.2249
      1275  1.2259  10021.8373
      187   1.573  13362.450
      52    3.14    0
      41    1.97   3344.14
      27    1.92  16703.06
      18    4.43   2281.23
      12    4.53   3185.19
      10    5.39   1059.38
      10    0.42   796.30 ] ;

```

MatrizR3=...

```

[ 1113  5.1499  3340.6124
      424  5.613  6681.225
      100  5.997  10021.837
      20   0.08  13362.45
      5    3.14    0
      3    0.43  16703.06 ] ;

```

MatrizR4=...

```

[20  3.58  3340.61
      16  4.05  6681.22
      6   4.46  10021.84
      2   4.84  13362.45 ] ;

```

## Matrizes Jupiter

```
%Matrizes de Júpiter
```

```
%Matriz L0
```

```
MatrizL0 = [ 59954691  0          0
            9695899  5.0619179  529.6909651
```

573610	1.444062	7.113547
306389	5.417347	1059.381930
97178	4.14265	632.78374
72903	3.64043	522.57742
64264	3.41145	103.09277
39806	2.29377	419.48464
38858	1.27232	316.39187
27965	1.78455	536.80451
13590	5.77481	1589.07290
8769	3.6300	949.1756
8246	3.5823	206.1855
7368	5.0810	735.8765
6263	0.0250	213.2991
6114	4.5132	1162.4747
5305	4.1863	1052.2684
5305	1.3067	14.2271
4905	1.3208	110.2063
4647	4.6996	3.9322
3045	4.3168	426.5982
2610	1.5667	846.0828
2028	1.0638	3.1814
1921	0.9717	639.8973
1765	2.1415	1066.4955
1723	3.8804	1265.5675
1633	3.5820	515.4639
1432	4.2968	625.6702
973	4.098	95.979
884	2.437	412.371
733	6.085	838.969
731	3.806	1581.959
709	1.293	742.990
692	6.134	2118.764
614	4.109	1478.867
582	4.540	309.278
495	3.756	323.505
441	2.958	454.909
417	1.036	2.488
390	4.897	1692.166
376	4.703	1368.660
341	5.715	533.623
330	4.740	0.048
262	1.877	0.963
261	0.820	380.128
257	3.724	199.072
244	5.220	728.763
235	1.227	909.819
220	1.651	543.918
207	1.855	525.759
202	1.807	1375.774
197	5.293	1155.361
175	3.730	942.062
175	3.226	1898.351
175	5.910	956.289
158	4.365	1795.258
151	3.906	74.782

```

149      4.377      1685.052
141      3.136      491.558
138      1.318      1169.588
131      4.169      1045.155
117      2.500      1596.186
117      3.389      0.521
106      4.554      526.510 ] ;

```

MatrizL1 = ...

```

[ 52993480757.0  0      0
489741      4.220667      529.690965
228919      6.026475      7.113547
27655       4.57266      1059.38193
20721       5.45939      522.57742
12106       0.16986      536.80451
6068        4.4242      103.0928
5434        3.9848      419.4846
4238        5.8901      14.2271
2212        5.2677      206.1855
1746        4.9267      1589.0729
1296        5.5513      3.1814
1173        5.8565      1052.2684
1163        0.5145      3.9322
1099        5.3070      515.4639
1007        0.4648      735.8765
1004        3.1504      426.5982
848         5.758      110.206
827         4.803      213.299
816         0.586      1066.495
725         5.518      639.897
568         5.989      625.670
474         4.132      412.371
413         5.737      95.979
345         4.242      632.784
336         3.732      1162.475
234         4.035      949.176
234         6.243      309.278
199         1.505      838.969
195         2.219      323.505
187         6.086      742.990
184         6.280      543.918
171         5.417      199.072
131         0.626      728.763
115         0.680      846.083
115         5.286      2118.764
108         4.493      956.289
80          5.82      1045.15
72          5.34      942.06
70          5.97      532.87
67          5.73      21.34
66          0.13      526.51
65          6.09      1581.96
59          0.59      1155.36
58          0.99      1596.19

```

```

57          5.97          1169.59
57          1.41          533.62
55          5.43          10.29
52          5.73          117.32
52          0.23          1368.66
50          6.08          525.76
47          3.63          1478.87
47          0.51          1265.57
40          4.16          1692.17
34          0.10          302.16
33          5.04          220.41
32          5.37          508.35
29          5.42          1272.68
29          3.36          4.67
29          0.76          88.87
25          1.61          831.86 ];

```

MatrizL2= ...

```

[ 47234  4.32148  7.11355
38966  0          0
30629  2.93021   529.69097
3189   1.0550    522.5774
2729   4.8455    536.8045
2723   3.4141    1059.3819
1721   4.1873    14.2271
383    5.768     419.485
378    0.760     515.464
367    6.055     103.093
337    3.786     3.181
308    0.694     206.186
218    3.814     1589.073
199    5.340     1066.495
197    2.484     3.932
156    1.406     1052.268
146    3.814     639.897
142    1.634     426.598
130    5.837     412.371
117    1.414     625.670
97     4.03      110.21
91     1.11      95.98
87     2.52      632.78
79     4.64      543.92
72     2.22      735.88
58     0.83      199.07
57     3.12      213.30
49     1.67      309.28
40     4.02      21.34
40     0.62      323.51
36     2.33      728.76
29     3.61      10.29
28     3.24      838.97
26     4.50      742.99
26     2.51      1162.47
25     1.22      1045.15
24     3.01      956.29

```

```

19      4.29      532.87
18      0.81      508.35
17      4.20      2118.76
17      1.83      526.51
15      5.81      1596.19
15      0.68      942.06
15      4.00      117.32
14      5.95      316.39
14      1.80      302.16
13      2.52      88.87
13      4.37      1169.59
11      4.44      525.76
10      1.72      1581.96
9       2.18      1155.36
9       3.29      220.41
9       3.32      831.86
8       5.76      846.08
8       2.71      533.62
7       2.18      1265.57
6       0.50      949.18 ] ;

```

MatrizL3 = ...

```
[ 6502  2.5986  7.1135
```

```

1357  1.3464  529.6910
471   2.475   14.227
417   3.245   536.805
353   2.974   522.577
155   2.076   1059.382
87    2.51    515.46
44    0       0
34    3.83    1066.50
28    2.45    206.19
24    1.28    412.37
23    2.98    543.92
20    2.10    639.90
20    1.40    419.48
19    1.59    103.09
17    2.30    21.34
17    2.60    1589.07
16    3.15    625.67
16    3.36    1052.27
13    2.76    95.98
13    2.54    199.07
13    6.27    426.60
9     1.76    10.29
9     2.27    110.21
7     3.43    309.28
7     4.04    728.76
6     2.52    508.35
5     2.91    1045.15
5     5.25    323.51
4     4.30    88.87
4     3.52    302.16

```

```

4      4.09      735.88
3      1.43      956.29
3      4.36      1596.19
3      1.25      213.30
3      5.02      838.97
3      2.24      117.32
2      2.90      742.99
2      2.36      942.06 ] ;

```

MatrizL4 =...

```
[ 669  0.853  7.114
```

```

114  3.142  0
100  0.743  14.227
50   1.65   536.80
44   5.82   529.69
32   4.86   522.58
15   4.29   515.46
9    0.71   1059.38
5    1.30   543.92
4    2.32   1066.50
4    0.48   21.34
3    3.00   412.37
2    0.40   639.90
2    4.26   199.07
2    4.91   625.67
2    4.26   206.19
1    5.26   1052.27
1    4.72   95.98
1    1.29   1589.07 ] ;

```

MatrizL5 =...

```
[ 50  5.26  7.11
```

```

16  5.25  14.23
4   0.01  536.80
2   1.10  522.58
1   3.14  0   ] ;

```

MatrizB0=...

```
[ 2268616  3.5585261  529.6909651
```

```

110090  0  0
109972  3.908093  1059.381930
8101    3.6051  522.5774
6438    0.3063  536.8045
6044    4.2588  1589.0729
1107    2.9853  1162.4747
944     1.675   426.598
942     2.936   1052.268
894     1.754   7.114
836     5.179   103.093

```

767	2.155	632.784
684	3.678	213.299
629	0.643	1066.495
559	0.014	846.083
532	2.703	110.206
464	1.173	949.176
431	2.608	419.485
351	4.611	2118.764
132	4.778	742.990
123	3.350	1692.166
116	1.387	323.505
115	5.049	316.392
104	3.701	515.464
103	2.319	1478.867
102	3.153	1581.959 ] ;

MatrizB1=...

[177352	5.701665	529.690965		
	3230	5.7794	1059.3819	
	3081	5.4746	522.5774	
	2212	4.7348	536.8045	
	1694	3.1416	0	
	346	4.746	1052.268	
	234	5.189	1066.495	
	196	6.186	7.114	
	150	3.927	1589.073	
	114	3.439	632.784	
	97	2.91	949.18	
	82	5.08	1162.47	
	77	2.51	103.09	
	77	0.61	419.48	
	74	5.50	515.46	
	61	5.45	213.30	
	50	3.95	735.88	
	46	0.54	110.21	
	45	1.90	846.08	
	37	4.70	543.92	
	36	6.11	316.39	
	32	4.92	1581.96 ] ;	

MatrizB2=...

[ 8094	1.4632	529.6910		
	813	3.1416	0	
	742	0.957	522.577	
	399	2.899	536.805	
	342	1.447	1059.382	
	74	0.41	1052.27	
	46	3.48	1066.50	
	30	1.93	1589.07	

```

29  0.99  515.46
23  4.27  7.11
14  2.92  543.92
12  5.22  632.78
11  4.88  949.18
6   6.21  1045.15 ] ;

```

MatrizB3=...

```
[ 252  3.381  529.691
```

```

122  2.733  522.577
49   1.04  536.80
11   2.31  1052.27
8    2.77  515.46
7    4.25  1059.38
6    1.78  1066.50
4    1.13  543.92
3    3.14  0 ] ;

```

MatrizB4=...

```
[15  4.53  522.58
```

```

5   4.47  529.69
4   5.44  536.80
3   0     0
2   4.52  515.46
1   4.20  1052.27 ] ;

```

MatrizB5=...

```
[ 1  0.09  522.58 ] ;
```

MatrizR0=...

```
[ 520887429  0
```

```

0
25209327  3.49108640  529.69096509
610600    3.841154    1059.381930
282029    2.574199    632.783739
187647    2.075904    522.577418
86793     0.71001     419.48464
72063     0.21466     536.80451
65517     5.97996     316.39187
30135     2.16132     949.17561
29135     1.67759     103.09277
23947     0.27458     7.11355
23453     3.54023     735.87651
22284     4.19363     1589.07290
13033     2.96043     1162.47470
12749     2.71550     1052.26838
9703      1.9067      206.1855

```

9161	4.4135	213.2991
7895	2.4791	426.5982
7058	2.1818	1265.5675
6138	6.2642	846.0828
5477	5.6573	639.8973
4170	2.0161	515.4639
4137	2.7222	625.6702
3503	0.5653	1066.4955
2617	2.0099	1581.9593
2500	4.5518	838.9693
2128	6.1275	742.9901
1912	0.8562	412.3711
1611	3.0887	1368.6603
1479	2.6803	1478.8666
1231	1.8904	323.5054
1217	1.8017	110.2063
1015	1.3867	454.9094
999	2.872	309.278
961	4.549	2118.764
886	4.148	533.623
821	1.593	1898.351
812	5.941	909.819
777	3.677	728.763
727	3.988	1155.361
655	2.791	1685.052
654	3.382	1692.166
621	4.823	956.289
615	2.276	942.062
562	0.081	543.918
542	0.284	525.759 ] ;

MatrizR1=...

[1271802	2.6493751	529.6909651
61662	3.00076	1059.38193
53444	3.89718	522.57742
41390	0	0
31185	4.88277	536.80451
11847	2.41330	419.48464
9166	4.7598	7.1135
3404	3.3469	1589.0729
3203	5.2108	735.8765
3176	2.7930	103.0928
2806	3.7422	515.4639
2677	4.3305	1052.2684
2600	3.6344	206.1855
2412	1.4695	426.5982
2101	3.9276	639.8973
1646	4.4163	1066.4955
1641	4.4163	625.6702
1050	3.1611	213.2991
1025	2.5543	412.3711
806	2.678	632.784
741	2.171	1162.475

```

677      6.250      838.969
567      4.577      742.990
485      2.469      949.176
469      4.710      543.918
445      0.403      323.505
416      5.368      728.763
402      4.605      309.278
347      4.681      14.227
338      3.168      956.289
261      5.343      846.083
247      3.923      942.062
220      4.842      1368.660
203      5.600      1155.361
200      4.439      1045.155
197      3.706      2118.764
196      3.759      199.072
184      4.265      95.979
180      4.402      532.872
170      4.846      526.510
146      6.130      533.623
133      1.322      110.206
132      4.512      525.759 ];

```

MatrizR2=...

```

[79645  1.35866  529.69097
      8252  5.7777  522.5774
      7030  3.2748  536.8045
      5314  1.8384  1059.3819
      1861  2.9768  7.1135
      964   5.480   515.464
      836   4.199   419.485
      498   3.142    0
      427   2.228   639.897
      406   3.783   1066.495
      377   2.242   1589.073
      363   5.368   206.186
      342   6.099   1052.268
      339   6.127   625.670
      333   0.003   426.598
      280   4.262   412.371
      257   0.963   632.784
      230   0.705   735.877
      201   3.069   543.918
      200   4.429   103.093
      139   2.932   14.227
      114   0.787   728.763
      95    1.70    838.97
      86    5.14    323.51
      83    0.06    309.28
      80    2.98    742.99
      75    1.60    956.29
      70    1.51    213.30

```

67	5.47	199.07	
62	6.10	1045.15	
56	0.96	1162.47	
52	5.58	942.06	
50	2.72	532.87	
45	5.52	508.35	
44	0.27	526.51	
40	5.95	95.98	] ;

MatrizR3=...

[ 3519 6.0580 529.6910

1073	1.6732	536.8045	
916	1.413	522.577	
342	0.523	1059.382	
255	1.196	7.114	
222	0.952	515.464	
90	3.14	0	
69	2.27	1066.50	
58	1.41	543.92	
58	0.53	639.90	
51	5.98	412.37	
47	1.58	625.67	
43	6.12	419.48	
37	1.18	14.23	
34	1.67	1052.27	
34	0.85	206.19	
31	1.04	1589.07	
30	4.63	426.60	
21	2.50	728.76	
15	0.89	199.07	
14	0.96	508.35	
13	1.50	1045.15	
12	2.61	735.88	
12	3.56	323.51	
11	1.79	309.28	
11	6.28	956.29	
10	6.26	103.09	
9	3.45	838.97	];

MatrizR4=...

[ 129 0.084 536.805

113	4.249	529.691	
83	3.30	522.58	
38	2.73	515.46	
27	5.69	7.11	
18	5.40	1059.38	
13	6.02	543.92	
9	0.77	1066.50	
8	5.68	14.23	
7	1.43	412.37	
6	5.12	639.90	
5	3.34	625.67	

```

3      3.40    1052.27
3      4.16    728.76
3      2.90    426.60 ];

MatrizR5=...
      [ 11  4.75  536.80
        4  5.92  522.58
        2  5.57  515.46
        2  4.30  543.92
        2  3.69   7.11
        2  4.13 1059.38
        2  5.49 1066.50 ];

```

### Matrizes Saturno

```

%Matrizes de Saturno

%Matriz L0
MatrizL0 = [87401354    0          0
            11107660    3.96205090    213.29909544
            1414151     4.5858152     7.1135470
            398379      0.521120      206.185548
            350769      3.303299      426.598191
            206816      0.246584      103.092774
            79271      3.84007      220.41264
            23990      4.66977      110.20632
            16574      0.43719      419.48464
            15820      0.93809      632.78374
            15054      2.71670      639.89729
            14907      5.76903      316.39187
            14610      1.56519      3.93215
            13160      4.44891      14.22709
            13005      5.98119      11.04570
            10725      3.12940      202.25340
            6126       1.7633      277.0350
            5863       0.2366      529.6910
            5228       4.2078      3.1814
            5020       3.1779      433.7117
            4593       0.6198      199.0720
            4006       2.2448      63.7359
            3874       3.2228      138.5175
            3269       0.7749      949.1756
            2954       0.9828      95.9792
            2461       2.0316      735.8765
            1758       3.2658      522.5774
            1640       5.5050      846.0828
            1581       4.3727      309.2783
            1391       4.0233      323.5054
            1124       2.8373      415.5525
            1087       4.1834      2.4477

```

1017	3.7170	227.5262
957	0.507	1265.567
853	3.421	175.166
849	3.191	209.367
789	5.007	0.963
749	2.144	853.196
744	5.253	224.345
687	1.747	1052.268
654	1.599	0.048
634	2.299	412.371
625	0.970	210.118
580	3.093	74.782
546	2.127	350.332
543	1.518	9.561
530	4.449	117.320
478	2.965	137.033
474	5.475	742.990
452	1.044	490.334
449	1.290	127.472
372	2.278	217.231
355	3.013	838.969
347	1.539	340.771
343	0.246	0.521
330	0.247	1581.959
322	0.961	203.738
322	2.572	647.011
309	3.495	216.480
287	2.370	351.817
278	0.400	211.815
249	1.470	1368.660
227	4.910	12.530
220	4.204	200.769
209	1.345	625.670
208	0.483	1162.475
208	1.283	39.357
204	6.011	265.989
185	3.503	149.563
184	0.973	4.193
182	5.491	2.921
174	1.863	0.751
165	0.440	5.417
149	5.736	52.690
148	1.535	5.629
146	6.231	195.140
140	4.295	21.341
131	4.068	10.295
125	6.277	1898.351
122	1.976	4.666
118	5.341	554.070
117	2.679	1155.361
114	5.594	1059.382
112	1.105	191.208
110	0.166	1.484
109	3.438	536.805
107	4.012	956.289

```

104      2.192      88.866
103      1.197      1685.052
101      4.965      269.921  ];

MatrizL1 = ...
[ 21354295596.0  0
1296855      1.8282054      213.2990954
564348      2.885001      7.113547
107679      2.277699      206.185548
98323      1.08070      426.59819
40255      2.04128      220.41264
19942      1.27955      103.09277
10512      2.74880      14.22709
6939      0.4049      639.8973
4803      2.4419      419.4846
4056      2.9217      110.2063
3769      3.6497      3.9322
3385      2.4169      3.1814
3302      1.2626      433.7117
3071      2.3274      199.0720
1953      3.5639      11.0457
1249      2.6280      95.9792
922      1.961      227.526
706      4.417      529.691
650      6.174      202.253
628      6.111      309.278
487      6.040      853.196
479      4.988      522.577
468      4.617      63.736
417      2.117      323.505
408      1.299      209.367
352      2.317      632.784
344      3.959      412.371
340      3.634      316.392
336      3.772      735.877
332      2.861      210.118
289      2.733      117.320
281      5.744      2.448
266      0.543      647.011
230      1.644      216.480
192      2.965      224.345
173      4.077      846.083
167      2.597      21.341
136      2.286      10.295
131      3.441      742.990
128      4.095      217.231
109      6.161      415.552
98      4.73      838.97
94      3.48      1052.27
92      3.95      88.87
87      1.22      440.83
83      3.11      625.67
78      6.24      302.16
67      0.29      4.67

```

66	5.65	9.56
62	4.29	127.47
62	1.83	195.14
58	2.48	191.96
57	5.02	137.03
55	0.28	74.78
54	5.13	490.33
51	1.46	536.80
47	1.18	149.56
47	5.15	515.46
46	2.23	956.29
44	2.71	5.42
40	0.41	269.92
40	3.89	728.76
38	0.65	422.67
38	2.53	12.53
37	3.78	2.92
35	6.08	5.63
34	3.21	1368.66
33	4.64	277.03
33	5.43	1066.50
33	0.30	351.82
32	4.39	1155.36
31	2.43	52.69
30	2.84	203.00
30	6.19	284.15
30	3.39	1059.38
29	2.03	330.62
28	2.74	265.99
26	4.51	340.77 ] ;

MatrizL2= ...

[ 116441	1.179879	7.113547	
91921	0.07425	213.29910	
90592	0	0	
15277	4.06492	206.18555	
10631	0.25778	220.41264	
10605	5.40964	426.59819	
4265	1.0460		14.2271
1216	2.9186		103.0928
1165	4.6094		639.8973
1082	5.6913		433.7117
1045	4.0421		199.0720
1020	0.6337		3.1814
634	4.388	419.485	
549	5.573	3.932	
457	1.268	110.206	
425	0.209	227.526	
274	4.288	95.979	
162	1.381	11.046	
129	1.566	309.278	
117	3.881	853.196	
105	4.900	647.011	
101	0.893	21.341	
96	2.91		316.39

95	5.63	412.37
85	5.73	209.37
83	6.05	216.48
82	1.02	117.32
75	4.76	210.12
67	0.46	522.58
66	0.48	10.29
64	0.35	323.51
61	4.88	632.78
53	2.75	529.69
46	5.69	440.83
45	1.67	202.25
42	5.71	88.87
32	0.07	63.74
32	1.67	302.16
31	4.16	191.96
27	0.83	224.34
25	5.66	735.88
20	5.94	217.23
18	4.90	625.67
17	1.63	742.99
16	0.58	515.46
14	0.21	838.97
14	3.76	195.14
12	4.72	203.00
12	0.13	234.64
12	3.12	846.08
11	5.92	536.80
11	5.60	728.76
11	3.20	1066.50
10	4.99	422.67
10	0.26	330.62
10	4.15	860.31
9	0.46	956.29
8	2.14	269.92
8	5.25	429.78
8	4.03	9.56
7	5.40	1052.27
6	4.46	284.15
6	5.93	405.26 ];

MatrizL3 = ...

[16039	5.73945	7.11355		
	4250	4.5854	213.2991	
	1907	4.7608	220.4126	
	1466	5.9133	206.1855	
	1162	5.6197	14.2271	
	1067	3.6082	426.5982	
	239	3.861	433.712	
	237	5.768	199.072	
	166	5.116	3.181	
	151	2.736	639.897	
	131	4.743	227.526	

63	0.23	419.48
62	4.74	103.09
40	5.47	21.34
40	5.96	95.98
39	5.83	110.21
28	3.01	647.01
25	0.99	3.93
19	1.92	853.20
18	4.97	10.29
18	1.03	412.37
18	4.20	216.48
18	3.32	309.28
16	3.90	440.83
16	5.62	117.32
13	1.18	88.87
11	5.58	11.05
11	5.93	191.96
10	3.95	209.37
9	3.39	302.16
8	4.88	323.51
7	0.38	632.78
6	2.25	522.58
6	1.06	210.12
5	4.64	234.64
4	3.14	0
4	2.31	515.46
3	2.20	860.31
3	0.59	529.69
3	4.93	224.34
3	0.42	625.67
2	4.77	330.62
2	3.35	429.78
2	3.20	202.25
2	1.19	1066.50
2	1.35	405.26
2	4.16	223.59
2	3.07	654.12 ] ;

MatrizL4 =...

[ 1662	3.9983	7.1135
257	2.984	220.413
236	3.902	14.227
149	2.741	213.299
114	3.142	0
110	1.515	206.186
68	1.72	426.60
40	2.05	433.71
38	1.24	199.07
31	3.01	227.53
15	0.83	639.90
9	3.71	21.34
6	2.42	419.48
6	1.16	647.01

```

4      1.45    95.98
4      2.12    440.83
3      4.09    110.21
3      2.77    412.37
3      3.01    88.87
3      0.00    853.20
3      0.39    103.09
2      3.78    117.32
2      2.83    234.64
2      5.08    309.28
2      2.24    216.48
2      5.19    302.16
1      1.55    191.96 ] ;

```

MatrizL5 =...

```

[ 124      2.259    7.114
34      2.16    14.23
28      1.20    220.41
6       1.22    227.53
5       0.24    433.71
4       6.23    426.60
3       2.97    199.07
3       4.29    206.19
2       6.25    213.30
1       5.28    639.90
1       0.24    440.83
1       3.14    0 ] ;

```

MatrizB0=...

```

[4330678  3.6028443      213.2990954
240348      2.852385      426.598191
84746      0      0
34116      0.57297      206.18555
30863      3.48442      220.41264
14734      2.11847      639.89729
9917       5.7900      419.4846
6994       4.7360      7.1135
4808       5.4331      316.3919
4788       4.9651      110.2063
3432       2.7326      433.7117
1506       6.0130      103.0928
1060       5.6310      529.6910
969        5.204      632.784
942        1.396      853.196
708        3.803      323.505
552        5.131      202.253
400        3.359      227.526
319        3.626      209.367
316        1.997      647.011
314        0.465      217.231
284        4.886      224.345

```

```

236      2.139      11.046
215      5.950      846.083
209      2.120      415.552
207      0.730      199.072
179      2.954      63.736
141      0.644      490.334
139      4.595      14.227
139      1.998      735.877
135      5.245      742.990
122      3.115      522.577
116      3.109      216.480
114      0.963      210.118 ] ;

```

MatrizB1=...

```

[397555      5.332900      213.299095
49479 3.14159 0
18572 6.09919 426.59819
14801 2.30586 206.18555
9644 1.6967      220.4126
3757 1.2543      419.4846
2717 5.9117      639.8973
1455 0.8516      433.7117
1291 2.9177      7.1135
853 0.436 316.392
298 0.919 632.784
292 5.316 853.196
284 1.619 227.526
275 3.889 103.093
172 0.052 647.011
166 2.444 199.072
158 5.209 110.206
128 1.207 529.691
110 2.457 217.231
82 2.76 210.12
81 2.86 14.23
69 1.66 202.25
65 1.26 216.48
61 1.25 209.37
59 1.82 323.51
46 0.82 440.83
36 1.82 224.34
34 2.84 117.32
33 1.31 412.37
32 1.19 846.08
27 4.65 1066.50
27 4.44 11.05 ];

```

MatrizB2=...

```

[ 20630      0.50482 213.29910

```

3720	3.9983	206.1855
1627	6.1819	220.4126
1346	0	0
706	3.039	419.485
365	5.099	426.598
330	5.279	433.712
219	3.828	639.897
139	1.043	7.114
104	6.157	227.526
93	1.98	316.39
71	4.15	199.07
52	2.88	632.78
49	4.43	647.01
41	3.16	853.20
29	4.53	210.12
24	1.12	14.23
21	4.35	217.23
20	5.31	440.83
18	0.85	110.21
17	5.68	216.48
16	4.26	103.09
14	3.00	412.37
12	2.53	529.69
8	3.32	202.25
7	5.56	209.37
7	0.29	323.51
6	1.16	117.32
6	3.61	860.31 ] ;

MatrizB3=...

[ 666	1.990	213.299
632	5.698	206.186
398	0	0
188	4.338	220.413
92	4.84	419.48
52	3.42	433.71
42	2.38	426.60
26	4.40	227.53
21	5.85	199.07
18	1.99	639.90
11	5.37	7.11
10	2.55	647.01
7	3.46	316.39
6	4.80	632.78
6	0.02	210.12
6	3.52	440.83
5	5.64	14.23
5	1.22	853.20
4	4.71	412.37
3	0.63	103.09

			2	3.72	216.48	]	;
MatrizB4=...							
[	80	1.12	206.19				
			32	3.12	213.30		
			17	2.48	220.41		
			12	3.14	0		
			9	0.38	419.48		
			6	1.56	433.71		
			5	2.63	227.53		
			5	1.28	199.07		
			1	1.43	426.60		
			1	0.67	647.01		
			1	1.72	440.83		
			1	6.18	639.90	]	;
MatrizB5=...							
[	8	2.82	206.19				
			1	0.51	220.41	]	;
MatrizR0=...							
[	955758136	0	0				
			52921382		2.39226220	213.29909544	
			1873680		5.2354961	206.1855484	
			1464664		1.6476305	426.5981909	
			821891		5.935200	316.391870	
			547507		5.015326	103.092774	
			371684		2.271148	220.412642	
			361778	3.139043	7.113547		
			140618		5.704067	632.783739	
			108975		3.293136	110.206321	
			69007	5.94100	419.48464		
			61053	0.94038	639.89729		
			48913	1.55733	202.25340		
			34144	0.19519	277.03499		
			32402	5.47085	949.17561		
			20937	0.46349	735.87651		
			20839	1.52103	433.71174		
			20747	5.33256	199.07200		
			15298	3.05944	529.69097		
			14296	2.60434	323.50542		
			12884	1.64892	138.51750		
			11993	5.98051	846.08283		
			11380	1.73106	522.57742		
			9796	5.2048	1265.5675		
			7753	5.8519	95.9792		
			6771	3.0043	14.2271		
			6466	0.1773	1052.2684		
			5850	1.4552	415.5525		
			5307	0.5974	63.7359		
			4696	2.1492	227.5262		

4044	1.6401	209.3669
3688	0.7802	412.3711
3461	1.8509	175.1661
3420	4.9455	1581.9593
3401	0.5539	350.3321
3376	3.6953	224.3448
2976	5.6847	210.1177
2885	1.3876	838.9693
2881	0.1796	853.1964
2508	3.5385	742.9901
2448	6.1841	1368.6603
2406	2.9656	117.3199
2174	0.0151	340.7709
2024	5.0541	11.0457 ];

MatrizR1=...

[ 6182981	0.2584352	213.2990954		
506578		0.711147		206.185548
341394		5.796358		426.598191
188491		0.472157		220.412642
186262		3.141593		0
143891		1.407449		7.113547
49621	6.01744		103.09277	
20928	5.09246		639.89729	
19953	1.17560		419.48464	
18840	1.60820		110.20632	
13877	0.75886		199.07200	
12893	5.94330		433.71174	
5397	1.2885		14.2271	
4869	0.8679		323.5054	
4247	0.3930		227.5262	
3252	1.2585		95.9792	
3081	3.4366		522.5774	
2909	4.6068		202.2534	
2856	2.1673		735.8765	
1988	2.4505		412.3711	
1941	6.0239		209.3669	
1581	1.2919		210.1177	
1340	4.3080		853.1964	
1316	1.2530		117.3199	
1203	1.8665		316.3919	
1091	0.0753		216.4805	
966	0.480		632.784	
954	5.152		647.011	
898	0.983		529.691	
882	1.885		1052.268	
874	1.402		224.345	
785	3.064		838.969	
740	1.382		625.670	
658	4.144		309.278	
650	1.725		742.990	
613	3.033		63.736	
599	2.549		217.231	

```

503          2.130          3.932 ];

MatrizR2=...
[436902      4.786717          213.299095
71923      2.50070      206.18555
49767      4.97168      220.41264
43221      3.86940      426.59819
29646      5.96310      7.11355
4721        2.4753          199.0720
4142        4.1067          433.7117
3789        3.0977          639.8973
2964        1.3721          103.0928
2556        2.8507          419.4846
2327         0              0
2208        6.2759          110.2063
2188        5.8555          14.2271
1957        4.9245          227.5262
924         5.464          323.505
706         2.971          95.979
546         4.129          412.371
431         5.178          522.577
405         4.173          209.367
391         4.481          216.480
374         5.834          117.320
361         3.277          647.011
356         3.192          210.118
326         2.269          853.196
207         4.022          735.877
204         0.088          202.253
180         3.597          632.784
178         4.097          440.825
154         3.135          625.670
148         0.136          302.165
133         2.594          191.958
132         5.933          309.278 ] ;

MatrizR3=...
[ 20315      3.02187      213.29910
8924        3.1914          220.4126
6909        4.3517          206.1855
4087        4.2241          7.1135
3879        2.0106          426.5982
1071        4.2036          199.0720
907         2.283          433.712
606         3.175          227.526
597         4.135          14.227
483         1.173          639.897
393         0              0
229         4.698          419.485
188         4.590          110.206

```

150	3.202	103.093	
121	3.768	323.505	
102	4.710	95.979	
101	5.819	412.371	
93	1.44		647.01
84	2.63		216.48
73	4.15		117.32
62	2.31		440.83
55	0.31		853.20
50	2.39		209.37
45	4.37		191.96
41	0.69		522.58
40	1.84		302.16
38	5.94		88.87
32	4.01		21.34 ] ;

MatrizR4=...

[1202	1.4150	220.4126	
	708	1.162	213.299
	516	6.240	206.186
	427	2.469	7.114
	268	0.187	426.598
	170	5.959	199.072
	150	0.480	433.712
	145	1.442	227.526
	121	2.405	14.227
	47	5.57	639.90
	19	5.86	647.01
	17	0.53	440.83
	16	2.90	110.21
	15	0.30	419.48
	14	1.30	412.37
	13	2.09	323.51
	11	0.22	95.98
	11	2.46	117.32
	10	3.14	0
	9	1.56	88.87
	9	2.28	21.34
	9	0.68	216.48
	8	1.27	234.64 ];

MatrizR5=...

[	129	5.913	220.413	
		32	0.69	7.11
		27	5.91	227.53
		20	4.95	433.71
		20	0.67	14.23
		14	2.67	206.19
		14	1.46	199.07
		13	4.59	426.60
		7	4.63	213.30
		5	3.61	639.90
		4	4.90	440.83
		3	4.07	647.01

3	4.66	191.96	
3	0.49	323.51	
3	3.18	419.48	
2	3.70	88.87	
2	3.32	95.98	
2	0.56	117.32	];

## Matrizes Terra

```
%Matrizes da Terra
```

```
MatrizL0 = ...
```

```
[175347046.0 0 0
3341656.0 4.6692568 6283.07585
34894.0 4.6261 12566.1517
3497.0 2.7441 5753.3849
3418.0 2.8289 3.5231
3136.0 3.6277 77713.7715
2676.0 4.4181 7860.4194
2343.0 6.1352 3930.2097
1324.0 0.7425 11506.7698
1273.0 2.0371 529.691
1199.0 1.1096 1577.3435
990 5.233 5884.927
902 2.045 26.298
857 3.508 398.149
780 1.179 5223.694
753 2.533 5507.553
505 4.583 18849.228
492 4.205 775.523
357 2.92 0.067
317 5.849 11790.629
284 1.899 796.298
271 0.315 10977.079
243 0.345 5486.778
206 4.806 2544.314
205 1.869 5573.143
202 2.458 6069.777 %Corrected from 202 2.4458 6069.777
156 0.833 213.299
132 3.411 2942.463
126 1.083 20.775
115 0.645 0.98
103 0.636 4694.003
102 0.976 15720.839
102 4.267 7.114
99 6.21 2146.17
98 0.68 155.42
86 5.98 161000.69
85 1.3 6275.96
85 3.67 71430.7
```

```
80 1.81 17260.15
79 3.04 12036.46
75 1.76 5088.63
74 3.50 3154.69
74 4.68 801.82
70 0.83 9437.76
62 3.98 8827.39
61 1.82 7084.9
57 2.78 6286.6
56 4.39 14143.5
56 3.47 6279.55
52 0.19 12139.55
52 1.33 1748.02
51 0.28 5856.48
49 0.49 1194.45
41 5.37 8429.24
41 2.4 19651.05
39 6.17 10447.39
37 6.04 10213.29
37 2.57 1059.38
36 1.71 2352.87
36 1.78 6812.77
33 0.59 17789.85
30 0.44 83996.85
30 2.74 1349.87
25 3.16 4690.48];
```

```
MatrizI1 =[628331966747.0 0 0
206059.0 2.678235 6283.07585
4303.0 2.6351 12566.1517
425.0 1.59 3.523
119.0 5.796 26.298
109.0 2.966 1577.344
93 2.59 18849.23
72 1.14 529.69
68 1.87 398.15
67 4.41 5507.55
59 2.89 5223.69
56 2.17 155.42
45 0.4 796.3
36 0.47 775.52
29 2.65 7.11
21 5.34 0.98
19 1.85 5486.78
19 4.97 213.3
17 2.99 6275.96
16 0.03 2544.31
16 1.43 2146.17
15 1.21 10977.08
12 2.83 1748.02
```

```
12 3.26 5088.63
12 5.27 1194.45
12 2.08 4694
11 0.77 553.57
10 1.3 6286.6           %Corrigido de 10 1.3 3286.6
10 4.24 1349.87
9 2.7 242.73
9 5.64 951.72
8 5.3 2352.87
6 2.65 9437.76
6 4.67 4690.48];
```

```
MatrizL2 =[52919.0 0 0
8720.0 1.0721 6283.0758
309.0 0.867 12566.152
27 0.05 3.52
16 5.19 26.3
16 3.68 155.42
10 0.76 18849.23
9 2.06 77713.77
7 0.83 775.52
5 4.66 1577.34
4 1.03 7.11
4 3.44 5573.14
3 5.14 796.3
3 6.05 5507.55
3 1.19 242.73
3 6.12 529.69
3 0.31 398.15
3 2.28 553.57
2 4.38 5223.69
2 3.75 0.98];
```

```
MatrizL3 =[289.0 5.844 6283.076
35 0 0
17 5.49 12566.15
3 5.2 155.42
1 4.72 3.52
1 5.3 18849.23
1 5.97 242.73];
```

```
MatrizL4 = [114.0 3.142 0
8 4.13 6283.08
1 3.84 12566.15];
```

```
MatrizL5 =[1 3.14 0];
```

```
MatrizB0=[280.0 3.199 84334.662
102.0 5.422 5507.553
80 3.88 5223.69
44 3.7 2352.87
32 4 1577.34];
```

```
MatrizB1=[9 3.9 5507.55
6 1.73 5223.69];
```

```
MatrizR0=[ 100013989.0 0 0
1670700.0 3.0984635 6283.07585
13956.0 3.05525 12566.1517
3084.0 5.1985 77713.7715
1628.0 1.1739 5753.3849
1576.0 2.8469 7860.4194
925.0 5.453 11506.77
542.0 4.564 3930.21
472.0 3.661 5884.927
346.0 0.964 5507.553
329.0 5.9 5223.694
307.0 0.299 5573.143
243.0 4.273 11790.629
212.0 5.847 1577.344
186.0 5.022 10977.079
175.0 3.012 18849.228
110.0 5.055 5486.778
98 0.89 6069.78
86 5.69 15720.84
86 1.27 161000.69
65 0.27 17260.15
63 0.92 529.69
57 2.01 83996.85
56 5.24 71430.7
49 3.25 2544.31
47 2.58 775.52
45 5.54 9437.76
43 6.01 6275.96
39 5.36 4694
38 2.39 8827.39
37 0.83 19651.05
37 4.9 12139.55
36 1.67 12036.46
35 1.84 2942.46
33 0.24 7084.9
32 0.18 5088.63
32 1.78 398.15
28 1.21 6286.6
28 1.9 6279.55
26 4.59 10447.39];
```

%Corrigido de 85 0.27 17260.15

```
MatrizR1=[ 103019.0 1.10749 6283.07585
1721.0 1.0644 12566.1517
```

```
702.0 3.142 0
32 1.02 18849.23
31 2.84 5507.55
25 1.32 5223.69
18 1.42 1577.34
10 5.91 10977.08
9 1.42 6275.96
9 0.27 5486.78];

MatrizR2=[4359.0 5.7846 6283.0758
124.0 5.579 12566.152
12 3.14 0
9 3.63 77713.77
6 1.87 5573.14
3 5.47 18849.23];

MatrizR3= [145.0 4.273 6283.076
7 3.92 12566.15];

MatrizR4 = [4 2.56 6283.08];
```

## Saturno Gráfico

### Conversão de graus decimais

```
function valor = sexonav( x )
%Conversão de valores em graus(decimal) para graus com minutos do grau
graus=fix(x);
partedecimal=x-graus;
minutos=partedecimal*60;
valor=[graus,minutos];
end
```

### Vetor Beta0

```
vetor_BETA0=[-53.7402
-59.3783
-52.8787
-51.3602
-5.4674
+54.3188
```

+54.3880

-32.9133

-24.5064

-22.3825

+44.3236

+25.6804

+29.3035

-40.6331

- 4.5699

+30.7363

-46.1513

-72.6798

-16.8161

-16.0270

-75.8239

+22.8643

+59.9061

+12.2669

-20.7836

+49.6802

+ 5.3851

+74.9223

+22.0999

-21.1357

-47.8312

-14.5009

-44.1375

+9.9651

-11.0519

+72.9876

+19.4060

-12.5856

-22.0800

-72.2357

+30.1255

- 3.4495

-36.2677

+ 6.6842

-16.0196

+35.8352

+ 0.4649

-31.1228

-42.5959

+ 7.1978

+46.6222

-13.7884

-39.6053

- 2.0545

-55.8708

+61.7328

```
+ 0.3330  
  
+66.1014  
  
-65.8402];
```

---

### Vetor\_estrelas

```
vetor_estrelas={'Acamar'  
  'Achernar'  
  'Acrux'  
  'Adhara'  
  'Aldebaran'  
  'Alioth'  
  'Alkaid'  
  'Al Na'ir'  
  'Alnilam'  
  'Alphard'  
  'Alphecca'  
  'Alpheratz'  
  'Altair'  
  'Ankaa'  
  'Antares'  
  'Arcturus'  
  'Atria'  
  'Avior'  
  'Bellatrix'  
  'Betelgeuse'  
  'Canopus'  
  'Capella'  
  'Deneb'  
  'Denebola'  
  'Diphda'  
  'Dubhe'  
  'Elnath'  
  'Eltanin'  
  'Enif'  
  'Fomalhaut'  
  'Gacrux'  
  'Gienah'  
  'Hadar'  
  'Hamal'  
  'Kaus Australis'  
  'Kochab'  
  'Markab'  
  'Menkar'  
  'Menkent'  
  'Miplacidus'  
  'Mirfak'
```

```
'Nunki '  
'Peacock'  
'Pollux'  
'Procyon'  
'Rasalhague'  
'Regulus'  
'Rigel'  
'Rigil Kentaurus'  
'Sabik'  
'Schedar'  
'Shaula'  
'Sirius'  
'Spica'  
'Suhail'  
'Vega'  
'Zubenelgenubi'  
'Polaris'  
'Octantis'  
'Sol'  
'Lua'  
'Venus'  
'Marte'  
'Jupiter'  
'Saturno'};
```

### **Vetor\_LAMBDA0**

```
vetor_LAMBDA0=[23.2723  
  
345.3117  
  
221.8701  
  
110.7630  
  
69.7892  
  
158.9334  
  
176.9331  
  
315.9070  
  
83.4636  
  
147.2792  
  
222.2959
```

14.3085

301.7765

345.4938

249.7623

204.2337

260.8962

173.1294

80.9464

88.7547

104.9614

81.8579

335.3293

171.6176

2.5835

135.1975

82.5750

267.9687

331.8850

333.8604

216.7397

190.7256

233.7925

37.6625

275.0787

133.3195

353.4857

44.3201

222.3086

211.9692

62.0810

282.3853

293.8176

113.2156

115.7855

262.4487

149.8292

76.8295

239.4793

257.9696

37.7838

264.5858

104.0816

203.8414

161.1877

285.3164

225.0827

88.5676

271.8706];

## Vetor\_MIU

```
vetor_MIU=[-0.00152
```

```
+0.00285
```

```
-0.00047
```

```
+0.00025
```

```
+0.00104
```

```
+0.00417
```

```
-0.00430
```

```
+0.00184
```

```
-0.00002
```

```
-0.00074
```

```
+0.00568
```

```
+0.00162
```

```
+0.01939
```

```
-0.00100
```

```
-0.00007
```

```
-0.00768
```

```
+0.00123
```

```
-0.00250
```

```
-0.00032
```

```
+0.00080
```

```
+0.00308
```

```
+0.00126
```

```
+0.00029
```

-0.01153

+0.00673

-0.00239

+0.00037

-0.00080

+0.00090

+0.00716

+0.00737

-0.00449

-0.00036

+0.00364

-0.00106

-0.00112

+0.00125

-0.00091

-0.00873

-0.01254

+0.00051

+0.00026

-0.00041

-0.01700

-0.01504

+0.00459

-0.00648

-0.00003

```
-0.13521  
+0.00084  
+0.00105  
+0.00007  
-0.01524  
-0.00075  
-0.00116  
+0.01403  
-0.00226  
+0.00098  
+0.00118];
```

### **Vetor\_MIU\_linha**

```
vetor_MIU_linha=[+0.00112  
-0.00275  
-0.00070  
+0.00010  
-0.00550  
+0.00194  
-0.00230  
-0.00536  
-0.00007  
+0.00067  
-0.00118  
-0.00575
```

+0.00733

-0.01237

-0.00061

-0.06288

-0.00075

-0.00013

-0.00037

+0.00026

+0.00076

-0.01191

-0.00002

-0.00849

-0.00191

-0.00343

-0.00491

-0.00055

-0.00029

-0.00802

-0.00543

-0.00128

-0.00076

-0.00569

-0.00346

-0.00088

-0.00182

```
-0.00197  
-0.01871  
-0.00329  
-0.00084  
-0.00156  
-0.00244  
-0.00436  
-0.03143  
-0.00609  
-0.00222  
-0.00007  
-0.02399  
+0.00275  
-0.00157  
-0.00079  
-0.03492  
-0.00118  
+0.00011  
+0.00709  
-0.00267  
-0.00118  
-0.00042];
```

### **Tabela**

```
function varout=tabela (ano,mes,dia,hora,minuto,segundo,lat,long)
```

```

hora_hora=hora+minuto/60+segundo/3600;
Matrizcoord=zeros(65,1);
for i= 1: 1: 59
    coord = aparente(i,dia,mes,ano,hora_hora);
    Matrizcoord(i+6,1)=coord(1);
    Matrizcoord(i+6,2)=coord(2);
end
dadossol=efemerides(dia,mes,ano,hora,minuto,segundo);
dadoslua=Lua(ano,mes,dia,hora,minuto,segundo);
dadosvenus=Venus(ano,mes,dia,hora,minuto,segundo);
dadosmarte=Marte(ano,mes,dia,hora,minuto,segundo);
dadosjupiter=Jupiter(ano,mes,dia,hora,minuto,segundo);
dadossaturno=Saturno(ano,mes,dia,hora,minuto,segundo);

Matrizcoord(1,1)=dadossol(1);
Matrizcoord(1,2)=dadossol(2);

Matrizcoord(2,1)=dadoslua(1);
Matrizcoord(2,2)=dadoslua(2);

Matrizcoord(3,1)=dadosvenus(1);
Matrizcoord(3,2)=dadosvenus(2);

Matrizcoord(4,1)=dadosmarte(1);
Matrizcoord(4,2)=dadosmarte(2);

Matrizcoord(5,1)=dadosjupiter(1);
Matrizcoord(5,2)=dadosjupiter(2);

Matrizcoord(6,1)=dadossaturno(1);
Matrizcoord(6,2)=dadossaturno(2);

nomes_astros;

for k=1:1:65

hori=horizontais(ano,mes,dia,hora,minuto,segundo,lat,long,Matrizcoord(k,1),Matrizcoord(k,2));
    C(k,1)=hori(1);
    C(k,2)=round(hori(2));

end

varout=C;
end

```

### Ler dados introduzidos

```

ano=(get(handles.editano,'string'));
ano=str2double(ano);

```

```

mes=(get(handles.editmes,'string'));
mes=str2double(mes);

dia=(get(handles.editdia,'string'));
dia=str2double(dia);

hora=get(handles.edit29,'string');
hora=str2double(hora);

minuto=get(handles.edit30,'string');
minuto=str2double(minuto);

segundo=get(handles.edit31,'string');
segundo=str2double(segundo);

hora_hora=hora+minuto/60+segundo/3600;

%Obter Latitudes
latitudegraus=get(handles.editlatg,'string');
latitudegraus=str2double(latitudegraus);

latitudeminutos=get(handles.editlatm,'string');
latitudeminutos=str2double(latitudeminutos);

latitudesinal=get(handles.latsinal,'string');
j=strfind(latitudesinal,'S');

latitudedecimal=latitudegraus+latitudeminutos/60;    %definir a latitude em
%graus em decimal
if j==1 ,latitudedecimal=-latitudedecimal;
end

%Obter Longitudes
longitudegraus=get(handles.editlongg,'string');
longitudegraus=str2double(longitudegraus);
longitudeminutos=get(handles.editlongm,'string');
longitudeminutos=str2double(longitudeminutos);

longsinal=get(handles.longsinal,'string');
f=strfind(longsinal,'E');

longitudedecimal=longitudegraus+longitudeminutos/60;
if f==1 ,longitudedecimal=-longitudedecimal;
end

```

## Gráfico - Planejamento

```

function grafico
thetag=linspace(0,360,45);
phig=linspace(0,180,45);
[thetag,phig]=meshgrid(thetag,phig);
rho=1;

x=rho*sind(phig).*cosd(thetag);
y=rho*sind(phig).*sind(thetag);
z=rho*cosd(phig);

colormap winter
Esfera=mesh(x,y,abs(z),'FaceAlpha', 0.3);

freezeColors
hold on

theta=linspace(0,2*pi,40);
phi=linspace(0,pi,40);
[theta,phi]=meshgrid(theta,phi);
rho=1;

%BUSSOLA

%pontos Norte:
AN=[-0.03, 0, 0];
BN=[0.03 ,0, 0];
CN=[0, -0.3, 0];
DN=[0, -0.04, 0.04];

%face inferior ABC
patch ([AN(1), BN(1) , CN(1) ] ,[ AN(2) , BN(2) , BN(2) ], [
AN(3) , BN(3) , CN(3) ], 'r');

%face anterior 1 ABD
patch ([AN(1), BN(1) , DN(1) ] ,[ AN(2) , BN(2) , DN(2) ], [
AN(3) , BN(3) , DN(3) ], 'r');

%face lateral 1 BDC
patch ([BN(1), DN(1) , CN(1) ] ,[ BN(2) , DN(2) , CN(2) ], [
BN(3) , DN(3) , CN(3) ], 'r');

%face lateral ADC
patch ([AN(1), DN(1) , CN(1) ] ,[ AN(2) , DN(2) , CN(2) ], [
AN(3) , DN(3) , CN(3) ], 'r');

%Pontos Sul
AS=[ -0.03, 0, 0];
BS=[0.03, 0, 0];
CS=[0, 0.3, 0];
DS=[0, 0.04, 0.04];

```

```

patch ([AS(1), BS(1) , CS(1) ] , [ AS(2) , BS(2) , BS(2) ] , [
AS(3) , BS(3) , CS(3) ], 'w');

%face anterior 1 ABD
patch ([AS(1), BS(1) , DS(1) ] , [ AS(2) , BS(2) , DS(2) ] , [
AS(3) , BS(3) , DS(3) ], 'w');

%face lateral 1 BDC
patch ([BS(1), DS(1) , CS(1) ] , [ BS(2) , DS(2) , CS(2) ] , [
BS(3) , DS(3) , CS(3) ], 'w');

%face lateral ADC
patch ([AS(1), DS(1) , CS(1) ] , [ AS(2) , DS(2) , CS(2) ] , [
AS(3) , DS(3) , CS(3) ], 'w');

hold on

N=text(0.035,-1.1,0, 'N');
N.FontSize=15;

S=text(0.035,1.1,0, 'S');
S.FontSize=15;

W=text(1.1,0,0, 'W');
W.FontSize=15;

E=text(-1.1,0, 'E');
E.FontSize=15;
% guidata(hObject, handles);

daspect ([1 1 1])
%colormap winter
colordef black
set(gca, 'Color', 'k')
set(gca, 'visible', 'on')
% camlight
view(0,91)
grid off
set(gca, 'visible', 'off')
set(gca, 'xtick', [])
set(gca, 'ytick', [])
set(gca, 'ztick', [])
rotate3d on
hManager = uigetmodemanager();
hManager.CurrentMode.ModeStateData.textState = 0;
end

```

## Definição de variáveis para o gráfico

```
%DEFINIÇÃO DE VARIÁVEIS GRÁFICAS
theta=linspace(0,2*pi,40);
phi=linspace(0,pi,40);
[theta,phi]=meshgrid(theta,phi);
rho=1;
```

## Sol gráfico

```
function sol_grafico(alturasol,azimutesol,theta,phi)

%Sol centro
sx=sind(90-alturasol)*cosd(270-azimutesol);
sy=sind(90-alturasol)*sind(270-azimutesol);
sz=cosd(90-alturasol);

%Sol forma
rho1=0.05;
xs=rho1*sin(phi).*cos(theta)+sx;
ys=rho1*sin(phi).*sin(theta)+sy;
zs=rho1*cos(phi)+sz;

%Sol cor
mapsol = [1, 0.843, 0];

Sol=surf(xs,ys,zs);
colormap (mapsol)
set(Sol, 'EdgeColor', 'interp')
set(Sol, 'EdgeColor', 'none', 'FaceAlpha',0.9,'EdgeLighting',
'none','FaceLighting','none')
freezeColors
end
```

## Lua Gráfico

```
function lua_grafico(alturalua,azimutelua,theta,phi)
%Centro da Lua
lx=sind(90-alturalua)*cosd(270-azimutelua);
ly=sind(90-alturalua)*sind(270-azimutelua);
lz=cosd(90-alturalua);

%Forma da Lua

rho1=0.03;
xl=rho1*sin(phi).*cos(theta)+lx;
yl=rho1*sin(phi).*sin(theta)+ly;
zl=rho1*cos(phi)+lz;
% Lua=plot3(xl,yl,zl,'w');
```

```

%Cor da Lua
maplua = [1, 1, 1];

Lua=surf(xl,y1,z1);
colormap (maplua)
set(Lua, 'EdgeColor', 'interp')
set(Lua, 'EdgeColor', 'none', 'FaceAlpha',0.9,'EdgeLighting',
'none','FaceLighting','none')
freezeColors
end

```

### Venus Gráfico

```

function venus_grafico(altura,azimute,theta,phi)
%venus centro
mx=sind(90-altura)*cosd(270-azimute);
my=sind(90-altura)*sind(270-azimute);
mz=cosd(90-altura);
%venus Forma
rho_m=0.02;
xm=rho_m*sin(phi).*cos(theta)+mx;
ym=rho_m*sin(phi).*sin(theta)+my;
zm=rho_m*cos(phi)+mz;

%venus cor
mapvenus = [ 0.9961 0.4, 0.3516];

%venus plot
venus=surf(xm,ym,zm);
colormap (mapvenus)
set(venus, 'EdgeColor', 'interp', 'FaceColor', 'interp')
freezeColors
end

```

### Marte Gráfico

```

function marte_grafico(altura,azimute,theta,phi)
%Marte centro
mx=sind(90-altura)*cosd(270-azimute);
my=sind(90-altura)*sind(270-azimute);
mz=cosd(90-altura);
%Marte Forma
rho_m=0.02;
xm=rho_m*sin(phi).*cos(theta)+mx;
ym=rho_m*sin(phi).*sin(theta)+my;
zm=rho_m*cos(phi)+mz;

%Marte cor
mapmarte = [ 0.85, 0, 0];

```

```

%Marte plot
Marte=surf(xm,ym,zm);
colormap (mapmarte)
set(Marte, 'EdgeColor', 'interp', 'FaceColor', 'interp')
freezeColors
end

```

### Jupiter Gráfico

```

function jupiter_grafico(altura,azimute,theta,phi)
%jupiter centro
mx=sind(90-altura)*cosd(270-azimute);
my=sind(90-altura)*sind(270-azimute);
mz=cosd(90-altura);
%jupiter Forma
rho_m=0.025;
xm=rho_m*sin(phi).*cos(theta)+mx;
ym=rho_m*sin(phi).*sin(theta)+my;
zm=rho_m*cos(phi)+mz;

%jupiter cor
mapjupiter = [ 0.9961    0.2,    0];

%jupiter plot
jupiter=surf(xm,ym,zm);
colormap (mapjupiter)
set(jupiter, 'EdgeColor', 'interp', 'FaceColor', 'interp')
freezeColors
end

```

### Saturno Gráfico

```

function saturno_grafico(altura,azimute,theta,phi)
aneis(altura,azimute)
%Saturno Experiência
% theta=linspace(0,2*pi,40);
% phi=linspace(0,pi,40);
% [theta,phi]=meshgrid(theta,phi);
% rho=1;
%saturno centro
mx=sind(90-altura)*cosd(270-azimute);
my=sind(90-altura)*sind(270-azimute);
mz=cosd(90-altura);
%saturno Forma
rho_m=0.025;
xm=rho_m*sin(phi).*cos(theta)+mx;
ym=rho_m*sin(phi).*sin(theta)+my;
zm=rho_m*cos(phi)+mz;

```

```

%saturno cor
mapsaturno = [ 0.97    0.87,    0.49];

%saturno plot
saturno=surf(xm,ym,zm);
colormap (mapsaturno)
set(saturno, 'EdgeColor', 'interp', 'FaceColor', 'interp')
daspect ([1 1 1])
freezeColors
end

```

### Saturno Anéis

```

%Experiencia aneis 3
function aneis(altura,azimute)
mx=sind(90-altura)*cosd(270-azimute);
my=sind(90-altura)*sind(270-azimute);
mz=cosd(90-altura);
%anel amarelo 1
R = 0.06; %outer radius
r = 0.04; %inner radius
steps = 100;
theta = 0:2*pi/steps:2*pi;
r_outer = ones(size(theta))*R;
r_inner = ones(size(theta))*r;
rr = [r_outer;r_inner];
theta = [theta;theta];
xx = rr.*cos(theta)+mx;
yy = rr.*sin(theta)+my;
zz = zeros(size(xx))+mz;
surf(xx,yy,zz, 'EdgeColor', 'none')
mapanel_1 = [ 0.89    0.89,    0.67];
colormap(mapanel_1)
freezeColors
hold on

%anel preto
% mx=sind(90)*cosd(270);
% my=sind(90)*sind(270);
% mz=cosd(90);
R = 0.061; %outer radius
r = 0.06; %inner radius
steps = 100;
theta = 0:2*pi/steps:2*pi;
r_outer = ones(size(theta))*R;
r_inner = ones(size(theta))*r;
rr = [r_outer;r_inner];
theta = [theta;theta];
xx = rr.*cos(theta)+mx;
yy = rr.*sin(theta)+my;

```

```

zz = zeros(size(xx))+mz;
surf(xx,yy,zz)
mapanel=[0,0,0];
colormap(mapanel)
freezeColors
hold on

%anel amarelo 2

R = 0.068; %outer radius
r = 0.061; %inner radius
steps = 100;
theta = 0:2*pi/steps:2*pi;
r_outer = ones(size(theta))*R;
r_inner = ones(size(theta))*r;
rr = [r_outer;r_inner];
theta = [theta;theta];
xx = rr.*cos(theta)+mx;
yy = rr.*sin(theta)+my;
zz = zeros(size(xx))+mz;
surf(xx,yy,zz, 'EdgeColor', 'none')
mapanel_2=[0.80,0.80,0.6];
colormap(mapanel_2)
freezeColors
hold on
end

```

### Estrela gráfico

```

function estrela_grafico(altura,azimute)
ex=sind(90-altura)*cosd(270-azimute);
ey=sind(90-altura)*sind(270-azimute);
ez=cosd(90-altura);

Estrela=plot3(ex,ey,ez, 'wp', 'MarkerFaceColor', 'w', 'MarkerSize',9);
hold on
end

```

### Rotinas de Cálculo dos desvios relativos ao NavPac

#### Estudo\_desvios

```
docs_datas
```

```

L=length(nomes_docs);

estudo=cell(33,4);

for k=1:1:L

estudo(k,:)=desvios(nomes_docs(k),Matriz_datas(k,1),Matriz_datas(k,2),Matriz_d
atas(k,3),Matriz_datas(k,4),Matriz_datas(k,5),Matriz_datas(k,6));

estudo_sem_polares(k,:)=desvios_sem_polares(nomes_docs(k),Matriz_datas(k,1),Ma
triz_datas(k,2),Matriz_datas(k,3),Matriz_datas(k,4),Matriz_datas(k,5),Matriz_d
atas(k,6));
end

estudo_sol=zeros(33,2);
estudo_lua=zeros(33,2);
estudo_polar=zeros(33,2);
estudo_octantis=zeros(33,2);
estudo_venus=zeros(33,2);
estudo_marte=zeros(33,2);
estudo_jupiter=zeros(33,2);
estudo_saturno=zeros(33,2);
estudo_bellatrix=zeros(33,2);
estudo_atria=zeros(33,2);
estudo_miaplacidus=zeros(33,2);
estudo_alnilam=zeros(33,2);
estudo_kochab=zeros(33,2);
estudo_procyon=zeros(33,2);

for k=1:1:L

paginas=pagina_desvios(nomes_docs(k),Matriz_datas(k,1),Matriz_datas(k,2),Matri
z_datas(k,3),Matriz_datas(k,4),Matriz_datas(k,5),Matriz_datas(k,6));
estudo_sol(k,:)=paginas(60,:);
estudo_lua(k,:)=paginas(65,:);

estudo_venus(k,:)=paginas(61,:);
estudo_marte(k,:)=paginas(62,:);
estudo_jupiter(k,:)=paginas(63,:);
estudo_saturno(k,:)=paginas(64,:);

estudo_polar(k,:)=paginas(44,:);
estudo_octantis(k,:)=paginas(54,:);
estudo_bellatrix(k,:)=paginas(19,:);
estudo_atria(k,:)=paginas(17,:);
estudo_miaplacidus(k,:)=paginas(40,:);
estudo_alnilam(k,:)=paginas(9,:);
estudo_kochab(k,:)=paginas(36,:);
estudo_procyon(k,:)=paginas(46,:);

end

```

```

estudo_cell=[datas, estudo];
titulos_estudo={'Data' 'Maior_Desvio_GHA' 'Astro_GHA' 'Maior_Desvio_Dec'
'Astro_Dec'};
estudo=cell2table(estudo_cell,'VariableNames',titulos_estudo)

% estudo_sem polares
estudo_sem_polares_cell=[datas, estudo_sem_polares];
titulos_estudo_sem_polares={'Data' 'Maior_Desvio_GHA' 'Astro_GHA'
'Maior_Desvio_Dec' 'Astro_Dec'};
estudo_sem_polares_cell=cell2table(estudo_sem_polares_cell,'VariableNames',titulos_estudo_sem_polares)

estudo_sol=num2cell(estudo_sol);
estudo_sol=[datas estudo_sol];
titulos_sol={'Data' 'Desvio_GHA' 'Desvio_Dec'};
estudo_sol=cell2table(estudo_sol,'VariableNames',titulos_sol)

estudo_lua=num2cell(estudo_lua);
estudo_lua=[datas estudo_lua];
titulos_lua={'Data' 'Desvio_GHA' 'Desvio_Dec'};
estudo_lua=cell2table(estudo_lua,'VariableNames',titulos_lua)

estudo_venus=num2cell(estudo_venus);
estudo_venus=[datas estudo_venus];
titulos_venus={'Data' 'Desvio_GHA' 'Desvio_Dec'};
estudo_venus=cell2table(estudo_venus,'VariableNames',titulos_venus)

estudo_marte=num2cell(estudo_marte);
estudo_marte=[datas estudo_marte];
titulos_marte={'Data' 'Desvio_GHA' 'Desvio_Dec'};
estudo_marte=cell2table(estudo_marte,'VariableNames',titulos_marte)

estudo_jupiter=num2cell(estudo_jupiter);
estudo_jupiter=[datas estudo_jupiter];
titulos_jupiter={'Data' 'Desvio_GHA' 'Desvio_Dec'};
estudo_jupiter=cell2table(estudo_jupiter,'VariableNames',titulos_jupiter)

estudo_saturno=num2cell(estudo_saturno);
estudo_saturno=[datas estudo_saturno];
titulos_saturno={'Data' 'Desvio_GHA' 'Desvio_Dec'};
estudo_saturno=cell2table(estudo_saturno,'VariableNames',titulos_saturno)

estudo_polar=num2cell(estudo_polar);
estudo_polar=[datas estudo_polar];
titulos_polar={'Data' 'Desvio_GHA' 'Desvio_Dec'};
estudo_polar=cell2table(estudo_polar,'VariableNames',titulos_polar)

estudo_octantis=num2cell(estudo_octantis);
estudo_octantis=[datas estudo_octantis];
titulos_octantis={'Data' 'Desvio_GHA' 'Desvio_Dec'};
estudo_octantis=cell2table(estudo_octantis,'VariableNames',titulos_octantis)

estudo_bellatrix=num2cell(estudo_bellatrix);

```

```

estudo_bellatrix=[datas estudo_bellatrix];
titulos_bellatrix={'Data' 'Desvio_GHA' 'Desvio_Dec'};
estudo_bellatrix=cell2table(estudo_bellatrix,'VariableNames',titulos_bellatrix
)

estudo_atria=num2cell(estudo_atria);
estudo_atria=[datas estudo_atria];
titulos_atria={'Data' 'Desvio_GHA' 'Desvio_Dec'};
estudo_atria=cell2table(estudo_atria,'VariableNames',titulos_atria)

estudo_miaplacidus=num2cell(estudo_miaplacidus);
estudo_miaplacidus=[datas estudo_miaplacidus];
titulos_miaplacidus={'Data' 'Desvio_GHA' 'Desvio_Dec'};
estudo_miaplacidus=cell2table(estudo_miaplacidus,'VariableNames',titulos_miapl
acidus)

estudo_alnilam=num2cell(estudo_alnilam);
estudo_alnilam=[datas estudo_alnilam];
titulos_alnilam={'Data' 'Desvio_GHA' 'Desvio_Dec'};
estudo_alnilam=cell2table(estudo_alnilam,'VariableNames',titulos_alnilam)

estudo_kochab=num2cell(estudo_kochab);
estudo_kochab=[datas estudo_kochab];
titulos_kochab={'Data' 'Desvio_GHA' 'Desvio_Dec'};
estudo_kochab=cell2table(estudo_kochab,'VariableNames',titulos_kochab)

estudo_procyon=num2cell(estudo_procyon);
estudo_procyon=[datas estudo_procyon];
titulos_procyon={'Data' 'Desvio_GHA' 'Desvio_Dec'};
estudo_procyon=cell2table(estudo_procyon,'VariableNames',titulos_procyon)

```

## docs\_datas

```

nomes_docs={'010110.txt'
'020610.txt'
'031210.txt'
'040311.txt'
'050811.txt'
'061111.txt'
'070212.txt'
'080512.txt'
'090912.txt'
'100413.txt'
'110713.txt'
'121013.txt'
'130114.txt'
'140614.txt'
'151214.txt'
'160315.txt'
'170815.txt'

```

```

'181115.txt'
'190216.txt'
'200516.txt'
'210916.txt'
'220417.txt'
'230717.txt'
'241017.txt'
'250118.txt'
'260618.txt'
'271218.txt'
'280319.txt'
'290819.txt'
'301119.txt'
'290220.txt'
'310520.txt'
'020920.txt'};

Matriz_datas= ...
[01 01 2010 12 0 0
02 06 2010 18 34 0
03 12 2010 20 51 0
04 03 2011 01 30 0
05 08 2011 11 0 0
06 11 2011 22 30 0
07 02 2012 02 0 0
08 05 2012 14 30 0
09 09 2012 21 0 0
10 04 2013 03 0 0
11 07 2013 10 30 0
12 10 2013 20 0 0
13 1 2014 2 0 0
14 06 2014 7 0 0
15 12 2014 21 0 0
16 03 2015 9 0 0
17 08 2015 17 0 0
18 11 2015 22 0 0
19 02 2016 05 0 0
20 05 2016 16 0 0
21 09 2016 21 0 0
22 4 2017 8 0 0
23 7 2017 19 0 0
24 10 2017 23 0 0
25 1 2018 0 0 0
26 06 2018 12 30 0
27 12 2018 22 0 0
28 03 2019 2 0 0
29 08 2019 7 0 0
30 11 2019 18 0 0
29 02 2020 0 0 0
31 05 2020 12 0 0
02 09 2020 23 0 0];

datas={'01/01/2010 - 12:00h'
'02/06/2010 - 18:00h'
'03/12/2010 - 20:51h'

```

```
'04/03/2011 - 01:30h'
'05/08/2011 - 11:00h'
'06/11/2011 - 22:30h'
'07/02/2012 - 02:00h'
'08/05/2012 - 14:30h'
'09/09/2012 - 21:00h'
'10/04/2013 - 03:00h'
'11/07/2013 - 10:30h'
'12/10/2013 - 20:00h'
'13/01/2014 - 02:00h'
'14/06/2014 - 07:00h'
'15/12/2014 - 21:00h'
'16/03/2015 - 09:00h'
'17/08/2015 - 17:00h'
'18/11/2015 - 22:00h'
'19/02/2016 - 05:00h'
'20/05/2016 - 16:00h'
'21/09/2016 - 21:00h'
'22/04/2017 - 08:00h'
'23/07/2017 - 19:00h'
'24/10/2017 - 23:00h'
'25/01/2018 - 00:00h'
'26/06/2018 - 12:30h'
'27/12/2018 - 22:00h'
'28/03/2019 - 02:00h'
'29/08/2019 - 07:00h'
'30/11/2019 - 18:00h'
'29/02/2020 - 00:00h'
'31/05/2020 - 12:00h'
'02/09/2020 - 23:00h'};
```

## desvios

```
function output=desvios(doc,dia,mes,ano,hora,minuto,segundo)
astros_navpac_file
z=pagina_desvios(doc,dia,mes,ano,hora,minuto,segundo);

z1=z(:,1);
z2=z(:,2);

gha_maior_erro=max(z1);
gha_astro_mais_errado= z1==max(z1);
gha_astro_mais_errado=astros_navpac(gha_astro_mais_errado);

dec_maior_erro=max(z2);
dec_astro_mais_errado= find(z2==max(z2));
dec_astro_mais_errado=astros_navpac(dec_astro_mais_errado(1));

output(1)=num2cell(gha_maior_erro);
output(2)=gha_astro_mais_errado(1);
```

```
output(3)=num2cell(dec_maior_erro);
output(4)=dec_astro_mais_errado(1);
end
```

### desvios\_sem\_polares

```
function output=desvios_sem_polares(doc,dia,mes,ano,hora,minuto,segundo)
astros_navpac_file
z=pagina_desvios(doc,dia,mes,ano,hora,minuto,segundo);

z1=z(:,1);
z2=z(:,2);
z1(44,1)=0;
z1(54,1)=0;

gha_maior_erro=max(z1);
gha_astro_mais_errado= z1==max(z1);

gha_astro_mais_errado=astros_navpac(gha_astro_mais_errado);

dec_maior_erro=max(z2);
dec_astro_mais_errado= find(z2==max(z2));
dec_astro_mais_errado=astros_navpac(dec_astro_mais_errado(1));

output(1)=num2cell(gha_maior_erro);
output(2)=gha_astro_mais_errado(1);
output(3)=num2cell(dec_maior_erro);
output(4)=dec_astro_mais_errado(1);
end
```

### astros\_navpac

```
astros_navpac={'Acamar'
'Achernar'
'Acrux'
'Adhara'
'Aldebaran'
'Alioth'
'Alkaid'
'Al Na'ir'
'Alnilam'
'Alphard'
```

'Alphecca'  
'Alpheratz'  
'Altair'  
'Ankaa'  
'Antares'  
'Arcturus'  
'Atria'  
'Avior'  
'Bellatrix'  
'Betelgeuse'  
'Canopus'  
'Capella'  
'Deneb'  
'Denebola'  
'Diphda'  
'Dubhe'  
'Elnath'  
'Eltanin'  
'Enif'  
'Fomalhaut'  
'Gacrux'  
'Gienah'  
'Hadar'  
'Hamal'  
'Kaus Australis'  
'Kochab'  
'Markab'  
'Menkar'  
'Menkent'  
'Miaplacidus'  
'Mirfak'  
'Nunki'  
'Peacock'  
'Polar'  
'Pollux'  
'Procyon'  
'Rasalhague'  
'Regulus'  
'Rigel'  
'Rigil Kentaurus'  
'Sabik'  
'Schedar'  
'Shaula'  
'Octantis'  
'Sirius'  
'Spica'  
'Suhail'  
'Vega'  
'Zubenelgenubi'  
'Sol'  
'Venus'  
'Marte'  
'Jupiter'  
'Saturno'  
'Lua'

```
'Aries'}];
```

### Pagina\_desvios

```
function z=pagina_desvios(doc,dia,mes,ano,hora,minuto,segundo)
doc=char(doc);

nav=navpac(doc);
meusdados=gerardata(dia,mes,ano,hora,minuto,segundo);

z=abs(nav-meusdados)*60;

z=round(z,1);
end
```

### navpac

```
function data=navpac(textfile)
textfile=char(textfile);

M = readtable(textfile,'ReadVariableNames',false);
% M = readtable(textfile);

a=table2array(M);
y=char(a);
gha_graus=str2num(y(:,25:29));
gha_min=str2num(y(:,30:35));
GHA_decimal=gha_graus+gha_min/60;

dec_graus=str2num(y(:,39:41));
dec_mins=str2num(y(:,42:47));
DEC_decimal=dec_graus+dec_mins/60;
DEC_decimal(66,1)=0;
ns=y(:,37:38);

%Detalhe: k=1 não é o primeiro sinal. mas sim k=2
for k=1:1:65
    if strcmp(ns(k+1,1),'S')==1 || strcmp(ns(k+1,2),'S')==1
        DEC_decimal(k,1)= -(DEC_decimal(k,1));
    end
end
GHA_decimal;
DEC_decimal;
data(:,1)=GHA_decimal;
data(:,2)=DEC_decimal;
```

end

### gerar data

```
function mydata=gerardata(dia,mes,ano,hora,minuto,segundo)
minuto_h=minuto/60;
segundo_h=segundo/3600;

horadec=hora+minuto_h+segundo_h;
mydata=zeros(66,4);
for k=1:1:43
    mydata(k,:)=aparente(k,dia,mes,ano,horadec);
end
mydata(44,:)=aparente(58,dia,mes,ano,horadec);
for k=45:1:53
    mydata(k,:)=aparente(k-1,dia,mes,ano,horadec);
end
mydata(54,:)=aparente(59,dia,mes,ano,horadec);
for k=55:1:59
    mydata(k,:)=aparente(k-2,dia,mes,ano,horadec);
end
% sol=efemerides(dia,mes,ano,hora,minuto,segundo);
sol=Sol(ano,mes,dia,hora,minuto,segundo);

mydata(60,:)= [0 sol(2) sol(3) 0];

tsid=TSG(ano,mes,dia,hora,minuto,segundo);

venus=Venus(ano,mes,dia,hora,minuto,segundo);
mydata(61,:)= [0 venus(2) wrapTo360(tsid-venus(1)) 0];

marte=Marte(ano,mes,dia,hora,minuto,segundo);
mydata(62,:)= [0 marte(2) wrapTo360(tsid-marte(1)) 0];

jupiter=Jupiter(ano,mes,dia,hora,minuto,segundo);
mydata(63,:)= [0 jupiter(2) wrapTo360(tsid-jupiter(1)) 0];

saturno=Saturno(ano,mes,dia,hora,minuto,segundo);
mydata(64,:)= [0 saturno(2) wrapTo360(tsid-saturno(1)) 0];

lua=Lua(ano,mes,dia,hora,minuto,segundo);
mydata(65,:)= [0 lua(2) wrapTo360(tsid-lua(1)) 0];

mydata(66,:)= [0 0 tsid 0];

mydata_gha=mydata(:,3);
mydata_dec=mydata(:,2);
mydata=[mydata_gha mydata_dec];
```

end

### Documentos de texto com dados do NavPac

01 01 10/1200

Acamar	+3.2	236	22.0	S	40	16.0
Achernar	+0.5	256	30.4	S	57	11.4
Acrux	+1.3	094	14.4	S	63	09.1
Adhara	+1.5	176	16.2	S	28	59.2
Aldebaran	+0.9	211	54.1	N	16	31.8
Alioth	+1.8	087	24.8	N	55	53.9
Alkaid	+1.9	074	03.0	N	49	15.4
Al Na'ir	+1.7	308	49.1	S	46	54.9
Alnilam	+1.7	196	50.7	S	01	11.7
Alphard	+2.0	139	00.4	S	08	42.2
Alphecca	+2.2	047	15.4	N	26	40.6
Alpheratz	+2.1	278	48.3	N	29	09.0
Altair	+0.8	343	13.1	N	08	53.7
Ankaa	+2.4	274	20.2	S	42	15.2
Antares	+1.0	033	31.8	S	26	27.2
Arcturus	0.0	067	00.2	N	19	07.6
Atria	+1.9	028	36.5	S	69	02.6
Avior	+1.9	155	20.6	S	59	32.4
Bellatrix	+1.6	199	36.5	N	06	21.5
Betelgeuse	Var.	192	05.8	N	07	24.5
Canopus	-0.7	184	58.8	S	52	42.1
Capella	+0.1	201	39.8	N	46	00.6
Deneb	+1.3	330	35.8	N	45	19.1
Denebola	+2.1	103	38.2	N	14	30.8
Diphda	+2.0	270	00.5	S	17	55.9
Dubhe	+1.8	114	56.3	N	61	41.4
Elnath	+1.7	199	17.5	N	28	37.0
Eltanin	+2.2	011	49.9	N	51	29.2
Enif	+2.4	314	51.9	N	09	55.4
Fomalhaut	+1.2	296	28.9	S	29	34.2
Gacrux	+1.6	093	06.0	S	57	10.0
Gienah	+2.6	096	57.0	S	17	35.9
Hadar	+0.6	069	54.0	S	60	25.1
Hamal	+2.0	249	05.6	N	23	30.8
Kaus Australis	+1.9	004	49.6	S	34	22.8
Kochab	+2.1	058	22.3	N	74	06.5
Markab	+2.5	294	43.1	N	15	15.7
Menkar	+2.5	235	19.6	N	04	07.8
Menkent	+2.1	069	12.9	S	36	25.1
Miaplacidus	+1.7	142	41.8	S	69	45.4
Mirfak	+1.8	229	45.8	N	49	54.0
Nunki	+2.0	357	03.9	S	26	17.1
Peacock	+1.9	334	25.7	S	56	42.3
Polaris	+2.0	239	49.7	N	89	18.8
Pollux	+1.1	164	32.5	N	28	00.0
Procyon	+0.4	166	04.1	N	05	11.9

Elaboração de um Almanaque Náutico

Rasalhague	+2.1	017	11.2	N 12	33.1	
Regulus	+1.4	128	48.0	N 11	54.9	
Rigel	+0.1	202	16.2	S 08	11.4	
Rigil Kentaurus	-0.3	060	57.8	S 60	52.4	
Sabik	+2.4	023	17.8	S 15	44.3	
Schedar	+2.2	270	45.7	N 56	35.9	
Shaula	+1.6	017	27.8	S 37	06.6	
S Octantis	+5.5	321	50.9	S 88	55.1	
Sirius	-1.5	179	37.7	S 16	43.8	
Spica	+1.0	079	36.1	S 11	12.9	
Suhail	+2.2	143	56.1	S 43	28.3	
Vega	0.0	001	43.2	N 38	47.5	
Zubenelgenubi	+2.8	058	10.5	S 16	05.0	
*Sun		359	06.9	S 22	59.2	SD 16.3*
*Venus	-4.0	001	46.4	S 23	37.3	HP 00.1*
*Mars	-0.8	138	37.8	N 18	47.8	HP 00.2*
*Jupiter	-2.1	312	00.9	S 13	34.4	*
*Saturn	+0.9	095	58.8	N 00	18.4	*
*Moon		168	34.1	N 21	51.7	HP 61.2*
Aries		281	02.1			

02-06-10/1834

Acamar	+3.2	124	57.7	S 40	15.6	
Achernar	+0.5	145	06.0	S 57	10.7	
Acruz	+1.3	342	49.1	S 63	09.8	
Adhara	+1.5	064	52.0	S 28	59.3	
Aldebaran	+0.9	100	29.6	N 16	31.8	
Alioth	+1.8	335	59.5	N 55	54.3	
Alkaid	+1.9	322	37.5	N 49	15.7	
Al Na'ir	+1.7	197	23.6	S 46	54.3	
Alnilam	+1.7	085	26.3	S 01	11.7	
Alphard	+2.0	027	35.8	S 08	42.4	
Alphecca	+2.2	295	49.9	N 26	40.7	
Alpheratz	+2.1	167	23.3	N 29	08.8	
Altair	+0.8	231	47.5	N 08	53.7	
Ankaa	+2.4	162	55.3	S 42	14.7	
Antares	+1.0	282	06.0	S 26	27.4	
Arcturus	0.0	315	34.8	N 19	07.6	
Atria	+1.9	277	09.3	S 69	02.8	
Avior	+1.9	043	56.8	S 59	32.9	
Bellatrix	+1.6	088	12.1	N 06	21.5	
Betelgeuse	Var.	080	41.4	N 07	24.5	
Canopus	-0.7	073	35.0	S 52	42.2	
Capella	+0.1	090	15.6	N 46	00.5	
Deneb	+1.3	219	10.2	N 45	18.9	
Denebola	+2.1	352	13.2	N 14	30.7	
Diphda	+2.0	158	35.6	S 17	55.6	
Dubhe	+1.8	003	31.5	N 61	41.9	
Elnath	+1.7	087	53.2	N 28	37.0	
Eltanin	+2.2	260	24.0	N 51	29.1	
Enif	+2.4	203	26.6	N 09	55.4	
Fomalhaut	+1.2	185	03.7	S 29	33.8	
Gacrux	+1.6	341	40.7	S 57	10.7	
Gienah	+2.6	345	31.8	S 17	36.2	
Hadar	+0.6	318	28.1	S 60	25.7	
Hamal	+2.0	137	40.9	N 23	30.7	
Kaus Australis	+1.9	253	23.7	S 34	22.7	

Elaboração de um Almanaque Náutico

Kochab	+2.1	306	55.9	N 74	06.8	
Markab	+2.5	183	18.0	N 15	15.7	
Menkar	+2.5	123	55.0	N 04	07.9	
Menkent	+2.1	317	47.3	S 36	25.5	
Miaplacidus	+1.7	031	18.2	S 69	46.0	
Mirfak	+1.8	118	21.5	N 49	53.8	
Nunki	+2.0	245	38.1	S 26	17.0	
Peacock	+1.9	222	59.6	S 56	41.8	
Polaris	+2.0	128	53.9	N 89	18.5	
Pollux	+1.1	053	08.0	N 28	00.1	
Procyon	+0.4	054	39.7	N 05	11.8	
Rasalhague	+2.1	265	45.5	N 12	33.1	
Regulus	+1.4	017	23.3	N 11	54.9	
Rigel	+0.1	090	51.9	S 08	11.4	
Rigil Kentaurus	-0.3	309	31.7	S 60	52.9	
Sabik	+2.4	271	52.1	S 15	44.3	
Schedar	+2.2	159	20.9	N 56	35.5	
Shaula	+1.6	266	01.9	S 37	06.7	
S Octantis	+5.5	209	59.4	S 88	54.4	
Sirius	-1.5	068	13.3	S 16	43.9	
Spica	+1.0	328	10.8	S 11	13.1	
Suhail	+2.2	032	31.7	S 43	28.8	
Vega	0.0	250	17.4	N 38	47.5	
Zubenelgenubi	+2.8	306	44.9	S 16	05.3	
*Sun		098	59.9	N 22	14.6	SD 15.8*
*Venus	-3.9	061	38.3	N 24	19.2	HP 00.1*
*Mars	+1.1	019	15.7	N 13	34.0	HP 00.1*
*Jupiter	-2.3	169	36.6	S 01	15.4	*
*Saturn	+1.0	350	38.5	N 03	04.7	*
*Moon		209	31.9	S 12	16.7	HP 54.3*
Aries		169	37.3			

031210/2051

Acamar	+3.2	340	38.8	S 40	15.6	
Achernar	+0.5	000	47.1	S 57	11.0	
Acrux	+1.3	198	31.4	S 63	09.4	
Adhara	+1.5	280	33.2	S 28	59.1	
Aldebaran	+0.9	316	10.7	N 16	32.0	
Alioth	+1.8	191	42.0	N 55	53.7	
Alkaid	+1.9	178	20.1	N 49	15.3	
Al Na'ir	+1.7	053	05.5	S 46	54.6	
Alnilam	+1.7	301	07.4	S 01	11.7	
Alphard	+2.0	243	17.3	S 08	42.4	
Alphecca	+2.2	151	32.4	N 26	40.6	
Alpheratz	+2.1	023	04.9	N 29	09.4	
Altair	+0.8	087	29.8	N 08	54.0	
Ankaa	+2.4	018	36.8	S 42	14.8	
Antares	+1.0	137	48.4	S 26	27.3	

Elaboração de um Almanaque Náutico

Arcturus	0.0	171	17.2	N 19	07.4	
Atria	+1.9	132	52.5	S 69	02.8	
Avior	+1.9	259	38.0	S 59	32.5	
Bellatrix	+1.6	303	53.2	N 06	21.6	
Betelgeuse	Var.	296	22.5	N 07	24.6	
Canopus	-0.7	289	16.0	S 52	42.0	
Capella	+0.1	305	56.2	N 46	00.5	
Deneb	+1.3	074	52.6	N 45	19.5	
Denebola	+2.1	207	55.1	N 14	30.5	
Diphda	+2.0	014	17.1	S 17	55.5	
Dubhe	+1.8	219	13.3	N 61	41.1	
Elnath	+1.7	303	34.1	N 28	37.0	
Eltanin	+2.2	116	07.1	N 51	29.3	
Enif	+2.4	059	08.6	N 09	55.7	
Fomalhaut	+1.2	040	45.5	S 29	33.9	
Gacrux	+1.6	197	22.9	S 57	10.3	
Gienah	+2.6	201	13.9	S 17	36.1	
Hadar	+0.6	174	10.7	S 60	25.4	
Hamal	+2.0	353	22.1	N 23	31.1	
Kaus Australis	+1.9	109	06.1	S 34	22.8	
Kochab	+2.1	162	40.3	N 74	06.4	
Markab	+2.5	038	59.7	N 15	16.1	
Menkar	+2.5	339	36.2	N 04	08.1	
Menkent	+2.1	173	29.6	S 36	25.3	
Miaplacidus	+1.7	246	59.4	S 69	45.5	
Mirfak	+1.8	334	02.1	N 49	54.2	
Nunki	+2.0	101	20.4	S 26	17.0	
Peacock	+1.9	078	41.9	S 56	42.1	
Polaris	+2.0	343	37.9	N 89	19.0	
Pollux	+1.1	268	49.2	N 27	59.8	
Procyon	+0.4	270	20.9	N 05	11.8	
Rasalhague	+2.1	121	28.0	N 12	33.2	
Regulus	+1.4	233	04.9	N 11	54.7	
Rigel	+0.1	306	33.0	S 08	11.3	
Rigil Kentaurus	-0.3	165	14.4	S 60	52.7	
Sabik	+2.4	127	34.5	S 15	44.3	
Schedar	+2.2	015	02.0	N 56	36.2	
Shaula	+1.6	121	44.3	S 37	06.7	
S Octantis	+5.5	065	50.6	S 88	54.9	
Sirius	-1.5	283	54.6	S 16	43.8	
Spica	+1.0	183	53.0	S 11	13.1	
Suhail	+2.2	248	13.2	S 43	28.5	
Vega	0.0	106	00.2	N 38	47.8	
Zubenelgenubi	+2.8	162	27.3	S 16	05.2	
*Sun		135	16.4	S 22	10.3	SD 16.2*
*Venus	-4.9	175	08.4	S 10	31.2	HP 00.4*
*Mars	+1.3	118	44.2	S 24	14.0	HP 00.1*
*Jupiter	-2.5	030	23.3	S 03	41.8	*
*Saturn	+0.9	190	48.0	S 03	43.2	*
*Moon		161	10.0	S 20	40.9	HP 58.8*
Aries		025	19.6			
040311/0130						
Acamar	+3.2	139	18.2	S 40	15.8	
Achernar	+0.5	159	26.8	S 57	11.0	

Elaboração de um Almanaque Náutico

---

Acrux	+1.3	357 09.1	S 63 09.7	
Adhara	+1.5	079 12.1	S 28 59.6	
Aldebaran	+0.9	114 49.8	N 16 31.9	
Alioth	+1.8	350 20.1	N 55 53.6	
Alkaid	+1.9	336 58.3	N 49 15.1	
Al Na'ir	+1.7	211 44.7	S 46 54.3	
Alnilam	+1.7	099 46.4	S 01 11.9	
Alphard	+2.0	041 55.9	S 08 42.7	
Alphecca	+2.2	310 10.8	N 26 40.3	
Alpheratz	+2.1	181 44.1	N 29 09.2	
Altair	+0.8	246 08.6	N 08 53.8	
Ankaa	+2.4	177 16.1	S 42 14.8	
Antares	+1.0	296 26.7	S 26 27.4	
Arcturus	0.0	329 55.5	N 19 07.2	
Atria	+1.9	291 30.1	S 69 02.6	
Avior	+1.9	058 16.7	S 59 33.0	
Bellatrix	+1.6	102 32.2	N 06 21.4	
Betelgeuse	Var.	095 01.5	N 07 24.4	
Canopus	-0.7	087 55.2	S 52 42.5	
Capella	+0.1	104 35.3	N 46 00.6	
Deneb	+1.3	233 31.5	N 45 19.1	
Denebola	+2.1	006 33.5	N 14 30.3	
Diphda	+2.0	172 56.3	S 17 55.6	
Dubhe	+1.8	017 51.4	N 61 41.2	
Elnath	+1.7	102 13.1	N 28 37.0	
Eltanin	+2.2	274 45.6	N 51 29.0	
Enif	+2.4	217 47.5	N 09 55.6	
Fomalhaut	+1.2	199 24.6	S 29 33.8	
Gacrux	+1.6	356 00.8	S 57 10.6	
Gienah	+2.6	359 52.2	S 17 36.4	
Hadar	+0.6	332 48.4	S 60 25.5	
Hamal	+2.0	152 01.3	N 23 31.0	
Kaus Australis	+1.9	267 44.6	S 34 22.6	
Kochab	+2.1	321 17.8	N 74 06.2	
Markab	+2.5	197 38.8	N 15 15.9	
Menkar	+2.5	138 15.4	N 04 07.9	
Menkent	+2.1	332 07.8	S 36 25.5	
Miaplacidus	+1.7	045 37.8	S 69 46.1	
Mirfak	+1.8	132 41.4	N 49 54.2	
Nunki	+2.0	259 59.0	S 26 16.9	
Peacock	+1.9	237 20.7	S 56 41.7	
Polaris	+2.0	142 45.0	N 89 19.1	
Pollux	+1.1	067 28.0	N 27 59.8	
Procyon	+0.4	068 59.7	N 05 11.6	
Rasalhague	+2.1	280 06.5	N 12 33.0	
Regulus	+1.4	031 43.4	N 11 54.5	
Rigel	+0.1	105 12.1	S 08 11.5	
Rigil Kentaurus	-0.3	323 52.2	S 60 52.7	
Sabik	+2.4	286 13.0	S 15 44.3	
Schedar	+2.2	173 41.5	N 56 36.1	
Shaula	+1.6	280 22.7	S 37 06.6	
S Octantis	+5.5	224 36.3	S 88 54.4	
Sirius	-1.5	082 33.5	S 16 44.2	
Spica	+1.0	342 31.2	S 11 13.4	
Suhail	+2.2	046 51.8	S 43 29.0	
Vega	0.0	264 38.8	N 38 47.4	
Zubenelgenubi	+2.8	321 05.6	S 16 05.4	
*Sun		199 31.9	S 06 37.9	SD 16.1*
*Venus	-4.1	239 34.1	S 18 57.5	HP 00.1*

Elaboração de um Almanaque Náutico

---

*Mars	+1.1	204 44.6	S 09 52.4	HP 00.1*
*Jupiter	-2.1	175 46.3	N 02 20.2	*
*Saturn	+0.5	348 07.6	S 03 51.2	*
*Moon		209 24.4	S 05 41.9	HP 54.2*
Aries		183 58.5		

050811/1100

Acamar	+3.2	073 58.6	S 40 15.2	
Achernar	+0.5	094 06.8	S 57 10.3	
Acrux	+1.3	291 50.5	S 63 10.1	
Adhara	+1.5	013 53.2	S 28 59.2	
Aldebaran	+0.9	049 30.3	N 16 31.9	
Alioth	+1.8	285 01.4	N 55 54.0	
Alkaid	+1.9	271 39.3	N 49 15.6	
Al Na'ir	+1.7	146 24.2	S 46 54.0	
Alnilam	+1.7	034 27.2	S 01 11.6	
Alphard	+2.0	336 37.0	S 08 42.6	
Alphecca	+2.2	244 51.3	N 26 40.8	
Alpheratz	+2.1	116 23.9	N 29 09.3	
Altair	+0.8	180 48.5	N 08 54.2	
Ankaa	+2.4	111 56.0	S 42 14.2	
Antares	+1.0	231 07.0	S 26 27.5	
Arcturus	0.0	264 36.3	N 19 07.5	
Atria	+1.9	226 09.7	S 69 03.1	
Avior	+1.9	352 58.5	S 59 32.8	
Bellatrix	+1.6	037 12.9	N 06 21.6	
Betelgeuse	Var.	029 42.3	N 07 24.5	
Canopus	-0.7	022 36.5	S 52 42.0	
Capella	+0.1	039 16.0	N 46 00.4	
Deneb	+1.3	168 11.2	N 45 19.5	
Denebola	+2.1	301 14.5	N 14 30.5	
Diphda	+2.0	107 36.3	S 17 55.1	
Dubhe	+1.8	312 33.1	N 61 41.4	
Elnath	+1.7	036 53.8	N 28 36.9	
Eltanin	+2.2	209 25.7	N 51 29.5	
Enif	+2.4	152 27.4	N 09 55.9	
Fomalhaut	+1.2	134 04.4	S 29 33.3	
Gacrux	+1.6	290 42.1	S 57 11.0	
Gienah	+2.6	294 33.2	S 17 36.5	
Hadar	+0.6	267 29.2	S 60 26.0	
Hamal	+2.0	086 41.5	N 23 31.0	
Kaus Australis	+1.9	202 24.5	S 34 22.7	
Kochab	+2.1	255 59.1	N 74 06.8	
Markab	+2.5	132 18.7	N 15 16.2	
Menkar	+2.5	072 55.7	N 04 08.2	
Menkent	+2.1	266 48.5	S 36 25.8	
Miaplacidus	+1.7	340 20.3	S 69 46.0	
Mirfak	+1.8	067 21.6	N 49 54.0	
Nunki	+2.0	194 38.9	S 26 16.8	
Peacock	+1.9	172 00.0	S 56 41.7	
Polaris	+2.0	077 04.9	N 89 18.6	
Pollux	+1.1	002 09.0	N 27 59.8	
Procyon	+0.4	003 40.7	N 05 11.7	
Rasalhague	+2.1	214 46.7	N 12 33.3	
Regulus	+1.4	326 24.5	N 11 54.6	

Elaboração de um Almanaque Náutico

Rigel	+0.1	039	52.8	S 08	11.2	
Rigil Kentaurus	-0.3	258	32.9	S 60	53.2	
Sabik	+2.4	220	53.1	S 15	44.3	
Schedar	+2.2	108	21.1	N 56	36.0	
Shaula	+1.6	215	02.7	S 37	06.7	
S Octantis	+5.5	158	40.7	S 88	54.3	
Sirius	-1.5	017	14.5	S 16	43.9	
Spica	+1.0	277	12.1	S 11	13.4	
Suhail	+2.2	341	33.2	S 43	28.9	
Vega	0.0	199	18.8	N 38	47.9	
Zubenelgenubi	+2.8	255	46.2	S 16	05.4	
*Sun		343	29.3	N 16	59.7	SD 15.8*
*Venus	-3.9	346	15.7	N 18	51.1	HP 00.1*
*Mars	+1.4	027	08.6	N 23	46.8	HP 00.1*
*Jupiter	-2.5	081	17.0	N 13	23.0	*
*Saturn	+0.9	286	03.4	S 02	50.4	*
*Moon		272	46.5	S 15	05.3	HP 59.5*
Aries		118	39.3			

061111/2230

Acamar	+3.2	338	36.3	S 40	15.3	
Achernar	+0.5	358	44.5	S 57	10.6	
Acrux	+1.3	196	28.7	S 63	09.8	
Adhara	+1.5	278	30.8	S 28	59.2	
Aldebaran	+0.9	314	07.9	N 16	32.0	
Alioth	+1.8	189	39.6	N 55	53.5	
Alkaid	+1.9	176	17.8	N 49	15.2	
Al Na'ir	+1.7	051	02.5	S 46	54.2	
Alnilam	+1.7	299	04.8	S 01	11.6	
Alphard	+2.0	241	14.8	S 08	42.6	
Alphecca	+2.2	149	29.9	N 26	40.6	
Alpheratz	+2.1	021	02.0	N 29	09.7	
Altair	+0.8	085	27.0	N 08	54.2	
Ankaa	+2.4	016	34.0	S 42	14.5	
Antares	+1.0	135	45.5	S 26	27.4	
Arcturus	0.0	169	14.6	N 19	07.3	
Atria	+1.9	130	48.9	S 69	03.0	
Avior	+1.9	257	35.9	S 59	32.6	
Bellatrix	+1.6	301	50.5	N 06	21.6	
Betelgeuse	Var.	294	19.8	N 07	24.5	
Canopus	-0.7	287	13.8	S 52	41.9	
Capella	+0.1	303	53.3	N 46	00.5	
Deneb	+1.3	072	49.9	N 45	19.8	
Denebola	+2.1	205	52.6	N 14	30.3	
Diphda	+2.0	012	14.3	S 17	55.2	
Dubhe	+1.8	217	10.9	N 61	40.9	
Elnath	+1.7	301	31.3	N 28	37.0	
Eltanin	+2.2	114	04.6	N 51	29.5	
Enif	+2.4	057	05.7	N 09	56.0	
Fomalhaut	+1.2	038	42.6	S 29	33.5	
Gacrux	+1.6	195	20.2	S 57	10.6	
Gienah	+2.6	199	11.3	S 17	36.4	
Hadar	+0.6	172	07.7	S 60	25.7	
Hamal	+2.0	351	19.2	N 23	31.3	

Elaboração de um Almanaque Náutico

Kaus Australis	+1.9	107	03.1	S	34	22.7	
Kochab	+2.1	160	38.6	N	74	06.4	
Markab	+2.5	036	56.8	N	15	16.4	
Menkar	+2.5	337	33.4	N	04	08.3	
Menkent	+2.1	171	26.9	S	36	25.6	
Miaplacidus	+1.7	244	57.5	S	69	45.7	
Mirfak	+1.8	331	59.0	N	49	54.2	
Nunki	+2.0	099	17.4	S	26	16.8	
Peacock	+1.9	076	38.6	S	56	41.9	
Polaris	+2.0	341	10.3	N	89	19.0	
Pollux	+1.1	266	46.5	N	27	59.7	
Procyon	+0.4	268	18.3	N	05	11.6	
Rasalhague	+2.1	119	25.3	N	12	33.3	
Regulus	+1.4	231	02.3	N	11	54.4	
Rigel	+0.1	304	30.4	S	08	11.2	
Rigil Kentaurus	-0.3	163	11.5	S	60	52.9	
Sabik	+2.4	125	31.7	S	15	44.3	
Schedar	+2.2	012	58.9	N	56	36.4	
Shaula	+1.6	119	41.3	S	37	06.7	
S Octantis	+5.5	063	32.8	S	88	54.6	
Sirius	-1.5	281	52.0	S	16	43.9	
Spica	+1.0	181	50.3	S	11	13.3	
Suhail	+2.2	246	10.8	S	43	28.7	
Vega	0.0	103	57.5	N	38	48.0	
Zubenelgenubi	+2.8	160	24.6	S	16	05.4	
*Sun		161	35.4	S	16	05.2	SD 16.1*
*Venus	-3.8	139	36.9	S	21	45.5	HP 00.1*
*Mars	+1.0	232	40.0	N	13	50.1	HP 00.1*
*Jupiter	-2.9	350	56.6	N	11	30.2	*
*Saturn	+0.7	181	05.5	S	06	51.7	*
*Moon		023	42.0	N	05	14.9	HP 54.2*
Aries		023	17.5				

070212/0200

Acamar	+3.2	121	56.0	S	40	15.6	
Achernar	+0.5	142	04.6	S	57	10.8	
Acrux	+1.3	339	46.9	S	63	09.9	
Adhara	+1.5	061	49.9	S	28	59.6	
Aldebaran	+0.9	097	27.3	N	16	31.9	
Alioth	+1.8	332	58.1	N	55	53.3	
Alkaid	+1.9	319	36.4	N	49	14.8	
Al Na'ir	+1.7	194	22.3	S	46	54.1	
Alnilam	+1.7	082	24.0	S	01	11.8	
Alphard	+2.0	024	33.6	S	08	42.9	
Alphecca	+2.2	292	48.8	N	26	40.2	
Alpheratz	+2.1	164	21.7	N	29	09.6	
Altair	+0.8	228	46.4	N	08	54.0	
Ankaa	+2.4	159	53.8	S	42	14.6	
Antares	+1.0	279	04.5	S	26	27.4	
Arcturus	0.0	312	33.4	N	19	06.9	
Atria	+1.9	274	07.5	S	69	02.6	
Avior	+1.9	040	54.7	S	59	33.1	
Bellatrix	+1.6	085	09.8	N	06	21.5	
Betelgeuse	Var.	077	39.0	N	07	24.4	
Canopus	-0.7	070	33.1	S	52	42.4	

Elaboração de um Almanaque Náutico

---

Capella	+0.1	087 12.5	N 46 00.6	
Deneb	+1.3	216 09.5	N 45 19.5	
Denebola	+2.1	349 11.3	N 14 30.0	
Diphda	+2.0	155 33.9	S 17 55.3	
Dubhe	+1.8	000 29.2	N 61 40.8	
Elnath	+1.7	084 50.5	N 28 37.0	
Eltanin	+2.2	257 23.9	N 51 29.1	
Enif	+2.4	200 25.3	N 09 55.9	
Fomalhaut	+1.2	182 02.3	S 29 33.5	
Gacrux	+1.6	338 38.6	S 57 10.8	
Gienah	+2.6	342 30.0	S 17 36.7	
Hadar	+0.6	315 26.1	S 60 25.7	
Hamal	+2.0	134 38.8	N 23 31.2	
Kaus Australis	+1.9	250 22.3	S 34 22.6	
Kochab	+2.1	303 56.8	N 74 06.0	
Markab	+2.5	180 16.5	N 15 16.3	
Menkar	+2.5	120 53.0	N 04 08.2	
Menkent	+2.1	314 45.5	S 36 25.7	
Miaplacidus	+1.7	028 15.9	S 69 46.2	
Mirfak	+1.8	115 18.6	N 49 54.4	
Nunki	+2.0	242 36.7	S 26 16.8	
Peacock	+1.9	219 58.2	S 56 41.6	
Polaris	+2.0	124 53.0	N 89 19.3	
Pollux	+1.1	050 05.5	N 27 59.6	
Procyon	+0.4	051 37.3	N 05 11.4	
Rasalhague	+2.1	262 44.5	N 12 33.0	
Regulus	+1.4	014 21.1	N 11 54.2	
Rigel	+0.1	087 49.7	S 08 11.5	
Rigil Kentaurus	-0.3	306 29.9	S 60 52.9	
Sabik	+2.4	268 50.7	S 15 44.3	
Schedar	+2.2	156 18.9	N 56 36.5	
Shaula	+1.6	263 00.4	S 37 06.6	
S Octantis	+5.5	207 07.1	S 88 54.3	
Sirius	-1.5	065 11.2	S 16 44.2	
Spica	+1.0	325 09.1	S 11 13.6	
Suhail	+2.2	029 29.6	S 43 29.1	
Vega	0.0	247 16.9	N 38 47.6	
Zubenelgenubi	+2.8	303 43.4	S 16 05.5	
*Sun		206 29.2	S 15 31.8	SD 16.2*
*Venus	-4.1	167 39.0	S 01 04.8	HP 00.1*
*Mars	-0.7	352 33.1	N 06 54.0	HP 00.2*
*Jupiter	-2.3	135 04.4	N 11 39.6	*
*Saturn	+0.6	318 15.2	S 08 52.9	*
*Moon		038 11.4	N 14 29.0	HP 57.9*
Aries		166 36.9		

080512/1430

Acamar	+3.2	039 38.7	S 40 15.4	
Achernar	+0.5	059 47.2	S 57 10.3	
AcruX	+1.3	257 29.0	S 63 10.4	
Adhara	+1.5	339 32.7	S 28 59.7	
Aldebaran	+0.9	015 09.9	N 16 31.9	
Alioth	+1.8	250 40.2	N 55 53.6	
Alkaid	+1.9	237 18.4	N 49 15.1	

Elaboração de um Almanaque Náutico

Al Na'ir	+1.7	112	04.1	S	46	53.8	
Alnilam	+1.7	000	06.8	S	01	11.8	
Alphard	+2.0	302	16.2	S	08	43.0	
Alphecca	+2.2	210	30.6	N	26	40.3	
Alpheratz	+2.1	082	03.9	N	29	09.4	
Altair	+0.8	146	28.2	N	08	54.1	
Ankaa	+2.4	077	36.1	S	42	14.2	
Antares	+1.0	196	46.2	S	26	27.5	
Arcturus	0.0	230	15.4	N	19	07.0	
Atria	+1.9	191	48.2	S	69	02.8	
Avior	+1.9	318	37.9	S	59	33.4	
Bellatrix	+1.6	002	52.5	N	06	21.5	
Betelgeuse	Var.	355	21.7	N	07	24.4	
Canopus	-0.7	348	16.2	S	52	42.5	
Capella	+0.1	004	55.4	N	46	00.6	
Deneb	+1.3	133	51.3	N	45	19.3	
Denebola	+2.1	266	53.6	N	14	30.1	
Diphda	+2.0	073	16.3	S	17	55.1	
Dubhe	+1.8	278	11.7	N	61	41.1	
Elnath	+1.7	002	33.3	N	28	37.0	
Eltanin	+2.2	175	05.4	N	51	29.1	
Enif	+2.4	118	07.3	N	09	55.9	
Fomalhaut	+1.2	099	44.3	S	29	33.2	
Gacrux	+1.6	256	20.7	S	57	11.3	
Gienah	+2.6	260	12.2	S	17	36.9	
Hadar	+0.6	233	07.7	S	60	26.1	
Hamal	+2.0	052	21.3	N	23	31.2	
Kaus Australis	+1.9	168	03.9	S	34	22.5	
Kochab	+2.1	221	38.1	N	74	06.3	
Markab	+2.5	097	58.6	N	15	16.3	
Menkar	+2.5	038	35.5	N	04	08.2	
Menkent	+2.1	232	27.4	S	36	26.0	
Miaplacidus	+1.7	305	59.2	S	69	46.5	
Mirfak	+1.8	033	01.3	N	49	54.2	
Nunki	+2.0	160	18.4	S	26	16.7	
Peacock	+1.9	137	39.7	S	56	41.3	
Polaris	+2.0	042	56.5	N	89	19.0	
Pollux	+1.1	327	48.2	N	27	59.7	
Procyon	+0.4	329	20.0	N	05	11.4	
Rasalhague	+2.1	180	26.2	N	12	33.0	
Regulus	+1.4	292	03.6	N	11	54.2	
Rigel	+0.1	005	32.4	S	08	11.4	
Rigil Kentaurus	-0.3	224	11.5	S	60	53.2	
Sabik	+2.4	186	32.5	S	15	44.4	
Schedar	+2.2	074	01.2	N	56	36.2	
Shaula	+1.6	180	42.0	S	37	06.6	
S Octantis	+5.5	124	30.4	S	88	53.9	
Sirius	-1.5	342	54.0	S	16	44.3	
Spica	+1.0	242	51.1	S	11	13.7	
Suhail	+2.2	307	12.4	S	43	29.4	
Vega	0.0	164	58.6	N	38	47.6	
Zubenelgenubi	+2.8	221	25.3	S	16	05.7	
*Sun		038	23.3	N	17	18.1	SD 15.8*
*Venus	-4.7	002	07.7	N	27	45.8	HP 00.4*
*Mars	+0.1	284	56.8	N	10	37.8	HP 00.1*
*Jupiter	-2.0	034	31.0	N	17	29.3	*
*Saturn	+0.4	240	36.8	S	06	56.3	*
*Moon		181	52.4	S	21	37.8	HP 60.4*
Aries		084	19.4				

090912/2100

Acamar	+3.2	259	37.1	S	40	15.0
Achernar	+0.5	279	45.3	S	57	10.1
Acrux	+1.3	117	29.1	S	63	10.3
Adhara	+1.5	199	31.7	S	28	59.2
Aldebaran	+0.9	235	08.5	N	16	32.0
Alioth	+1.8	110	40.2	N	55	53.6
Alkaid	+1.9	097	18.3	N	49	15.2
Al Na'ir	+1.7	332	02.5	S	46	53.8
Alnilam	+1.7	220	05.5	S	01	11.6
Alphard	+2.0	162	15.5	S	08	42.8
Alphecca	+2.2	070	30.2	N	26	40.6
Alpheratz	+2.1	302	02.3	N	29	09.8
Altair	+0.8	006	27.1	N	08	54.4
Ankaa	+2.4	297	34.4	S	42	14.0
Antares	+1.0	056	45.5	S	26	27.5
Arcturus	0.0	090	15.0	N	19	07.2
Atria	+1.9	051	47.9	S	69	03.2
Avior	+1.9	178	37.3	S	59	32.9
Bellatrix	+1.6	222	51.2	N	06	21.7
Betelgeuse	Var.	215	20.5	N	07	24.5
Canopus	-0.7	208	15.1	S	52	41.9
Capella	+0.1	224	53.8	N	46	00.4
Deneb	+1.3	353	50.0	N	45	19.9
Denebola	+2.1	126	53.1	N	14	30.1
Diphda	+2.0	293	14.7	S	17	54.8
Dubhe	+1.8	138	11.6	N	61	40.9
Elnath	+1.7	222	31.9	N	28	36.9
Eltanin	+2.2	035	04.8	N	51	29.7
Enif	+2.4	338	05.9	N	09	56.3
Fomalhaut	+1.2	319	42.8	S	29	33.1
Gacrux	+1.6	116	20.6	S	57	11.1
Gienah	+2.6	120	11.7	S	17	36.7
Hadar	+0.6	093	07.7	S	60	26.2
Hamal	+2.0	272	19.7	N	23	31.4
Kaus Australis	+1.9	028	03.0	S	34	22.6
Kochab	+2.1	081	39.2	N	74	06.5
Markab	+2.5	317	57.1	N	15	16.7
Menkar	+2.5	258	34.0	N	04	08.5
Menkent	+2.1	092	27.1	S	36	26.0
Miaplacidus	+1.7	165	59.1	S	69	46.1
Mirfak	+1.8	252	59.5	N	49	54.2
Nunki	+2.0	020	17.4	S	26	16.7
Peacock	+1.9	357	38.2	S	56	41.6
Polaris	+2.0	262	05.7	N	89	18.9
Pollux	+1.1	187	47.2	N	27	59.6
Procyon	+0.4	189	19.0	N	05	11.5
Rasalhague	+2.1	040	25.5	N	12	33.4
Regulus	+1.4	152	03.0	N	11	54.3
Rigel	+0.1	225	31.1	S	08	11.1
Rigil Kentaurus	-0.3	084	11.4	S	60	53.4
Sabik	+2.4	046	31.7	S	15	44.3
Schedar	+2.2	293	59.2	N	56	36.5
Shaula	+1.6	040	41.2	S	37	06.7

Elaboração de um Almanaque Náutico

S Octantis	+5.5	344	11.8	S 88	54.2	
Sirius	-1.5	202	52.9	S 16	43.9	
Spica	+1.0	102	50.7	S 11	13.6	
Suhail	+2.2	167	11.8	S 43	29.0	
Vega	0.0	024	57.7	N 38	48.1	
Zubenelgenubi	+2.8	081	24.8	S 16	05.6	
*Sun		135	44.5	N 04	54.6	SD 15.9*
*Venus	-4.2	178	47.0	N 18	04.9	HP 00.2*
*Mars	+1.2	085	41.1	S 15	47.7	HP 00.1*
*Jupiter	-2.4	230	04.3	N 21	50.3	*
*Saturn	+0.8	098	18.5	S 08	18.5	*
*Moon		212	04.0	N 20	38.6	HP 54.9*
Aries		304	18.6			

100413/0300

Acamar	+3.2	198	49.6	S 40	15.3	
Achernar	+0.5	218	58.2	S 57	10.2	
Acrux	+1.3	056	39.5	S 63	10.5	
Adhara	+1.5	138	43.4	S 28	59.8	
Aldebaran	+0.9	174	20.5	N 16	32.0	
Alioth	+1.8	049	51.0	N 55	53.2	
Alkaid	+1.9	036	29.3	N 49	14.7	
Al Na'ir	+1.7	271	15.0	S 46	53.6	
Alnilam	+1.7	159	17.4	S 01	11.9	
Alphard	+2.0	101	26.8	S 08	43.3	
Alphecca	+2.2	009	41.6	N 26	40.1	
Alpheratz	+2.1	241	14.7	N 29	09.7	
Altair	+0.8	305	39.2	N 08	54.2	
Ankaa	+2.4	236	47.0	S 42	14.0	
Antares	+1.0	355	56.9	S 26	27.6	
Arcturus	0.0	029	26.3	N 19	06.7	
Atria	+1.9	350	58.5	S 69	02.8	
Avior	+1.9	117	48.6	S 59	33.6	
Bellatrix	+1.6	162	03.1	N 06	21.5	
Betelgeuse	Var.	154	32.3	N 07	24.3	
Canopus	-0.7	147	27.1	S 52	42.6	
Capella	+0.1	164	05.6	N 46	00.6	
Deneb	+1.3	293	02.5	N 45	19.5	
Denebola	+2.1	066	04.2	N 14	29.7	
Diphda	+2.0	232	27.1	S 17	54.9	
Dubhe	+1.8	077	22.1	N 61	40.7	
Elnath	+1.7	161	43.7	N 28	37.0	
Eltanin	+2.2	334	16.8	N 51	29.0	
Enif	+2.4	277	18.2	N 09	56.1	
Fomalhaut	+1.2	258	55.2	S 29	33.0	
Gacrux	+1.6	055	31.2	S 57	11.4	
Gienah	+2.6	059	22.9	S 17	37.1	
Hadar	+0.6	032	18.2	S 60	26.2	
Hamal	+2.0	211	32.0	N 23	31.4	
Kaus Australis	+1.9	327	14.6	S 34	22.5	
Kochab	+2.1	020	49.7	N 74	05.9	
Markab	+2.5	257	09.5	N 15	16.5	
Menkar	+2.5	197	46.2	N 04	08.3	
Menkent	+2.1	031	38.1	S 36	26.2	
Miaplacidus	+1.7	105	10.0	S 69	46.7	

Elaboração de um Almanaque Náutico

Mirfak	+1.8	192	11.7	N 49	54.5	
Nunki	+2.0	319	29.2	S 26	16.6	
Peacock	+1.9	296	50.4	S 56	41.2	
Polaris	+2.0	201	46.4	N 89	19.4	
Pollux	+1.1	126	58.6	N 27	59.5	
Procyon	+0.4	128	30.6	N 05	11.2	
Rasalhague	+2.1	339	37.2	N 12	33.0	
Regulus	+1.4	091	14.2	N 11	53.9	
Rigel	+0.1	164	43.1	S 08	11.5	
Rigil Kentaurus	-0.3	023	22.0	S 60	53.3	
Sabik	+2.4	345	43.3	S 15	44.4	
Schedar	+2.2	233	12.0	N 56	36.5	
Shaula	+1.6	339	52.7	S 37	06.6	
S Octantis	+5.5	283	40.8	S 88	53.7	
Sirius	-1.5	142	04.6	S 16	44.4	
Spica	+1.0	042	01.8	S 11	14.0	
Suhail	+2.2	106	23.1	S 43	29.6	
Vega	0.0	324	09.7	N 38	47.6	
Zubenelgenubi	+2.8	020	36.0	S 16	05.9	
*Sun		224	39.8	N 07	58.3	SD 16.0*
*Venus	-3.9	221	17.2	N 08	09.3	HP 00.1*
*Mars	+1.2	222	49.4	N 08	11.4	HP 00.1*
*Jupiter	-2.1	171	26.1	N 22	02.6	*
*Saturn	+0.2	025	29.3	S 12	08.1	*
*Moon		228	40.5	N 09	09.0	HP 56.1*
Aries		243	30.8			

110713/1030

Acamar	+3.2	042	18.5	S 40	14.9	
Achernar	+0.5	062	26.8	S 57	09.8	
Acrux	+1.3	260	09.3	S 63	10.8	
Adhara	+1.5	342	12.9	S 28	59.5	
Aldebaran	+0.9	017	49.6	N 16	32.0	
Alioth	+1.8	253	20.7	N 55	53.5	
Alkaid	+1.9	239	58.8	N 49	15.0	
Al Na'ir	+1.7	114	43.3	S 46	53.4	
Alnilam	+1.7	002	46.6	S 01	11.7	
Alphard	+2.0	304	56.3	S 08	43.2	
Alphecca	+2.2	213	10.7	N 26	40.4	
Alpheratz	+2.1	084	43.3	N 29	09.9	
Altair	+0.8	149	07.8	N 08	54.5	
Ankaa	+2.4	080	15.6	S 42	13.6	
Antares	+1.0	199	25.9	S 26	27.6	
Arcturus	0.0	232	55.6	N 19	06.9	
Atria	+1.9	194	27.2	S 69	03.1	
Avior	+1.9	321	18.6	S 59	33.4	
Bellatrix	+1.6	005	32.3	N 06	21.6	
Betelgeuse	Var.	358	01.6	N 07	24.4	
Canopus	-0.7	350	56.7	S 52	42.2	
Capella	+0.1	007	34.8	N 46	00.5	
Deneb	+1.3	136	31.0	N 45	19.8	
Denebola	+2.1	269	33.7	N 14	29.8	
Diphda	+2.0	075	55.8	S 17	54.6	
Dubhe	+1.8	280	52.1	N 61	40.8	

Elaboração de um Almanaque Náutico

Elnath	+1.7	005	12.9	N 28	36.9	
Eltanin	+2.2	177	45.6	N 51	29.5	
Enif	+2.4	120	46.8	N 09	56.4	
Fomalhaut	+1.2	102	23.7	S 29	32.7	
Gacrux	+1.6	259	00.9	S 57	11.6	
Gienah	+2.6	262	52.3	S 17	37.1	
Hadar	+0.6	235	47.6	S 60	26.5	
Hamal	+2.0	055	00.8	N 23	31.5	
Kaus Australis	+1.9	170	43.3	S 34	22.5	
Kochab	+2.1	224	19.5	N 74	06.3	
Markab	+2.5	100	38.1	N 15	16.8	
Menkar	+2.5	041	15.2	N 04	08.5	
Menkent	+2.1	235	07.4	S 36	26.3	
Miaplacidus	+1.7	308	40.4	S 69	46.6	
Mirfak	+1.8	035	40.5	N 49	54.3	
Nunki	+2.0	162	57.9	S 26	16.6	
Peacock	+1.9	140	18.6	S 56	41.2	
Polaris	+2.0	044	57.3	N 89	19.0	
Pollux	+1.1	330	28.1	N 27	59.5	
Procyon	+0.4	332	00.0	N 05	11.3	
Rasalhague	+2.1	183	06.1	N 12	33.2	
Regulus	+1.4	294	43.7	N 11	54.0	
Rigel	+0.1	008	12.3	S 08	11.2	
Rigil Kentaurus	-0.3	226	51.4	S 60	53.6	
Sabik	+2.4	189	12.2	S 15	44.4	
Schedar	+2.2	076	40.3	N 56	36.5	
Shaula	+1.6	183	21.5	S 37	06.7	
S Octantis	+5.5	126	44.5	S 88	53.7	
Sirius	-1.5	345	34.0	S 16	44.2	
Spica	+1.0	245	31.2	S 11	13.9	
Suhail	+2.2	309	52.8	S 43	29.5	
Vega	0.0	167	38.5	N 38	48.0	
Zubenelgenubi	+2.8	224	05.2	S 16	05.9	
*Sun		336	07.2	N 22	02.9	SD 15.7*
*Venus	-3.9	307	23.8	N 17	25.8	HP 00.1*
*Mars	+1.6	358	34.4	N 23	55.5	HP 00.1*
*Jupiter	-1.9	353	12.2	N 23	11.7	*
*Saturn	+0.6	233	36.6	S 10	48.5	*
*Moon		302	19.7	N 08	48.4	HP 54.8*
Aries		087	00.0			

121013/2000

Acamar	+3.2	276	51.1	S 40	14.9	
Achernar	+0.5	296	59.4	S 57	09.9	
Acrux	+1.3	134	42.9	S 63	10.4	
Adhara	+1.5	216	45.6	S 28	59.4	
Aldebaran	+0.9	252	22.2	N 16	32.1	
Alioth	+1.8	127	54.4	N 55	53.1	
Alkaid	+1.9	114	32.5	N 49	14.8	
Al Na'ir	+1.7	349	16.5	S 46	53.6	
Alnilam	+1.7	237	19.3	S 01	11.6	
Alphard	+2.0	179	29.3	S 08	43.1	
Alphecca	+2.2	087	44.4	N 26	40.4	
Alpheratz	+2.1	319	16.2	N 29	10.2	
Altair	+0.8	023	41.3	N 08	54.6	

Elaboração de um Almanaque Náutico

Ankaa	+2.4	314	48.4	S	42	13.8	
Antares	+1.0	073	59.5	S	26	27.6	
Arcturus	0.0	107	29.2	N	19	06.8	
Atria	+1.9	069	01.5	S	69	03.2	
Avior	+1.9	195	51.3	S	59	33.0	
Bellatrix	+1.6	240	04.9	N	06	21.7	
Betelgeuse	Var.	232	34.2	N	07	24.5	
Canopus	-0.7	225	29.2	S	52	42.0	
Capella	+0.1	242	07.2	N	46	00.4	
Deneb	+1.3	011	04.5	N	45	20.2	
Denebola	+2.1	144	07.0	N	14	29.7	
Diphda	+2.0	310	28.6	S	17	54.5	
Dubhe	+1.8	155	25.3	N	61	40.4	
Elnath	+1.7	239	45.5	N	28	37.0	
Eltanin	+2.2	052	19.5	N	51	29.7	
Enif	+2.4	355	20.0	N	09	56.6	
Fomalhaut	+1.2	336	56.8	S	29	32.8	
Gacrux	+1.6	133	34.5	S	57	11.3	
Gienah	+2.6	137	25.7	S	17	37.0	
Hadar	+0.6	110	21.5	S	60	26.3	
Hamal	+2.0	289	33.4	N	23	31.7	
Kaus Australis	+1.9	045	16.9	S	34	22.6	
Kochab	+2.1	098	54.5	N	74	06.1	
Markab	+2.5	335	11.1	N	15	17.0	
Menkar	+2.5	275	47.8	N	04	08.7	
Menkent	+2.1	109	41.0	S	36	26.2	
Miaplacidus	+1.7	183	13.2	S	69	46.2	
Mirfak	+1.8	270	12.9	N	49	54.5	
Nunki	+2.0	037	31.4	S	26	16.6	
Peacock	+1.9	014	52.1	S	56	41.5	
Polaris	+2.0	278	48.7	N	89	19.2	
Pollux	+1.1	205	00.8	N	27	59.4	
Procyon	+0.4	206	32.8	N	05	11.3	
Rasalhague	+2.1	057	39.7	N	12	33.3	
Regulus	+1.4	169	16.8	N	11	53.9	
Rigel	+0.1	242	45.0	S	08	11.1	
Rigil Kentaurus	-0.3	101	25.3	S	60	53.4	
Sabik	+2.4	063	45.8	S	15	44.3	
Schedar	+2.2	311	12.9	N	56	36.9	
Shaula	+1.6	057	55.1	S	37	06.7	
S Octantis	+5.5	001	22.9	S	88	54.1	
Sirius	-1.5	220	06.7	S	16	44.1	
Spica	+1.0	120	04.6	S	11	13.9	
Suhail	+2.2	184	25.7	S	43	29.1	
Vega	0.0	042	12.2	N	38	48.2	
Zubenelgenubi	+2.8	098	38.8	S	16	05.8	
*Sun		123	24.6	S	07	41.3	SD 16.0*
*Venus	-4.3	078	26.0	S	24	01.1	HP 00.2*
*Mars	+1.6	170	28.9	N	13	18.8	HP 00.1*
*Jupiter	-2.3	210	29.3	N	21	58.2	*
*Saturn	+0.6	101	58.8	S	13	11.3	*
*Moon		019	19.6	S	14	47.4	HP 59.3*
Aries		321	33.3				

130114/0200

Acamar	+3.2	097	46.7	S	40	15.2
Achernar	+0.5	117	55.3	S	57	10.3
Acrux	+1.3	315	37.4	S	63	10.4
Adhara	+1.5	037	40.7	S	28	59.7
Aldebaran	+0.9	073	17.5	N	16	32.1
Alioth	+1.8	308	49.2	N	55	52.7
Alkaid	+1.9	295	27.5	N	49	14.3
Al Na'ir	+1.7	170	12.5	S	46	53.7
Alnilam	+1.7	058	14.5	S	01	11.8
Alphard	+2.0	000	24.2	S	08	43.3
Alphecca	+2.2	268	39.7	N	26	40.0
Alpheratz	+2.1	140	12.0	N	29	10.2
Altair	+0.8	204	37.0	N	08	54.5
Ankaa	+2.4	135	44.3	S	42	14.0
Antares	+1.0	254	54.9	S	26	27.6
Arcturus	0.0	288	24.3	N	19	06.5
Atria	+1.9	249	56.8	S	69	02.8
Avior	+1.9	016	46.0	S	59	33.4
Bellatrix	+1.6	061	00.1	N	06	21.6
Betelgeuse	Var.	053	29.4	N	07	24.4
Canopus	-0.7	046	24.3	S	52	42.4
Capella	+0.1	063	02.3	N	46	00.6
Deneb	+1.3	192	00.5	N	45	20.0
Denebola	+2.1	325	02.0	N	14	29.4
Diphda	+2.0	131	24.4	S	17	54.7
Dubhe	+1.8	336	19.7	N	61	40.2
Elnath	+1.7	060	40.6	N	28	37.0
Eltanin	+2.2	233	15.3	N	51	29.2
Enif	+2.4	176	15.8	N	09	56.5
Fomalhaut	+1.2	157	52.6	S	29	33.0
Gacrux	+1.6	314	29.1	S	57	11.3
Gienah	+2.6	318	20.6	S	17	37.2
Hadar	+0.6	291	16.2	S	60	26.1
Hamal	+2.0	110	29.1	N	23	31.8
Kaus Australis	+1.9	226	12.5	S	34	22.5
Kochab	+2.1	279	49.4	N	74	05.6
Markab	+2.5	156	07.0	N	15	17.0
Menkar	+2.5	096	43.3	N	04	08.6
Menkent	+2.1	290	36.0	S	36	26.1
Miaplacidus	+1.7	004	07.5	S	69	46.5
Mirfak	+1.8	091	08.3	N	49	54.7
Nunki	+2.0	218	27.0	S	26	16.6
Peacock	+1.9	195	48.1	S	56	41.3
Polaris	+2.0	099	54.1	N	89	19.7
Pollux	+1.1	025	55.7	N	27	59.3
Procyon	+0.4	027	27.8	N	05	11.1
Rasalhague	+2.1	238	35.2	N	12	33.1
Regulus	+1.4	350	11.7	N	11	53.7
Rigel	+0.1	063	40.2	S	08	11.4
Rigil Kentaurus	-0.3	282	20.2	S	60	53.2
Sabik	+2.4	244	41.2	S	15	44.4
Schedar	+2.2	132	09.0	N	56	37.1
Shaula	+1.6	238	50.6	S	37	06.6
S Octantis	+5.5	182	38.6	S	88	53.9
Sirius	-1.5	041	01.9	S	16	44.4
Spica	+1.0	300	59.7	S	11	14.0
Suhail	+2.2	005	20.6	S	43	29.4
Vega	0.0	223	07.9	N	38	47.9

Elaboração de um Almanaque Náutico

Zubenelgenubi	+2.8	279	33.9	S 16	05.9	
*Sun		207	52.3	S 21	30.6	SD 16.3*
*Venus	-4.4	211	26.9	S 16	28.1	HP 00.5*
*Mars	+0.6	306	11.5	S 04	25.9	HP 00.1*
*Jupiter	-2.7	036	43.9	N 22	47.9	*
*Saturn	+0.6	272	56.4	S 15	58.6	*
*Moon		064	17.3	N 19	28.2	HP 54.3*
Aries		142	28.9			

140614/0700

Acamar	+3.2	322	48.4	S 40	14.8	
Achernar	+0.5	342	56.9	S 57	09.6	
Acrux	+1.3	180	38.4	S 63	11.1	
Adhara	+1.5	262	42.6	S 28	59.8	
Aldebaran	+0.9	298	19.2	N 16	32.1	
Alioth	+1.8	173	50.2	N 55	53.1	
Alkaid	+1.9	160	28.4	N 49	14.7	
Al Na'ir	+1.7	035	13.0	S 46	53.1	
Alnilam	+1.7	283	16.3	S 01	11.8	
Alphard	+2.0	225	25.8	S 08	43.5	
Alphecca	+2.2	133	40.4	N 26	40.1	
Alpheratz	+2.1	005	13.0	N 29	10.1	
Altair	+0.8	069	37.5	N 08	54.5	
Ankaa	+2.4	000	45.4	S 42	13.4	
Antares	+1.0	119	55.3	S 26	27.7	
Arcturus	0.0	153	25.1	N 19	06.6	
Atria	+1.9	114	55.9	S 69	03.1	
Avior	+1.9	241	48.4	S 59	33.7	
Bellatrix	+1.6	286	01.9	N 06	21.6	
Betelgeuse	Var.	278	31.1	N 07	24.4	
Canopus	-0.7	271	26.6	S 52	42.4	
Capella	+0.1	288	04.2	N 46	00.5	
Deneb	+1.3	057	00.9	N 45	19.9	
Denebola	+2.1	190	03.2	N 14	29.5	
Diphda	+2.0	356	25.6	S 17	54.3	
Dubhe	+1.8	201	21.3	N 61	40.6	
Elnath	+1.7	285	42.4	N 28	37.0	
Eltanin	+2.2	098	15.5	N 51	29.3	
Enif	+2.4	041	16.5	N 09	56.5	
Fomalhaut	+1.2	022	53.4	S 29	32.5	
Gacrux	+1.6	179	30.1	S 57	11.9	
Gienah	+2.6	183	21.7	S 17	37.5	
Hadar	+0.6	156	16.6	S 60	26.7	
Hamal	+2.0	335	30.4	N 23	31.7	
Kaus Australis	+1.9	091	12.7	S 34	22.4	
Kochab	+2.1	144	49.5	N 74	06.0	
Markab	+2.5	021	07.8	N 15	17.0	
Menkar	+2.5	321	44.8	N 04	08.6	
Menkent	+2.1	155	36.7	S 36	26.6	
Miaplacidus	+1.7	229	10.2	S 69	47.0	
Mirfak	+1.8	316	10.0	N 49	54.5	
Nunki	+2.0	083	27.4	S 26	16.5	
Peacock	+1.9	060	48.0	S 56	40.9	
Polaris	+2.0	325	18.7	N 89	19.3	

Elaboração de um Almanaque Náutico

Pollux	+1.1	250	57.5	N 27	59.4	
Procyon	+0.4	252	29.6	N 05	11.1	
Rasalhague	+2.1	103	35.7	N 12	33.1	
Regulus	+1.4	215	13.2	N 11	53.7	
Rigel	+0.1	288	42.0	S 08	11.3	
Rigil Kentaurus	-0.3	147	20.4	S 60	53.8	
Sabik	+2.4	109	41.6	S 15	44.4	
Schedar	+2.2	357	10.1	N 56	36.7	
Shaula	+1.6	103	50.8	S 37	06.7	
S Octantis	+5.5	047	11.0	S 88	53.4	
Sirius	-1.5	266	03.7	S 16	44.4	
Spica	+1.0	166	00.6	S 11	14.3	
Suhail	+2.2	230	22.4	S 43	29.8	
Vega	0.0	088	08.3	N 38	47.9	
Zubenelgenubi	+2.8	144	34.6	S 16	06.1	
*Sun		284	56.9	N 23	15.5	SD 15.7*
*Venus	-3.9	320	21.9	N 15	40.6	HP 00.1*
*Mars	-0.2	175	50.4	S 05	05.0	HP 00.2*
*Jupiter	-1.8	252	33.8	N 21	48.4	*
*Saturn	+0.3	141	35.4	S 14	46.8	*
*Moon		088	28.0	S 18	20.3	HP 60.4*
Aries		007	30.2			

151214/2100

Acamar	+3.2	354	43.5	S 40	14.9	
Achernar	+0.5	014	52.0	S 57	09.9	
Acrux	+1.3	212	34.4	S 63	10.6	
Adhara	+1.5	294	37.6	S 28	59.7	
Aldebaran	+0.9	330	14.1	N 16	32.2	
Alioth	+1.8	205	46.3	N 55	52.5	
Alkaid	+1.9	192	24.7	N 49	14.2	
Al Na'ir	+1.7	067	08.9	S 46	53.4	
Alnilam	+1.7	315	11.2	S 01	11.7	
Alphard	+2.0	257	21.1	S 08	43.5	
Alphecca	+2.2	165	36.7	N 26	39.9	
Alpheratz	+2.1	037	08.6	N 29	10.6	
Altair	+0.8	101	33.8	N 08	54.7	
Ankaa	+2.4	032	40.9	S 42	13.7	
Antares	+1.0	151	51.6	S 26	27.7	
Arcturus	0.0	185	21.3	N 19	06.3	
Atria	+1.9	146	53.0	S 69	03.0	
Avior	+1.9	273	43.3	S 59	33.4	
Bellatrix	+1.6	317	56.8	N 06	21.6	
Betelgeuse	Var.	310	26.1	N 07	24.4	
Canopus	-0.7	303	21.4	S 52	42.3	
Capella	+0.1	319	58.7	N 46	00.6	
Deneb	+1.3	088	57.3	N 45	20.4	
Denebola	+2.1	221	58.9	N 14	29.2	
Diphda	+2.0	028	21.0	S 17	54.4	
Dubhe	+1.8	233	16.7	N 61	39.9	
Elnath	+1.7	317	37.2	N 28	37.0	
Eltanin	+2.2	130	12.5	N 51	29.4	
Enif	+2.4	073	12.5	N 09	56.8	
Fomalhaut	+1.2	054	49.2	S 29	32.6	
Gacrux	+1.6	211	26.0	S 57	11.5	

Elaboração de um Almanaque Náutico

Gienah	+2.6	215	17.5	S 17	37.4	
Hadar	+0.6	188	13.0	S 60	26.4	
Hamal	+2.0	007	25.6	N 23	32.0	
Kaus Australis	+1.9	123	09.1	S 34	22.5	
Kochab	+2.1	176	47.4	N 74	05.5	
Markab	+2.5	053	03.6	N 15	17.3	
Menkar	+2.5	353	40.0	N 04	08.8	
Menkent	+2.1	187	32.8	S 36	26.4	
Miaplacidus	+1.7	261	05.0	S 69	46.6	
Mirfak	+1.8	348	04.6	N 49	54.9	
Nunki	+2.0	115	23.7	S 26	16.5	
Peacock	+1.9	092	44.4	S 56	41.2	
Polaris	+2.0	356	22.0	N 89	19.8	
Pollux	+1.1	282	52.4	N 27	59.1	
Procyon	+0.4	284	24.6	N 05	11.0	
Rasalhague	+2.1	135	32.1	N 12	33.2	
Regulus	+1.4	247	08.6	N 11	53.5	
Rigel	+0.1	320	37.0	S 08	11.3	
Rigil Kentaurus	-0.3	179	17.0	S 60	53.4	
Sabik	+2.4	141	38.0	S 15	44.4	
Schedar	+2.2	029	05.3	N 56	37.4	
Shaula	+1.6	135	47.2	S 37	06.7	
S Octantis	+5.5	079	21.9	S 88	53.8	
Sirius	-1.5	297	58.7	S 16	44.4	
Spica	+1.0	197	56.6	S 11	14.2	
Suhail	+2.2	262	17.6	S 43	29.5	
Vega	0.0	120	04.9	N 38	48.1	
Zubenelgenubi	+2.8	176	30.8	S 16	06.0	
*Sun		136	11.1	S 23	17.4	SD 16.2*
*Venus	-3.9	122	18.7	S 24	10.4	HP 00.1*
*Mars	+1.0	088	12.8	S 19	22.7	HP 00.1*
*Jupiter	-2.3	254	16.4	N 14	44.7	*
*Saturn	+0.5	162	04.4	S 18	05.6	*
*Moon		211	56.5	S 04	02.6	HP 55.0*
Aries		039	26.3			

160315/0900

Acamar	+3.2	263	56.0	S 40	15.0	
Achernar	+0.5	284	04.8	S 57	09.8	
Acrux	+1.3	121	45.4	S 63	11.0	
Adhara	+1.5	203	49.7	S 29	00.0	
Aldebaran	+0.9	239	26.4	N 16	32.1	
Alioth	+1.8	114	57.5	N 55	52.5	
Alkaid	+1.9	101	35.9	N 49	14.1	
Al Na'ir	+1.7	336	21.0	S 46	53.1	
Alnilam	+1.7	224	23.4	S 01	11.9	
Alphard	+2.0	166	32.9	S 08	43.8	
Alphecca	+2.2	074	48.1	N 26	39.7	
Alpheratz	+2.1	306	20.9	N 29	10.4	
Altair	+0.8	010	45.5	N 08	54.5	
Ankaa	+2.4	301	53.3	S 42	13.5	
Antares	+1.0	061	03.0	S 26	27.7	
Arcturus	0.0	094	32.7	N 19	06.1	
Atria	+1.9	056	03.6	S 69	02.8	

Elaboração de um Almanaque Náutico

Avior	+1.9	182	55.3	S	59	33.9	
Bellatrix	+1.6	227	09.0	N	06	21.5	
Betelgeuse	Var.	219	38.3	N	07	24.3	
Canopus	-0.7	212	33.8	S	52	42.7	
Capella	+0.1	229	11.0	N	46	00.7	
Deneb	+1.3	358	09.2	N	45	20.0	
Denebola	+2.1	131	10.4	N	14	29.1	
Diphda	+2.0	297	33.3	S	17	54.4	
Dubhe	+1.8	142	28.0	N	61	40.0	
Elnath	+1.7	226	49.4	N	28	37.0	
Eltanin	+2.2	039	23.9	N	51	29.1	
Enif	+2.4	342	24.5	N	09	56.7	
Fomalhaut	+1.2	324	01.4	S	29	32.5	
Gacrux	+1.6	120	37.2	S	57	11.9	
Gienah	+2.6	124	29.0	S	17	37.7	
Hadar	+0.6	097	23.9	S	60	26.6	
Hamal	+2.0	276	37.9	N	23	31.9	
Kaus Australis	+1.9	032	20.6	S	34	22.3	
Kochab	+2.1	085	57.8	N	74	05.4	
Markab	+2.5	322	15.7	N	15	17.2	
Menkar	+2.5	262	52.3	N	04	08.7	
Menkent	+2.1	096	44.1	S	36	26.6	
Miaplacidus	+1.7	170	16.8	S	69	47.1	
Mirfak	+1.8	257	17.1	N	49	54.9	
Nunki	+2.0	024	35.3	S	26	16.4	
Peacock	+1.9	001	56.1	S	56	40.9	
Polaris	+2.0	266	07.2	N	89	19.9	
Pollux	+1.1	192	04.4	N	27	59.2	
Procyon	+0.4	193	36.6	N	05	10.8	
Rasalhague	+2.1	044	43.6	N	12	33.0	
Regulus	+1.4	156	20.3	N	11	53.3	
Rigel	+0.1	229	49.2	S	08	11.4	
Rigil Kentaurus	-0.3	088	27.8	S	60	53.6	
Sabik	+2.4	050	49.4	S	15	44.5	
Schedar	+2.2	298	17.9	N	56	37.2	
Shaula	+1.6	044	58.6	S	37	06.6	
S Octantis	+5.5	348	35.0	S	88	53.3	
Sirius	-1.5	207	10.9	S	16	44.6	
Spica	+1.0	107	08.0	S	11	14.5	
Suhail	+2.2	171	29.5	S	43	30.0	
Vega	0.0	029	16.5	N	38	47.8	
Zubenelgenubi	+2.8	085	42.2	S	16	06.2	
*Sun		312	48.9	S	01	48.5	SD 16.1*
*Venus	-3.9	281	57.2	N	11	03.4	HP 00.1*
*Mars	+1.3	291	24.0	N	06	59.3	HP 00.1*
*Jupiter	-2.4	172	26.2	N	17	43.4	*
*Saturn	+0.4	065	15.3	S	19	02.9	*
*Moon		008	17.8	S	15	25.1	HP 59.7*
Aries		308	38.3				
170815/1700							
Acamar	+3.2	176	02.6	S	40	14.4	
Achernar	+0.5	196	10.9	S	57	09.2	
Acrux	+1.3	033	53.3	S	63	11.3	
Adhara	+1.5	115	57.0	S	28	59.6	

Elaboração de um Almanaque Náutico

Aldebaran	+0.9	151	33.1	N 16	32.2	
Alioth	+1.8	027	05.3	N 55	52.8	
Alkaid	+1.9	013	43.4	N 49	14.5	
Al Na'ir	+1.7	248	26.9	S 46	52.9	
Alnilam	+1.7	136	30.4	S 01	11.6	
Alphard	+2.0	078	40.3	S 08	43.6	
Alphecca	+2.2	346	55.1	N 26	40.1	
Alpheratz	+2.1	218	27.0	N 29	10.6	
Altair	+0.8	282	51.9	N 08	54.9	
Ankaa	+2.4	213	59.4	S 42	13.0	
Antares	+1.0	333	09.7	S 26	27.8	
Arcturus	0.0	006	39.9	N 19	06.4	
Atria	+1.9	328	09.9	S 69	03.3	
Avior	+1.9	095	03.3	S 59	33.6	
Bellatrix	+1.6	139	15.9	N 06	21.7	
Betelgeuse	Var.	131	45.2	N 07	24.4	
Canopus	-0.7	124	41.2	S 52	42.2	
Capella	+0.1	141	17.8	N 46	00.4	
Deneb	+1.3	270	15.3	N 45	20.5	
Denebola	+2.1	043	17.8	N 14	29.2	
Diphda	+2.0	209	39.6	S 17	53.9	
Dubhe	+1.8	054	36.1	N 61	40.1	
Elnath	+1.7	138	56.3	N 28	37.0	
Eltanin	+2.2	311	30.5	N 51	29.7	
Enif	+2.4	254	30.7	N 09	57.0	
Fomalhaut	+1.2	236	07.4	S 29	32.1	
Gacrux	+1.6	032	44.9	S 57	12.1	
Gienah	+2.6	036	36.4	S 17	37.7	
Hadar	+0.6	009	31.2	S 60	27.0	
Hamal	+2.0	188	44.3	N 23	32.0	
Kaus Australis	+1.9	304	26.9	S 34	22.4	
Kochab	+2.1	358	05.9	N 74	05.9	
Markab	+2.5	234	21.9	N 15	17.5	
Menkar	+2.5	174	58.8	N 04	09.0	
Menkent	+2.1	008	51.3	S 36	26.8	
Miaplacidus	+1.7	082	25.5	S 69	47.0	
Mirfak	+1.8	169	23.5	N 49	54.6	
Nunki	+2.0	296	41.6	S 26	16.4	
Peacock	+1.9	274	01.8	S 56	40.9	
Polaris	+2.0	177	42.9	N 89	19.4	
Pollux	+1.1	104	11.6	N 27	59.1	
Procyon	+0.4	105	43.8	N 05	11.0	
Rasalhague	+2.1	316	50.3	N 12	33.3	
Regulus	+1.4	068	27.7	N 11	53.4	
Rigel	+0.1	141	56.1	S 08	11.1	
Rigil Kentaurus	-0.3	000	35.1	S 60	54.0	
Sabik	+2.4	322	56.1	S 15	44.4	
Schedar	+2.2	210	23.6	N 56	37.2	
Shaula	+1.6	317	05.0	S 37	06.7	
S Octantis	+5.5	260	07.7	S 88	53.4	
Sirius	-1.5	119	18.0	S 16	44.3	
Spica	+1.0	019	15.2	S 11	14.4	
Suhail	+2.2	083	37.1	S 43	29.8	
Vega	0.0	301	23.0	N 38	48.3	
Zubenelgenubi	+2.8	357	49.2	S 16	06.2	
*Sun		073	58.7	N 13	21.6	SD 15.8*
*Venus	-4.1	079	28.3	N 06	45.3	HP 00.5*
*Mars	+1.7	092	28.7	N 19	53.3	HP 00.1*
*Jupiter	-1.7	067	04.0	N 11	48.1	*

Elaboração de um Almanaque Náutico

*Saturn	+0.5	344 04.5	S 17 57.0	*
*Moon		042 23.0	N 00 24.5	HP 54.0*
Aries		220 45.3		

181115/2200

Acamar	+3.2	342 54.3	S 40 14.6	
Achernar	+0.5	003 02.8	S 57 09.6	
Acrux	+1.3	200 45.2	S 63 10.9	
Adhara	+1.5	282 48.5	S 28 59.7	
Aldebaran	+0.9	318 24.7	N 16 32.3	
Alioth	+1.8	193 57.4	N 55 52.3	
Alkaid	+1.9	180 35.7	N 49 14.1	
Al Na'ir	+1.7	055 19.3	S 46 53.1	
Alnilam	+1.7	303 22.0	S 01 11.7	
Alphard	+2.0	245 32.0	S 08 43.7	
Alphecca	+2.2	153 47.6	N 26 39.9	
Alpheratz	+2.1	025 19.1	N 29 10.9	
Altair	+0.8	089 44.4	N 08 54.9	
Ankaa	+2.4	020 51.5	S 42 13.3	
Antares	+1.0	140 02.1	S 26 27.8	
Arcturus	0.0	173 32.1	N 19 06.1	
Atria	+1.9	135 02.9	S 69 03.2	
Avior	+1.9	261 54.6	S 59 33.5	
Bellatrix	+1.6	306 07.5	N 06 21.7	
Betelgeuse	Var.	298 36.8	N 07 24.4	
Canopus	-0.7	291 32.5	S 52 42.3	
Capella	+0.1	308 09.1	N 46 00.6	
Deneb	+1.3	077 08.0	N 45 20.7	
Denebola	+2.1	210 09.7	N 14 29.0	
Diphda	+2.0	016 31.6	S 17 54.0	
Dubhe	+1.8	221 27.6	N 61 39.6	
Elnath	+1.7	305 47.7	N 28 37.0	
Eltanin	+2.2	118 23.4	N 51 29.6	
Enif	+2.4	061 23.1	N 09 57.1	
Fomalhaut	+1.2	042 59.7	S 29 32.3	
Gacrux	+1.6	199 36.8	S 57 11.8	
Gienah	+2.6	203 28.3	S 17 37.6	
Hadar	+0.6	176 23.5	S 60 26.7	
Hamal	+2.0	355 36.1	N 23 32.2	
Kaus Australis	+1.9	111 19.5	S 34 22.4	
Kochab	+2.1	164 59.1	N 74 05.5	
Markab	+2.5	041 14.1	N 15 17.7	
Menkar	+2.5	341 50.6	N 04 09.0	
Menkent	+2.1	175 43.5	S 36 26.6	
Miaplacidus	+1.7	249 16.5	S 69 46.8	
Mirfak	+1.8	336 15.0	N 49 54.9	
Nunki	+2.0	103 34.1	S 26 16.4	
Peacock	+1.9	080 54.5	S 56 41.0	
Polaris	+2.0	344 08.7	N 89 19.9	
Pollux	+1.1	271 03.1	N 27 59.0	
Procyon	+0.4	272 35.4	N 05 10.9	
Rasalhague	+2.1	123 42.8	N 12 33.3	
Regulus	+1.4	235 19.4	N 11 53.3	
Rigel	+0.1	308 47.7	S 08 11.1	
Rigil Kentaurus	-0.3	167 27.5	S 60 53.7	

Elaboração de um Almanaque Náutico

Sabik	+2.4	129	48.5	S 15	44.4	
Schedar	+2.2	017	15.6	N 56	37.7	
Shaula	+1.6	123	57.6	S 37	06.7	
S Octantis	+5.5	067	17.1	S 88	53.6	
Sirius	-1.5	286	09.6	S 16	44.4	
Spica	+1.0	186	07.3	S 11	14.5	
Suhail	+2.2	250	28.6	S 43	29.7	
Vega	0.0	108	15.7	N 38	48.3	
Zubenelgenubi	+2.8	164	41.5	S 16	06.2	
*Sun		153	42.0	S 19	18.4	SD 16.2*
*Venus	-4.3	196	32.3	S 02	52.9	HP 00.2*
*Mars	+1.6	203	44.8	S 00	07.1	HP 00.1*
*Jupiter	-1.9	216	51.4	N 05	07.7	*
*Saturn	+0.5	143	03.6	S 19	42.3	*
*Moon		064	32.1	S 11	18.3	HP 58.5*
Aries		027	37.6			

190216/0500

Acamar	+3.2	178	52.7	S 40	14.9	
Achernar	+0.5	199	01.4	S 57	09.7	
Acrux	+1.3	036	42.1	S 63	11.1	
Adhara	+1.5	118	46.4	S 29	00.1	
Aldebaran	+0.9	154	22.8	N 16	32.2	
Alioth	+1.8	029	54.4	N 55	52.2	
Alkaid	+1.9	016	32.9	N 49	13.8	
Al Na'ir	+1.7	251	17.6	S 46	53.0	
Alnilam	+1.7	139	19.9	S 01	11.9	
Alphard	+2.0	081	29.5	S 08	44.0	
Alphecca	+2.2	349	45.0	N 26	39.6	
Alpheratz	+2.1	221	17.4	N 29	10.7	
Altair	+0.8	285	42.3	N 08	54.7	
Ankaa	+2.4	216	49.9	S 42	13.4	
Antares	+1.0	335	59.6	S 26	27.8	
Arcturus	0.0	009	29.5	N 19	05.8	
Atria	+1.9	331	00.0	S 69	02.9	
Avior	+1.9	097	52.1	S 59	34.0	
Bellatrix	+1.6	142	05.5	N 06	21.5	
Betelgeuse	Var.	134	34.7	N 07	24.3	
Canopus	-0.7	127	30.5	S 52	42.7	
Capella	+0.1	144	07.1	N 46	00.7	
Deneb	+1.3	273	06.2	N 45	20.3	
Denebola	+2.1	046	07.1	N 14	28.8	
Diphda	+2.0	212	29.9	S 17	54.1	
Dubhe	+1.8	057	24.6	N 61	39.7	
Elnath	+1.7	141	45.7	N 28	37.0	
Eltanin	+2.2	314	21.1	N 51	29.1	
Enif	+2.4	257	21.2	N 09	56.9	
Fomalhaut	+1.2	238	57.9	S 29	32.3	
Gacrux	+1.6	035	33.8	S 57	12.0	
Gienah	+2.6	039	25.7	S 17	37.9	
Hadar	+0.6	012	20.4	S 60	26.7	
Hamal	+2.0	191	34.4	N 23	32.2	
Kaus Australis	+1.9	307	17.2	S 34	22.3	
Kochab	+2.1	000	55.6	N 74	05.2	

Elaboração de um Almanaque Náutico

Markab	+2.5	237	12.4	N 15	17.5	
Menkar	+2.5	177	48.8	N 04	08.9	
Menkent	+2.1	011	40.8	S 36	26.7	
Miaplacidus	+1.7	085	13.7	S 69	47.2	
Mirfak	+1.8	172	13.3	N 49	55.1	
Nunki	+2.0	299	31.9	S 26	16.4	
Peacock	+1.9	276	52.6	S 56	40.8	
Polaris	+2.0	180	36.8	N 89	20.1	
Pollux	+1.1	107	00.8	N 27	59.0	
Procyon	+0.4	108	33.1	N 05	10.7	
Rasalhague	+2.1	319	40.5	N 12	33.0	
Regulus	+1.4	071	16.9	N 11	53.1	
Rigel	+0.1	144	45.7	S 08	11.4	
Rigil Kentaurus	-0.3	003	24.5	S 60	53.7	
Sabik	+2.4	325	46.1	S 15	44.5	
Schedar	+2.2	213	14.3	N 56	37.6	
Shaula	+1.6	319	55.2	S 37	06.6	
S Octantis	+5.5	263	26.5	S 88	53.2	
Sirius	-1.5	122	07.4	S 16	44.7	
Spica	+1.0	022	04.7	S 11	14.7	
Suhail	+2.2	086	26.2	S 43	30.1	
Vega	0.0	304	13.5	N 38	47.9	
Zubenelgenubi	+2.8	000	38.9	S 16	06.4	
*Sun		251	31.6	S 11	28.7	SD 16.2*
*Venus	-3.9	278	45.6	S 19	47.7	HP 00.1*
*Mars	+0.5	352	04.9	S 17	18.2	HP 00.1*
*Jupiter	-2.4	051	40.2	N 05	03.2	*
*Saturn	+0.5	329	15.7	S 20	57.4	*
*Moon		114	13.4	N 17	30.4	HP 56.7*
Aries		223	35.5			

200516

Acamar	+3.2	074	01.6	S 40	14.5	
Achernar	+0.5	094	10.2	S 57	09.2	
Acrux	+1.3	291	50.7	S 63	11.6	
Adhara	+1.5	013	55.5	S 29	00.0	
Aldebaran	+0.9	049	31.7	N 16	32.2	
Alioth	+1.8	285	03.0	N 55	52.5	
Alkaid	+1.9	271	41.3	N 49	14.1	
Al Na'ir	+1.7	146	25.6	S 46	52.6	
Alnilam	+1.7	034	28.9	S 01	11.8	
Alphard	+2.0	336	38.5	S 08	44.0	
Alphecca	+2.2	244	53.3	N 26	39.7	
Alpheratz	+2.1	116	25.9	N 29	10.6	
Altair	+0.8	180	50.4	N 08	54.8	
Ankaa	+2.4	111	58.4	S 42	12.9	
Antares	+1.0	231	07.7	S 26	27.9	
Arcturus	0.0	264	37.9	N 19	06.0	
Atria	+1.9	226	07.2	S 69	03.1	
Avior	+1.9	353	01.7	S 59	34.2	
Bellatrix	+1.6	037	14.5	N 06	21.6	
Betelgeuse	Var.	029	43.7	N 07	24.3	
Canopus	-0.7	022	40.0	S 52	42.7	
Capella	+0.1	039	16.3	N 46	00.6	
Deneb	+1.3	168	14.2	N 45	20.2	

Elaboração de um Almanaque Náutico

Denebola	+2.1	301	15.8	N 14	28.9	
Diphda	+2.0	107	38.4	S 17	53.9	
Dubhe	+1.8	312	33.6	N 61	40.0	
Elnath	+1.7	036	54.8	N 28	37.0	
Eltanin	+2.2	209	29.0	N 51	29.2	
Enif	+2.4	152	29.4	N 09	57.0	
Fomalhaut	+1.2	134	06.2	S 29	32.0	
Gacrux	+1.6	290	42.4	S 57	12.5	
Gienah	+2.6	294	34.3	S 17	38.1	
Hadar	+0.6	267	28.6	S 60	27.1	
Hamal	+2.0	086	43.1	N 23	32.1	
Kaus Australis	+1.9	202	25.2	S 34	22.3	
Kochab	+2.1	256	03.4	N 74	05.5	
Markab	+2.5	132	20.7	N 15	17.5	
Menkar	+2.5	072	57.6	N 04	09.0	
Menkent	+2.1	266	49.1	S 36	27.0	
Miaplacidus	+1.7	340	23.5	S 69	47.5	
Mirfak	+1.8	067	22.2	N 49	54.9	
Nunki	+2.0	194	39.9	S 26	16.3	
Peacock	+1.9	172	00.3	S 56	40.5	
Polaris	+2.0	076	03.0	N 89	19.8	
Pollux	+1.1	002	09.9	N 27	59.1	
Procyon	+0.4	003	42.2	N 05	10.7	
Rasalhague	+2.1	214	48.6	N 12	33.0	
Regulus	+1.4	326	25.7	N 11	53.1	
Rigel	+0.1	039	54.8	S 08	11.3	
Rigil Kentaurus	-0.3	258	32.6	S 60	54.1	
Sabik	+2.4	220	54.2	S 15	44.5	
Schedar	+2.2	108	22.8	N 56	37.3	
Shaula	+1.6	215	03.1	S 37	06.7	
S Octantis	+5.5	158	13.0	S 88	52.9	
Sirius	-1.5	017	16.5	S 16	44.7	
Spica	+1.0	277	13.2	S 11	14.8	
Suhail	+2.2	341	35.3	S 43	30.3	
Vega	0.0	199	21.5	N 38	48.0	
Zubenelgenubi	+2.8	255	47.1	S 16	06.5	
*Sun		060	51.6	N 20	09.5	SD 15.8*
*Venus	-3.9	065	30.3	N 18	27.8	HP 00.1*
*Mars	-2.0	238	37.3	S 21	41.1	HP 00.3*
*Jupiter	-2.1	313	28.9	N 07	43.9	*
*Saturn	+0.1	225	44.8	S 20	41.4	*
*Moon		253	22.4	S 12	43.4	HP 54.1*
Aries		118	44.3			

21/09/16 2100

Acamar	+3.2	271	26.2	S 40	14.2	
Achernar	+0.5	291	34.5	S 57	09.1	
Acrux	+1.3	129	17.1	S 63	11.4	
Adhara	+1.5	211	20.6	S 28	59.6	
Aldebaran	+0.9	246	56.5	N 16	32.4	
Alioth	+1.8	122	29.2	N 55	52.4	
Alkaid	+1.9	109	07.5	N 49	14.1	
Al Na'ir	+1.7	343	50.4	S 46	52.7	
Alnilam	+1.7	231	53.9	S 01	11.6	

Elaboração de um Almanaque Náutico

Alphard	+2.0	174	03.9	S	08	43.8	
Alphecca	+2.2	082	19.2	N	26	39.9	
Alpheratz	+2.1	313	50.6	N	29	11.0	
Altair	+0.8	018	15.7	N	08	55.1	
Ankaa	+2.4	309	23.0	S	42	12.9	
Antares	+1.0	068	33.5	S	26	27.9	
Arcturus	0.0	102	03.8	N	19	06.1	
Atria	+1.9	063	33.4	S	69	03.4	
Avior	+1.9	190	27.2	S	59	33.6	
Bellatrix	+1.6	234	39.4	N	06	21.8	
Betelgeuse	Var.	227	08.7	N	07	24.5	
Canopus	-0.7	220	05.0	S	52	42.1	
Capella	+0.1	236	40.9	N	46	00.5	
Deneb	+1.3	005	39.4	N	45	20.8	
Denebola	+2.1	138	41.6	N	14	28.9	
Diphda	+2.0	305	03.2	S	17	53.6	
Dubhe	+1.8	149	59.6	N	61	39.6	
Elnath	+1.7	234	19.6	N	28	37.0	
Eltanin	+2.2	046	54.9	N	51	29.7	
Enif	+2.4	349	54.5	N	09	57.3	
Fomalhaut	+1.2	331	31.0	S	29	31.9	
Gacrux	+1.6	128	08.7	S	57	12.3	
Gienah	+2.6	132	00.1	S	17	37.9	
Hadar	+0.6	104	55.0	S	60	27.1	
Hamal	+2.0	284	07.7	N	23	32.4	
Kaus Australis	+1.9	039	50.6	S	34	22.4	
Kochab	+2.1	093	30.9	N	74	05.6	
Markab	+2.5	329	45.6	N	15	17.9	
Menkar	+2.5	270	22.3	N	04	09.2	
Menkent	+2.1	104	15.0	S	36	27.0	
Miaplacidus	+1.7	177	49.5	S	69	47.0	
Mirfak	+1.8	264	46.6	N	49	54.9	
Nunki	+2.0	032	05.3	S	26	16.4	
Peacock	+1.9	009	25.3	S	56	40.8	
Polaris	+2.0	272	35.0	N	89	19.7	
Pollux	+1.1	199	35.0	N	27	58.9	
Procyon	+0.4	201	07.3	N	05	10.9	
Rasalhague	+2.1	052	14.2	N	12	33.3	
Regulus	+1.4	163	51.3	N	11	53.2	
Rigel	+0.1	237	19.7	S	08	11.0	
Rigil Kentaurus	-0.3	095	58.9	S	60	54.1	
Sabik	+2.4	058	19.9	S	15	44.5	
Schedar	+2.2	305	47.1	N	56	37.7	
Shaula	+1.6	052	28.8	S	37	06.8	
S Octantis	+5.5	355	25.1	S	88	53.3	
Sirius	-1.5	214	41.6	S	16	44.3	
Spica	+1.0	114	39.0	S	11	14.7	
Suhail	+2.2	179	00.9	S	43	29.9	
Vega	0.0	036	47.2	N	38	48.4	
Zubenelgenubi	+2.8	093	13.0	S	16	06.4	
*Sun		136	48.7	N	00	16.9	SD 15.9*
*Venus	-3.9	110	11.7	S	10	30.3	HP 00.1*
*Mars	0.0	050	04.7	S	25	54.3	HP 00.1*
*Jupiter	-1.7	133	17.1	S	00	03.4	*
*Saturn	+0.5	066	30.3	S	20	38.1	*
*Moon		247	55.8	N	16	36.8	HP 59.6*
Aries		316	09.8				

2017/04/22 0800

Acamar	+3.2	285	51.1	S	40	14.4
Achernar	+0.5	305	59.8	S	57	09.1
Acrux	+1.3	143	39.9	S	63	11.8
Adhara	+1.5	225	44.9	S	29	00.2
Aldebaran	+0.9	261	20.9	N	16	32.4
Alioth	+1.8	136	52.2	N	55	52.1
Alkaid	+1.9	123	30.7	N	49	13.7
Al Na'ir	+1.7	358	15.0	S	46	52.5
Alnilam	+1.7	246	18.2	S	01	11.8
Alphard	+2.0	188	27.7	S	08	44.3
Alphecca	+2.2	096	42.8	N	26	39.4
Alpheratz	+2.1	328	15.4	N	29	10.9
Altair	+0.8	032	39.9	N	08	54.8
Ankaa	+2.4	323	47.9	S	42	12.8
Antares	+1.0	082	57.0	S	26	28.0
Arcturus	0.0	116	27.3	N	19	05.6
Atria	+1.9	077	56.1	S	69	03.1
Avior	+1.9	204	51.2	S	59	34.3
Bellatrix	+1.6	249	03.7	N	06	21.6
Betelgeuse	Var.	241	32.9	N	07	24.3
Canopus	-0.7	234	29.5	S	52	42.8
Capella	+0.1	251	05.2	N	46	00.8
Deneb	+1.3	020	04.0	N	45	20.3
Denebola	+2.1	153	05.0	N	14	28.5
Diphda	+2.0	319	27.9	S	17	53.7
Dubhe	+1.8	164	22.5	N	61	39.6
Elnath	+1.7	248	43.9	N	28	37.1
Eltanin	+2.2	061	18.9	N	51	29.1
Enif	+2.4	004	19.0	N	09	57.2
Fomalhaut	+1.2	345	55.7	S	29	31.8
Gacrux	+1.6	142	31.6	S	57	12.7
Gienah	+2.6	146	23.6	S	17	38.4
Hadar	+0.6	119	17.7	S	60	27.3
Hamal	+2.0	298	32.4	N	23	32.4
Kaus Australis	+1.9	054	14.5	S	34	22.3
Kochab	+2.1	107	53.4	N	74	05.1
Markab	+2.5	344	10.2	N	15	17.7
Menkar	+2.5	284	46.9	N	04	09.2
Menkent	+2.1	118	38.3	S	36	27.2
Miaplacidus	+1.7	192	13.1	S	69	47.7
Mirfak	+1.8	279	11.3	N	49	55.2
Nunki	+2.0	046	29.3	S	26	16.3
Peacock	+1.9	023	49.5	S	56	40.4
Polaris	+2.0	287	36.8	N	89	20.2
Pollux	+1.1	213	58.9	N	27	58.9
Procyon	+0.4	215	31.4	N	05	10.6
Rasalhague	+2.1	066	38.1	N	12	32.9
Regulus	+1.4	178	14.9	N	11	52.8
Rigel	+0.1	251	44.1	S	08	11.3
Rigil Kentaurus	-0.3	110	21.7	S	60	54.2
Sabik	+2.4	072	43.6	S	15	44.6
Schedar	+2.2	320	12.3	N	56	37.7
Shaula	+1.6	066	52.4	S	37	06.7
S Octantis	+5.5	009	59.7	S	88	52.7
Sirius	-1.5	229	05.8	S	16	44.8

Elaboração de um Almanaque Náutico

Spica	+1.0	129 02.5	S 11 15.1	
Suhail	+2.2	193 24.7	S 43 30.5	
Vega	0.0	051 11.3	N 38 47.9	
Zubenelgenubi	+2.8	107 36.5	S 16 06.7	
*Sun		300 22.7	N 12 17.9	SD 15.9*
*Venus	-4.7	333 44.4	N 01 48.4	HP 00.4*
*Mars	+1.6	272 14.2	N 20 51.0	HP 00.1*
*Jupiter	-2.4	134 49.0	S 05 00.9	*
*Saturn	+0.3	063 10.6	S 22 03.2	*
*Moon		351 53.0	S 09 27.5	HP 57.7*
Aries		330 34.3		

2017/07/23 1900

Acamar	+3.2	181 58.5	S 40 14.0	
Achernar	+0.5	202 06.9	S 57 08.7	
Acrux	+1.3	039 48.5	S 63 11.9	
Adhara	+1.5	121 52.9	S 28 59.8	
Aldebaran	+0.9	157 28.6	N 16 32.5	
Alioth	+1.8	033 00.6	N 55 52.3	
Alkaid	+1.9	019 38.9	N 49 14.0	
Al Na'ir	+1.7	254 22.0	S 46 52.3	
Alnilam	+1.7	142 26.0	S 01 11.6	
Alphard	+2.0	084 35.8	S 08 44.1	
Alphecca	+2.2	352 50.7	N 26 39.7	
Alpheratz	+2.1	224 22.6	N 29 11.1	
Altair	+0.8	288 47.3	N 08 55.1	
Ankaa	+2.4	219 55.0	S 42 12.5	
Antares	+1.0	339 04.8	S 26 28.1	
Arcturus	0.0	012 35.3	N 19 05.8	
Atria	+1.9	334 03.6	S 69 03.5	
Avior	+1.9	100 59.6	S 59 34.0	
Bellatrix	+1.6	145 11.4	N 06 21.8	
Betelgeuse	Var.	137 40.7	N 07 24.5	
Canopus	-0.7	130 37.6	S 52 42.3	
Capella	+0.1	147 12.9	N 46 00.6	
Deneb	+1.3	276 11.2	N 45 20.7	
Denebola	+2.1	049 13.2	N 14 28.6	
Diphda	+2.0	215 35.2	S 17 53.3	
Dubhe	+1.8	060 31.1	N 61 39.6	
Elnath	+1.7	144 51.6	N 28 37.1	
Eltanin	+2.2	317 26.5	N 51 29.5	
Enif	+2.4	260 26.2	N 09 57.4	
Fomalhaut	+1.2	242 02.8	S 29 31.6	
Gacrux	+1.6	038 40.1	S 57 12.8	
Gienah	+2.6	042 31.7	S 17 38.3	
Hadar	+0.6	015 26.0	S 60 27.5	
Hamal	+2.0	194 39.8	N 23 32.5	
Kaus Australis	+1.9	310 21.9	S 34 22.4	
Kochab	+2.1	004 02.1	N 74 05.5	
Markab	+2.5	240 17.5	N 15 18.0	

Elaboração de um Almanaque Náutico

Menkar	+2.5	180	54.4	N 04	09.4	
Menkent	+2.1	014	46.4	S 36	27.4	
Miaplacidus	+1.7	088	22.1	S 69	47.5	
Mirfak	+1.8	175	18.7	N 49	55.0	
Nunki	+2.0	302	36.7	S 26	16.3	
Peacock	+1.9	279	56.5	S 56	40.5	
Polaris	+2.0	183	20.2	N 89	19.8	
Pollux	+1.1	110	06.9	N 27	58.9	
Procyon	+0.4	111	39.3	N 05	10.7	
Rasalhague	+2.1	322	45.7	N 12	33.1	
Regulus	+1.4	074	23.0	N 11	53.0	
Rigel	+0.1	147	51.8	S 08	11.0	
Rigil Kentaurus	-0.3	006	29.9	S 60	54.5	
Sabik	+2.4	328	51.2	S 15	44.6	
Schedar	+2.2	216	19.2	N 56	37.7	
Shaula	+1.6	322	59.9	S 37	06.8	
S Octantis	+5.5	265	43.6	S 88	52.8	
Sirius	-1.5	125	13.7	S 16	44.5	
Spica	+1.0	025	10.5	S 11	15.0	
Suhail	+2.2	089	33.0	S 43	30.3	
Vega	0.0	307	18.8	N 38	48.3	
Zubenelgenubi	+2.8	003	44.4	S 16	06.7	
*Sun		103	22.5	N 19	54.5	SD 15.7*
*Venus	-4.0	146	19.7	N 21	10.2	HP 00.1*
*Mars	+1.7	102	03.8	N 20	44.9	HP 00.1*
*Jupiter	-1.9	031	31.8	S 05	09.4	*
*Saturn	+0.2	325	19.1	S 21	55.3	*
*Moon		098	21.5	N 17	06.5	HP 60.1*
Aries		226	42.1			

2017/10/24 2300

Acamar	+3.2	333	47.6	S 40	14.0	
Achernar	+0.5	353	55.9	S 57	08.9	
Acrux	+1.3	191	38.4	S 63	11.6	
Adhara	+1.5	273	42.0	S 28	59.7	
Aldebaran	+0.9	309	17.6	N 16	32.6	
Alioth	+1.8	184	50.6	N 55	51.9	
Alkaid	+1.9	171	29.0	N 49	13.6	
Al Na'ir	+1.7	046	11.8	S 46	52.6	
Alnilam	+1.7	294	15.1	S 01	11.5	
Alphard	+2.0	236	25.2	S 08	44.1	
Alphecca	+2.2	144	40.8	N 26	39.6	
Alpheratz	+2.1	016	12.0	N 29	11.4	
Altair	+0.8	080	37.3	N 08	55.2	
Ankaa	+2.4	011	44.3	S 42	12.7	
Antares	+1.0	130	54.8	S 26	28.1	
Arcturus	0.0	164	25.3	N 19	05.6	
Atria	+1.9	125	54.4	S 69	03.5	
Avior	+1.9	252	48.7	S 59	33.7	
Bellatrix	+1.6	297	00.5	N 06	21.8	
Betelgeuse	Var.	289	29.8	N 07	24.5	
Canopus	-0.7	282	26.5	S 52	42.2	
Capella	+0.1	299	01.6	N 46	00.7	
Deneb	+1.3	068	01.3	N 45	21.0	

Elaboração de um Almanaque Náutico

Denebola	+2.1	201 02.8	N 14 28.5	
Diphda	+2.0	007 24.5	S 17 53.4	
Dubhe	+1.8	212 20.5	N 61 39.2	
Elnath	+1.7	296 40.5	N 28 37.1	
Eltanin	+2.2	109 17.0	N 51 29.6	
Enif	+2.4	052 16.0	N 09 57.6	
Fomalhaut	+1.2	033 52.4	S 29 31.7	
Gacrux	+1.6	190 30.0	S 57 12.5	
Gienah	+2.6	194 21.4	S 17 38.2	
Hadar	+0.6	167 16.2	S 60 27.3	
Hamal	+2.0	346 29.0	N 23 32.7	
Kaus Australis	+1.9	102 12.0	S 34 22.4	
Kochab	+2.1	155 53.3	N 74 05.1	
Markab	+2.5	032 07.1	N 15 18.2	
Menkar	+2.5	332 43.5	N 04 09.5	
Menkent	+2.1	166 36.4	S 36 27.2	
Miaplacidus	+1.7	240 11.1	S 69 47.1	
Mirfak	+1.8	327 07.5	N 49 55.3	
Nunki	+2.0	094 26.7	S 26 16.3	
Peacock	+1.9	071 46.5	S 56 40.7	
Polaris	+2.0	334 30.7	N 89 20.2	
Pollux	+1.1	261 56.0	N 27 58.8	
Procyon	+0.4	263 28.5	N 05 10.7	
Rasalhague	+2.1	114 35.8	N 12 33.2	
Regulus	+1.4	226 12.4	N 11 52.8	
Rigel	+0.1	299 40.9	S 08 10.9	
Rigil Kentaurus	-0.3	158 20.2	S 60 54.3	
Sabik	+2.4	120 41.3	S 15 44.6	
Schedar	+2.2	008 08.4	N 56 38.2	
Shaula	+1.6	114 50.1	S 37 06.9	
S Octantis	+5.5	057 41.6	S 88 53.2	
Sirius	-1.5	277 02.9	S 16 44.4	
Spica	+1.0	177 00.3	S 11 15.0	
Suhail	+2.2	241 22.2	S 43 30.0	
Vega	0.0	099 09.0	N 38 48.4	
Zubenelgenubi	+2.8	155 34.4	S 16 06.7	
*Sun		168 58.1	S 12 04.3	SD 16.1*
*Venus	-3.9	185 52.7	S 03 45.3	HP 00.1*
*Mars	+1.8	196 48.3	N 00 30.3	HP 00.1*
*Jupiter	-1.7	167 16.1	S 11 35.8	*
*Saturn	+0.5	115 02.6	S 22 17.6	*
*Moon		109 09.5	S 19 27.7	HP 54.2*
Aries		018 31.9		

2018/01/25 0000

Acamar	+3.2	079 31.0	S 40 14.4	
Achernar	+0.5	099 39.7	S 57 09.1	
Acrux	+1.3	297 20.5	S 63 11.6	
Adhara	+1.5	019 24.8	S 29 00.0	
Aldebaran	+0.9	055 00.7	N 16 32.5	
Alioth	+1.8	290 32.9	N 55 51.5	
Alkaid	+1.9	277 11.5	N 49 13.2	
Al Na'ir	+1.7	151 55.4	S 46 52.6	
Alnilam	+1.7	039 58.1	S 01 11.7	
Alphard	+2.0	342 07.8	S 08 44.3	

Elaboração de um Almanaque Náutico

Alphecca	+2.2	250	23.6	N 26	39.2	
Alpheratz	+2.1	121	55.5	N 29	11.4	
Altair	+0.8	186	20.6	N 08	55.0	
Ankaa	+2.4	117	28.0	S 42	12.8	
Antares	+1.0	236	37.7	S 26	28.1	
Arcturus	0.0	270	08.0	N 19	05.3	
Atria	+1.9	231	37.1	S 69	03.2	
Avior	+1.9	358	31.2	S 59	34.1	
Bellatrix	+1.6	042	43.5	N 06	21.7	
Betelgeuse	Var.	035	12.8	N 07	24.4	
Canopus	-0.7	028	09.4	S 52	42.6	
Capella	+0.1	044	44.6	N 46	00.9	
Deneb	+1.3	173	44.9	N 45	20.7	
Denebola	+2.1	306	45.4	N 14	28.2	
Diphda	+2.0	113	08.0	S 17	53.5	
Dubhe	+1.8	318	02.6	N 61	39.0	
Elnath	+1.7	042	23.5	N 28	37.2	
Eltanin	+2.2	215	00.3	N 51	29.1	
Enif	+2.4	157	59.5	N 09	57.5	
Fomalhaut	+1.2	139	36.0	S 29	31.8	
Gacrux	+1.6	296	12.2	S 57	12.6	
Gienah	+2.6	300	04.0	S 17	38.4	
Hadar	+0.6	272	58.5	S 60	27.2	
Hamal	+2.0	092	12.3	N 23	32.8	
Kaus Australis	+1.9	207	55.1	S 34	22.4	
Kochab	+2.1	261	35.5	N 74	04.7	
Markab	+2.5	137	50.6	N 15	18.1	
Menkar	+2.5	078	26.8	N 04	09.4	
Menkent	+2.1	272	19.0	S 36	27.2	
Miaplacidus	+1.7	345	53.2	S 69	47.5	
Mirfak	+1.8	072	50.8	N 49	55.5	
Nunki	+2.0	200	09.9	S 26	16.3	
Peacock	+1.9	177	30.1	S 56	40.6	
Polaris	+2.0	080	31.2	N 89	20.6	
Pollux	+1.1	007	38.7	N 27	58.8	
Procyon	+0.4	009	11.3	N 05	10.5	
Rasalhague	+2.1	220	18.9	N 12	32.9	
Regulus	+1.4	331	55.0	N 11	52.6	
Rigel	+0.1	045	23.9	S 08	11.1	
Rigil Kentaurus	-0.3	264	02.6	S 60	54.1	
Sabik	+2.4	226	24.3	S 15	44.7	
Schedar	+2.2	113	52.2	N 56	38.3	
Shaula	+1.6	220	33.1	S 37	06.8	
S Octantis	+5.5	163	41.9	S 88	52.9	
Sirius	-1.5	022	45.8	S 16	44.7	
Spica	+1.0	282	43.0	S 11	15.2	
Suhail	+2.2	347	04.8	S 43	30.4	
Vega	0.0	204	52.3	N 38	48.0	
Zubenelgenubi	+2.8	261	17.1	S 16	06.8	
*Sun		176	57.0	S 19	01.5	SD 16.2*
*Venus	-3.9	172	46.9	S 19	15.1	HP 00.1*
*Mars	+1.3	247	15.1	S 19	14.8	HP 00.1*
*Jupiter	-1.9	255	55.8	S 16	48.2	*
*Saturn	+0.5	209	48.7	S 22	29.4	*
*Moon		088	59.1	N 08	31.6	HP 58.1*
Aries		124	15.1			

2018/06/26 1230

Acamar	+3.2	057	21.1	S	40	13.8
Achernar	+0.5	077	29.5	S	57	08.4
Acrux	+1.3	275	10.4	S	63	12.3
Adhara	+1.5	357	15.2	S	29	00.0
Aldebaran	+0.9	032	50.8	N	16	32.6
Alioth	+1.8	268	22.6	N	55	52.0
Alkaid	+1.9	255	01.0	N	49	13.7
Al Na'ir	+1.7	129	44.2	S	46	52.1
Alnilam	+1.7	017	48.2	S	01	11.6
Alphard	+2.0	319	58.0	S	08	44.4
Alphecca	+2.2	228	12.8	N	26	39.4
Alpheratz	+2.1	099	44.9	N	29	11.3
Altair	+0.8	164	09.6	N	08	55.1
Ankaa	+2.4	095	17.4	S	42	12.2
Antares	+1.0	214	26.7	S	26	28.2
Arcturus	0.0	247	57.4	N	19	05.4
Atria	+1.9	209	24.9	S	69	03.5
Avior	+1.9	336	22.2	S	59	34.3
Bellatrix	+1.6	020	33.7	N	06	21.8
Betelgeuse	Var.	013	03.0	N	07	24.5
Canopus	-0.7	006	00.2	S	52	42.5
Capella	+0.1	022	34.9	N	46	00.7
Deneb	+1.3	151	33.7	N	45	20.7
Denebola	+2.1	284	35.2	N	14	28.3
Diphda	+2.0	090	57.5	S	17	53.1
Dubhe	+1.8	295	52.8	N	61	39.4
Elnath	+1.7	020	13.7	N	28	37.2
Eltanin	+2.2	192	49.0	N	51	29.4
Enif	+2.4	135	48.6	N	09	57.6
Fomalhaut	+1.2	117	25.1	S	29	31.3
Gacrux	+1.6	274	02.0	S	57	13.2
Gienah	+2.6	277	53.8	S	17	38.7
Hadar	+0.6	250	47.7	S	60	27.8
Hamal	+2.0	070	02.1	N	23	32.7
Kaus Australis	+1.9	185	43.9	S	34	22.4
Kochab	+2.1	239	24.4	N	74	05.2
Markab	+2.5	115	39.8	N	15	18.2
Menkar	+2.5	056	16.7	N	04	09.5
Menkent	+2.1	250	08.3	S	36	27.7
Miaplacidus	+1.7	323	44.7	S	69	47.8
Mirfak	+1.8	050	40.8	N	49	55.2
Nunki	+2.0	177	58.7	S	26	16.2
Peacock	+1.9	155	18.4	S	56	40.3
Polaris	+2.0	058	38.9	N	89	20.1
Pollux	+1.1	345	28.9	N	27	58.8
Procyon	+0.4	347	01.5	N	05	10.6
Rasalhague	+2.1	198	07.9	N	12	33.0
Regulus	+1.4	309	45.1	N	11	52.7
Rigel	+0.1	023	14.1	S	08	11.0
Rigil Kentaurus	-0.3	241	51.7	S	60	54.7
Sabik	+2.4	204	13.3	S	15	44.7
Schedar	+2.2	091	41.6	N	56	37.9
Shaula	+1.6	198	21.9	S	37	06.9
S Octantis	+5.5	140	58.9	S	88	52.5
Sirius	-1.5	000	36.1	S	16	44.6

## Elaboração de um Almanaque Náutico

---

Spica	+1.0	260 32.5	S 11 15.4	
Suhail	+2.2	324 55.3	S 43 30.6	
Vega	0.0	182 41.2	N 38 48.2	
Zubenelgenubi	+2.8	239 06.4	S 16 07.0	
*Sun		006 47.1	N 23 20.7	SD 15.7*
*Venus	-4.0	324 26.3	N 18 15.8	HP 00.1*
*Mars	-2.0	149 05.9	S 22 26.7	HP 00.3*
*Jupiter	-2.3	240 32.6	S 14 51.7	*
*Saturn	0.0	185 39.8	S 22 27.0	*
*Moon		206 33.1	S 18 45.8	HP 54.6*
Aries		102 05.0		

2018/12/27 2200

Acamar	+3.2	021 35.1	S 40 14.0	
Achernar	+0.5	041 43.7	S 57 08.8	
AcruX	+1.3	239 25.0	S 63 11.9	
Adhara	+1.5	321 29.1	S 29 00.0	
Aldebaran	+0.9	357 04.7	N 16 32.7	
Alioth	+1.8	232 37.4	N 55 51.2	
Alkaid	+1.9	219 16.1	N 49 13.0	
Al Na'ir	+1.7	093 59.2	S 46 52.4	
Alnilam	+1.7	342 02.1	S 01 11.6	
Alphard	+2.0	284 12.1	S 08 44.5	
Alphecca	+2.2	192 28.0	N 26 39.1	
Alpheratz	+2.1	063 59.5	N 29 11.8	
Altair	+0.8	128 24.8	N 08 55.2	
Ankaa	+2.4	059 31.9	S 42 12.5	
Antares	+1.0	178 41.9	S 26 28.2	
Arcturus	0.0	212 12.4	N 19 05.0	
Atria	+1.9	173 40.8	S 69 03.4	
Avior	+1.9	300 35.9	S 59 34.1	
Bellatrix	+1.6	344 47.6	N 06 21.9	
Betelgeuse	Var.	337 16.8	N 07 24.5	
Canopus	-0.7	330 13.9	S 52 42.5	
Capella	+0.1	346 48.3	N 46 00.9	
Deneb	+1.3	115 49.2	N 45 21.1	
Denebola	+2.1	248 49.7	N 14 27.9	
Diphda	+2.0	055 12.0	S 17 53.2	
Dubhe	+1.8	260 06.8	N 61 38.7	
Elnath	+1.7	344 27.4	N 28 37.3	
Eltanin	+2.2	157 04.9	N 51 29.3	
Enif	+2.4	100 03.6	N 09 57.8	
Fomalhaut	+1.2	081 39.9	S 29 31.5	
Gacrux	+1.6	238 16.6	S 57 12.8	
Gienah	+2.6	242 08.3	S 17 38.7	
Hadar	+0.6	215 02.7	S 60 27.5	
Hamal	+2.0	034 16.3	N 23 33.1	
Kaus Australis	+1.9	149 59.2	S 34 22.4	
Kochab	+2.1	203 40.8	N 74 04.5	
Markab	+2.5	079 54.6	N 15 18.5	
Menkar	+2.5	020 30.8	N 04 09.7	
Menkent	+2.1	214 23.2	S 36 27.5	
Miaplacidus	+1.7	287 58.1	S 69 47.5	
Mirfak	+1.8	014 54.5	N 49 55.7	

Elaboração de um Almanaque Náutico

Nunki	+2.0	142	14.0	S 26	16.3	
Peacock	+1.9	119	33.8	S 56	40.5	
Polaris	+2.0	022	07.9	N 89	20.8	
Pollux	+1.1	309	42.7	N 27	58.7	
Procyon	+0.4	311	15.4	N 05	10.5	
Rasalhague	+2.1	162	23.2	N 12	32.9	
Regulus	+1.4	273	59.3	N 11	52.4	
Rigel	+0.1	347	28.0	S 08	11.0	
Rigil Kentaurus	-0.3	206	06.9	S 60	54.4	
Sabik	+2.4	168	28.5	S 15	44.7	
Schedar	+2.2	055	56.1	N 56	38.7	
Shaula	+1.6	162	37.1	S 37	06.9	
S Octantis	+5.5	105	31.6	S 88	52.9	
Sirius	-1.5	324	50.0	S 16	44.7	
Spica	+1.0	224	47.3	S 11	15.4	
Suhail	+2.2	289	09.3	S 43	30.4	
Vega	0.0	146	56.8	N 38	48.2	
Zubenelgenubi	+2.8	203	21.4	S 16	07.0	
*Sun		149	41.6	S 23	17.7	SD 16.3*
*Venus	-4.7	198	16.5	S 14	21.6	HP 00.2*
*Mars	+0.4	068	44.9	S 01	28.6	HP 00.1*
*Jupiter	-1.8	176	54.5	S 21	27.1	*
*Saturn	+0.5	144	31.7	S 22	30.1	*
*Moon		256	49.0	N 08	53.9	HP 59.5*
Aries		066	19.9			

2019/03/28 0200

Acamar	+3.2	170	27.9	S 40	14.0	
Achernar	+0.5	190	36.6	S 57	08.6	
Acrux	+1.3	028	16.4	S 63	12.3	
Adhara	+1.5	110	21.7	S 29	00.3	
Aldebaran	+0.9	145	57.3	N 16	32.7	
Alioth	+1.8	021	28.9	N 55	51.3	
Alkaid	+1.9	008	07.6	N 49	13.0	
Al Na'ir	+1.7	242	51.4	S 46	52.0	
Alnilam	+1.7	130	54.7	S 01	11.7	
Alphard	+2.0	073	04.3	S 08	44.7	
Alphecca	+2.2	341	19.7	N 26	38.9	
Alpheratz	+2.1	212	52.0	N 29	11.6	
Altair	+0.8	277	16.8	N 08	55.0	
Ankaa	+2.4	208	24.5	S 42	12.3	
Antares	+1.0	327	33.5	S 26	28.3	
Arcturus	0.0	001	04.1	N 19	04.9	
Atria	+1.9	322	31.6	S 69	03.3	
Avior	+1.9	089	28.4	S 59	34.6	
Bellatrix	+1.6	133	40.1	N 06	21.8	
Betelgeuse	Var.	126	09.3	N 07	24.4	
Canopus	-0.7	119	06.7	S 52	42.8	
Capella	+0.1	135	41.0	N 46	01.0	
Deneb	+1.3	264	41.3	N 45	20.7	
Denebola	+2.1	037	41.6	N 14	27.8	
Diphda	+2.0	204	04.5	S 17	53.1	
Dubhe	+1.8	048	58.5	N 61	38.9	
Elnath	+1.7	133	20.0	N 28	37.3	
Eltanin	+2.2	305	56.5	N 51	29.0	

Elaboração de um Almanaque Náutico

Enif	+2.4	248	55.8	N 09	57.6	
Fomalhaut	+1.2	230	32.2	S 29	31.3	
Gacrux	+1.6	027	08.2	S 57	13.2	
Gienah	+2.6	031	00.2	S 17	39.0	
Hadar	+0.6	003	54.0	S 60	27.7	
Hamal	+2.0	183	08.9	N 23	33.0	
Kaus Australis	+1.9	298	50.9	S 34	22.3	
Kochab	+2.1	352	31.5	N 74	04.5	
Markab	+2.5	228	47.0	N 15	18.3	
Menkar	+2.5	169	23.4	N 04	09.6	
Menkent	+2.1	003	14.8	S 36	27.7	
Miaplacidus	+1.7	076	50.6	S 69	48.0	
Mirfak	+1.8	163	47.3	N 49	55.7	
Nunki	+2.0	291	05.8	S 26	16.2	
Peacock	+1.9	268	25.6	S 56	40.2	
Polaris	+2.0	171	35.6	N 89	20.8	
Pollux	+1.1	098	35.1	N 27	58.7	
Procyon	+0.4	100	07.8	N 05	10.3	
Rasalhague	+2.1	311	15.0	N 12	32.7	
Regulus	+1.4	062	51.3	N 11	52.3	
Rigel	+0.1	136	20.6	S 08	11.1	
Rigil Kentaurus	-0.3	354	58.1	S 60	54.6	
Sabik	+2.4	317	20.2	S 15	44.8	
Schedar	+2.2	204	48.9	N 56	38.4	
Shaula	+1.6	311	28.7	S 37	06.8	
S Octantis	+5.5	254	20.0	S 88	52.3	
Sirius	-1.5	113	42.5	S 16	44.9	
Spica	+1.0	013	39.1	S 11	15.7	
Suhail	+2.2	078	01.5	S 43	30.9	
Vega	0.0	295	48.6	N 38	47.9	
Zubenelgenubi	+2.8	352	13.1	S 16	07.2	
*Sun		208	40.8	N 02	49.2	SD 16.0*
*Venus	-4.0	241	24.2	S 11	37.0	HP 00.1*
*Mars	+1.4	159	47.0	N 20	33.5	HP 00.1*
*Jupiter	-2.2	311	38.6	S 22	40.3	*
*Saturn	+0.6	284	00.6	S 21	35.4	*
*Moon		298	38.6	S 21	45.3	HP 55.1*
Aries		215	12.3			

2020/02/29 0000

Acamar	+3.2	113	31.2	S 40	13.9	
Achernar	+0.5	133	39.9	S 57	08.5	
Acrux	+1.3	331	19.8	S 63	12.4	
Adhara	+1.5	053	24.9	S 29	00.3	
Aldebaran	+0.9	089	00.3	N 16	32.8	
Alioth	+1.8	324	32.4	N 55	50.9	
Alkaid	+1.9	311	11.2	N 49	12.6	
Al Na'ir	+1.7	185	54.6	S 46	51.9	
Alnilam	+1.7	073	57.8	S 01	11.6	
Alphard	+2.0	016	07.5	S 08	44.9	
Alphecca	+2.2	284	23.3	N 26	38.7	
Alpheratz	+2.1	155	55.3	N 29	11.9	
Altair	+0.8	220	20.2	N 08	55.2	
Ankaa	+2.4	151	27.7	S 42	12.1	

Elaboração de um Almanaque Náutico

Antares	+1.0	270	36.9	S	26	28.4	
Arcturus	0.0	304	07.6	N	19	04.6	
Atria	+1.9	265	34.6	S	69	03.4	
Avior	+1.9	032	31.9	S	59	34.6	
Bellatrix	+1.6	076	43.2	N	06	21.9	
Betelgeuse	Var.	069	12.4	N	07	24.4	
Canopus	-0.7	062	10.1	S	52	42.7	
Capella	+0.1	078	43.8	N	46	01.1	
Deneb	+1.3	207	45.0	N	45	20.9	
Denebola	+2.1	340	44.9	N	14	27.5	
Diphda	+2.0	147	07.8	S	17	52.9	
Dubhe	+1.8	352	01.6	N	61	38.5	
Elnath	+1.7	076	22.9	N	28	37.4	
Eltanin	+2.2	249	00.4	N	51	28.9	
Enif	+2.4	191	59.2	N	09	57.9	
Fomalhaut	+1.2	173	35.5	S	29	31.1	
Gacrux	+1.6	330	11.6	S	57	13.4	
Gienah	+2.6	334	03.5	S	17	39.2	
Hadar	+0.6	306	57.3	S	60	27.9	
Hamal	+2.0	126	12.0	N	23	33.3	
Kaus Australis	+1.9	241	54.1	S	34	22.3	
Kochab	+2.1	295	35.8	N	74	04.2	
Markab	+2.5	171	50.3	N	15	18.6	
Menkar	+2.5	112	26.6	N	04	09.9	
Menkent	+2.1	306	18.2	S	36	27.9	
Miaplacidus	+1.7	019	54.2	S	69	48.1	
Mirfak	+1.8	106	50.1	N	49	55.9	
Nunki	+2.0	234	09.1	S	26	16.2	
Peacock	+1.9	211	28.7	S	56	40.1	
Polaris	+2.0	114	12.9	N	89	21.1	
Pollux	+1.1	041	38.1	N	27	58.6	
Procyon	+0.4	043	10.9	N	05	10.2	
Rasalhague	+2.1	254	18.5	N	12	32.6	
Regulus	+1.4	005	54.5	N	11	52.0	
Rigel	+0.1	079	23.8	S	08	11.0	
Rigil Kentaurus	-0.3	298	01.4	S	60	54.7	
Sabik	+2.4	260	23.6	S	15	44.9	
Schedar	+2.2	147	52.0	N	56	38.8	
Shaula	+1.6	254	32.0	S	37	06.9	
S Octantis	+5.5	197	16.7	S	88	52.3	
Sirius	-1.5	056	45.7	S	16	44.9	
Spica	+1.0	316	42.4	S	11	15.9	
Suhail	+2.2	021	04.9	S	43	31.0	
Vega	0.0	238	52.3	N	38	47.9	
Zubenelgenubi	+2.8	295	16.4	S	16	07.4	
*Sun		176	52.0	S	07	52.2	SD 16.1*
*Venus	-4.3	136	08.3	N	10	23.2	HP 00.2*
*Mars	+1.1	238	51.2	S	23	33.0	HP 00.1*
*Jupiter	-2.0	227	18.9	S	22	02.7	*
*Saturn	+0.7	218	06.2	S	20	33.9	*
*Moon		121	02.9	N	10	10.7	HP 54.4*
Aries		158	16.2				

2019/08/29 0700

Acamar +3.2 037 26.9 S 40 13.4

Elaboração de um Almanaque Náutico

Achernar	+0.5	057 35.2	S 57 08.1	
Acrux	+1.3	255 17.1	S 63 12.5	
Adhara	+1.5	337 21.4	S 28 59.8	
Aldebaran	+0.9	012 56.5	N 16 32.8	
Alioth	+1.8	248 29.3	N 55 51.6	
Alkaid	+1.9	235 07.8	N 49 13.3	
Al Na'ir	+1.7	109 49.9	S 46 51.9	
Alnilam	+1.7	357 54.1	S 01 11.4	
Alphard	+2.0	300 04.2	S 08 44.5	
Alphecca	+2.2	208 19.4	N 26 39.3	
Alpheratz	+2.1	079 50.8	N 29 11.9	
Altair	+0.8	144 15.8	N 08 55.4	
Ankaa	+2.4	075 23.1	S 42 11.9	
Antares	+1.0	194 33.0	S 26 28.4	
Arcturus	0.0	228 04.0	N 19 05.2	
Atria	+1.9	189 30.9	S 69 03.8	
Avior	+1.9	316 28.8	S 59 34.2	
Bellatrix	+1.6	000 39.5	N 06 22.0	
Betelgeuse	Var.	353 08.8	N 07 24.6	
Canopus	-0.7	346 06.5	S 52 42.2	
Capella	+0.1	002 40.2	N 46 00.8	
Deneb	+1.3	131 40.2	N 45 21.2	
Denebola	+2.1	264 41.6	N 14 28.0	
Diphda	+2.0	071 03.3	S 17 52.7	
Dubhe	+1.8	275 59.1	N 61 38.9	
Elnath	+1.7	000 19.3	N 28 37.3	
Eltanin	+2.2	172 55.9	N 51 29.6	
Enif	+2.4	115 54.6	N 09 58.0	
Fomalhaut	+1.2	097 30.9	S 29 31.0	
Gacrux	+1.6	254 08.6	S 57 13.3	
Gienah	+2.6	258 00.2	S 17 38.9	
Hadar	+0.6	230 54.2	S 60 28.1	
Hamal	+2.0	050 07.7	N 23 33.2	
Kaus Australis	+1.9	165 49.9	S 34 22.4	
Kochab	+2.1	219 32.5	N 74 04.9	
Markab	+2.5	095 45.7	N 15 18.7	
Menkar	+2.5	036 22.5	N 04 09.9	
Menkent	+2.1	230 14.8	S 36 27.9	
Miaplacidus	+1.7	303 51.7	S 69 47.7	
Mirfak	+1.8	030 46.1	N 49 55.5	
Nunki	+2.0	158 04.8	S 26 16.2	
Peacock	+1.9	135 24.0	S 56 40.3	
Polaris	+2.0	037 56.6	N 89 20.4	
Pollux	+1.1	325 34.8	N 27 58.7	
Procyon	+0.4	327 07.5	N 05 10.5	
Rasalhague	+2.1	178 14.4	N 12 33.1	
Regulus	+1.4	289 51.3	N 11 52.4	
Rigel	+0.1	003 20.0	S 08 10.7	
Rigil Kentaurus	-0.3	221 58.3	S 60 55.0	
Sabik	+2.4	184 19.6	S 15 44.8	
Schedar	+2.2	071 47.2	N 56 38.5	
Shaula	+1.6	178 28.0	S 37 07.0	
S Octantis	+5.5	120 47.6	S 88 52.6	
Sirius	-1.5	340 42.1	S 16 44.5	
Spica	+1.0	240 39.0	S 11 15.6	
Suhail	+2.2	305 01.7	S 43 30.5	
Vega	0.0	162 47.8	N 38 48.5	
Zubenelgenubi	+2.8	219 12.8	S 16 07.2	
*Sun		284 44.0	N 09 25.9	SD 15.8*

Elaboração de um Almanaque Náutico

*Venus	-3.9	280	18.7	N 09	12.3	HP 00.1*
*Mars	+1.7	283	01.7	N 09	56.5	HP 00.1*
*Jupiter	-2.2	188	27.2	S 22	14.6	*
*Saturn	+0.3	156	46.4	S 22	26.5	*
*Moon		299	35.1	N 17	40.0	HP 61.1*
Aries		082	11.9			

2019/11/30 1800

Acamar	+3.2	294	33.5	S 40	13.7
Achernar	+0.5	314	42.0	S 57	08.5
Acrux	+1.3	152	23.7	S 63	12.2
Adhara	+1.5	234	27.7	S 28	59.9
Aldebaran	+0.9	270	02.9	N 16	32.9
Alioth	+1.8	145	36.0	N 55	51.0
Alkaid	+1.9	132	14.7	N 49	12.8
Al Na'ir	+1.7	006	57.2	S 46	52.1
Alnilam	+1.7	255	00.5	S 01	11.4
Alphard	+2.0	197	10.6	S 08	44.6
Alphecca	+2.2	105	26.6	N 26	39.0
Alpheratz	+2.1	336	57.8	N 29	12.1
Altair	+0.8	041	23.1	N 08	55.4
Ankaa	+2.4	332	30.1	S 42	12.2
Antares	+1.0	091	40.2	S 26	28.4
Arcturus	0.0	125	10.9	N 19	04.8
Atria	+1.9	086	38.6	S 69	03.6
Avior	+1.9	213	34.8	S 59	34.1
Bellatrix	+1.6	257	45.9	N 06	22.0
Betelgeuse	Var.	250	15.1	N 07	24.6
Canopus	-0.7	243	12.7	S 52	42.3
Capella	+0.1	259	46.4	N 46	00.9
Deneb	+1.3	028	47.7	N 45	21.3
Denebola	+2.1	161	48.2	N 14	27.7
Diphda	+2.0	328	10.2	S 17	52.8
Dubhe	+1.8	173	05.3	N 61	38.4
Elnath	+1.7	257	25.6	N 28	37.3
Eltanin	+2.2	070	03.6	N 51	29.4
Enif	+2.4	013	01.8	N 09	58.1
Fomalhaut	+1.2	354	38.0	S 29	31.2
Gacrux	+1.6	151	15.3	S 57	13.1
Gienah	+2.6	155	06.9	S 17	38.9
Hadar	+0.6	128	01.1	S 60	27.8
Hamal	+2.0	307	14.4	N 23	33.4
Kaus Australis	+1.9	062	57.3	S 34	22.4
Kochab	+2.1	116	40.1	N 74	04.4
Markab	+2.5	352	52.9	N 15	18.8
Menkar	+2.5	293	29.1	N 04	10.0
Menkent	+2.1	127	21.7	S 36	27.7
Miaplacidus	+1.7	200	57.4	S 69	47.6
Mirfak	+1.8	287	52.5	N 49	55.9
Nunki	+2.0	055	12.1	S 26	16.3
Peacock	+1.9	032	31.6	S 56	40.4
Polaris	+2.0	294	43.2	N 89	20.9
Pollux	+1.1	222	41.0	N 27	58.6

Elaboração de um Almanaque Náutico

Procyon	+0.4	224	13.8	N 05	10.4	
Rasalhague	+2.1	075	21.6	N 12	32.9	
Regulus	+1.4	186	57.7	N 11	52.2	
Rigel	+0.1	260	26.4	S 08	10.8	
Rigil Kentaurus	-0.3	119	05.3	S 60	54.7	
Sabik	+2.4	081	26.8	S 15	44.8	
Schedar	+2.2	328	54.2	N 56	38.9	
Shaula	+1.6	075	35.3	S 37	07.0	
S Octantis	+5.5	018	12.8	S 88	52.7	
Sirius	-1.5	237	48.5	S 16	44.6	
Spica	+1.0	137	45.8	S 11	15.7	
Suhail	+2.2	202	08.0	S 43	30.5	
Vega	0.0	059	55.3	N 38	48.3	
Zubenelgenubi	+2.8	116	19.8	S 16	07.2	
*Sun		092	50.6	S 21	40.5	SD 16.2*
*Venus	-3.9	062	51.5	S 24	45.7	HP 00.1*
*Mars	+1.7	123	55.1	S 13	25.6	HP 00.1*
*Jupiter	-1.8	069	48.2	S 23	17.8	*
*Saturn	+0.6	049	52.4	S 22	08.5	*
*Moon		037	56.8	S 22	09.3	HP 55.9*
Aries		339	19.0			

2020/05/31 1200

Acamar	+3.2	024	41.6	S 40	13.4	
Achernar	+0.5	044	50.1	S 57	07.9	
Acrux	+1.3	242	30.2	S 63	12.9	
Adhara	+1.5	324	35.7	S 29	00.2	
Aldebaran	+0.9	000	10.8	N 16	32.8	
Alioth	+1.8	235	42.7	N 55	51.3	
Alkaid	+1.9	222	21.3	N 49	13.0	
Al Na'ir	+1.7	097	04.2	S 46	51.6	
Alnilam	+1.7	345	08.4	S 01	11.5	
Alphard	+2.0	287	18.1	S 08	44.9	
Alphecca	+2.2	195	33.2	N 26	38.9	
Alpheratz	+2.1	067	05.3	N 29	11.9	
Altair	+0.8	131	29.9	N 08	55.3	
Ankaa	+2.4	062	37.7	S 42	11.7	
Antares	+1.0	181	46.6	S 26	28.5	
Arcturus	0.0	215	17.6	N 19	04.7	
Atria	+1.9	176	43.5	S 69	03.7	
Avior	+1.9	303	43.1	S 59	34.7	
Bellatrix	+1.6	347	53.8	N 06	21.9	
Betelgeuse	Var.	340	23.0	N 07	24.5	
Canopus	-0.7	333	21.1	S 52	42.6	
Capella	+0.1	349	54.4	N 46	01.0	
Deneb	+1.3	118	54.6	N 45	20.9	
Denebola	+2.1	251	55.3	N 14	27.6	
Diphda	+2.0	058	17.8	S 17	52.6	
Dubhe	+1.8	263	12.3	N 61	38.8	
Elnath	+1.7	347	33.5	N 28	37.3	
Eltanin	+2.2	160	10.0	N 51	29.1	
Enif	+2.4	103	08.9	N 09	58.0	
Fomalhaut	+1.2	084	45.3	S 29	30.8	
Gacrux	+1.6	241	21.9	S 57	13.8	

Elaboração de um Almanaque Náutico

---

Gienah	+2.6	245 13.8	S 17 39.3	
Hadar	+0.6	218 07.2	S 60 28.3	
Hamal	+2.0	037 22.2	N 23 33.2	
Kaus Australis	+1.9	153 03.7	S 34 22.3	
Kochab	+2.1	206 45.5	N 74 04.6	
Markab	+2.5	083 00.2	N 15 18.7	
Menkar	+2.5	023 36.9	N 04 10.0	
Menkent	+2.1	217 28.2	S 36 28.2	
Miaplacidus	+1.7	291 05.7	S 69 48.3	
Mirfak	+1.8	018 00.6	N 49 55.7	
Nunki	+2.0	145 18.6	S 26 16.2	
Peacock	+1.9	122 37.9	S 56 39.9	
Polaris	+2.0	025 34.4	N 89 20.7	
Pollux	+1.1	312 48.7	N 27 58.6	
Procyon	+0.4	314 21.6	N 05 10.3	
Rasalhague	+2.1	165 28.2	N 12 32.7	
Regulus	+1.4	277 05.1	N 11 52.1	
Rigel	+0.1	350 34.4	S 08 10.9	
Rigil Kentaurus	-0.3	209 11.3	S 60 55.1	
Sabik	+2.4	171 33.3	S 15 44.9	
Schedar	+2.2	059 02.0	N 56 38.5	
Shaula	+1.6	165 41.6	S 37 07.0	
S Octantis	+5.5	108 02.6	S 88 52.0	
Sirius	-1.5	327 56.4	S 16 44.8	
Spica	+1.0	227 52.6	S 11 16.0	
Suhail	+2.2	292 15.7	S 43 31.1	
Vega	0.0	150 01.9	N 38 48.1	
Zubenelgenubi	+2.8	206 26.4	S 16 07.5	
*Sun		000 33.3	N 22 01.1	SD 15.8*
*Venus	-4.1	355 13.3	N 23 53.5	HP 00.5*
*Mars	0.0	084 47.3	S 09 06.2	HP 00.1*
*Jupiter	-2.6	130 34.4	S 21 00.2	*
*Saturn	+0.4	125 32.2	S 19 56.7	*
*Moon		248 47.6	N 05 26.7	HP 59.5*
Aries		069 26.6		