

UNIVERSIDADE FEDERAL DE UBERLÂNDIA  
INSTITUTO DE GEOGRAFIA - IG  
CURSO DE ENGENHARIA DE AGRIMENSURA E CARTOGRÁFICA

FELIPE MUNDIM DE OLIVEIRA

**ELABORAÇÃO DE UM PROGRAMA COMPUTACIONAL NO SCILAB PARA  
CALCULAR LEVANTAMENTOS TOPOGRÁFICOS PLANIALTIMÉTRICOS**

Monte Carmelo

2022

FELIPE MUNDIM DE OLIVEIRA

**ELABORAÇÃO DE UM PROGRAMA COMPUTACIONAL NO SCILAB PARA  
CALCULAR LEVANTAMENTOS TOPOGRÁFICOS PLANIALTIMÉTRICOS**

Trabalho de Conclusão de Curso apresentado ao Instituto de Geografia da Universidade Federal de Uberlândia como requisito parcial para obtenção do título de bacharel em Engenharia de Agrimensura e Cartográfica.

Orientadora: Prof<sup>a</sup>. Msc. Suelem Farias Soares Martins.

Monte Carmelo

2022

FELIPE MUNDIM DE OLIVEIRA

**ELABORAÇÃO DE UM PROGRAMA COMPUTACIONAL NO SCILAB PARA  
CALCULAR LEVANTAMENTOS TOPOGRÁFICOS PLANIALTIMÉTRICOS**

Trabalho de Conclusão de Curso apresentado ao Instituto de Geografia da Universidade Federal de Uberlândia como requisito parcial para obtenção do título de bacharel em Engenharia de Agrimensura e Cartográfica.

Orientadora: Prof<sup>a</sup>. Msc. Suelem Farias Martins.

Monte Carmelo, 13/01/2022.

Banca Examinadora:

---

Prof<sup>a</sup>. Msc. Suelem Farias Soares Martins(Orientadora)  
Universidade Federal de Uberlândia (FECIV / UFU).

---

Prof<sup>a</sup>. Msc. Jaqueline Vicente Matsuoka (Membro Convidado)  
(UNIFUCAMP)

---

Eng. Agrimensor e Cartógrafo. Weldon Martins dos Santos (Membro Convidado)  
Universidade Federal de Uberlândia (IG/UFU)

---

Prof<sup>a</sup>. Dra. Luziane Ribeiro Indjai (Membro Convidado)  
Universidade Federal de Uberlândia (IG/UFU)

## **AGRADECIMENTOS**

Agradeço primeiramente a Deus pela vida e saúde.

A minha mãe e meu pai, pelos conselhos, paciência e carinho durante toda a graduação e a vida. Tornando possível a realização deste sonho.

Meu irmão pelos conselhos e por estar do meu lado em todos os momentos da minha vida, junto a minha irmã pela paciência durante toda a graduação.

A todos os amigos e familiares que contribuíram para realização deste trabalho e do curso, o meu muito obrigado a todos. E aos meus professores que eu tive o prazer de ser aluno o meu agradecimento.

Agradeço muito a minha orientadora, a professora e mestre Suelem Farias, pelos inúmeros conselhos e ajudas, para finalização deste trabalho, muito obrigado. Também aos professores, o Doutor Gabriel Guimarães pelos conselhos na banca do Trabalho de conclusão de curso 1 (TCC 1), e também a professora mestre Jaqueline Matsuoka pela participação e sugestões, o engenheiro agrimensor e cartógrafo da UFU Weldon Martins dos Santos pelas sugestões na banca. E também a professora e doutora Luziane Ribeiro Indjai pela participação e sugestões na banca.

## RESUMO

Este trabalho tem como o objetivo realizar a elaboração de um programa na linguagem scilab capaz de realizar os cálculos de um levantamento topográfico planialtimétrico, utilizando uma interface gráfica para introduzir dados, vindo de campo, para calcular e gerar resultados das técnicas de poligonal fechada e nivelamento geométrico composto a partir das equações e dos dados trazidos de campo demonstrados na literatura. A metodologia deste trabalho baseou-se na NBR13133 – Execução de levantamento topográfico, na literatura a partir de apostilas e livros amplamente utilizados entre estudantes, professores e profissionais da área de topografia. Utilizando os seus métodos e dados para validação dos resultados, sendo desenvolvido uma interface gráfica dentro do software scilab, com objetivo de facilitar a interpretação visual para o cálculo do programa que foi dividido em três partes: “Cálculos de (Azimute, Ângulos e Distância), cálculo da poligonal fechada e cálculo do nivelamento geométrico”. Os resultados utilizando o teodolito para a poligonal fechada com 5 vértices em comparação com o trabalho de referência foram idênticos, no que diz respeito as diferenças de erro angular que resultaram em  $0^0$  graus. O erro linear teve uma diferença de 0,00123764 metro ou aproximadamente 1 milímetro e a diferença para área foi de  $0,315 \text{ m}^2$ . Já o nivelamento geométrico composto, utilizando o método de visadas iguais, foi calculado utilizando 36 pontos para nivelamento e para o contranivelamento que obtiveram um erro altimétrico de 0,0086 metro ou aproximadamente 8 milímetros, uma diferença de 0 metro com o trabalho de referência, todos dentro da tolerância estabelecida. Portanto, conclui-se que foi finalizado de forma satisfatória a interface gráfica do programa, a organização dos dados para serem calculados dentro das tolerâncias estabelecidas pela literatura e as normas técnicas. Sendo este programa disponibilizado gratuito como o software scilab, gerando conhecimento na área de topografia e para aprendizagem desta linguagem scilab.

**Palavras-chave:** Software Scilab. Levantamento Topográfico. Poligonal Fechada. Nivelamento Geométrico.

## LISTA DE FIGURAS

Figura 1 - Poligonal Fechada.....	16
Figura 2 - Quadrantes do Azimute.....	17
Figura 3 - Classificação das poligonais segundo a NBR13133.....	21
Figura 4 - Método de Visadas Iguais.....	23
Figura 5 - Erro de colimação e curvatura terrestre.....	23
Figura 6 - Fluxograma com as etapas do desenvolvimento deste trabalho.....	28
Figura 7 - Processo da instalação do software e o modulo GUI.....	29
Figura 8 - O criador de interface gráfica e criação de um elemento gráfico.....	31
Figura 9 - Geração do código da interface gráfica desenvolvida no guibuilder.....	32
Figura 10 - Equipamentos que podem obter os dados para estes trabalho na topografia.....	33
Figura 11 - Modelo do txt da Poligonal para inserir no programa.....	35
Figura 12 - Cálculos Realizados a partir dos dados de Entrada.....	36
Figura 13 - Modelo do txt do Nivelamento geométrico composto.....	37
Figura 14 - Cálculos realizados do nivelamento a partir dos dados e leituras de entrada.....	38
Figura 15 - Interface Gráfica do Programa no Scilab.....	39
Figura 16 - Dados de entrada e resultados na interface gráfica.....	40
Figura 17 - Dados de entrada da poligonal fechada na interface gráfica.....	42
Figura 18 - Dados de entrada do arquivo(txt) da poligonal.....	42
Figura 19 - Resultados do cálculo da poligonal na interface do programa.....	43
Figura 20 - Distribuição dos Pontos com os dados utilizados.....	44
Figura 21 - Comparação dos resultados da poligonal do programa com a literatura.....	45
Figura 22 - Dados Iniciais do Nivelamento.....	47
Figura 23 - Dados de Campo do Nivelamento Geométrico composto.....	48
Figura 24 - Resultados do cálculo do nivelamento na interface gráfica.....	49
Figura 25 - Comparação das distâncias e desníveis do nivelamento.....	51
Figura 26 - Croqui das altitudes dos pontos do nivelamento.....	52

## LISTA DE GRÁFICOS

Gráfico 1 - Comparação do erro linear.....	46
Gráfico 2 - Comparação das Coordenadas Corrigidas.....	46
Gráfico 3 - Comparação do cálculo de áreas.....	47

## LISTA DE TABELAS

Tabela 1 - Classificação dos Teodolitos.....	21
Tabela 2 - Classificação das Estações Totais.....	21
Tabela 3 - Classificação dos níveis segundo a NBR 13133.....	22
Tabela 4 - Especificações do formato dos dados inseridos para calcular.....	34
Tabela 5 - Dados de Entrada utilizados para cálculos com o as leituras do teodolito.....	34
Tabela 6 - Especificações do formato dos dados iniciais da Poligonal Fechada.....	35
Tabela 7 - Formato dos dados iniciais do cálculo do método de visadas iguais.....	36
Tabela 8 - Comparação dos resultados do scilab com a literatura.....	41



## **LISTA DE QUADROS**

Quadro 1 - Resultados de Erros, Tolerância e Desnível do Nivelamento.....	50
---	----

## SIGLAS E ABREVIATURAS

ABNT	Associação Brasileira de Normas Técnicas
ATOMS	<i>Automatic modules Management</i>
ENPC	<i>École Nationale des Ponts et Chaussées</i>
GUI	<i>Graphical User Interface</i>
INRIA	<i>Institut National de Recherche en Informatique et en Automatique</i>
PD	Pontaria Direta
PI	Pontaria Inversa
UFU	Universidade Federal de Uberlândia

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO.....</b>	<b>12</b>
<b>1.1</b>	<b>Objetivos.....</b>	<b>13</b>
<b>1.1.1</b>	<b>Objetivo Geral.....</b>	<b>13</b>
<b>1.1.2</b>	<b>Objetivo Especifico.....</b>	<b>13</b>
<b>1.2</b>	<b>Justificativa.....</b>	<b>13</b>
<b>2</b>	<b>REVISÃO BIBLIOGRÁFICA.....</b>	<b>14</b>
<b>2.1</b>	<b>Topografia.....</b>	<b>14</b>
<b>2.1.1</b>	<b>Levantamento Planimétrico.....</b>	<b>15</b>
2.1.1.1	Poligonal Fechada.....	16
<b>2.1.2</b>	<b>Levantamento Altimétrico.....</b>	<b>21</b>
2.1.2.1	Nivelamento Geométrico.....	22
<b>2.2</b>	<b>Sistema Topográfico Local.....</b>	<b>25</b>
<b>2.3</b>	<b>Software.....</b>	<b>25</b>
<b>2.3.1</b>	<b>Scilab.....</b>	<b>26</b>
<b>3</b>	<b>MATERIAL E MÉTODOS.....</b>	<b>27</b>
<b>3.1</b>	<b>Material.....</b>	<b>28</b>
<b>3.1.1</b>	<b>Scilab.....</b>	<b>28</b>
<b>3.2</b>	<b>Métodos.....</b>	<b>30</b>
<b>3.2.1</b>	<b>Levantamento Topográfico Planialtimétrico.....</b>	<b>30</b>
3.2.1.1	Cálculos de Azimute, Ângulos e Distância.....	33
3.2.1.2	Poligonal Fechada.....	35
3.2.1.3	Nivelamento Geométrico.....	36
<b>4</b>	<b>RESULTADOS E DISCUSSÃO.....</b>	<b>38</b>
<b>4.1</b>	<b>Levantamento Topográfico Planialtimétrico.....</b>	<b>38</b>
<b>4.1.1</b>	<b>Cálculos de Azimute, Ângulos e Distância.....</b>	<b>40</b>
<b>4.1.2</b>	<b>Poligonal Fechada.....</b>	<b>41</b>
<b>4.1.3</b>	<b>Nivelamento Geométrico.....</b>	<b>47</b>
<b>5</b>	<b>CONCLUSÃO.....</b>	<b>52</b>
	<b>REFERÊNCIAS.....</b>	<b>54</b>
	<b>ANEXOS A – DADOS DE ENTRADA.....</b>	<b>56</b>

<b>APÊNDICE A – CÓDIGO DESTE TRABALHO NO SCILAB.....</b>	<b>58</b>
<b>APÊNDICE B - RESULTADOS DA POLIGONAL FECHADA.....</b>	<b>83</b>
<b>APÊNDICE C - RESULTADOS DO NIVELAMENTO.....</b>	<b>84</b>

## 1 INTRODUÇÃO

A topografia é uma ciência que tem como objetivo representar uma porção da superfície terrestre, através das medições de ângulos, distâncias e desníveis, para se obter as coordenadas (X, Y, Z) (VEIGA, ZANETTI e FAGGION, 2012).

E de acordo com Veiga, Zanetti e Faggion (2012), o levantamento planialtimétrico determina a posição planimétrica (coordenadas X e Y) e a altitude de pontos (coordenada Z), empregando-se técnicas muito precisas como é o caso da poligonação com o método de poligonal fechada que é um dos métodos mais utilizados para a determinação de coordenadas dos pontos (X, Y) em topografia, além do método de visadas iguais do nivelamento geométrico, que é o método mais preciso e de larga aplicação na engenharia, para se determinar as altitudes dos pontos (Z). Sendo que os levantamentos topográficos seguem as normas da NBR 13133.

A partir das coordenadas (X, Y, Z), permite representar o relevo por perfil topográfico, por pontos cotados e por curvas de nível na topografia, do qual os perfis topográficos possuem inúmeras aplicações em engenharia, em locais destinados a implantação de obras em rodovias, ferrovias e canais (CORDINI, 2015).

Sendo que este levantamento topográfico pode ser realizado utilizando os equipamentos como teodolito ou estação total para poligonação e o nível para o nivelamento geométrico, com o auxílio fundamental de um software para se calcular o levantamento. Sendo que equipamentos como o teodolito óptico e o nível óptico, por exemplo, que não possuem um gravador de dados acoplado, um software para processamento desses dados se torna muito importante principalmente se for gratuito.

E criando um programa em um software de código aberto e gratuito como o scilab, que possui como característica ser um software de computação numérica, com uma sofisticada estrutura de dados, além da geração de gráficos 2D e 3D, então qualquer programa desenvolvido neste software gera um grande conhecimento, por possuir um ambiente computacional muito robusto e ser bastante utilizado profissionalmente e também cientificamente nas engenharias (SANTOS, 2016).

Desta forma utilizando a linguagem do software do mesmo nome scilab, é possível calcular o levantamento topográfico planialtimétrico, com os métodos mais precisos e utilizados na engenharia como a poligonal fechada e o método de visadas iguais com nivelamento geométrico composto, juntamente com os cálculos dos ângulos e direções a partir do instrumento teodolito, com

a utilização de um software de código aberto, gerando muito conhecimento sem restrições. A partir de dados obtidos através do levantamento da literatura para a validação dos resultados.

## **1.1 Objetivos**

### **1.1.1 Objetivo Geral**

Elaborar um programa computacional em scilab para calcular o levantamento topográfico planialtimétrico.

### **1.1.2 Objetivo Especifico**

- ✓ Realizar a revisão bibliográfica sobre os métodos de levantamento topográfico planialtimétrico;
- ✓ Escolher os dados da literatura que serão organizados e calculados para validar o software;
- ✓ Criar uma interface gráfica para introdução dos dados;
- ✓ Implementar as equações no cálculo do azimute, ângulos, direções, poligonal fechada e nivelamento geométrico composto;
- ✓ Comparar os resultados do software scilab com os da literatura;
- ✓ Disponibilizar o código completo para estudos futuros.

## **1.2 Justificativa**

A Topografia é umas das ciências que mais geram dados, que após serem organizados e calculados em programas, permitem obter informações e conhecimento sobre a área estudada como por exemplo, a altitude do relevo, monitoramento de fenômenos naturais e planejamento urbano.

Sendo o levantamento topográfico planialtimétrico, uma das técnicas mais utilizadas para de obter as coordenadas 3D dos pontos.

E segundo ABNT (1994), o levantamento planialtimétrico tem como destino o conhecimento do terreno como o relevo, limites, áreas e localização. Com informações sobre o terreno para estudos preliminares de projetos básicos.

Para Veiga, Zanetti, Faggion (2012), é extremamente prático o processo computacional para se obter gráficos e o cálculo de áreas, a partir de métodos analíticos, como e o caso da poligonal fechada.

Conforme Campos (2010), o scilab é um software de computação científica, totalmente gratuito e com o código fonte aberto, e que tem como uma grande vantagem possuir no software o seu interpretador da linguagem em scilab. O que qualquer programa gerado no scilab, pode-se gerar um conhecimento muito grande.

Assim, a elaboração de um programa para o cálculo do levantamento topográfico planialtimétrico, com os métodos mais utilizados e precisos da topografia como o da poligonal fechada e nivelamento geométrico, com acesso a código fonte, gera um grande conhecimento nas áreas de topografia, como também na programação com a utilização da linguagem e dos complementos do software scilab, muito utilizados cientificamente e no mercado de trabalho. E poderá ser aplicado tanto no mercado de trabalho como no meio acadêmico, com aplicação na topografia, e para geração de outros produtos na engenharia.

## **2 REVISÃO BIBLIOGRÁFICA**

### **2.1 Topografia**

De acordo com Veiga, Zanetti, Faggion (2012), a topografia tem o seu significado etimologicamente de TOPOS em grego lugar e GRAPHEN como descrição, assim a topografia tem o significado de descrição do lugar. Do qual a topografia tem como objetivo principal a execução do

levantamento para medições de ângulos, distâncias e desníveis, que permite se representar uma determinada porção da superfície terrestre com uma escala adequada.

Ainda segundo Cordini (2014), a topografia é uma ciência com base na geometria e trigonometria plana, empregando medições de distâncias horizontais ou inclinadas, de ângulos horizontais e verticais, e a orientação com o azimute e a diferenças de nível, com a finalidade de se obter a representação, a partir de pontos com as projeções ortogonais sobre um plano de referência, que descreve a forma, dimensão e posição relativa de uma determinada porção do terreno.

A topografia se divide em topometria e topologia. Sendo a planimetria e altimetria estudada dentro da topometria, que é responsável pelas análises dos procedimentos clássicos da medição de distâncias, ângulos e desníveis, com o propósito da determinação do posicionamento relativo dos pontos topográficos. Já a topologia tem como objetivo, estudar as formas da superfície terrestre com as leis que regem o seu modelado. (VEIGA, ZANETTI e FAGGION, 2012).

### **2.1.1 Levantamento Planimétrico**

Segundo Veiga, Zanetti, Faggion (2012), O levantamento planimétrico procura determinar a posição dos pontos de coordenadas (X, Y), onde é necessário de início, determinar pontos de apoio (poligonal principal) ao levantamento, e a partir destes, são levantados os demais pontos (irradiações) permitindo representar a área levantada. Aplicando técnicas de levantamento topográfico planimétrico.

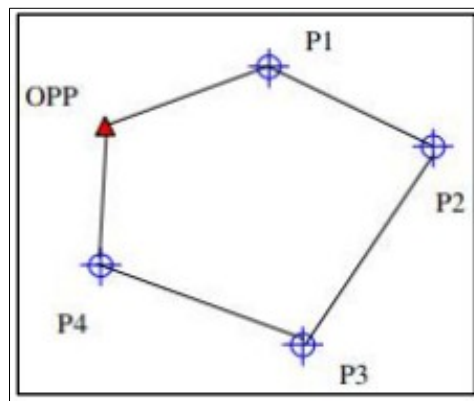
Ainda, de acordo com Veiga, Zanetti, Faggion (2012), às técnicas de poligonização são as mais empregadas para se determinar as coordenadas (X, Y) dos pontos na topografia. Do qual para realizar o levantamento topográfico de uma poligonal, é necessário aplicar o método de caminhamento, que percorre o contorno de um percurso definido por uma série de pontos, onde são medidos todos os ângulos, distâncias e uma orientação inicial. Partindo de um ponto de apoio com coordenadas conhecidas no levantamento, e a partir desta se calcula as coordenadas de todos os outros pontos da formação da poligonal. Sendo que as poligonais levantadas em campo podem ser fechadas, enquadradas ou abertas.



### 2.1.1.1 Poligonal Fechada

Esta é a poligonal que parte de um ponto com coordenadas conhecidas (OPP) e retorna ao mesmo ponto (Figura 1). E este procedimento possibilita a verificação dos erros de fechamento angular e linear (VEIGA; ZANETTI; FAGGION 2012).

Figura 1 - Poligonal Fechada



Fonte: Veiga, Zanetti e Faggion, 2012.

Destacando que é necessário verificar se o erro cometido está na tolerância estabelecida, para logo em seguida classificar a poligonal segundo as normas técnicas que e a NBR 13.133. Sendo fundamental a verificação do erro angular e linear, para dar qualidade e confiabilidade no trabalho desenvolvido e se apresentar o erro maior que a tolerância, deve-se voltar ao campo (RODRIGUES, 2008).

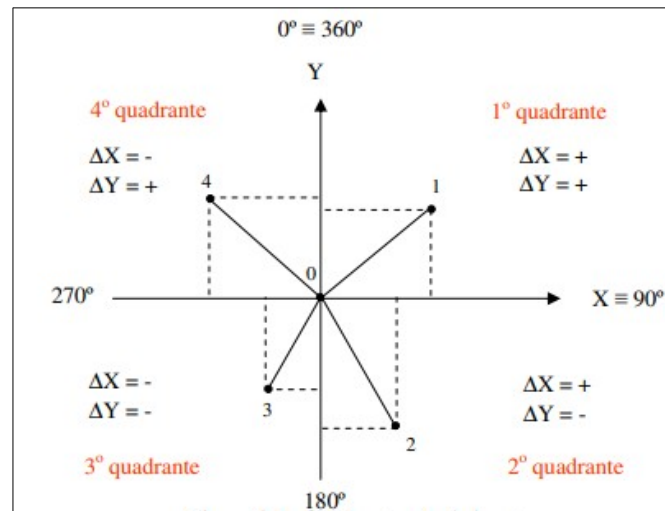
Para Veiga, Zanetti e Faggion (2012), através dos dados medidos em campo como ângulos e distâncias, a orientação inicial e as coordenadas do ponto de partida, é possível calcular as coordenadas de todos os pontos da poligonal. Podendo-se calcular o azimute de partida a partir das coordenadas planimétricas de dois pontos, como mostrado nas equações abaixo, onde o azimute de uma direção é medido a partir do Norte, no sentido horário, e que varia segundo o quadrante conforme mostrado na Figura 2.

$$Az_{01} = \arctg\left(\frac{\Delta X}{\Delta Y}\right) \quad (2.1)$$

$$\Delta X = X_1 - X_0 \quad (2.2)$$

$$\Delta Y = Y_1 - Y_0 \quad (2.3)$$

Figura 2 - Quadrantes do Azimute



Fonte: Veiga, Zanetti e Faggion, 2012.

Para se calcular o ângulo vertical de dados do teodolito, pode-se calcular este ângulo a partir de uma PD e uma PI ao mesmo alvo no teodolito, como demonstrado na equação abaixo (VEIGA; ZANETTI; FAGGION 2012).

$$Z = \frac{360^0 + ZPD - ZPI}{2} \quad (2.4)$$

Onde:

Z: o Ângulo Vertical isento do Erro de Verticalidade.

PD: Pontaria Direta;

PI: Pontaria Inversa.

E o cálculo dos ângulos horizontais a partir da instalação do teodolito, e feita através das seguintes fórmulas (VEIGA; ZANETTI; FAGGION 2012):

$$AH = \frac{LPD + LPI}{2} \pm 90^0 \quad (2.5)$$

Sendo: + se  $PD > PI$

- se  $PD < PI$

Onde:

PD: Pontaria Direta;  
 PI: Pontaria Inversa;  
 LPD: Leitura em PD;  
 LPI: Leitura em PI.

Já para se calcular a distância horizontal, que é obtida através do equipamento teodolito com a utilização da taqueometria, é necessário adotar uma mira estadimétrica (VEIGA; ZANETTI; FAGGION 2012):

$$Dh = G * K * \text{sen}^2 Z \quad (2.6)$$

Onde:

G: Número Gerador da Mira Real: (G = Leitura Superior - Leitura Inferior);  
 K: é a constante estadimétrica do instrumento, definida pelo fabricante igual a 100;  
 Z: Ângulo Vertical.

Na verificação do erro de fechamento angular, é muito importante verificar os ângulos medidos, e como é um polígono fechado é possível verificar se houve algum erro na medição dos ângulos. E seu cálculo se da seguinte forma (VEIGA; ZANETTI; FAGGION 2012):

$$EA = \Sigma AH - (n + 2) \times 180^\circ - \text{Para os ângulos Externos} \quad (2.7)$$

$$EA = \Sigma AH - (n - 2) \times 180^\circ - \text{Para os ângulos internos} \quad (2.8)$$

Onde:

EA: Erro Angular.  
 AH: Ângulos horizontais.  
 $\Sigma AH$ : Somatório dos ângulos medidos.  
 n: Número de pontos da poligonal.

$$TA = p * \sqrt{m} \quad (2.9)$$

m: número de ângulos medidos na poligonal;  
 TA: Tolerância Angular;

Verificação se há o erro Angular

$|EA| \leq |TA|$  então OK!

Correção do erro angular

$$c = - (EA / n) \quad (2.10)$$

$$AHC = AH + c \quad (2.11)$$

Sendo que o:

AHC: Ângulos Horizontais Corrigidos;

c: Correção do erro angular.

E segundo Veiga, Zanetti e Faggion (2012), os próximos cálculos da poligonal fechada são os cálculos dos azimutes, das coordenadas parciais, da verificação do erro de fechamento linear e das coordenadas finais como descrito das equações abaixo:

Cálculo dos Azimutes

Se a poligonal e no sentido Horário então a fórmula e

$$AZI(\text{Azimute}) = AZ_{i-1,i} + AH_i - 180^0 \quad (2.12)$$

Se for no sentido Anti - Horário da Poligonal

$$AZI(\text{Azimute}) = AZ_{i-1,i} - AH_i + 180^0 \quad (2.13)$$

Onde,

i varia de 0 a (n - 1);

$AZI_{i-1,i}$ : Azimute inicial da direção;

n: é o número de estações da poligonal;

se  $i + 1 > n$  então  $i = 0$ ;

se  $i - 1 < n$  então  $i = n$ .

Cálculo das coordenadas provisórias

$$X_i = X_{i-1} + d_{i-1,i} \cdot \text{ksen}(Az_{i-1,i}) \quad (2.14)$$

$$Y_i = Y_{i-1} + d_{i-1,i} \cdot \text{ksen}(Az_{i-1,i}) \quad (2.15)$$

Verificação do erro linear

$$e_x = X_{\text{OPP}}^{\text{Calculado}} - X_{\text{OPP}} \quad (2.16)$$

$$e_y = Y_{\text{OPP}}^{\text{Calculado}} - Y_{\text{OPP}} \quad (2.17)$$

$$EP = \sqrt{(ex^2 + ey^2)} \quad (2.18)$$

$$P = \Sigma d \quad (2.19)$$

$$Z_{\text{escala}} = \frac{P}{EP} \quad (2.20)$$

Onde,

elx: erro do eixo x;

ely: erro do eixo y;

EP: erro planimétrico;

d: distâncias da poligonal;

$\Sigma d$ : somatório das distâncias;

P: é o perímetro da poligonal;

Z\_escala: erro em forma de escala.

Tlinear: tolerância linear.

Verificar se há o erro linear

Erro planimétrico(Z) < tolerância linear (Tlinear)

Cálculo das coordenadas corrigidas

$$Cxi = - ex * \frac{d_{i-1,i}}{\Sigma d} \quad (2.21)$$

$$Cyi = - ey * \frac{d_{i-1,i}}{\Sigma d} \quad (2.22)$$

$$X_i^c = X_{i-1}^c + d_{i-1,i} \cdot \text{sen}(Az_{i-1,i}) + Cyi \quad (2.23)$$

$$Y_i^c = Y_{i-1}^c + d_{i-1,i} \cdot \text{cos}(Az_{i-1,i}) + Cxi \quad (2.24)$$

Onde

Cxi: correção para as coordenadas Xi;

Cyi: correção para as coordenadas Yi;

$d_{i-1,i}$ : distância parcial.

Do qual a poligonal fechada possibilita a verificação e controle dos erros no levantamento, com os equipamentos utilizados em campo, seja teodolito ou estação total. Na Tabela 1 e Tabela 2

são apresentados as precisões da estação total e do Teodolito (VEIGA; ZANETTI; FAGGION 2012).

Tabela 1 - Classificação dos Teodolitos

Classes de Teodolitos	Desvio-padrão	Precisão Angular
1 – Precisão Baixa		$\leq \pm 30''$
2 – Precisão Média		$\leq \pm 07''$
3 – Precisão Alta		$\leq \pm 02''$

Fonte: Veiga, Zanetti e Faggion, 2012.

Tabela 2 - Classificação das Estações Totais

Classes de Estações Totais	Desvio – Padrão	Precisão Angular	Desvio – Padrão	Precisão Linear
1 – Precisão Baixa		$\leq 30''$		(5 mm + 10 ppm * d)
2 – Precisão Média		$\leq 07''$		(5 mm + 5 ppm * d)
3 – Precisão Alta		$\leq 02''$		(3 mm + 3 ppm * d)

Fonte: Veiga, Zanetti e Faggion, 2012.

Segundo Rodrigues (2008), a NBR 13133 classifica as poligonais em função do erro linear ocorrido em campo, utilizando também o somatório das distâncias horizontais da poligonal (L), que é definida como a extensão da poligonal. E como demonstrado na Figura 3 abaixo.

Figura 3 - Classificação das poligonais segundo a NBR13133

Classes	IP	IIP	IIIP	IVP	VP	IPRC	IIPRC
Erro linear (em m)	$0,10 \cdot \sqrt{L(Km)}$	$0,30 \cdot \sqrt{L(Km)}$	$0,42 \cdot \sqrt{L(Km)}$	$0,56 \cdot \sqrt{L(Km)}$	$2,20 \cdot \sqrt{L(Km)}$	$0,07 \cdot \sqrt{L(Km)}$	$0,30 \cdot \sqrt{L(Km)}$

Fonte: Rodrigues, 2008.

E finalizando o cálculo da poligonal fechada, e obtendo as coordenadas (X, Y), pode-se calcular a área da poligonal, pois a avaliação de áreas é uma atividade comum na topografia (VEIGA; ZANETTI; FAGGION 2012). Sendo que a equação empregada para calcular a área é definida assim (CINTRA, 1996):

$$2A = \sum (y_i * x_{i+1}) - \sum (x_i * y_{i+1}) \quad (2.25)$$

### 2.1.2 Levantamento Altimétrico

Este levantamento tem por finalidade, determinar as alturas relativas de uma superfície de referência, dos pontos de apoio ou dos pontos de detalhes, presume-se o conhecimento de suas

posições planimétricas, objetivando representar altimetricamente a superfície levantada (ABNT, 1994).

Para determinação do valor da altitude dos pontos, é necessário obter os desníveis entre os pontos. Conhecendo o valor de referência inicial para se calcular as demais altitudes, sendo estes métodos denominados na topografia de nivelamento, que permite determinar os desníveis com precisão de centímetros e até milímetros. Considerando a finalidade do trabalho para escolha do método (VEIGA; ZANETTI; FAGGION 2012).

### 2.1.2.1 Nivelamento Geométrico

Este nivelamento visa determinar as diferenças de nível, empregando instrumentos que fornecem retas do plano horizontal. Fazendo a interseção do plano com a mira, onde são colocados sucessivamente nos planos topográficos de estudo, permitindo determinar as alturas de leituras nos seus respectivos pontos, obtendo diferença de nível na diferença entre os valores encontrados. Utilizando diferentes métodos de nivelamento geométrico, que conforme o instrumento utilizado é possível classificar o nivelamento conforme a sua precisão (Tabela 3) (COMASTRI, TULER, 2013).

Tabela 3 - Classificação dos níveis segundo a NBR 13133

Classes de níveis	Desvio-padrão
1 - Precisão Baixa	$> \pm 10$ mm/km
2 - Precisão Média	$\leq \pm 10$ mm/km
3 - Precisão Alta	$\leq \pm 3$ mm/km
4 - Precisão Muito Alta	$\leq \pm 1$ mm/km

Fonte: ABNT (1994).

O nivelamento geométrico pode ser simples ou composto, sendo que no simples o desnível entre os pontos de interesse, é determinado apenas com uma única instalação do equipamento, um único lance. Já o nivelamento geométrico composto e o desnível entre os pontos a ser determinado através de vários lances (VEIGA, ZANETTI, FAGGION, 2012).

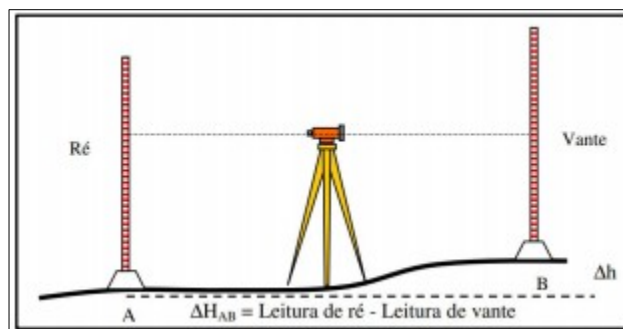
E segundo Veiga, Zanetti, Faggion (2012), é possível dividir o nivelamento geométrico em quatro métodos:

- ✓ Visadas iguais;
- ✓ Visadas extremas;

- ✓ Visadas recíprocas;
- ✓ Visadas equidistantes.

O método de visadas iguais é um dos métodos de nivelamento mais empregados na topografia, por ser o método mais preciso. O seu desenvolvimento é feito a partir da instalação de duas miras colocadas à mesma distância do nível, sobre os pontos a ser determinado o desnível, como descrito na Figura 4. E assim sendo efetuadas as leituras, que visa determinar o desnível através da diferença entre a leitura de ré e vante (VEIGA, ZANETTI, FAGGION, 2012).

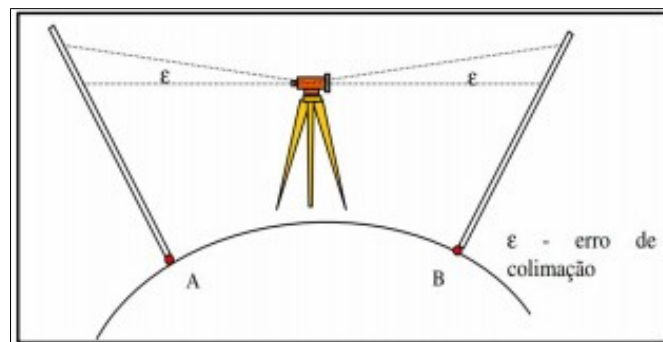
Figura 4 - Método de Visadas Iguais



Fonte: Veiga, Zanetti, Faggion (2012).

Sendo a grande vantagem deste método a minimização de erros causados pela curvatura terrestre e de colimação do nível como mostrado na (Figura 5) (VEIGA, ZANETTI, FAGGION, 2012).

Figura 5 - Erro de colimação e curvatura terrestre



Fonte: Veiga, Zanetti, Faggion (2012).

Assim para Veiga, Zanetti, Faggion (2012), calcula-se o método de visadas do nivelamento geométrico a partir das seguintes fórmulas abaixo:

$$dre = (FS_{RE} - FI_{RE}) * 100; \quad (2.26)$$

Onde,

dre: distância de Re;



FS: Fio superior;

FI: Fio Inferior.

$$dvante = (FS\_Vante - FI\_Vante) * 100 \quad (2.27)$$

$$nivelado\_re = FS\_RE - FI\_RE \quad (2.28)$$

$$nivelado\_vante = FS\_VANTE - FI\_VANTE \quad (2.29)$$

$$Desnível\ R-V = (FN\_Re - FN\_Vante) \quad (2.30)$$

Onde,

R: Re;

V: Vante;

FN: Fio nivelador.

$$h = h^0 + Desnível\_R-V \quad (2.31)$$

Sendo que,

$h^0$ : Altitude Conhecida;

$h$  = Altitude.

Para se verificar o erro no procedimento de campo utilizando este método em um levantamento, emprega-se o contranivelamento, que e necessário se realizar os seguintes cálculos (VEIGA; ZANETTI; FAGGION 2014):

$$\text{Tolerância altimétrica} = np * \sqrt{(k)} \quad (2.32)$$

Onde,

$np$ : é um valor em centímetros ou milímetros;

$k$ : é a distância média nivelada em quilômetros.

$$Dm = (DNIV + DCON)/2 \quad (2.33)$$

DNIV: Distância nivelada (nivelamento);

DCON: Distância nivelada (contranivelamento);

Distância média nivelada ( $Dm$ ).

$$\text{Erro} = |\Delta HNIV| - |\Delta HCON| \quad (2.34)$$

$\Delta\text{HNIV}$  : Desnível do nivelamento;

$\Delta\text{HCON}$  : Desnível do contra-nivelamento  $\Delta\text{HCON}$ ;

(Erro) < (Tolerância altimétrica) Ok!

## 2.2 Sistema Topográfico Local

Este sistema de projeção é utilizado para levantamentos topográficos com apoio em rede de referência cadastral pelo método direto clássico, visando a representar as posições relativas dos acidentes levantados a partir de medições angulares, lineares, horizontais e verticais (ABNT, 1998).

Sendo que neste sistema representa em planta as posições relativas de pontos de um levantamento topográfico com origem em um ponto de coordenadas geodésicas conhecidas, com todos os ângulos e distâncias de sua determinação com representação sobre o plano tangente à superfície de referência (elipsoide de referência) do sistema geodésico adotado (ABNT, 1998).

## 2.3 Software

Segundo Sommerville (2011), os softwares são programas de computador com documentação associada, e produtos de software podem ser desenvolvidos para um cliente específico ou para o mercado em geral. O software se tornou fundamental no mundo moderno, com aplicações importantes nas áreas de infraestruturas e serviços nacionais e internacionais, controlados por sistemas computacionais. Sendo uma necessidade das pessoas criarem programas para seus negócios para simplificar seu trabalho, além dos cientistas e engenheiros que desenvolvem programas para processar seus dados experimentais.

E conforme Sommerville (2011), um processo de software é um conjunto de atividades relacionadas levando a produção de um produto de software. Nestas atividades se desenvolve o desenvolvimento de software do zero, com uma linguagem padrão de programação como C ou Java. E em todos os processos devem incluir às quatro atividades fundamentais na engenharia de software como:

- ✓ Especificação de software: que e funcionalidade do software e as restrições no seu funcionamento.
- ✓ Projeto e implementação de software: Atender as especificações do software.
- ✓ Validação de software: validar o software de acordo com a necessidade dos clientes.
- ✓ Evolução do software: incluir no software as mudanças de acordo com os clientes.

E o software é responsável por servir como uma interface entre as necessidades dos usuários e os dispositivos de hardware, onde no sistema operacional como o windows, por exemplo, se encontra uma diversidade de softwares com funções diferenciadas. Sendo que nos primeiros programas de computador foram-se desenvolvidos através de uma linguagem chamada de “linguagem de máquina”. Onde este cenário só começou a mudar quando surgiram as primeiras linguagens de montagem e as linguagens de alto nível, e estas linguagens facilitaram a construção de programas (PEREIRA, VISSOTTO, FRANCISCATTO, 2015).

Durante a execução de um programa feito em uma linguagem de alto nível, o interpretador traduz cada instrução e a executa imediatamente, onde temos como exemplos destas linguagens a Basic, PHP, Perl entre outras (PEREIRA, VISSOTTO, FRANCISCATTO, 2015).

Podendo o software ser livre ou fechado, sendo que o modelo aberto de desenvolvimento e distribuição de software traz inúmeras vantagens técnicas em relação ao software fechado. E umas destas vantagens são que os softwares abertos promovem a colaboração, a troca de conhecimentos, informações e ideias entre a comunidade de usuários de um determinado pacote ou sistema, é o caso de maior sucesso do software livre até hoje e o Linux (KON, 2001).

### 2.3.1 Scilab

O Scilab foi desenvolvido desde 1990 pelos pesquisadores do INRIA (*Institut National de Recherche en Informatique et en Automatique*) e do ENPC (*École Nationale des Ponts et Chaussées*) na França, sendo o scilab um software de computação científica gratuito e que possui o código fonte aberto, utilizado para visualização, e possui o seu próprio interpretador da sua linguagem em Scilab (CAMPOS, 2010).

E segundo Santos (2016), o Scilab é um software para computação numérica, muito semelhante ao Matlab, que ainda fornece um ambiente computacional extremamente robusto e de

forma gratuita, dispondo de sofisticadas estruturas de dados com ferramentas para geração de gráficos 2D e 3D, com simulações e otimizações. Sendo este software muito difundido e utilizado profissionalmente e cientificamente nas engenharias. Tendo como características utilizar a sua própria linguagem de Scilab, que é do mesmo nome do Software.

A linguagem Scilab utilizada no software é baseada na linguagem C, sendo esta uma linguagem de programação de alto nível, interpretada e não compilada, orientada à análise numérica. Tendo como característica a diferenciação entre letras maiúsculas e minúsculas, com operadores aritméticos, lógicos e relacionais com a manipulação de arquivos e comandos de controle de fluxo, como repetições e condicionais (SANTOS, 2016).

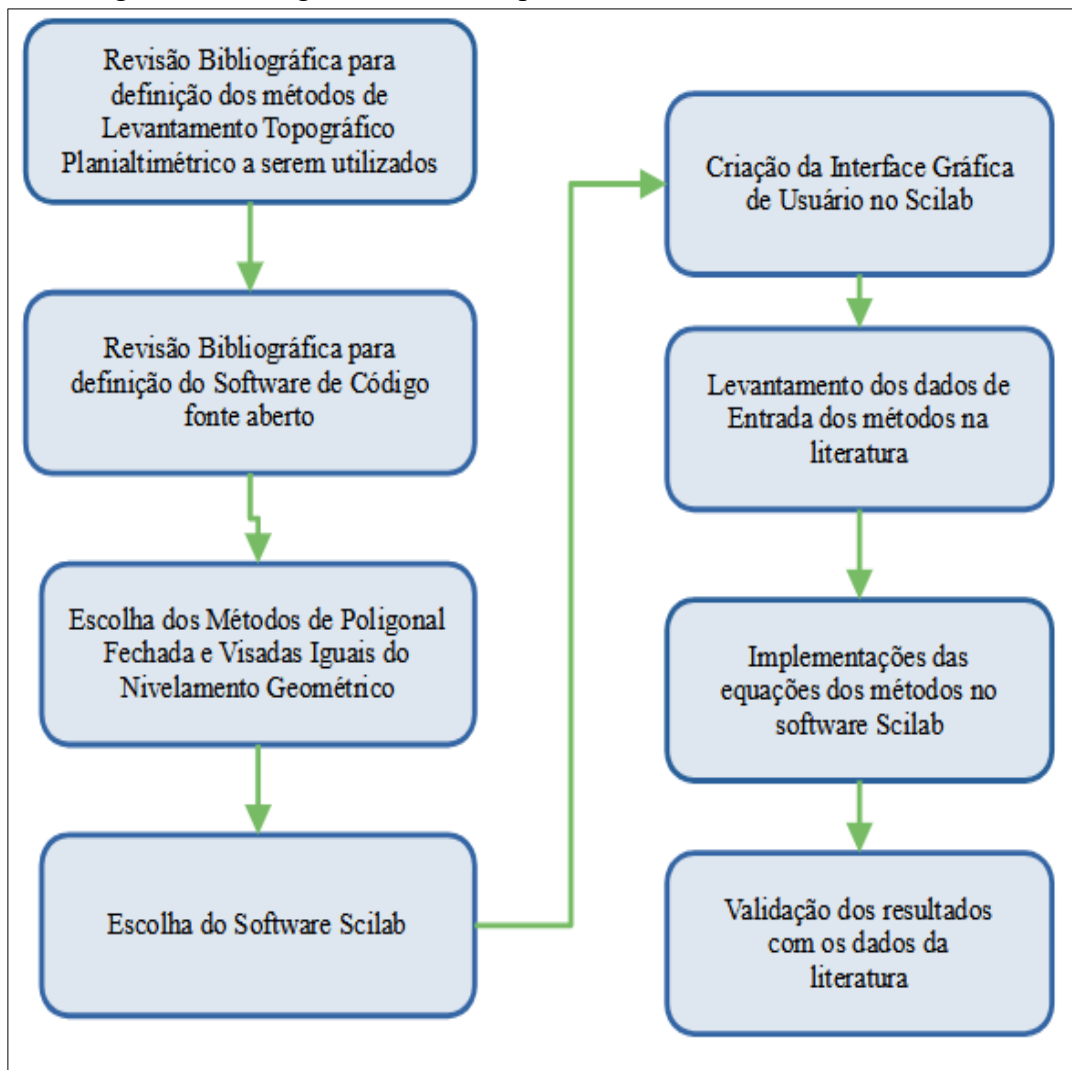
Segundo Scilab(2018), o scilab possui também caixas de ferramentas chamadas ATOMS (*Automatic modules Management for Scilab*) sendo repositório para módulos, permitindo a instalação ou remoção dos módulos com diferentes aplicações como, por exemplo, GUI Builder (*Graphic User Interface Builder*), que é um construtor de interface de usuário gráfico muito eficiente, que gera o próprio código da interface automático.

### **3 MATERIAL E MÉTODOS**

Neste trabalho foi realizada a revisão bibliográfica para definição dos métodos de levantamento topográfico a serem utilizados e, a escolha do software de código aberto, para criar um programa que utiliza os dados da literatura de referência para calcular e validar os seus resultados.

A Figura 6 abaixo, ilustra o fluxograma com todas as etapas para o desenvolvimento deste trabalho.

Figura 6 - Fluxograma com as etapas do desenvolvimento deste trabalho



Fonte: O autor, 2021.

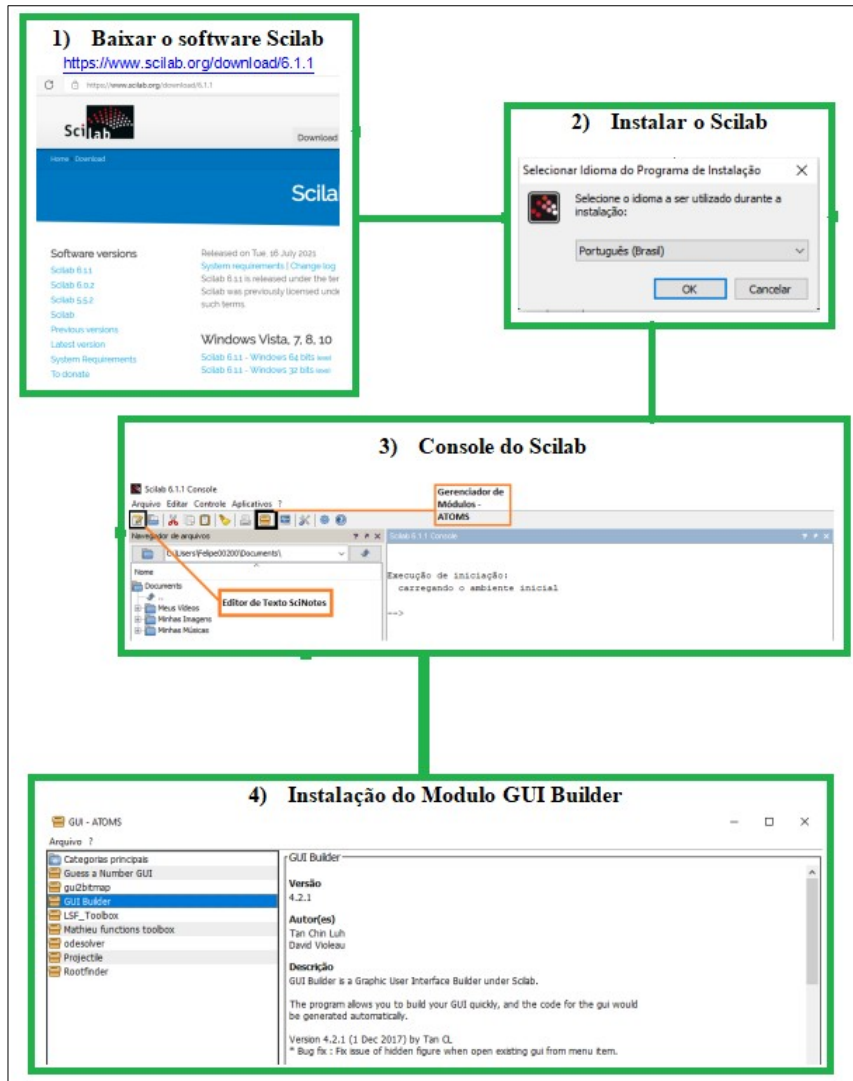
### 3.1 Material

#### 3.1.1 Scilab

O software scilab, serviu para calcular o levantamento topográfico planialtimétrico, por ser um software bastante utilizado na engenharia, com uma grande estrutura para processamento de dados.

Foi criada a interface gráfica dentro do scilab, utilizando o módulo *GUI Builder*, nas caixas de ferramentas *Atoms*, que também gera os códigos da interface criada automaticamente. Na Figura 7 mostra o processo de download do scilab até a instalação do módulo dentro do software.

Figura 7 - Processo da instalação do software e o modulo GUI.



Fonte: O autor, 2021.

Ainda segundo Santos (2016), no scilab não é necessário declarar as bibliotecas, nem a presença de uma função principal, como ocorre nas linguagens C ou C++, por exemplo. Além de possuir as funções empregadas neste trabalho a `clear` e `clc`:

**clear**: limpa a memória do Scilab, apagando todas as variáveis criadas anteriormente;

**clc**: limpa a tela do console, apagando todos os registros antigos.

Sendo que este software utilizado para a elaboração deste trabalho, foi empregado com as seguintes especificações como:

- ✓ Versão do Software: Scilab 6.1.1;
- ✓ Sistema operacional: Windows 64 bits;
- ✓ Modulo do Scilab utilizado: ATOMS - GUI Builder - Versão 4.2.1.

## 3.2 Métodos

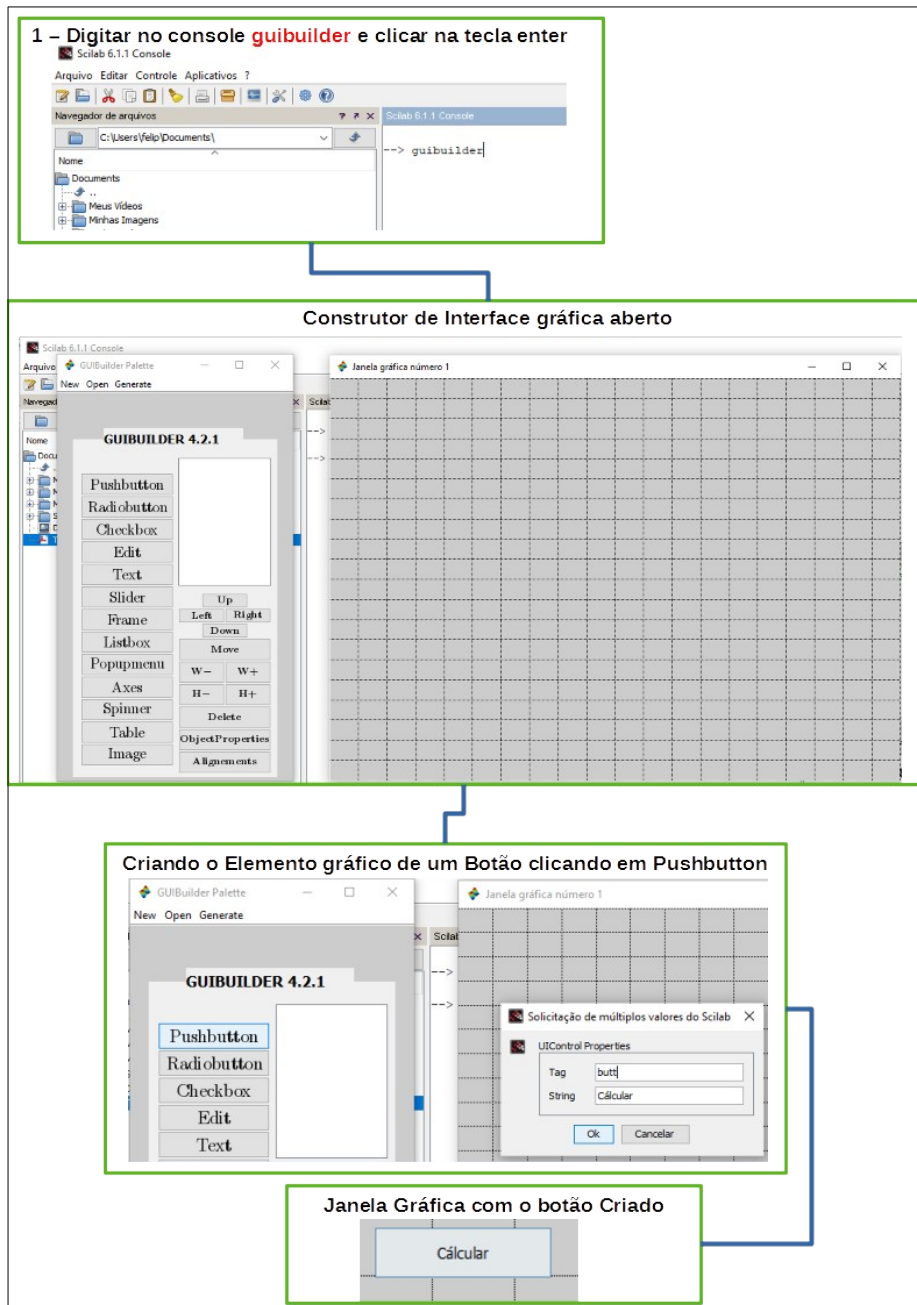
### 3.2.1 Levantamento Topográfico Planialtimétrico

E o resultado do cálculo do levantamento topográfico é de grande importância para representação de uma determinada região da superfície terrestre e também na geração de conhecimento da topografia e da linguagem scilab. Assim foi criado este programa para o cálculo a partir de dados inseridos pelo usuário. Sendo desenvolvido uma interface gráfica para se fazer estes cálculos tão importantes que e a base de vários projetos na engenharia.

Na criação desta interface scilab, foi instado o módulo do *gui builder* no scilab como mostrado na figura Figura 7, que e um criador de interface gráfica para o usuário, e assim foi elaborado a interface de forma prática para serem inseridos os dados e efetuar os cálculos a partir das equações inseridas no software, com métodos comprovados e de ampla aplicação na engenharia.

E neste módulo *gui*, pode ser desenvolver diferentes interfaces para aplicações tanto na engenharia como para outras ciências para se realizar cálculos numéricos, por exemplo. Além do software permiti a partir deste módulo desenvolver aplicações avançadas como a criação de interfaces como tabelas, botões, textos e análises de imagens, entre outras, onde cada elemento criado podem definir todas as suas propriedades como tamanho, posição e fontes, e este criador de interface pode ser visto aberto conforme mostrado na Figura 8.

Figura 8 - O criador de interface gráfica e criação de um elemento gráfico.

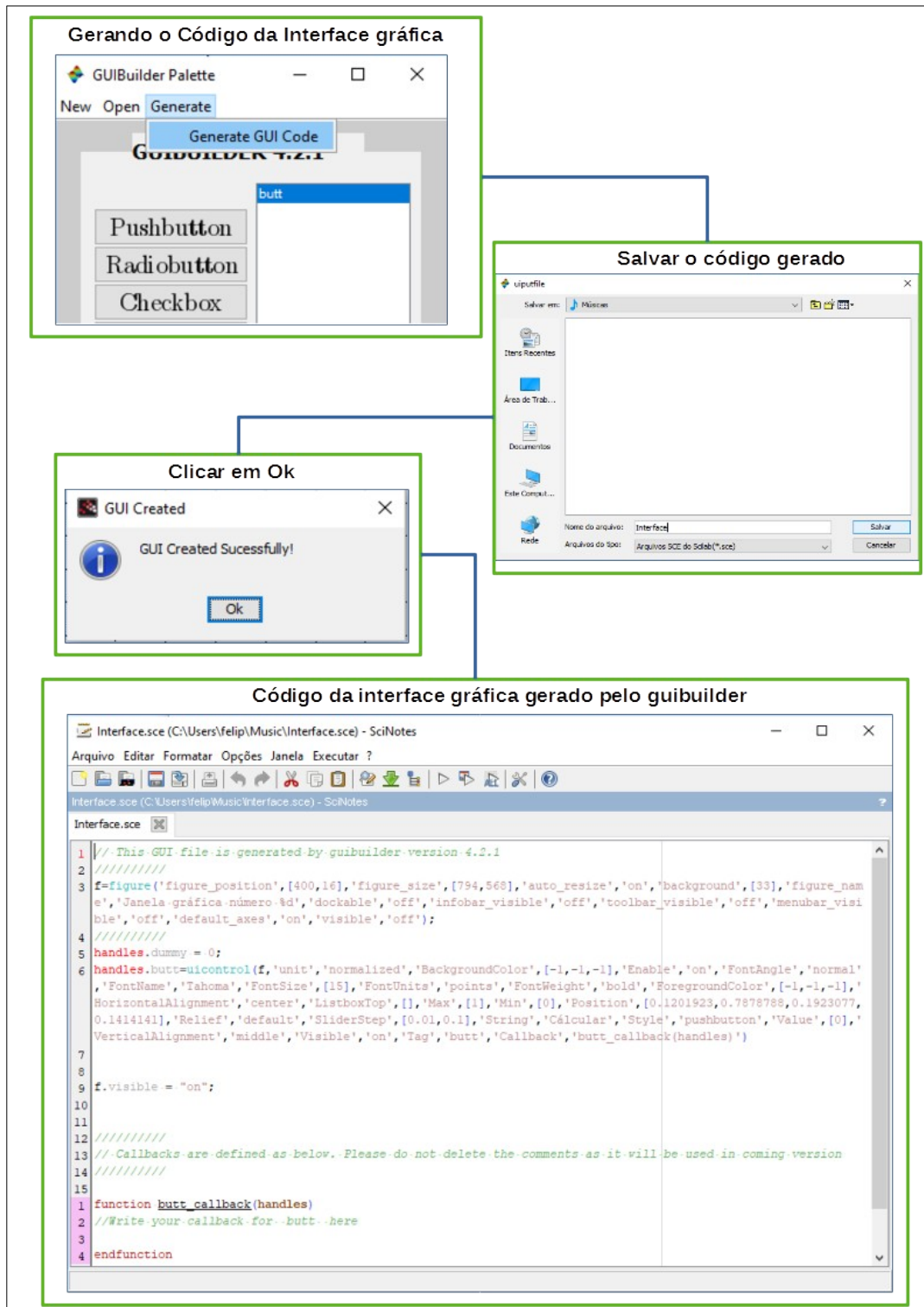


Fonte: O autor, 2021.

E na Figura 9 mostra-se como e feito para gerar o código automaticamente da interface criada dentro do módulo instalado no software scilab.



Figura 9 - Geração do código da interface gráfica desenvolvida no guibuilder



Fonte: O autor, 2021.

Concluída a criação da interface gráfica do levantamento topográfico planialtimétrico e gerado o código, que foi desenvolvido em três partes os cálculos do levantamento:

- ✓ Cálculos de Azimute, Ângulos e Distância;

- ✓ Poligonal Fechada;
- ✓ Nivelamento Geométrico Composto.

Onde estes dados utilizados da literatura, vêm a partir do levantamento de campo empregando alguns dos equipamentos mostrados na Figura 10.

Figura 10 - Equipamentos que podem obter os dados para estes trabalho na topografia



Fonte: O autor, 2021.

### 3.2.1.1 Cálculos de Azimute, Ângulos e Distância

E com as leituras do teodolito, são realizados os cálculos do Azimute de Partida, Ângulo Horizontal, Ângulo Vertical e Distância Horizontal. Sendo que para se fazer os cálculos é necessário inserir os dados de entrada conforme especificado no Tabela 4 abaixo.

Tabela 4 - Especificações do formato dos dados inseridos para calcular

Cálculos de:	Dados de Entrada	Exemplos	
		Fora do Scilab	No Programa Scilab
Azimute	X (m)	100	100
	Y (m)	100	100
Ângulo Horizontal	PD_HR (graus.minseg)	122° 31' 35"	122.3135
	PI_HR (graus.minseg)	123° 35' 38"	123.538
Ângulo Vertical	PD_HR (graus.minseg)	122° 31' 35"	122.3135
	PI_HR (graus.minseg)	123° 35' 38"	123.538
Distância Horizontal	Fio Superior	1,456	1.456
	Fio Inferior	2,345	2.345
	Ângulo Vertical (graus.minseg)	156° 34' 36"	156.3436

Fonte: o autor, (2021).

E para se calcular este programa no software scilab do levantamento topográfico planialtimétrico, serão utilizados todos os dados da literatura, para se realizar a validação dos resultados. Assim para se calcular esta primeira parte com as leituras do teodolito, foram inseridos os seguintes dados com o nome do autor conforme mostrado na Tabela 5.

Tabela 5 - Dados de Entrada utilizados para cálculos com o as leituras do teodolito

Cálculos do:	Dados de Entrada	Veiga, Zanetti, Faggion (2012) Almeida, Freitas e Machado (2005)	
Azimute	X (m)	459.234	233.786
	Y (m)	301.459	502.591
Ângulo Horizontal	PD_HR (graus.minseg)	0	
	PI_HR (graus.minseg)	181.0000	
Ângulo Vertical	PD_VZ (graus.minseg)	88.5654	
	PI_VZ (graus.minseg)	271.0506	
Distância Horizontal	Fio Superior	3.068	
	Fio Inferior	2	
	Ângulo Vertical (graus.minseg)	88.54	

Fonte: o autor, (2021).

### 3.2.1.2 Poligonal Fechada

Nesta segunda parte é feita o cálculo de um dos métodos mais utilizados na topografia para se obter as coordenadas (X, Y), por permitir também calcular os erros lineares e angulares. E para este cálculo é necessário os dados de entrada como mostrado na Tabela 6 abaixo, sendo estes dados de entrada utilizados da literatura, e deveram ser inseridos na interface gráfica do programa criado no devido formato como mostrado.

Tabela 6 - Especificações do formato dos dados iniciais da Poligonal Fechada

Indicadores	Dados Iniciais da Poligonal Fechada	
	Fora do Scilab	No Programa Scilab
Azimute de Partida (graus.minseg)	106° 52' 07"	106.5207
Coordenada X_inicial	224,19	224.19
Coordenada Y_inicial	589,25	589.25
Tolerância Angular (graus.minseg)	00° 00' 10"	00.0010
Tolerância Linear (Escala)	2000	2000
Sentido da Poligonal	Horário	Horário
Ângulos Medidos	Externo	Externo

Fonte: Adaptado de Veiga, Zanetti, Faggion (2012).

Após digitados os dados iniciais da poligonal, e inserido o txt contendo a caderneta com os dados de campo como a estação, vante, tipo, ângulo horizontal, ângulo vertical e distância horizontal, sendo que se não houver os dados do ângulo vertical os dados deveram ser inseridos como (0), conforme mostrado na Figura 11.

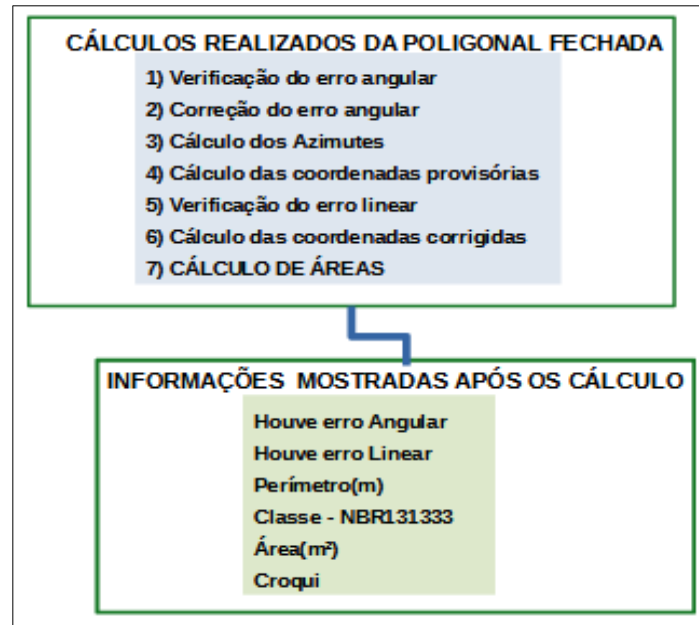
Figura 11 - Modelo do txt da Poligonal para inserir no programa

Arquivo	Editar	Formatar	Exibir	Ajuda		
OPP	1	P	301.2983	0	100.18	
1	2	P	246.4725	0	115.80	
2	3	P	261.2934	0	116.68	
3	4	P	301.4511	0	91.65	
4	OPP	P	148.2831	0	89.06	

Fonte: Adaptado de Veiga, Zanetti, Faggion (2012).

Sendo também realizado a partir do cálculo da poligonal fechada, o cálculo de áreas e também o desenho do croqui da poligonal fechada. E a partir dos dados iniciais e do arquivo txt inseridos do programa, e calculados na poligonal as seguintes seqüências de cálculos como mostrado na Figura 12.

Figura 12 - Cálculos Realizados a partir dos dados de Entrada.



Fonte: o autor, (2021).

### 3.2.1.3 Nivelamento Geométrico

Esta última parte é feita o cálculo do método mais preciso e de grande aplicação na engenharia para determinação das altitudes (Z). Devido a sua grande precisão, e empregando o método de visadas iguais do nivelamento geométrico composto, utilizando o contranivelamento que se permite verificar o erro, como é o caso deste trabalho. Sendo necessário inserir os dados de entrada como mostrado na Tabela 7 abaixo.

Tabela 7 - Formato dos dados iniciais do cálculo do método de visadas iguais

Indicadores	Dados Iniciais da Poligonal Fechada	
	Fora do Scilab	No Programa Scilab
Coordenada Z_inicial (m)	900,000	900.0
Tolerância Altimétrica (mm)	200	200.0

Fonte: Adaptado de Veiga, Zanetti, Faggion (2012).

E após de inseridos os dados iniciais da coordenada e a tolerância altimétrica do método do nivelamento, e inserido o arquivo txt com os dados obtidos em campo com o levantamento, utilizando o nível e a mira estadimétrica. Assim foi feito o arquivo txt com os dados para se fazer o cálculo no software como mostrado na Figura 13 abaixo, com os dados utilizados da literatura.

Figura 13 - Modelo do txt do Nivelamento geométrico composto

Arquivo	Editar	Formatar	Exibir	Ajuda		
RN217	N	0.747	0.555	1.686	1.481	
A1	N	1.057	0.856	1.395	1.197	
A2	N	1.694	1.5	1.535	1.334	
A3	N	1.608	1.408	1.624	1.425	
A4	N	1.835	1.641	1.195	0.995	
A5	N	1.748	1.55	1.415	1.221	
A6	N	1.523	1.331	1.545	1.34	
A7	N	1.756	1.562	1.352	1.158	
A8	N	2.115	1.905	0.585	0.394	
A9	N	3.09	2.866	0.43	0.2	
HV04	C	0.438	0.208	3.097	2.872	
B1	C	0.602	0.41	2.131	1.922	
B2	C	1.345	1.149	1.747	1.554	
B3	C	1.547	1.345	1.528	1.333	
B4	C	1.421	1.226	1.753	1.556	

Estação

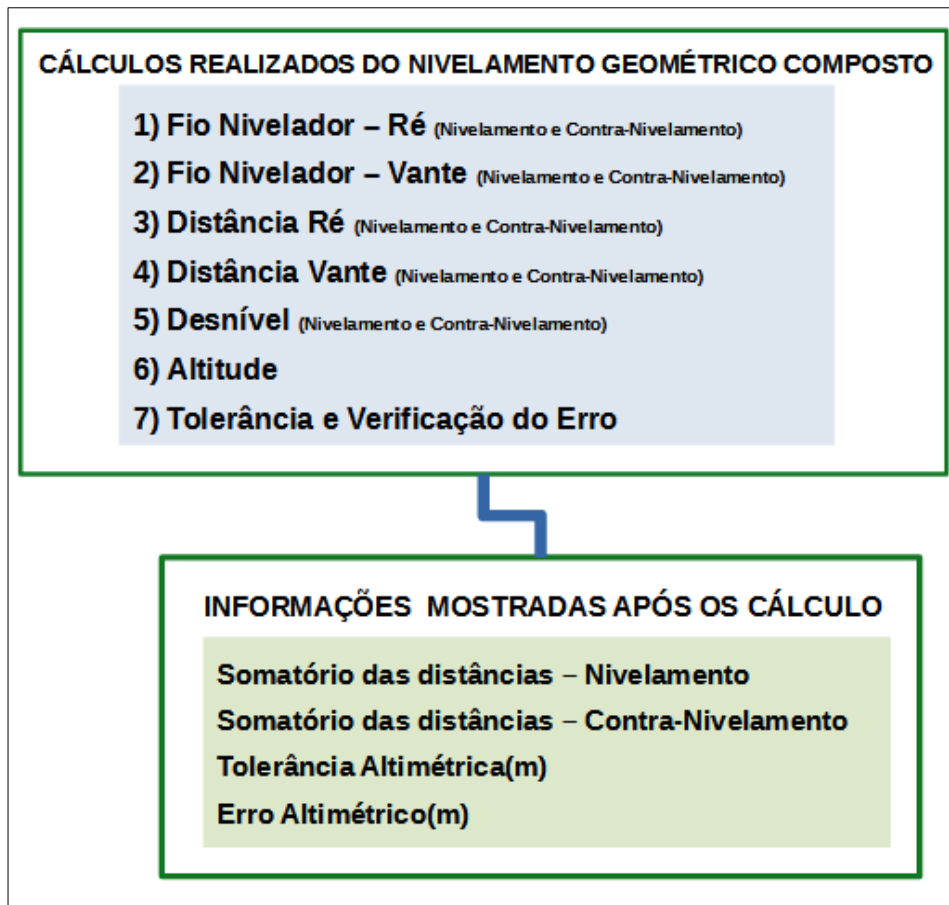
N = Nivelamento  
ou  
C = Contra-Nivelamento

Ré - Fio Superior  
Ré - Fio Inferior  
Vante - Fio Superior  
Vante - Fio Inferior

Fonte: Adaptado de Veiga, Zanetti, Faggion (2012).

Assim com os dados iniciais inseridos na interface gráfica do programa na parte do nivelamento geométrico composto com o método de visadas iguais, e feito o cálculo utilizando a linguagem scilab, onde seguiu os seguintes passos para se calcular a altitudes dos pontos, como mostrado na Figura 14.

Figura 14 - Cálculos realizados do nivelamento a partir dos dados e leituras de entrada



Fonte: Adaptado de Veiga, Zanetti, Faggion (2014).

## 4 RESULTADOS E DISCUSSÃO

### 4.1 Levantamento Topográfico Planialtimétrico

Para o iniciar o programa foi elaborado uma interface gráfica para inserir os dados de entrada conforme explicado anteriormente nos métodos, assim utilizando o módulo do scilab “*guibuilder*” foi criado a interface conforme (Figura 15) abaixo, que também gerou o código da interface para depois calcular o levantamento topográfico planialtimétrico.

Figura 15 - Interface Gráfica do Programa no Scilab

Cálculo do Levantamento Topográfico Planialtimétrico

Sobre

**Cálculos de Azimute, Ângulos e Distância**

**Cálculo do Azimute - Coordenadas Cartesianas**

X(m)      Y (m)

Ponto 1

Ponto 2

Clique para calcular o Azimute

---

Ângulo Horizontal	Ângulo Vertical
PD_HR(graus.minseg) <input type="text"/>	PD_VZ (graus.minseg) <input type="text"/>
PI_HR(graus.minseg) <input type="text"/>	PI_VZ(graus.minseg) <input type="text"/>

Clique para calcular

---

**Distância Horizontal(m)**

Fio Superior(m)

Fio Inferior(m)

Ângulo Vertical(graus.minseg)

Clique para calcular

**Cálculo da Poligonal Fechada**

1º Passo - Digite os dados Iniciais

Azimute de Partida (graus.minseg):

Coordenada X\_inicial:

Coordenada Y\_inicial:

Tolerância Angular(graus.minseg):

Tolerância Linear(Escala):

Sentido da Poligonal:

Ângulos Medidos:

Exportar Resultados em txt:

2º Passo - Clique aqui para Inserir txt poligonal

**Cálculo do Nivelamento Geométrico - Visadas Iguais**

1º Passo - Digite os dados Iniciais

Coordenada Z\_Inicial:

Tolerância Altimétrica(mm):

Exportar Resultados em txt:

2º Passo - Clique aqui para inserir txt nivelamento

Fonte: o autor, (2021).



Assim com a interface gráfica criada, e possível com os dados inseridos mostrar os resultados também na tela e exportar em formato txt. Podendo o usuário calcular este importante levantamento de forma prática. Sendo que todo o código para criação da interface está descrita no apêndice A deste trabalho.

#### 4.1.1 Cálculos de Azimute, Ângulos e Distância

Nesta primeira parte e feita o cálculo de dados vindos do levantamento de campo feito com o teodolito óptico, por exemplo, onde e feita o levantamento de forma de coleta de dados manual. Assim para este programa foram utilizados os dados da literatura para se fazer a validação do programa como mostrado na Figura 16, os dados de entrada com os resultados na interface do programa.

Figura 16 - Dados de entrada e resultados na interface gráfica

**Cálculos de Azimute, Ângulos e Distância**

**Cálculo do Azimute - Coordenadas Cartesianas**

	X(m)	Y (m)
Ponto 1	459.234	233.786
Ponto 2	301.459	502.591

**Clique para calcular o Azimute**

329.58921

**Ângulo Horizontal**

PD_HR(graus.minseg)	0
PI_HR(graus.minseg)	180.0000

**Clique para calcular**

0

**Ângulo Vertical**

PD_VZ (graus.minseg)	88.5654
PI_VZ(graus.minseg)	271.0506

**Clique para calcular**

88.931667

**Distância Horizontal(m)**

Fio Superior(m)	3.068
Fio Inferior(m)	2
Ângulo Vertical(graus.minseg)	88.54

**Clique para calcular**

106.76064

Fonte: O autor, 2021.

E com os resultados obtidos no programa é possível fazer a comparação, devido aos dados de entrada serem tirados da literatura, assim como mostrado na Tabela 8 abaixo e com a comparação dos resultados.

Tabela 8 - Comparação dos resultados do scilab com a literatura

Cálculos	RESULTADOS			
	Fonte dos dados de Entrada	Literatura	Programa Scila	Diferença (Literatura x Programa)
Azimute (graus decimais)	Veiga, Zanetti, Faggion (2014).	329.5891667	329.58921	-0.0000433
Ângulo Horizontal (graus decimais)	Veiga, Zanetti, Faggion (2014).	0	0	0
Ângulo Vertical (graus decimais)	Veiga, Zanetti, Faggion (2014).	88.93166667	88.931667	-0.00000033
Distância Horizontal (m)	Almeida, Freitas e Machado (2005)	106.76	106.76064	-000.00064

Fonte: o autor, (2021).

Segundo a tabela de comparação dos cálculos acima, é possível verificar que as diferenças entre os resultados da literatura e os obtidos no software são satisfatórios, com diferenças como mostrado no cálculo de azimute de (0,0000433), e no cálculo do ângulo vertical com a diferença de (0,00000033) devido ao arredondamento e do caso do cálculo da distância horizontal houve a diferença de (0,00064) devido ao arredondamento, feito pela literatura, ressaltando que o programa scilab utiliza como separador o ponto.

#### 4.1.2 Poligonal Fechada

O início do cálculo da poligonal fechada se dá partir da introdução dos dados de entrada na interface gráfica, com o azimute de partida, coordenadas conhecidas, tolerâncias e o sentido da poligonal como mostrado na Figura 17. Com a caderneta em formato (txt) de dados obtidos em campo, como estação, vante, ângulos e distâncias (Figura 18). Com estes dados é possível calcular as coordenadas finais de cada ponto da poligonal fechada, e fazendo a comparação dos resultados obtidos no software com o da literatura, devido os dados inseridos neste trabalho serem tirados da literatura. Sendo assim é possível se fazer a validação dos resultados, com a utilização de métodos já comprovados e amplamente utilizados na engenharia e na topografia.

Figura 17 - Dados de entrada da poligonal fechada na interface gráfica.

**Cálculo da Poligonal Fechada**

**1º Passo - Digite os dados Iniciais**

Azimute de Partida (graus.minseg): 106.5207

Coordenada X\_inicial: 224.19

Coordenada Y\_inicial: 589.25

Tolerância Angular (graus.minseg): 00.0010

Tolerância Linear (Escala): 2000

Sentido da Poligonal: Horário

Ângulos Medidos: Externo

Exportar Resultados em txt: Não

**2º Passo - Clique aqui para Inserir txt poligonal**

Fonte: o autor, (2021).

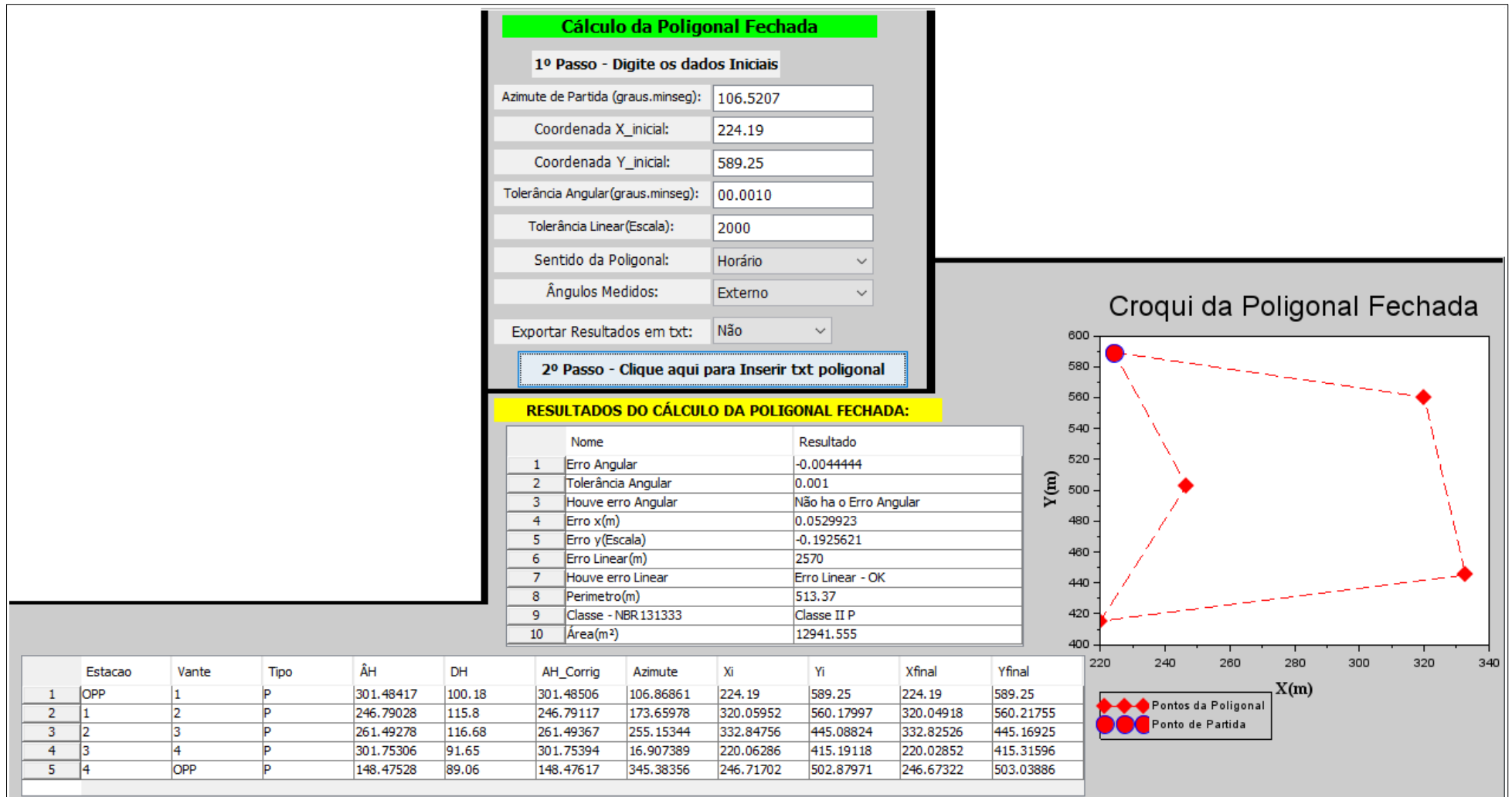
Figura 18 - Dados de entrada do arquivo(txt) da poligonal

entrada - poligonal - Bloco de Notas						
Arquivo	Editar	Formatar	Exibir	Ajuda		
OPP	1	P	301.2903	0	100.18	
1	2	P	246.4725	0	115.80	
2	3	P	261.2934	0	116.68	
3	4	P	301.4511	0	91.65	
4	OPP	P	148.2831	0	89.06	

Fonte: Adaptado de Veiga, Zanetti, Faggion (2012).

Assim inseridos os dados de entrada no programa scilab, foi feito todo o cálculo da poligonal fechada, seguindo o passo a passo dos métodos, como descritos nas equações acima das referências, onde foram calculados os ângulos corrigidos, azimutes, coordenadas, os erros angulares e lineares uma das grandes vantagens deste método, também com o croqui da poligonal fechada, e este resultado apresentado na própria tela do programa como representado na Figura 19.

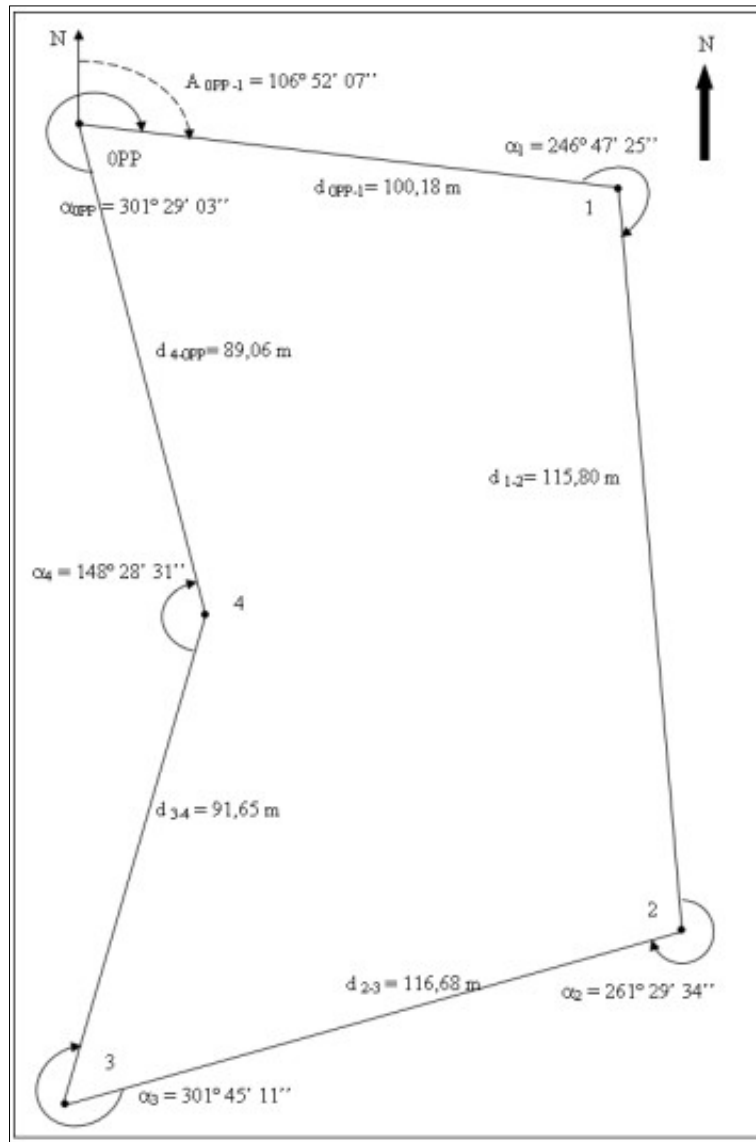
Figura 19 - Resultados do cálculo da poligonal na interface do programa



Fonte: o autor(2021).

A poligonal com os dados utilizados da literatura possui 5 pontos, sendo o ponto que possui as coordenadas e azimuth conhecido é o ponto (OPP) o ponto de partida, onde se inicia e fecha a poligonal. Seguindo o sentido horário do caminhar com ângulos medidos externamente como indicada na Figura 20.

Figura 20 - Distribuição dos Pontos com os dados utilizados



Fonte: Veiga, Zanetti, Faggion (2012).

E com os resultados gerados é possível fazer a validação dos dados obtidos dos cálculos no programa com o da literatura. Como a área calculada no programa de 12.941,555 m<sup>2</sup> e 513,37 metros de perímetro, e outros resultados como ângulos, azimutes, coordenadas e os erros como descrito na Figura 21.

Figura 21 - Comparação dos resultados da poligonal do programa com a literatura

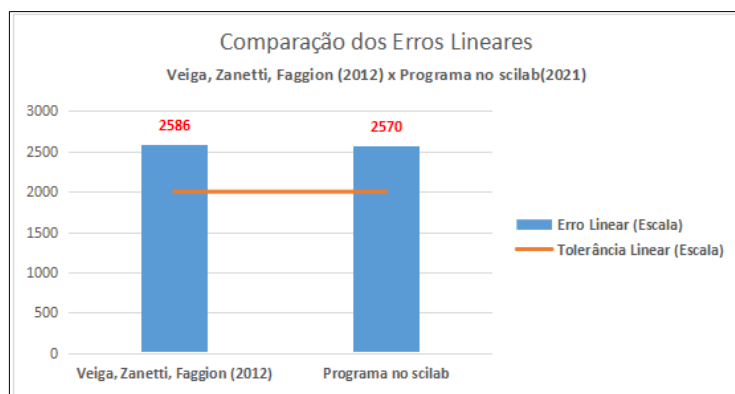
RESULTADOS	PONTOS	Ângulos horizontais Corrigidos	Azimutes	Coordenada Provisorias (x) (metros)	Coordenada Provisorias (y) (metros)	Coordenada Corrigidas (X) (metros)	Coordenada Corrigidas (Y) (metros)	Erro Angular	Erro Linear	Perimetro	Área (m <sup>2</sup> )
Veiga, Zanetti, Faggion (2012)	OPP	301° 29' 07"	106° 52' 07"	224,190	589,250	224,19	589,25	- 0° 0' 16"	0,19848306	513,37	12941,24
	1	246° 47' 28"	173° 39' 35"	320,060	560,180	320,05	560,22				
	2	261° 29' 37"	255° 09' 12"	332,848	445,090	332,82	445,17				
	3	301° 45' 14"	16° 54' 26"	220,067	415,193	220,03	415,32				
	4	148° 28' 34"	345° 23' 00"	246,721	502,882	246,67	503,04				
OPP	301° 29' 07"	106° 52' 07"	224,247	589,060	224,19	589,25					
Programa no scilab	OPP	301°29' 6.22"	106° 52' 7"	224,19	589,25	224,19000	589,25	- 0° 0' 16"	0,1997207	513,37	12941,555
	1	246° 47' 28.2"	173° 39' 35.2"	320,05952	560,17997	320,04918	560,21755				
	2	261° 29' 37.2"	255° 9' 12.38"	332,84756	445,08824	332,82526	445,16925				
	3	301° 45' 14.1"	16° 54' 26.6"	220,06286	415,19118	220,02852	415,31596				
	4	148° 28' 34.2"	345° 23' 0.82"	246,71702	502,87971	246,67322	503,03886				
OPP	301°29' 6.22"	106° 52' 7"	224,24299	589,05744	224,19000	589,25					
Diferença	OPP	0° 0' 0"	0° 0' 0"	0	0	0	0	0	0,00123764	0	0,315
	1	0° 0' 0.2"	0° 0' 0.2"	0,00048	0,00003	0,00082	0,00245				
	2	0° 0' 0.2"	0° 0' 0.38"	0,00044	0,00176	0,00526	0,00075				
	3	0° 0' 0.1"	0° 0' 0.6"	0,00414	0,00182	0,00148	0,00404				
	4	0° 0' 0.78"	0° 0' 0.82"	0,00398	0,00229	0,00322	0,00114				
OPP	0°0' 0"	0°0' 0"	0,00401	0,00256	0	0					

Fonte: O autor(2021).

Conforme apresentado na figura Figura 21 acima, houve diferenças na comparação dos resultados gerados pelo programa scilab com a literatura nos ângulos horizontais corrigidos e azimutes, nas casas dos segundos decimais, devido a maior casas decimais do programa scilab sem fazer o arredondamento, onde a maior diferença foi do azimute do ponto 4 de ( $0^{\circ}0' 0.82''$ ), e com um erro angular igual ao gerado da literatura.

E como representado no Gráfico 1 abaixo, é possível analisar que a utilização dos métodos e os dados da literatura neste programa desenvolvido em scilab, houve diferença no erro linear com a literatura, mais dentro tolerância estabelecida na literatura. Onde o erro planimétrico em escala tem que ser maior que a tolerância linear.

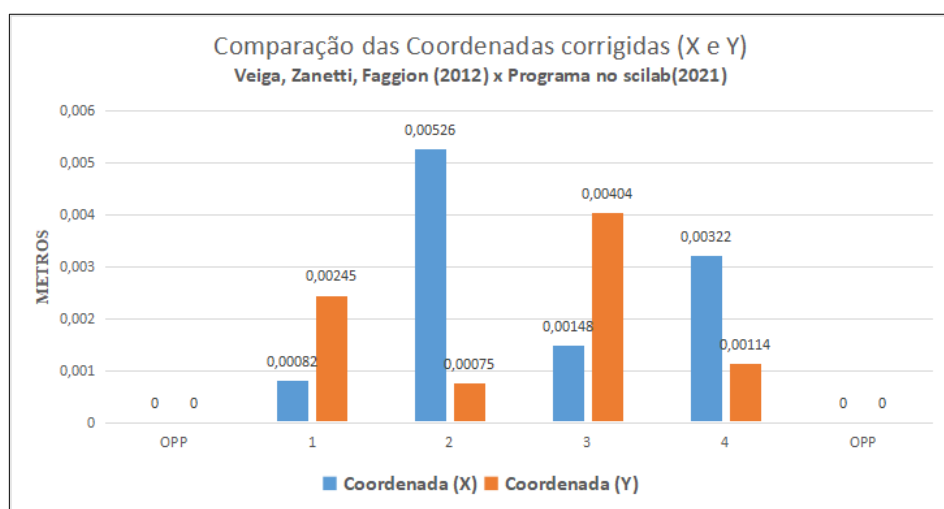
Gráfico 1 - Comparação do erro linear



Fonte: O autor(2021).

Como os erros angulares e lineares foram dentro das tolerâncias estabelecidas na literatura e as normas técnicas da NBR13133, calcula-se então as correções e as coordenadas finais, que neste trabalho apresentou uma diferença com a literatura como a maior do ponto 2 de (0,00526 metro). Como se pode observar no Gráfico 2.

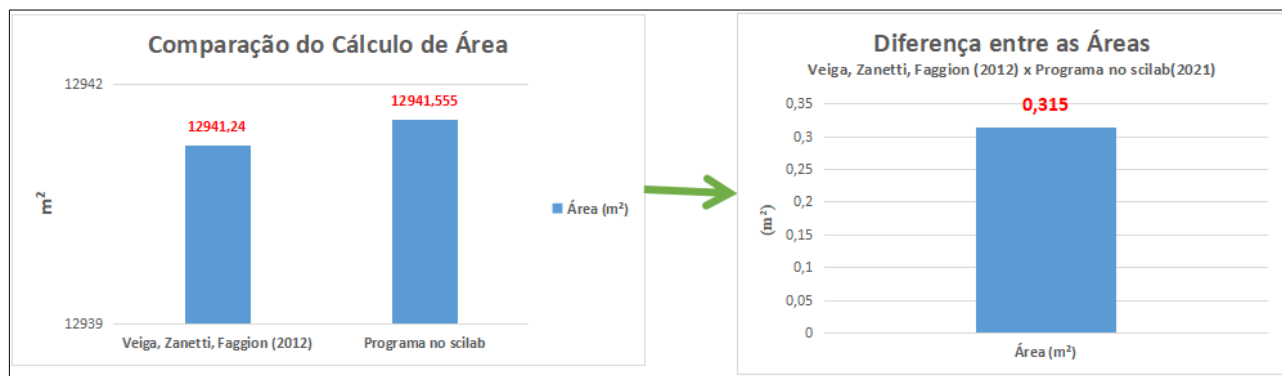
Gráfico 2 - Comparação das Coordenadas Corrigidas



Fonte: O autor(2021).

Assim, com a poligonal fechada, calculada com as coordenadas corrigidas finais, é possível calcular a área da poligonal e fazer a comparação como mostrado no Gráfico 3.

Gráfico 3 - Comparação do cálculo de áreas



Fonte: O autor(2021).

Sendo de fundamental importância evitar erros grosseiros, como os erros causados na introdução dos dados para se calcular a poligonal fechada, que modificam e interfere, nos resultados deste método topográfico.

#### 4.1.3 Nivelamento Geométrico

No cálculo do nivelamento geométrico composto foi empregado o método de visadas iguais para calcular as altitudes e o contranivelamento para calcular o erro, então foi elaborado a interface gráfica para se introduzir os dados iniciais como mostrado na Figura 22, com o arquivo txt, com os formatos dos dados dos 36 pontos do nivelamento e contranivelamento Figura 23.

Figura 22 - Dados Iniciais do Nivelamento

A interface gráfica tem um título em uma barra verde: 'Cálculo do Nivelamento Geométrico - Visadas Iguais'. Abaixo, há um subtítulo: '1º Passo - Digite os dados Iniciais'. O formulário contém três campos de entrada: 'Coordenada Z\_Inicial:' com o valor 100, 'Tolerância Altimétrica(mm):' com o valor 12, e 'Exportar Resultados em txt:' com o valor Sim e uma seta para baixo. Abaixo dos campos, há um botão azul com o texto: '2º Passo - Clique aqui para inserir txt nivelamento'.

Fonte: o autor (2021).



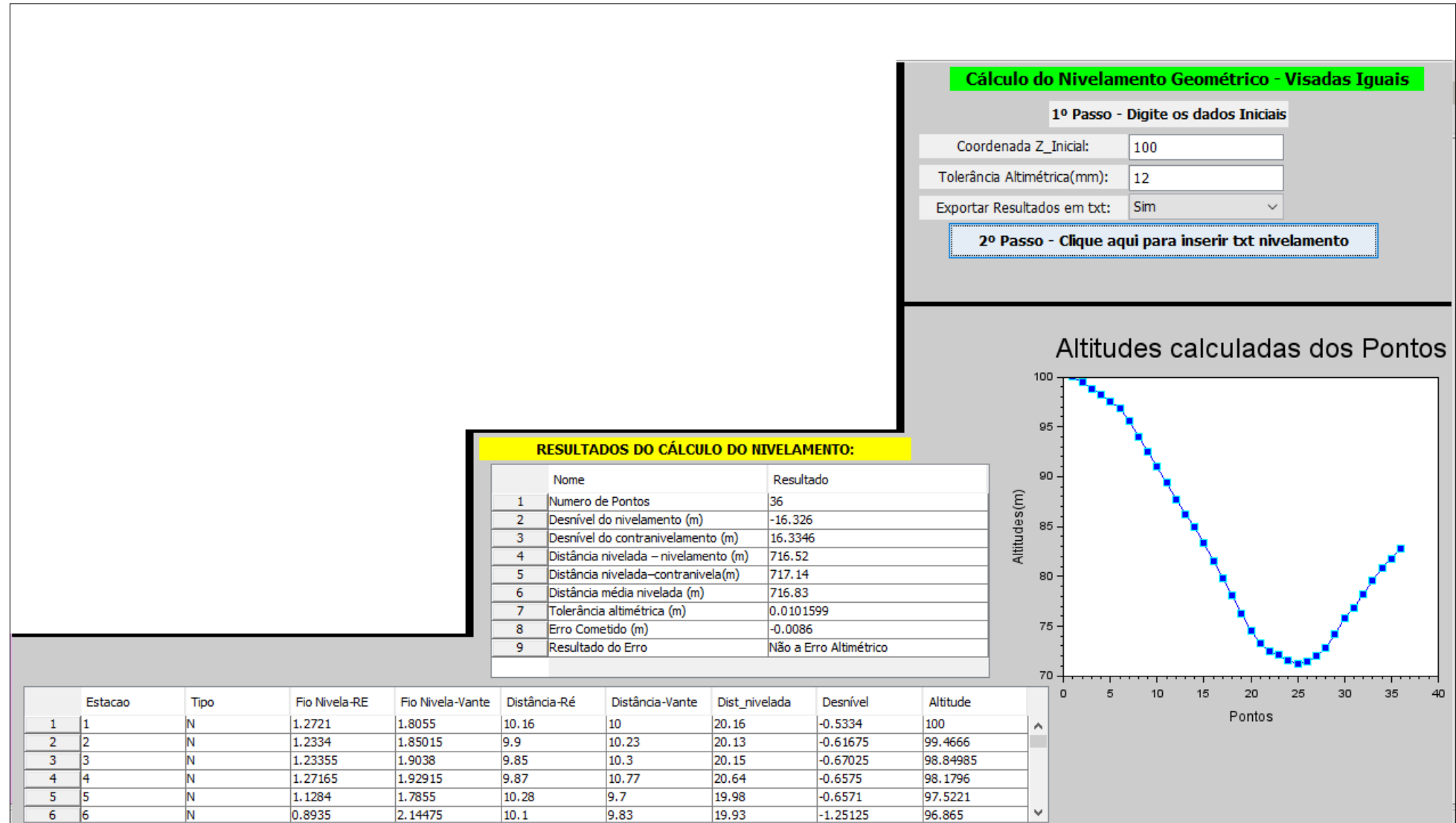
Figura 23 - Dados de Campo do Nivelamento Geométrico composto.

dados entrada - nivelamento - Bloco de Notas					
Arquivo	Editar	Formatar	Exibir	Ajuda	
1	N	1.3229	1.2213	1.8555	1.7555
2	N	1.2829	1.1839	1.9013	1.799
3	N	1.2828	1.1843	1.9553	1.8523
4	N	1.321	1.2223	1.983	1.8753
5	N	1.1798	1.077	1.834	1.737
6	N	0.944	0.843	2.1939	2.0956
7	N	0.7236	0.6259	2.322	2.2183
8	N	0.8239	0.7246	2.3536	2.2536
9	N	0.9099	0.8036	2.428	2.3326
10	N	0.702	0.602	2.28	2.18
11	N	0.6289	0.53	2.272	2.1673
12	N	0.7926	0.696	2.281	2.176
13	N	0.8398	0.7398	2.1906	2.0903
14	N	0.597	0.502	2.21	2.119
15	N	0.849	0.753	2.574	2.4699
16	N	0.6956	0.5944	2.458	2.3586
17	N	0.7655	0.667	2.474	2.3736
18	N	0.7478	0.649	2.552	2.45
19	N	0.6593	0.5593	2.412	2.3123
20	N	0.7699	0.6686	2.0206	1.9209
21	N	1.0086	0.9053	1.84	1.7436
22	N	1.4113	1.3089	1.7856	1.6866
23	N	1.2413	1.1573	1.7888	1.702
24	N	1.2503	1.1512	1.5403	1.4406
25	N	1.5478	1.4458	1.366	1.269
26	N	1.8986	1.7963	1.348	1.2499
27	N	2.011	1.9029	1.1803	1.0806
28	N	2.1845	2.0838	0.7773	0.6793
29	N	2.4093	2.3089	0.8419	0.741
30	N	2.071	1.971	1.031	0.922
31	N	2.3039	2.2039	0.9385	0.8436
32	N	2.427	2.3253	1.0906	0.9899
33	N	2.3426	2.2386	0.9786	0.8838
34	N	2.0158	1.9158	1.1456	1.0459
35	N	2.147	2.0476	1.088	0.9875
36	N	2.098	2.013	1.2399	1.1472
1	C	1.356	1.262	2.215	2.1299
2	C	1.0942	0.994	2.1526	2.0536

Fonte: Adaptado de Roza (2017).

Na Figura 24 abaixo é mostrado o resultado do cálculo do nivelamento geométrico na própria interface do programa criado, de forma bem detalhada.

Figura 24 - Resultados do cálculo do nivelamento na interface gráfica



Fonte: o autor (2021).

No cálculo do nivelamento foi utilizado da literatura 36 pontos no nivelamento e contranivelamento, com dados iniciais de uma cota arbitrária de 100 metros e uma tolerância altimétrica estabelecida pela literatura de 12 milímetros. Sendo feito o cálculo que obteve um desnível somado do nivelamento de (-16,326 m) e um desnível do contranivelamento de (16,3346 m), com descrito na Quadro 1 com a comparação dos erros e tolerâncias.

Quadro 1 - Resultados de Erros, Tolerância e Desnível do Nivelamento

<b>Cálculos</b>	<b>Roza (2017)</b>	<b>Este trabalho</b>	<b>Diferenças</b>
<b>Desnível do nivelamento (metros)</b>	-16,326	-16,326	0
<b>Desnível do contranivelamento (metros)</b>	16,3346	16,3346	0
<b>Distância nivelada do nivelamento (metros)</b>	716,25	716,52	0,27000
<b>Distância nivelada do contranivelamento (metros)</b>	717,14	717,14	0
<b>Distância média nivelada (metros)</b>	716,695	716,83	0,13500
<b>Erro cometido (metros)</b>	0,0086	0,0086	0
<b>Tolerância Altimétrica (metros)</b>	0,010158	0,0101599	0,0000019

Fonte: o autor (2021).

Como observado o erro calculado neste programa desenvolvido foi dentro da tolerância estabelecida pela literatura, com diferenças que não afetaram a qualidade da precisão do nivelamento analisado. O cálculo aqui descrito alcançou resultados satisfatórios com a diferenças em cada pontos dos desníveis, por exemplo a partir da 5 casas decimais. Como demonstrado na Figura 25 abaixo.

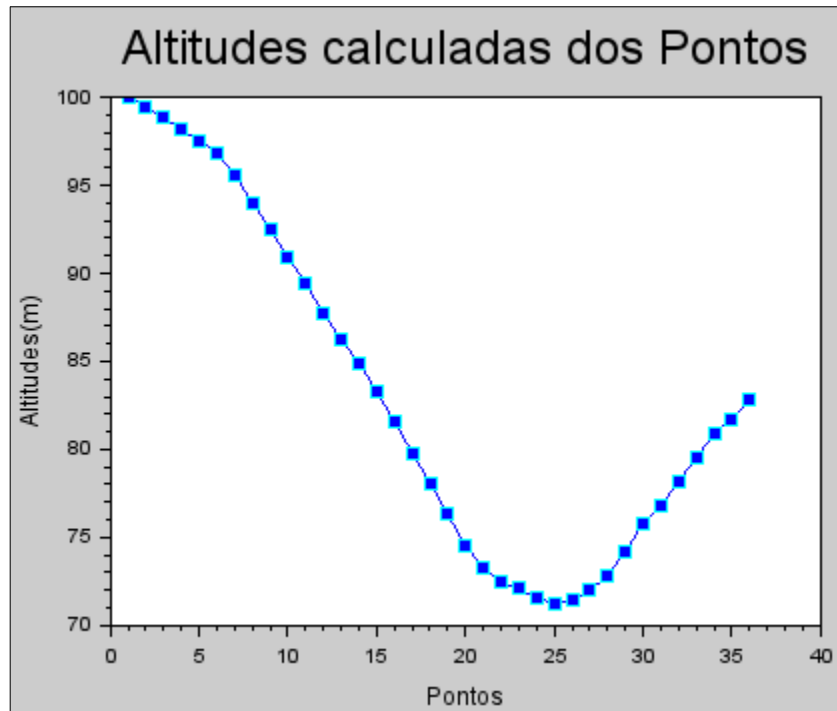
Figura 25 - Comparação das distâncias e desníveis do nivelamento

NIVELAMENTO						
ESTAÇÃO	Distância Nivelada (metros)			Desníveis (metros)		
	Roza (2017)	Este Trabalho	Diferença	Roza (2017)	Este Trabalho	Diferença
1	20,16	20,16	0	-0,5334	-0,5334	0
2	20,13	20,13	0	-0,6168	-0,61675	0,00005
3	20,15	20,15	0	-0,6703	-0,67025	0,00005
4	20,64	20,64	0	-0,6575	-0,6575	0
5	19,98	19,98	0	-0,6571	-0,6571	0
6	19,93	19,93	0	-1,2513	-1,25125	0,00005
7	20,14	20,14	0	-1,5954	-1,5954	0
8	19,93	19,93	0	-1,5294	-1,52935	0,00005
9	20,17	20,17	0	-1,5236	-1,52355	0,00005
10	20	20	0	-1,578	-1,578	0
11	20,09	20,36	0,27	-1,6402	-1,6402	0
12	20,16	20,16	0	-1,4842	-1,4842	0
13	20,03	20,03	0	-1,3507	-1,35065	0,00005
14	18,6	18,6	0	-1,615	-1,615	0
15	20,01	20,01	0	-1,721	-1,72095	0,00005
16	20,06	20,06	0	-1,7633	-1,7633	0
17	19,89	19,89	0	-1,7076	-1,70755	0,00005
18	20,08	20,08	0	-1,8026	-1,8026	0
19	19,97	19,97	0	-1,7529	-1,75285	0,00005
20	20,1	20,1	0	-1,2515	-1,2515	0
21	19,97	19,97	0	-0,8349	-0,83485	0,00005
22	20,14	20,14	0	-0,376	-0,376	0
23	17,08	17,08	0	-0,5461	-0,5461	0
24	19,88	19,88	0	-0,2897	-0,2897	0
25	19,9	19,9	0	0,1793	0,1793	0
26	20,04	20,04	0	0,5485	0,5485	0
27	20,78	20,78	0	0,8265	0,8265	0
28	19,87	19,87	0	1,4059	1,40585	0,00005
29	20,13	20,13	0	1,5677	1,56765	0,00005
30	20,9	20,9	0	1,0445	1,0445	0
31	19,49	19,49	0	1,3629	1,36285	0,00005
32	20,24	20,24	0	1,3359	1,3359	0
33	19,88	19,88	0	1,3594	1,3594	0
34	19,97	19,97	0	0,8701	0,87005	0,00005
35	19,99	19,99	0	1,0596	1,05955	0,00005
36	17,77	17,77	0	0,862	0,86195	0,00005

Fonte: o autor (2021).

Assim com os desníveis calculados conforme os métodos da literatura e respeitando com rigor as normas com as suas tolerâncias, foi feito no próprio scilab na interface do programa, o croqui com as altitudes de cada ponto deste nivelamento como representado na Figura 26.

Figura 26 - Croqui das altitudes dos pontos do nivelamento



Fonte: o autor (2021).

Segundo Veiga, Zanetti, Faggion (2012), as medições dos desníveis é um dos principais objetivos do levantamento topográfico, com a coleta de dados em campo de modo a representar uma determinada superfície terrestre. Considerando a precisão do método utilizado para se obter estes desníveis, o que neste trabalho foi empregado o nivelamento geométrico composto com o método de visadas iguais, onde o desnível entre os pontos será determinado a partir de vários lances.

## 5 CONCLUSÃO

Este trabalho foi realizado utilizando as equações dos métodos de alta precisão, comprovados e de grande aplicação na engenharia como a poligonal fechada e o nivelamento geométrico composto com o método de visadas iguais, com dados utilizados nos cálculos vindos da literatura. Assim se pode gerar um produto final muito satisfatório dentro de um software livre e muito utilizado como o scilab.

Na parte de todos os cálculos houve algumas diferenças envolvendo a parte dos arredondamentos, como as diferenças nas casas dos segundos decimais dos ângulos e azimutes, e as diferenças a partir de milímetros nas coordenadas e nos erros que interfere mesmo de forma pequena mais diferencia os resultados, mais mesmo assim em todos os cálculos como, por exemplo, da poligonal fechada e nivelamento geométrico, ficou na tolerância estabelecida pela literatura utilizada.

E este trabalho tem uma contribuição no desenvolvimento de um programa no cálculo de um levantamento topográfico utilizando os métodos de poligonal fechada para obter as coordenadas (X, Y), e o cálculo dos desníveis e altitudes (Z) com o método de visadas iguais com o nivelamento geométrico composto. Que são amplamente utilizados na topografia, e que com este programa é possível calcular com estes métodos, inúmeros pontos de campo com acesso total ao programa criado no scilab, e disponibilizado nos apêndices deste trabalho.

Pode-se concluir que o programa desenvolvido em linguagem scilab, atendeu a todos os objetivos pretendidos, deste a criação da interface gráfica, aos cálculos e também os resultados. Com a possibilidade de estudos futuros sobre o código criado neste trabalho.

E são sugeridos para trabalhos futuros a partir deste programa, a aplicação do ajustamento de observações na poligonal e do nivelamento.

## REFERÊNCIAS

ALMEIDA, Ariclo Pulinho Pires de, FREITAS, José Carlos de Paula, e MACHADO, Maria Márcia Magela. **Topografia - 1 - fundamentos, teoria e prática**. Instituto de geociências da universidade federal de minas gerais. 2005. 173 p. Apostila. Disponível em: <<https://www.csr.ufmg.br/geoprocessamento/publicacoes/Apostila%20Top1.pdf>>. Acesso em: 06 out. 2021.

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. **NBR 13133**: Execução de levantamento topográfico. 1 ed. Rio de Janeiro - RJ, 1994. 35 p. Disponível em: <<http://www.carto.eng.uerj.br/cdecart/download/NBR13133.pdf>>. Acesso em: 07 ago. 2021.

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. **NBR 14166**: Rede de Referência Cadastral Municipal - Procedimento . 1 ed. Rio de Janeiro - RJ, 1998. 23 p. Disponível em: <<http://www.carto.eng.uerj.br/cdecart/download/NBR14166.pdf>>. Acesso em: 08 ago. 2021.

CAMPOS, Frederico Ferreira. **Fundamentos de Scilab**. Belo Horizonte: Universidade Federal de Minas Gerais, 2010. 170 f. Apostila. Disponível em: <[https://www.ime.unicamp.br/~encpos/VIII\\_EnCPos/Apostila\\_Scilab.pdf](https://www.ime.unicamp.br/~encpos/VIII_EnCPos/Apostila_Scilab.pdf)>. Acesso em: 12 set. 2021.

COMASTRI, J.A.; TULER, J.C. **Topografia; Altimetria**, 3ed, Viçosa: UFV, 1999. 200p.

CORDINI, Jucilei. **Altimetria: teoria e métodos visando a representação do relevo**. UNIVERSIDADE FEDERAL DE SANTA CATARINA, 2014. 103 p. Apostila. Disponível em: <<https://topografia.paginas.ufsc.br/files/2015/09/Altimetria-Apostila.pdf>>. Acesso em: 06 ago. 2021.

KON, Fabio. **O Software Aberto e a Questão Social. Relatório Técnico**. 2001, USP. 10 p. Disponível em: <<https://www.ime.usp.br/~kon/papers/open-software.pdf>>. Acesso em: 25 out. 2021.

PEREIRA, Adriana Soares; VISSOTTO, Elisa Maria; FRANCISCATTO, Roberto. **Sistemas Operacionais**. Frederico Westphalen - RS: 2015. 110 p. Disponível em: <[http://roberto.cfw.ufsm.br/images/uploads/sistemas\\_operacionais.pdf](http://roberto.cfw.ufsm.br/images/uploads/sistemas_operacionais.pdf)>. Acesso em: 20 out. 2021.

RODRIGUES, Dalto Domingos. **TOPOGRAFIA: planimetria para engenheiros Agrimensores e Cartógrafos**. Viçosa: 2008. Apostila. Disponível em: <[https://www.eam.ufv.br/?page\\_id=384](https://www.eam.ufv.br/?page_id=384)>. Acesso em: 11 out. 2021.

ROZA, Fábio Muniz. **AVALIAÇÃO DO DESEMPENHO DO NIVELAMENTO GNSS NA ZONA URBANA DE MONTE CARMELO/MG**. 2017. 65 f. TCC (Graduação) - Curso de Engenharia de Agrimensura e Cartográfica, Instituto de Geografia, Universidade Federal de Uberlândia, Monte Carmelo, 2017. Disponível em:

<<https://repositorio.ufu.br/bitstream/123456789/20961/3/AvaliacaoDesempenhoNivelamento.pdf>>. Acesso em: 15 set. 2021.

SANTOS, Afonso de Paula dos. **EAM480 - Programação aplicada a Eng<sup>a</sup> de Agrimensura e Cartográfica**. UFV, 2016. 126 p. Notas de Aula. Disponível em: <[https://www.eam.ufv.br/wp-content/uploads/2016/08/EAM480\\_Notas\\_Aulas\\_2016.pdf](https://www.eam.ufv.br/wp-content/uploads/2016/08/EAM480_Notas_Aulas_2016.pdf)>. Acesso em: 12 ago. 2021.

SCILAB. **Using ATOMS**. 2018. Disponível em: <<https://wiki.scilab.org/Modules>. Acesso em: 26 nov. 2021>.

SOMMERVILLE, Ian. **Software Engineering**. 9. ed. São Paulo: Pearson Prentice Hall, 2011. 544 p. Kalinka Oliveira; Ivan Bosnic. Disponível em: <<http://www.facom.ufu.br/~william/Disciplinas%202018-2/BSI-GSI030-EngenhariaSoftware/Livro/engenhariaSoftwareSommerville.pdf>>. Acesso em: 12 nov. 2021.

VEIGA, Luis Augusto Koenig; ZANETTI, Maria Aparecida Zehnpfennig; FAGGION, Pedro Luis. **FUNDAMENTOS DE TOPOGRAFIA**. Universidade Federal do Paraná, 2012. 288 p. Apostila. Disponível em: <[http://www.cartografica.ufpr.br/docs/topo2/apos\\_topo.pdf](http://www.cartografica.ufpr.br/docs/topo2/apos_topo.pdf)>. Acesso em: 06 ago. 2021.

VEIGA, Luis Augusto Koenig; ZANETTI, Maria Aparecida Zehnpfennig; FAGGION, Pedro Luis. **FUNDAMENTOS DE TOPOGRAFIA**. Universidade Federal do Paraná, 2014. 288 p. Apostila. Disponível em: <<http://www.cartografica.ufpr.br/portal/wp-content/uploads/2015/08/Material-de-Apoio-Topo-II-2015.pdf>>. Acesso em: 07 ago. 2021.



## ANEXOS A – DADOS DE ENTRADA

<b>POLIGONAL FECHADA</b>					
<b>Fonte: Veiga, Zanetti, Faggion (2012).</b>					
<b>ESTAÇÃO</b>	<b>VANTE</b>	<b>TIPO</b>	<b>ÂNGULO HORIZONTAL</b>	<b>ÂNGULO VERTICAL</b>	<b>DISTÂNCIA HORIZONTAL</b>
OPP	1	P	301.2903	0	100.18
1	2	P	246.4725	0	115.80
2	3	P	261.2934	0	116.68
3	4	P	301.4511	0	91.65
4	OPP	P	148.2831	0	89.06

<b>NIVELAMENTO GEOMÉTRICO COMPOSTO – MÉTODO DE VISADAS IGUAIS</b>					
<b>Fonte: Roza (2017)</b>					
<b>ESTAÇÃO</b>	<b>TIPO</b>	<b>RE – Fio Superior</b>	<b>RE – Fio Inferior</b>	<b>Vante – Fio Superior</b>	<b>Vante – Fio Inferior</b>
1	N	1.3229	1.2213	1.8555	1.7555
2	N	1.2829	1.1839	1.9013	1.799
3	N	1.2828	1.1843	1.9553	1.8523
4	N	1.321	1.2223	1.983	1.8753
5	N	1.1798	1.077	1.834	1.737
6	N	0.944	0.843	2.1939	2.0956
7	N	0.7236	0.6259	2.322	2.2183
8	N	0.8239	0.7246	2.3536	2.2536
9	N	0.9099	0.8036	2.428	2.3326
10	N	0.702	0.602	2.28	2.18
11	N	0.6289	0.53	2.272	2.1673
12	N	0.7926	0.696	2.281	2.176
13	N	0.8398	0.7398	2.1906	2.0903
14	N	0.597	0.502	2.21	2.119
15	N	0.849	0.753	2.574	2.4699
16	N	0.6956	0.5944	2.458	2.3586
17	N	0.7655	0.667	2.474	2.3736
18	N	0.7478	0.649	2.552	2.45
19	N	0.6593	0.5593	2.412	2.3123
20	N	0.7699	0.6686	2.0206	1.9209
21	N	1.0086	0.9053	1.84	1.7436
22	N	1.4113	1.3089	1.7856	1.6866
23	N	1.2413	1.1573	1.7888	1.702
24	N	1.2503	1.1512	1.5403	1.4406
25	N	1.5478	1.4458	1.366	1.269
26	N	1.8986	1.7963	1.348	1.2499
27	N	2.011	1.9029	1.1803	1.0806
28	N	2.1845	2.0838	0.7773	0.6793

29	N	2.4093	2.3089	0.8419	0.741
30	N	2.071	1.971	1.031	0.922
31	N	2.3039	2.2039	0.9385	0.8436
32	N	2.427	2.3253	1.0906	0.9899
33	N	2.3426	2.2386	0.9786	0.8838
34	N	2.0158	1.9158	1.1456	1.0459
35	N	2.147	2.0476	1.088	0.9875
36	N	2.098	2.013	1.2399	1.1472
1	C	1.356	1.262	2.215	2.1299
2	C	1.0942	0.994	2.1526	2.0536
3	C	1.017	0.919	1.889	1.7876
4	C	0.9796	0.8813	2.3373	2.236
5	C	1.0438	0.937	2.3746	2.278
6	C	0.9476	0.849	2.3076	2.212
7	C	1.0699	0.9676	2.112	2.0135
8	C	0.8638	0.763	2.4306	2.3306
9	C	0.8262	0.7289	2.2346	2.1346
10	C	1.0146	0.9163	1.8446	1.742
11	C	1.646	1.541	2.19	2.093
12	C	1.436	1.3363	1.6163	1.5176
13	C	1.608	1.5086	1.3263	1.225
14	C	1.935	1.8416	1.3848	1.299
15	C	1.9426	1.8426	1.564	1.4638
16	C	2.0099	1.9099	1.1716	1.0715
17	C	2.075	1.9796	0.826	0.7166
18	C	2.5499	2.4489	0.7976	0.6963
19	C	2.473	2.3726	0.672	0.5716
20	C	2.4183	2.3175	0.702	0.6036
21	C	2.3815	2.28	0.6169	0.513
22	C	2.4676	2.3626	0.7436	0.6445
23	C	2.3766	2.28	0.7556	0.666
24	C	2.234	2.135	0.8869	0.784
25	C	2.299	2.1976	0.8139	0.7173
26	C	2.353	2.2516	0.711	0.612
27	C	2.265	2.168	0.6939	0.591
28	C	2.3578	2.258	0.838	0.7338
29	C	2.453	2.35	0.9188	0.8176
30	C	2.166	2.0673	0.5763	0.4745
31	C	2.178	2.076	0.9226	0.8236
32	C	1.818	1.718	1.162	1.063
33	C	1.8315	1.73	1.1745	1.074
34	C	1.9799	1.8763	1.308	1.207
35	C	1.8446	1.7443	1.2285	1.1279
36	C	1.7789	1.68	1.2435	1.1455

## APÊNDICE A – CÓDIGO DESTE TRABALHO NO SCILAB

```
clear; //Limpar a memória
clc; //Limpar a tela do console

//Criação da Interface Gráfica com o criador de interface "guibuilder"
//Codigo gerado da interface automaticamente no guibuilder version 4.2.1
// This GUI file is generated by guibuilder version 4.2.1
//////////

f=figure('figure_position',[0,0],'figure_size',
[1280,750],'auto_resize','on','background',[33],'figure_name','Cálculo do
Levantamento Topográfico
Planialtimétrico','dockable','off','info_bar_visible','off','toolbar_visible
','off','menubar_visible','on','default_axes','on','visible','off');
//////////

//Remoção do Menu da Janela Grafica
delmenu(f.figure_id,gettext("&?"));
delmenu(f.figure_id,gettext("&Edit"));
delmenu(f.figure_id,gettext("&Tools"));
toolbar(f.figure_id,"off");
delmenu(f.figure_id,gettext("&File"));

//Criação do Novo Menu
menu2 = uimenu("parent",f, "label",gettext("About"));

//Inserindo informação no menu 1 criado
function Funcionamento_Scilab()
informacoes_geral = sprintf(gettext("Neste Programa deve -se utilizar
os dados nos seguintes formatos:\n\nUtilizar (.)pontos para números\n
Utilizar para Ângulos = Ex:233graus 22minutos e 10segundos = (no scilab =
233.2210)\n\nUtilizar distância = Ex: 233.333\n\nFormato do arquivo .txt da
Poligonal Fechada (*apenas dados Numéricos)\nEstação Vante Tipo_(P)
Ângulo_Horizontal Ângulo_vertical Distância_Horizontal\n\n Formato do
arquivo .txt do Metodo de Visadas Iguais do Nivelamento Geométrico (*apenas
dados Numericos)\n Estacao Tipo(N ou C) Ré_FS Ré_FI Vante_FS Vante_FI\
\n"));
messagebox(informacoes_geral, gettext("About"), "info", "modal");
endfunction
uimenu("parent",menu2, "label",gettext("About"),
"callback","Funcionamento_Scilab()");
sleep(600);

handles.dummy = 0;

handles.text35=icontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-
1],'Enable','on','FontAngle','normal','FontName','Tahoma','FontSize',
[10],'FontUnits','points','FontWeight','normal','ForegroundColor',[-1,-1,-
1],'HorizontalAlignment','center','ListboxTop',[],'Max',[1],'Min',
[0],'Position',
[0,0.6630435,0.08,0.0391304],'Relief','default','SliderStep',
[0.01,0.1],'String','PD_HR(graus.minseg)','Style','text','Value',
[0],'VerticalAlignment','middle','Visible','on','Tag','text35','Callback','
')
```

```
handles.text36=uicontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-1], 'Enable','on','FontAngle','normal','FontName','Tahoma','FontSize',[10], 'FontUnits','points','FontWeight','normal','ForegroundColor',[-1,-1,-1], 'HorizontalAlignment','center','ListboxTop',[], 'Max',[1], 'Min',[0], 'Position',[0,0.6230435,0.08,0.0391304], 'Relief','default','SliderStep',[0.01,0.1], 'String','PI_HR (graus.minseg)', 'Style','text','Value',[0], 'VerticalAlignment','middle','Visible','on','Tag','text36','Callback','')
```

```
handles.edit35=uicontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-1], 'Enable','on','FontAngle','normal','FontName','Tahoma','FontSize',[11], 'FontUnits','points','FontWeight','normal','ForegroundColor',[-1,-1,-1], 'HorizontalAlignment','left','ListboxTop',[], 'Max',[1], 'Min',[0], 'Position',[0.0814,0.6630435,0.07,0.0391304], 'Relief','default','SliderStep',[0.01,0.1], 'String','','Style','edit','Value',[0], 'VerticalAlignment','middle','Visible','on','Tag','edit35','Callback','')
```

```
handles.edit36=uicontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-1], 'Enable','on','FontAngle','normal','FontName','Tahoma','FontSize',[11], 'FontUnits','points','FontWeight','normal','ForegroundColor',[-1,-1,-1], 'HorizontalAlignment','left','ListboxTop',[], 'Max',[1], 'Min',[0], 'Position',[0.0814,0.6230435,0.07,0.0391304], 'Relief','default','SliderStep',[0.01,0.1], 'String','','Style','edit','Value',[0], 'VerticalAlignment','middle','Visible','on','Tag','edit36','Callback','')
```

```
handles.text37=uicontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-1], 'Enable','on','FontAngle','normal','FontName','Tahoma','FontSize',[11], 'FontUnits','points','FontWeight','bold','ForegroundColor',[-1,-1,-1], 'HorizontalAlignment','center','ListboxTop',[], 'Max',[1], 'Min',[0], 'Position',[0,0.7095652,0.15,0.025], 'Relief','default','SliderStep',[0.01,0.1], 'String','Ângulo Horizontal', 'Style','text','Value',[0], 'VerticalAlignment','middle','Visible','on','Tag','text37','Callback','')
```

```
handles.text38=uicontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-1], 'Enable','on','FontAngle','normal','FontName','Tahoma','FontSize',[10], 'FontUnits','points','FontWeight','normal','ForegroundColor',[-1,-1,-1], 'HorizontalAlignment','left','ListboxTop',[], 'Max',[1], 'Min',[0], 'Position',[0.16,0.6630435,0.08,0.0391304], 'Relief','default','SliderStep',[0.01,0.1], 'String','PD_VZ (graus.minseg)', 'Style','text','Value',[0], 'VerticalAlignment','middle','Visible','on','Tag','text38','Callback','')
```

```
handles.text39=uicontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-1], 'Enable','on','FontAngle','normal','FontName','Tahoma','FontSize',[10], 'FontUnits','points','FontWeight','normal','ForegroundColor',[-1,-1,-1], 'HorizontalAlignment','left','ListboxTop',[], 'Max',[1], 'Min',[0], 'Position',[0.16,0.6230435,0.08,0.0391304], 'Relief','default','SliderStep',[0.01,0.1], 'String','PI_VZ (graus.minseg)', 'Style','text','Value',[0], 'VerticalAlignment','middle','Visible','on','Tag','text39','Callback','')
```

```
handles.edit37=uicontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-1], 'Enable','on','FontAngle','normal','FontName','Tahoma','FontSize',[11], 'FontUnits','points','FontWeight','normal','ForegroundColor',[-1,-1,-1], 'HorizontalAlignment','left','ListboxTop',[], 'Max',[1], 'Min',[0], 'Position',[0.2411,0.6630435,0.07,0.0391304], 'Relief','default','SliderStep',[0.01,0.1], 'String','','Style','edit','Value',[0], 'VerticalAlignment','middle','Visible','on','Tag','edit37','Callback','')
```

```
handles.edit38=uicontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-1], 'Enable','on','FontAngle','normal','FontName','Tahoma','FontSize',[11], 'FontUnits','points','FontWeight','normal','ForegroundColor',[-1,-1,-1], 'HorizontalAlignment','left','ListboxTop',[], 'Max',[1], 'Min',[0], 'Position',[0.2411,0.6230435,0.07,0.0391304], 'Relief','default','SliderStep',[0.01,0.1], 'String','','Style','edit','Value',[0], 'VerticalAlignment','middle','Visible','on','Tag','edit38','Callback','')
```

```
handles.text40=uicontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-1], 'Enable','on','FontAngle','normal','FontName','Tahoma','FontSize',[11], 'FontUnits','points','FontWeight','bold','ForegroundColor',[-1,-1,-1], 'HorizontalAlignment','center','ListboxTop',[], 'Max',[1], 'Min',[0], 'Position',[0.16,0.7095652,0.15,0.025], 'Relief','default','SliderStep',[0.01,0.1], 'String','Ângulo Vertical','Style','text','Value',[0], 'VerticalAlignment','middle','Visible','on','Tag','text40','Callback','')
```

```
handles.edit39=uicontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-1], 'Enable','on','FontAngle','normal','FontName','Tahoma','FontSize',[10], 'FontUnits','points','FontWeight','normal','ForegroundColor',[-1,-1,-1], 'HorizontalAlignment','left','ListboxTop',[], 'Max',[1], 'Min',[0], 'Position',[0.112,0.44,0.08,0.0391304], 'Relief','default','SliderStep',[0.01,0.1], 'String','','Style','edit','Value',[0], 'VerticalAlignment','middle','Visible','on','Tag','edit39','Callback','')
```

```
handles.edit40=uicontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-1], 'Enable','on','FontAngle','normal','FontName','Tahoma','FontSize',[10], 'FontUnits','points','FontWeight','normal','ForegroundColor',[-1,-1,-1], 'HorizontalAlignment','left','ListboxTop',[], 'Max',[1], 'Min',[0], 'Position',[0.112,0.40,0.08,0.039], 'Relief','default','SliderStep',[0.01,0.1], 'String','','Style','edit','Value',[0], 'VerticalAlignment','middle','Visible','on','Tag','edit40','Callback','')
```

```
handles.edit41=uicontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-1], 'Enable','on','FontAngle','normal','FontName','Tahoma','FontSize',[10], 'FontUnits','points','FontWeight','normal','ForegroundColor',[-1,-1,-1], 'HorizontalAlignment','left','ListboxTop',[], 'Max',[1], 'Min',[0], 'Position',[0.112,0.36,0.08,0.0391304], 'Relief','default','SliderStep',[0.01,0.1], 'String','','Style','edit','Value',[0], 'VerticalAlignment','middle','Visible','on','Tag','edit41','Callback','')
```

```
handles.text41=uicontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-1], 'Enable','on','FontAngle','normal','FontName','Tahoma','FontSize',[10], 'FontUnits','points','FontWeight','normal','ForegroundColor',[-1,-1,-1], 'HorizontalAlignment','center','ListboxTop',[], 'Max',[1], 'Min',
```

```
[0], 'Position', [0.00, 0.44, 0.11, 0.0391304], 'Relief', 'default', 'SliderStep',
[0.01, 0.1], 'String', 'Fio Superior (m)', 'Style', 'text', 'Value',
[0], 'VerticalAlignment', 'middle', 'Visible', 'on', 'Tag', 'text41', 'Callback',
')
```

```
handles.text42=icontrol(f, 'unit', 'normalized', 'BackgroundColor', [-1, -1, -
1], 'Enable', 'on', 'FontAngle', 'normal', 'FontName', 'Tahoma', 'FontSize',
[10], 'FontUnits', 'points', 'FontWeight', 'normal', 'ForegroundColor', [-1, -1, -
1], 'HorizontalAlignment', 'center', 'ListboxTop', [], 'Max', [1], 'Min',
[0], 'Position', [0.00, 0.40, 0.11, 0.0391304], 'Relief', 'default', 'SliderStep',
[0.01, 0.1], 'String', 'Fio Inferior (m)', 'Style', 'text', 'Value',
[0], 'VerticalAlignment', 'middle', 'Visible', 'on', 'Tag', 'text42', 'Callback',
')
```

```
handles.text43=icontrol(f, 'unit', 'normalized', 'BackgroundColor', [-1, -1, -
1], 'Enable', 'on', 'FontAngle', 'normal', 'FontName', 'Tahoma', 'FontSize',
[10], 'FontUnits', 'points', 'FontWeight', 'normal', 'ForegroundColor', [-1, -1, -
1], 'HorizontalAlignment', 'center', 'ListboxTop', [], 'Max', [1], 'Min',
[0], 'Position', [0.00, 0.36, 0.11, 0.0391304], 'Relief', 'default', 'SliderStep',
[0.01, 0.1], 'String', 'Ângulo Vertical (graus.minseg)', 'Style', 'text', 'Value',
[0], 'VerticalAlignment', 'middle', 'Visible', 'on', 'Tag', 'text43', 'Callback',
')
```

```
handles.text44=icontrol(f, 'unit', 'normalized', 'BackgroundColor', [-1, -1, -
1], 'Enable', 'on', 'FontAngle', 'normal', 'FontName', 'Tahoma', 'FontSize',
[11], 'FontUnits', 'points', 'FontWeight', 'bold', 'ForegroundColor', [-1, -1, -
1], 'HorizontalAlignment', 'center', 'ListboxTop', [], 'Max', [1], 'Min',
[0], 'Position',
[0.00, 0.48, 0.1921, 0.0434783], 'Relief', 'default', 'SliderStep',
[0.01, 0.1], 'String', 'Distância Horizontal (m)', 'Style', 'text', 'Value',
[0], 'VerticalAlignment', 'middle', 'Visible', 'on', 'Tag', 'text44', 'Callback',
')
```

```
handles.butt12=icontrol(f, 'unit', 'normalized', 'BackgroundColor', [-1, -1, -
1], 'Enable', 'on', 'FontAngle', 'normal', 'FontName', 'Tahoma', 'FontSize',
[12], 'FontUnits', 'points', 'FontWeight', 'bold', 'ForegroundColor', [-1, -1, -
1], 'HorizontalAlignment', 'center', 'ListboxTop', [], 'Max', [1], 'Min',
[0], 'Position', [0.006, 0.58, 0.12, 0.04], 'Relief', 'default', 'SliderStep',
[0.01, 0.1], 'String', 'Clique para calcular', 'Style', 'pushbutton', 'Value',
[0], 'VerticalAlignment', 'middle', 'Visible', 'on', 'Tag', 'butt12', 'Callback',
'AH(handles)')
```

```
handles.butt13=icontrol(f, 'unit', 'normalized', 'BackgroundColor', [-1, -1, -
1], 'Enable', 'on', 'FontAngle', 'normal', 'FontName', 'Tahoma', 'FontSize',
[12], 'FontUnits', 'points', 'FontWeight', 'bold', 'ForegroundColor', [-1, -1, -
1], 'HorizontalAlignment', 'center', 'ListboxTop', [], 'Max', [1], 'Min',
[0], 'Position', [0.175, 0.58, 0.12, 0.04], 'Relief', 'default', 'SliderStep',
[0.01, 0.1], 'String', 'Clique para calcular', 'Style', 'pushbutton', 'Value',
[0], 'VerticalAlignment', 'middle', 'Visible', 'on', 'Tag', 'butt13', 'Callback',
'AV(handles)')
```

```
handles.edit42=icontrol(f, 'unit', 'normalized', 'BackgroundColor', [-1, -1, -
1], 'Enable', 'on', 'FontAngle', 'normal', 'FontName', 'Tahoma', 'FontSize',
[10], 'FontUnits', 'points', 'FontWeight', 'normal', 'ForegroundColor', [-1, -1, -
1], 'HorizontalAlignment', 'center', 'ListboxTop', [], 'Max', [1], 'Min',
[0], 'Position', [0.19, 0.542, 0.08, 0.0391304], 'Relief', 'default', 'SliderStep',
[0.01, 0.1], 'String', '', 'Style', 'edit', 'Value',
[0], 'VerticalAlignment', 'middle', 'Visible', 'on', 'Tag', 'edit42', 'Callback',
')
```

```

handles.edit43=uicontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-1],
'Enable','on','FontAngle','normal','FontName','Tahoma','FontSize',
[10],'FontUnits','points','FontWeight','normal','ForegroundColor',[-1,-1,-1],
'HorizontalAlignment','center','ListboxTop',[],'Max',[1],'Min',
[0],'Position',[0.03,0.542,0.08,0.0391304],'Relief','default','SliderStep',
[0.01,0.1],'String','','Style','edit','Value',
[0],'VerticalAlignment','middle','Visible','on','Tag','edit43','Callback','
')

handles.butt14=uicontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-1],
'Enable','on','FontAngle','normal','FontName','Tahoma','FontSize',
[12],'FontUnits','points','FontWeight','bold','ForegroundColor',[-1,-1,-1],
'HorizontalAlignment','center','ListboxTop',[],'Max',[1],'Min',
[0],'Position',[0.01,0.32,0.12,0.04],'Relief','default','SliderStep',
[0.01,0.1],'String','Clique para calcular','Style','pushbutton','Value',
[0],'VerticalAlignment','middle','Visible','on','Tag','butt14','Callback','
DH(handles)')

handles.edit44=uicontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-1],
'Enable','on','FontAngle','normal','FontName','Tahoma','FontSize',
[10],'FontUnits','points','FontWeight','normal','ForegroundColor',[-1,-1,-1],
'HorizontalAlignment','center','ListboxTop',[],'Max',[1],'Min',
[0],'Position',[0.03,0.28,0.08,0.0391304],'Relief','default','SliderStep',
[0.01,0.1],'String','','Style','edit','Value',
[0],'VerticalAlignment','middle','Visible','on','Tag','edit44','Callback','
')

handles.textlinha1=uicontrol(f,'unit','normalized','BackgroundColor',
[0,0,0],'Enable','on','FontAngle','normal','FontName','Tahoma','FontSize',
[11],'FontUnits','points','FontWeight','normal','ForegroundColor',[-1,-1,-1],
'HorizontalAlignment','left','ListboxTop',[],'Max',[1],'Min',
[0],'Position',[0.315,0.27,0.0058565,0.80],'Relief','default','SliderStep',
[0.01,0.1],'String','UnName1','Style','text','Value',
[0],'VerticalAlignment','middle','Visible','on','Tag','textlinha1','Callbac
k','')

handles.textlinha2=uicontrol(f,'unit','normalized','BackgroundColor',
[0,0,0],'Enable','on','FontAngle','normal','FontName','Tahoma','FontSize',
[11],'FontUnits','points','FontWeight','normal','ForegroundColor',[-1,-1,-1],
'HorizontalAlignment','left','ListboxTop',[],'Max',[1],'Min',
[0],'Position',
[0.6145095,0.5305674,0.0058565,0.4751773],'Relief','default','SliderStep',
[0.01,0.1],'String','UnName2','Style','text','Value',
[0],'VerticalAlignment','middle','Visible','on','Tag','textlinha2','Callbac
k','')

handles.textlinha3=uicontrol(f,'unit','normalized','BackgroundColor',
[0,0,0],'Enable','on','FontAngle','normal','FontName','Tahoma','FontSize',
[11],'FontUnits','points','FontWeight','normal','ForegroundColor',[-1,-1,-1],
'HorizontalAlignment','left','ListboxTop',[],'Max',[1],'Min',
[0],'Position',[0,0.27,0.32,0.0070922],'Relief','default','SliderStep',
[0.01,0.1],'String','UnName3','Style','text','Value',
[0],'VerticalAlignment','middle','Visible','on','Tag','textlinha3','Callbac
k','')

handles.textlinha4=uicontrol(f,'unit','normalized','BackgroundColor',
[0,0,0],'Enable','on','FontAngle','normal','FontName','Tahoma','FontSize',
[11],'FontUnits','points','FontWeight','normal','ForegroundColor',[-1,-1,-1],
'HorizontalAlignment','left','ListboxTop',[],'Max',[1],'Min',
[0],'Position',[0.32,0.53,0.30,0.0070922],'Relief','default','SliderStep',

```

```
[0.01,0.1],'String','UnName3','Style','text','Value',  
[0],'VerticalAlignment','middle','Visible','on','Tag','textlinha4','Callbac  
k','')
```

```
handles.text33=uicontrol(f,'unit','normalized','BackgroundColor',  
[0,0,0],'Enable','on','FontAngle','normal','FontName','Tahoma','FontSize',  
[11],'FontUnits','points','FontWeight','normal','ForegroundColor',[-1,-1,-  
1],'HorizontalAlignment','left','ListboxTop',[],'Max',[1],'Min',  
[0],'Position',[0.62,0.69,0.38,0.0070922],'Relief','default','SliderStep',  
[0.01,0.1],'String','UnName3','Style','text','Value',  
[0],'VerticalAlignment','middle','Visible','on','Tag','text33','Callback','  
)
```

```
handles.text4=uicontrol(f,'unit','normalized','BackgroundColor',  
[0,1,0],'Enable','on','FontAngle','normal','FontName','Tahoma','FontSize',  
[15],'FontUnits','points','FontWeight','bold','ForegroundColor',[-1,-1,-  
1],'HorizontalAlignment','center','ListboxTop',[],'Max',[1],'Min',  
[0],'Position',  
[0.0153734,0.9716312,0.25,0.0269504],'Relief','default','SliderStep',  
[0.01,0.1],'String','Cálculos de Azimute, Ângulos e  
Distância','Style','text','Value',  
[0],'VerticalAlignment','middle','Visible','on','Tag','text4','Callback',''  
)
```

```
handles.text5=uicontrol(f,'unit','normalized','BackgroundColor',  
[0,1,0],'Enable','on','FontAngle','normal','FontName','Tahoma','FontSize',  
[15],'FontUnits','points','FontWeight','bold','ForegroundColor',[-1,-1,-  
1],'HorizontalAlignment','center','ListboxTop',[],'Max',[1],'Min',  
[0],'Position',  
[0.33,0.9673759,0.2510981,0.0283688],'Relief','default','SliderStep',  
[0.01,0.1],'String','Cálculo da Poligonal Fechada','Style','text','Value',  
[0],'VerticalAlignment','middle','Visible','on','Tag','text5','Callback',''  
)
```

```
handles.text6=uicontrol(f,'unit','normalized','BackgroundColor',  
[0,1,0],'Enable','on','FontAngle','normal','FontName','Tahoma','FontSize',  
[15],'FontUnits','points','FontWeight','bold','ForegroundColor',[-1,-1,-  
1],'HorizontalAlignment','center','ListboxTop',[],'Max',[1],'Min',  
[0],'Position',  
[0.6515373,0.964539,0.329429,0.0312057],'Relief','default','SliderStep',  
[0.01,0.1],'String','Cálculo do Nivelamento Geométrico - Visadas  
Iguais','Style','text','Value',  
[0],'VerticalAlignment','middle','Visible','on','Tag','text6','Callback',''  
)
```

```
handles.text7=uicontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-  
1],'Enable','on','FontAngle','normal','FontName','Tahoma','FontSize',  
[12],'FontUnits','points','FontWeight','bold','ForegroundColor',[-1,-1,-  
1],'HorizontalAlignment','center','ListboxTop',[],'Max',[1],'Min',  
[0],'Position',  
[0.0297804,0.9262411,0.224451,0.0255319],'Relief','default','SliderStep',  
[0.01,0.1],'String','Cálculo do Azimute - Coordenadas  
Cartesianas','Style','text','Value',  
[0],'VerticalAlignment','middle','Visible','on','Tag','text7','Callback',''  
)
```

```
handles.text8=uicontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-  
1],'Enable','on','FontAngle','normal','FontName','Tahoma','FontSize',  
[11],'FontUnits','points','FontWeight','normal','ForegroundColor',[-1,-1,-  
1],'HorizontalAlignment','left','ListboxTop',[],'Max',[1],'Min',
```



```
[0], 'Position',  
[0,0.8593617,0.0497804,0.0255319], 'Relief', 'default', 'SliderStep',  
[0.01,0.1], 'String', 'Ponto 1', 'Style', 'text', 'Value',  
[0], 'VerticalAlignment', 'middle', 'Visible', 'on', 'Tag', 'text8', 'Callback', ''  
)
```

```
handles.text9=uicontrol(f, 'unit', 'normalized', 'BackgroundColor', [-1,-1,-  
1], 'Enable', 'on', 'FontAngle', 'normal', 'FontName', 'Tahoma', 'FontSize',  
[11], 'FontUnits', 'points', 'FontWeight', 'normal', 'ForegroundColor', [-1,-1,-  
1], 'HorizontalAlignment', 'left', 'ListboxTop', [], 'Max', [1], 'Min',  
[0], 'Position',  
[0,0.8293617,0.0497804,0.0255319], 'Relief', 'default', 'SliderStep',  
[0.01,0.1], 'String', 'Ponto 2', 'Style', 'text', 'Value',  
[0], 'VerticalAlignment', 'middle', 'Visible', 'on', 'Tag', 'text9', 'Callback', ''  
)
```

```
handles.edit1=uicontrol(f, 'unit', 'normalized', 'BackgroundColor', [-1,-1,-  
1], 'Enable', 'on', 'FontAngle', 'normal', 'FontName', 'Tahoma', 'FontSize',  
[11], 'FontUnits', 'points', 'FontWeight', 'normal', 'ForegroundColor', [-1,-1,-  
1], 'HorizontalAlignment', 'left', 'ListboxTop', [], 'Max', [1], 'Min',  
[0], 'Position',  
[0.0519766,0.8251064,0.101757,0.0269504], 'Relief', 'default', 'SliderStep',  
[0.01,0.1], 'String', '', 'Style', 'edit', 'Value',  
[0], 'VerticalAlignment', 'middle', 'Visible', 'on', 'Tag', 'edit1', 'Callback', ''  
)
```

```
handles.edit2=uicontrol(f, 'unit', 'normalized', 'BackgroundColor', [-1,-1,-  
1], 'Enable', 'on', 'FontAngle', 'normal', 'FontName', 'Tahoma', 'FontSize',  
[11], 'FontUnits', 'points', 'FontWeight', 'normal', 'ForegroundColor', [-1,-1,-  
1], 'HorizontalAlignment', 'left', 'ListboxTop', [], 'Max', [1], 'Min',  
[0], 'Position',  
[0.0519766,0.8551064,0.101757,0.0269504], 'Relief', 'default', 'SliderStep',  
[0.01,0.1], 'String', '', 'Style', 'edit', 'Value',  
[0], 'VerticalAlignment', 'middle', 'Visible', 'on', 'Tag', 'edit2', 'Callback', ''  
)
```

```
handles.text10=uicontrol(f, 'unit', 'normalized', 'BackgroundColor', [-1,-1,-  
1], 'Enable', 'on', 'FontAngle', 'normal', 'FontName', 'Tahoma', 'FontSize',  
[11], 'FontUnits', 'points', 'FontWeight', 'normal', 'ForegroundColor', [-1,-1,-  
1], 'HorizontalAlignment', 'center', 'ListboxTop', [], 'Max', [1], 'Min',  
[0], 'Position',  
[0.0778917,0.8836879,0.0549048,0.0312057], 'Relief', 'default', 'SliderStep',  
[0.01,0.1], 'String', 'X(m)', 'Style', 'text', 'Value',  
[0], 'VerticalAlignment', 'middle', 'Visible', 'on', 'Tag', 'text10', 'Callback', ''  
)
```

```
handles.edit3=uicontrol(f, 'unit', 'normalized', 'BackgroundColor', [-1,-1,-  
1], 'Enable', 'on', 'FontAngle', 'normal', 'FontName', 'Tahoma', 'FontSize',  
[11], 'FontUnits', 'points', 'FontWeight', 'normal', 'ForegroundColor', [-1,-1,-  
1], 'HorizontalAlignment', 'left', 'ListboxTop', [], 'Max', [1], 'Min',  
[0], 'Position',  
[0.1619766,0.8551064,0.101757,0.0269504], 'Relief', 'default', 'SliderStep',  
[0.01,0.1], 'String', '', 'Style', 'edit', 'Value',  
[0], 'VerticalAlignment', 'middle', 'Visible', 'on', 'Tag', 'edit3', 'Callback', ''  
)
```

```
handles.edit4=uicontrol(f, 'unit', 'normalized', 'BackgroundColor', [-1,-1,-  
1], 'Enable', 'on', 'FontAngle', 'normal', 'FontName', 'Tahoma', 'FontSize',  
[11], 'FontUnits', 'points', 'FontWeight', 'normal', 'ForegroundColor', [-1,-1,-  
1], 'HorizontalAlignment', 'left', 'ListboxTop', [], 'Max', [1], 'Min',
```

```

[0], 'Position',
[0.1619766,0.8251064,0.101757,0.0269504], 'Relief', 'default', 'SliderStep',
[0.01,0.1], 'String', '', 'Style', 'edit', 'Value',
[0], 'VerticalAlignment', 'middle', 'Visible', 'on', 'Tag', 'edit4', 'Callback', ''
)

handles.text11=uicontrol(f, 'unit', 'normalized', 'BackgroundColor', [-1,-1,-
1], 'Enable', 'on', 'FontAngle', 'normal', 'FontName', 'Tahoma', 'FontSize',
[11], 'FontUnits', 'points', 'FontWeight', 'normal', 'ForegroundColor', [-1,-1,-
1], 'HorizontalAlignment', 'center', 'ListboxTop', [], 'Max', [1], 'Min',
[0], 'Position',
[0.1878917,0.8836879,0.0549048,0.0312057], 'Relief', 'default', 'SliderStep',
[0.01,0.1], 'String', 'Y (m)', 'Style', 'text', 'Value',
[0], 'VerticalAlignment', 'middle', 'Visible', 'on', 'Tag', 'text11', 'Callback', ''
)

handles.butt1=uicontrol(f, 'unit', 'normalized', 'BackgroundColor', [-1,-1,-
1], 'Enable', 'on', 'FontAngle', 'normal', 'FontName', 'Tahoma', 'FontSize',
[13], 'FontUnits', 'points', 'FontWeight', 'bold', 'ForegroundColor', [-1,-1,-
1], 'HorizontalAlignment', 'center', 'ListboxTop', [], 'Max', [1], 'Min',
[0], 'Position',
[0.0292826,0.7758865,0.2057101,0.0439716], 'Relief', 'default', 'SliderStep',
[0.01,0.1], 'String', 'Clique para calcular o
Azimute', 'Style', 'pushbutton', 'Value',
[0], 'VerticalAlignment', 'middle', 'Visible', 'on', 'Tag', 'butt1', 'Callback', 'b
utt1_AZ(handles)')

handles.text15=uicontrol(f, 'unit', 'normalized', 'BackgroundColor', [-1,-1,-
1], 'Enable', 'on', 'FontAngle', 'normal', 'FontName', 'Tahoma', 'FontSize',
[12], 'FontUnits', 'points', 'FontWeight', 'bold', 'ForegroundColor', [-1,-1,-
1], 'HorizontalAlignment', 'center', 'ListboxTop', [], 'Max', [1], 'Min',
[0], 'Position',
[0.35,0.9177305,0.1669107,0.0340426], 'Relief', 'default', 'SliderStep',
[0.01,0.1], 'String', '1° Passo - Digite os dados
Iniciais', 'Style', 'text', 'Value',
[0], 'VerticalAlignment', 'middle', 'Visible', 'on', 'Tag', 'text15', 'Callback', ''
)

handles.text16=uicontrol(f, 'unit', 'normalized', 'BackgroundColor', [-1,-1,-
1], 'Enable', 'on', 'FontAngle', 'normal', 'FontName', 'Tahoma', 'FontSize',
[11], 'FontUnits', 'points', 'FontWeight', 'normal', 'ForegroundColor', [-1,-1,-
1], 'HorizontalAlignment', 'center', 'ListboxTop', [], 'Max', [1], 'Min',
[0], 'Position',
[0.325,0.8794326,0.1449341,0.0312057], 'Relief', 'default', 'SliderStep',
[0.01,0.1], 'String', 'Azimute de Partida
(graus.minseg):', 'Style', 'text', 'Value',
[0], 'VerticalAlignment', 'middle', 'Visible', 'on', 'Tag', 'text16', 'Callback', ''
)

handles.text17=uicontrol(f, 'unit', 'normalized', 'BackgroundColor', [-1,-1,-
1], 'Enable', 'on', 'FontAngle', 'normal', 'FontName', 'Tahoma', 'FontSize',
[12], 'FontUnits', 'points', 'FontWeight', 'normal', 'ForegroundColor', [-1,-1,-
1], 'HorizontalAlignment', 'center', 'ListboxTop', [], 'Max', [1], 'Min',
[0], 'Position',
[0.325,0.8394326,0.1449341,0.0312057], 'Relief', 'default', 'SliderStep',
[0.01,0.1], 'String', 'Coordenada X_inicial:', 'Style', 'text', 'Value',
[0], 'VerticalAlignment', 'middle', 'Visible', 'on', 'Tag', 'text17', 'Callback', ''
)

```

```

handles.text18=uicontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-1],
'Enable','on','FontAngle','normal','FontName','Tahoma','FontSize',
[12],'FontUnits','points','FontWeight','normal','ForegroundColor',[-1,-1,-1],
'HorizontalAlignment','center','ListboxTop',[],'Max',[1],'Min',
[0],'Position',
[0.325,0.7994326,0.1449341,0.0312057],'Relief','default','SliderStep',
[0.01,0.1],'String','Coordenada Y_inicial:', 'Style','text','Value',
[0],'VerticalAlignment','middle','Visible','on','Tag','text18','Callback','
')

handles.text19=uicontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-1],
'Enable','on','FontAngle','normal','FontName','Tahoma','FontSize',
[11],'FontUnits','points','FontWeight','normal','ForegroundColor',[-1,-1,-1],
'HorizontalAlignment','center','ListboxTop',[],'Max',[1],'Min',
[0],'Position',
[0.325,0.7594326,0.1449341,0.0312057],'Relief','default','SliderStep',
[0.01,0.1],'String','Tolerância Angular(graus.minseg):', 'Style','text','Value',
[0],'VerticalAlignment','middle','Visible','on','Tag','text19','Callback','
')

handles.text20=uicontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-1],
'Enable','on','FontAngle','normal','FontName','Tahoma','FontSize',
[11],'FontUnits','points','FontWeight','normal','ForegroundColor',[-1,-1,-1],
'HorizontalAlignment','center','ListboxTop',[],'Max',[1],'Min',
[0],'Position',
[0.325,0.7194326,0.1449341,0.0312057],'Relief','default','SliderStep',
[0.01,0.1],'String','Tolerância Linear(Escala):', 'Style','text','Value',
[0],'VerticalAlignment','middle','Visible','on','Tag','text20','Callback','
')

handles.text21=uicontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-1],
'Enable','on','FontAngle','normal','FontName','Tahoma','FontSize',
[12],'FontUnits','points','FontWeight','normal','ForegroundColor',[-1,-1,-1],
'HorizontalAlignment','center','ListboxTop',[],'Max',[1],'Min',
[0],'Position',
[0.325,0.6794326,0.1449341,0.0312057],'Relief','default','SliderStep',
[0.01,0.1],'String','Sentido da Poligonal:', 'Style','text','Value',
[0],'VerticalAlignment','middle','Visible','on','Tag','text21','Callback','
')

handles.text22=uicontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-1],
'Enable','on','FontAngle','normal','FontName','Tahoma','FontSize',
[12],'FontUnits','points','FontWeight','normal','ForegroundColor',[-1,-1,-1],
'HorizontalAlignment','center','ListboxTop',[],'Max',[1],'Min',
[0],'Position',
[0.325,0.6394326,0.1449341,0.0312057],'Relief','default','SliderStep',
[0.01,0.1],'String','Ângulos Medidos:', 'Style','text','Value',
[0],'VerticalAlignment','middle','Visible','on','Tag','text22','Callback','
')

handles.edit5=uicontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-1],
'Enable','on','FontAngle','normal','FontName','Tahoma','FontSize',
[12],'FontUnits','points','FontWeight','normal','ForegroundColor',[-1,-1,-1],
'HorizontalAlignment','left','ListboxTop',[],'Max',[1],'Min',
[0],'Position',
[0.471,0.8765957,0.1076135,0.0340426],'Relief','default','SliderStep',
[0.01,0.1],'String','','Style','edit','Value',
[0],'VerticalAlignment','middle','Visible','on','Tag','edit5','Callback','
')

```

```
handles.edit6=uicontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-1], 'Enable','on','FontAngle','normal','FontName','Tahoma','FontSize',[12], 'FontUnits','points','FontWeight','normal','ForegroundColor',[-1,-1,-1], 'HorizontalAlignment','left','ListboxTop',[], 'Max',[1], 'Min',[0], 'Position',[0.471,0.8365957,0.1076135,0.0340426], 'Relief','default','SliderStep',[0.01,0.1], 'String','','Style','edit','Value',[0], 'VerticalAlignment','middle','Visible','on','Tag','edit6','Callback','')
)
```

```
handles.edit7=uicontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-1], 'Enable','on','FontAngle','normal','FontName','Tahoma','FontSize',[12], 'FontUnits','points','FontWeight','normal','ForegroundColor',[-1,-1,-1], 'HorizontalAlignment','left','ListboxTop',[], 'Max',[1], 'Min',[0], 'Position',[0.471,0.7965957,0.1076135,0.0340426], 'Relief','default','SliderStep',[0.01,0.1], 'String','','Style','edit','Value',[0], 'VerticalAlignment','middle','Visible','on','Tag','edit7','Callback','')
)
```

```
handles.edit8=uicontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-1], 'Enable','on','FontAngle','normal','FontName','Tahoma','FontSize',[12], 'FontUnits','points','FontWeight','normal','ForegroundColor',[-1,-1,-1], 'HorizontalAlignment','left','ListboxTop',[], 'Max',[1], 'Min',[0], 'Position',[0.471,0.7565957,0.1076135,0.0340426], 'Relief','default','SliderStep',[0.01,0.1], 'String','','Style','edit','Value',[0], 'VerticalAlignment','middle','Visible','on','Tag','edit8','Callback','')
)
```

```
handles.edit9=uicontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-1], 'Enable','on','FontAngle','normal','FontName','Tahoma','FontSize',[12], 'FontUnits','points','FontWeight','normal','ForegroundColor',[-1,-1,-1], 'HorizontalAlignment','left','ListboxTop',[], 'Max',[1], 'Min',[0], 'Position',[0.471,0.7165957,0.1076135,0.0340426], 'Relief','default','SliderStep',[0.01,0.1], 'String','','Style','edit','Value',[0], 'VerticalAlignment','middle','Visible','on','Tag','edit9','Callback','')
)
```

```
handles.menu1=uicontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-1], 'Enable','on','FontAngle','normal','FontName','Tahoma','FontSize',[12], 'FontUnits','points','FontWeight','normal','ForegroundColor',[-1,-1,-1], 'HorizontalAlignment','left','ListboxTop',[], 'Max',[1], 'Min',[0], 'Position',[0.471,0.6765957,0.1076135,0.0340426], 'Relief','default','SliderStep',[0.01,0.1], 'String','Horário|Anti-Horário','Style','popupmenu','Value',[0], 'VerticalAlignment','middle','Visible','on','Tag','menu1','Callback','')
)
```

```
handles.menu2=uicontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-1], 'Enable','on','FontAngle','normal','FontName','Tahoma','FontSize',[12], 'FontUnits','points','FontWeight','normal','ForegroundColor',[-1,-1,-1], 'HorizontalAlignment','left','ListboxTop',[], 'Max',[1], 'Min',[0], 'Position',[0.471,0.6365957,0.1076135,0.0340426], 'Relief','default','SliderStep',[0.01,0.1], 'String','Externo|Interno','Style','popupmenu','Value',[0], 'VerticalAlignment','middle','Visible','on','Tag','menu2','Callback','')
)
```

```
handles.text23=uicontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-1], 'Enable','on','FontAngle','normal','FontName','Tahoma','FontSize',[12], 'FontUnits','points','FontWeight','bold','ForegroundColor',[-1,-1,-1], 'HorizontalAlignment','center','ListboxTop',[], 'Max',[1], 'Min',[0], 'Position',[0.72,0.9177305,0.1669107,0.0340426], 'Relief','default','SliderStep',[0.01,0.1], 'String','1° Passo - Digite os dados Iniciais', 'Style','text', 'Value',[0], 'VerticalAlignment','middle', 'Visible','on', 'Tag','text23', 'Callback','')
```

```
handles.text24=uicontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-1], 'Enable','on','FontAngle','normal','FontName','Tahoma','FontSize',[12], 'FontUnits','points','FontWeight','normal','ForegroundColor',[-1,-1,-1], 'HorizontalAlignment','center','ListboxTop',[], 'Max',[1], 'Min',[0], 'Position',[0.63,0.8794326,0.1449341,0.0312057], 'Relief','default','SliderStep',[0.01,0.1], 'String','Coordenada Z_Inicial:', 'Style','text', 'Value',[0], 'VerticalAlignment','middle', 'Visible','on', 'Tag','text24', 'Callback','')
```

```
handles.text25=uicontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-1], 'Enable','on','FontAngle','normal','FontName','Tahoma','FontSize',[12], 'FontUnits','points','FontWeight','normal','ForegroundColor',[-1,-1,-1], 'HorizontalAlignment','center','ListboxTop',[], 'Max',[1], 'Min',[0], 'Position',[0.63,0.8394326,0.1449341,0.0312057], 'Relief','default','SliderStep',[0.01,0.1], 'String','Tolerância Altimétrica(mm):', 'Style','text', 'Value',[0], 'VerticalAlignment','middle', 'Visible','on', 'Tag','text25', 'Callback','')
```

```
handles.edit10=uicontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-1], 'Enable','on','FontAngle','normal','FontName','Tahoma','FontSize',[12], 'FontUnits','points','FontWeight','normal','ForegroundColor',[-1,-1,-1], 'HorizontalAlignment','left','ListboxTop',[], 'Max',[1], 'Min',[0], 'Position',[0.7755,0.8765957,0.1076135,0.0340426], 'Relief','default','SliderStep',[0.01,0.1], 'String','', 'Style','edit', 'Value',[0], 'VerticalAlignment','middle', 'Visible','on', 'Tag','edit10', 'Callback','')
```

```
handles.edit11=uicontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-1], 'Enable','on','FontAngle','normal','FontName','Tahoma','FontSize',[12], 'FontUnits','points','FontWeight','normal','ForegroundColor',[-1,-1,-1], 'HorizontalAlignment','left','ListboxTop',[], 'Max',[1], 'Min',[0], 'Position',[0.7755,0.8365957,0.1076135,0.0340426], 'Relief','default','SliderStep',[0.01,0.1], 'String','', 'Style','edit', 'Value',[0], 'VerticalAlignment','middle', 'Visible','on', 'Tag','edit11', 'Callback','')
```

```
handles.butt4=uicontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-1], 'Enable','on','FontAngle','normal','FontName','Tahoma','FontSize',[13], 'FontUnits','points','FontWeight','bold','ForegroundColor',[-1,-1,-1], 'HorizontalAlignment','center','ListboxTop',[], 'Max',[1], 'Min',[0], 'Position',[0.65,0.75,0.30,0.048227], 'Relief','default','SliderStep',[0.01,0.1], 'String','2° Passo - Clique aqui para inserir txt nivelamento', 'Style','pushbutton', 'Value',
```

```
[0], 'VerticalAlignment', 'middle', 'Visible', 'on', 'Tag', 'butt1', 'Callback', 'butt4_NG(handles)')
```

```
handles.axes1= newaxes();handles.axes1.margins = [ 0 0 0  
0];handles.axes1.axes_bounds = [0.73,0.40,0.260,0.380];
```

```
handles.text102=uicontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-1], 'Enable', 'on', 'FontAngle', 'normal', 'FontName', 'Tahoma', 'FontSize', [12], 'FontUnits', 'points', 'FontWeight', 'normal', 'ForegroundColor', [-1,-1,-1], 'HorizontalAlignment', 'center', 'ListboxTop', [], 'Max', [1], 'Min', [0], 'Position', [0.325,0.59,0.1449341,0.0312057], 'Relief', 'default', 'SliderStep', [0.01,0.1], 'String', 'Exportar Resultados em txt:', 'Style', 'text', 'Value', [0], 'VerticalAlignment', 'middle', 'Visible', 'on', 'Tag', 'text22', 'Callback', '')
```

```
handles.menu102=uicontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-1], 'Enable', 'on', 'FontAngle', 'normal', 'FontName', 'Tahoma', 'FontSize', [12], 'FontUnits', 'points', 'FontWeight', 'normal', 'ForegroundColor', [-1,-1,-1], 'HorizontalAlignment', 'left', 'ListboxTop', [], 'Max', [1], 'Min', [0], 'Position', [0.471,0.59,0.08,0.0340426], 'Relief', 'default', 'SliderStep', [0.01,0.1], 'String', 'Sim|Não', 'Style', 'popupmenu', 'Value', [0], 'VerticalAlignment', 'middle', 'Visible', 'on', 'Tag', 'menu2', 'Callback', '' )
```

```
handles.butt3=uicontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-1], 'Enable', 'on', 'FontAngle', 'normal', 'FontName', 'Tahoma', 'FontSize', [12], 'FontUnits', 'points', 'FontWeight', 'bold', 'ForegroundColor', [-1,-1,-1], 'HorizontalAlignment', 'center', 'ListboxTop', [], 'Max', [1], 'Min', [0], 'Position', [0.34,0.535,0.2628111,0.048227], 'Relief', 'default', 'SliderStep', [0.01,0.1], 'String', '2° Passo - Clique aqui para Inserir txt poligonal', 'Style', 'pushbutton', 'Value', [0], 'VerticalAlignment', 'middle', 'Visible', 'on', 'Tag', 'butt3', 'Callback', 'butt1_PF(handles)')
```

```
handles.text105=uicontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-1], 'Enable', 'on', 'FontAngle', 'normal', 'FontName', 'Tahoma', 'FontSize', [12], 'FontUnits', 'points', 'FontWeight', 'normal', 'ForegroundColor', [-1,-1,-1], 'HorizontalAlignment', 'center', 'ListboxTop', [], 'Max', [1], 'Min', [0], 'Position', [0.63,0.80,0.1449341,0.0312057], 'Relief', 'default', 'SliderStep', [0.01,0.1], 'String', 'Exportar Resultados em txt:', 'Style', 'text', 'Value', [0], 'VerticalAlignment', 'middle', 'Visible', 'on', 'Tag', 'text25', 'Callback', '' )
```

```
handles.menu105=uicontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-1], 'Enable', 'on', 'FontAngle', 'normal', 'FontName', 'Tahoma', 'FontSize', [12], 'FontUnits', 'points', 'FontWeight', 'normal', 'ForegroundColor', [-1,-1,-1], 'HorizontalAlignment', 'left', 'ListboxTop', [], 'Max', [1], 'Min', [0], 'Position', [0.7755,0.80,0.1076135,0.0340426], 'Relief', 'default', 'SliderStep', [0.01,0.1], 'String', 'Sim|Não', 'Style', 'popupmenu', 'Value', [0], 'VerticalAlignment', 'middle', 'Visible', 'on', 'Tag', 'menu2', 'Callback', '' )
```

```
f.visible = "on";
```

```
//Função para Calcular o Azimute  
function butt1_AZ(handles)
```

```

handles.edit12=uicontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-
1],'Enable','on','FontAngle','normal','FontName','Tahoma','FontSize',
[12],'FontUnits','points','FontWeight','normal','ForegroundColor',[-1,-1,-
1],'HorizontalAlignment','center','ListboxTop',[],'Max',[1],'Min',
[0],'Position',[0.07,0.745,0.12,0.03],'Relief','default','SliderStep',
[0.01,0.1],'String','','Style','edit','Value',
[0],'VerticalAlignment','middle','Visible','on','Tag','edit12','Callback','
')

//Coordenadas Digitadas pelo Usuario
X0 = handles.edit2.string; Y0 = handles.edit3.string;
X1 = handles.edit1.string; Y1 = handles.edit4.string;
//Convertendo a string para formato real
X00 = strtod(X0);Y00 = strtod(Y0);
X11 = strtod(X1);Y11 = strtod(Y1);

//Cálculo do Azimute sem o quadrante
dis_x = (X11 - X00);
dis_y = (Y11 - Y00);
Azimute0 = abs(atan((dis_x) / (dis_y))*(180/%pi))

//1° quadrante
if (X11 - X00) > 0 & (Y11 - Y00) >= 0 then
    Azimute_Partida = Azimute0;
//2° quadrante
elseif (X11 - X00) >= 0 & (Y11 - Y00) < 0 then
    num_quad = 180;
    Azimute_Partida = 180 - Azimute0;
//3° quadrante
elseif (X11 - X00) < 0 & (Y11 - Y00) < 0 then
    Azimute_Partida = 180 + Azimute0;
//4° quadrante
elseif (X11 - X00) < 0 & (Y11 - Y00) >= 0
    Azimute_Partida = 360 - Azimute0;

end

//Mostrar o Resultado na Interface Gráfica
handles.edit12.string = string(Azimute_Partida)
endfunction

//Cálculo do Ângulo Horizontal
function AH(handles)
    PD_HR = handles.edit35.string;
    PI_HR = handles.edit36.string;
    //Convertendo a string para formato real
    PD_HR0 = strtod(PD_HR)
    PI_HR0 = strtod(PI_HR)

    //Convertendo (PD_HR0)de Gruas, Minutos e Segundos para GrausDecimais
    G_PD_HR0 = int((PD_HR0));
    M_PD_HR0 = int((PD_HR0 - G_PD_HR0)* 100);
    S_PD_HR0 = (((PD_HR0 - G_PD_HR0)* 100) - M_PD_HR0)*100;
    PD_HR1 = G_PD_HR0 + (M_PD_HR0/60)+ (S_PD_HR0/3600);
    //Converter GrausDecimais em Radianos
    PD_HR1_RAD = (PD_HR1*%pi)/180;

    //Convertendo (PI_HR0)de Gruas, Minutos e Segundos para GrausDecimais
    G_PI_HR0 = int((PI_HR0));
    M_PI_HR0 = int((PI_HR0 - G_PI_HR0)* 100);
    S_PI_HR0 = (((PI_HR0 - G_PI_HR0)* 100) - M_PI_HR0)*100;

```

```

PI_HR1 = G_PI_HR0 + (M_PI_HR0/60) + (S_PI_HR0/3600);
//Converter GrausDecimais em Radianos
PI_HR1_RAD = (PI_HR1*pi)/180;

//Cálculo do Ângulo Horizontal
if PD_HR1 > PI_HR1 then
AH0 = ((PD_HR1 + PI_HR1)/2)+90
elseif PD_HR1 < PI_HR1
AH0 = ((PD_HR1 + PI_HR1)/2)-90
end

//Mostrar o Resultado na Interface Gráfica
handles.edit43.string = string(AH0)
endfunction

//Cálculo do Ângulo Vertical
function AV(handles)
PD_VZ = handles.edit37.string;
PI_VZ = handles.edit38.string;
//Convertendo string para numero real
PD_VZ0 = strtod(PD_VZ);
PI_VZ0 = strtod(PI_VZ);

//Convertendo (PD_VZ0)de Gruas, Minutos e Segundos para GrausDecimais
G_PD_VZ0 = int((PD_VZ0));
M_PD_VZ0 = int ((PD_VZ0 - G_PD_VZ0) * 100);
S_PD_VZ0 = (((PD_VZ0 - G_PD_VZ0) * 100) - M_PD_VZ0) * 100;
PD_VZ1 = G_PD_VZ0 + (M_PD_VZ0/60) + (S_PD_VZ0/3600);
//Converter GrausDecimais em Radianos
PD_VZ1_RAD = (PD_VZ1*pi)/180;

//Convertendo (PI_VZ0)de Gruas, Minutos e Segundos para GrausDecimais
G_PI_VZ0 = int((PI_VZ0));
M_PI_VZ0 = int ((PI_VZ0 - G_PI_VZ0) * 100);
S_PI_VZ0 = (((PI_VZ0 - G_PI_VZ0) * 100) - M_PI_VZ0) * 100;
PI_VZ1 = G_PI_VZ0 + (M_PI_VZ0/60) + (S_PI_VZ0/3600);
//Converter GrausDecimais em Radianos
PI_VZ1_RAD = (PI_VZ1*pi)/180;

//Cálculo do o ângulo vertical isento do erro de verticalidade
Z = (360 + PD_VZ1 - PI_VZ1)/2;
Z_rad = (Z*pi)/180
//Mostrando o resultado
handles.edit42.string = string(Z)
endfunction

//Cálculo da Distância Horizontal
function DH(handles)
//Lendo os dados
FS = handles.edit39.string;
FI = handles.edit40.string;
ANGV = handles.edit41.string;
//Convertendo de string para numero real
FS0 = strtod(FS);
FI0 = strtod(FI);
AV0 = strtod(ANGV);
//Convertendo (Ângulo Vertical)de Gruas, Minutos e Segundos para
GrausDecimais
G = int((AV0));
M = int ((AV0 - G) * 100);

```



```

S = (((AV0 - G) * 100) - M)*100;
Z = G + (M/60) + (S/3600);
//Converter GrausDecimais em Radianos
Z_RAD = (Z*pi)/180;

//Cálculo da Distância Horizontal
DistH = (FS0 - FI0) * 100 .* sin(Z_RAD)^2;
//Mostrando os resultados
handles.edit44.string = string(DistH)

endfunction

//Cálculo da Poligonal Fechada
function butt1_PF(handles)
handles.text30=uicontrol(f,'unit','normalized','BackgroundColor',
[1,1,0],'Enable','on','FontAngle','normal','FontName','Tahoma','FontSize',
[12],'FontUnits','points','FontWeight','bold','ForegroundColor',[-1,-1,-
1],'HorizontalAlignment','center','ListboxTop',[],'Max',[1],'Min',
[0],'Position',[0.325,0.495,0.30,0.03],'Relief','default','SliderStep',
[0.01,0.1],'String','RESULTADOS DO CÁLCULO DA POLIGONAL
FECHADA:','Style','text','Value',
[0],'VerticalAlignment','middle','Visible','on','Tag','text30','Callback','
')
//Lendo o arquivo txt de entrada
Poligonal_irrad_entrada = uigetfile("*.txt",pwd(),"Selecione o
arquivo da Poligonal Fechada");
pol_irrad_entrada = mopen(Poligonal_irrad_entrada,"r");
i = 1;
while ~feof(pol_irrad_entrada)
[num, Estacao(i),...
Vante(i),...
Tipo(i),...
Ângulo_Horizontal(i),...
Ângulo_Vertical(i),...
dh(i)] = m fscanff(pol_irrad_entrada,"%s %s %s %s %s %s")
i = i + 1;
end
//-----Conversão de String para
float-----
//Lendo as variaveis inseridas pelo usuario em formato de string
Azimute_Graus_0 = handles.edit5.string
coord_x0_0 = handles.edit6.string
coord_y0_0 = handles.edit7.string
TA_0 = handles.edit8.string
Tol_Line_Escala_0 = handles.edit9.string
Sent_da_Poligonal=get(handles.menu1,'Value')
Ang_Medidos = get(handles.menu2,'Value')
//Converter para valores numericos
Azimute_Graus = strtod(Azimute_Graus_0);
coord_x0 = strtod(coord_x0_0);
coord_y0 = strtod(coord_y0_0);
TA = strtod(TA_0);
Tol_Line_Escala = strtod(Tol_Line_Escala_0);

//Convertendo (Azimute de Partida)de Gruas, Minutos e Segundos para
GrausDecimais
//separando em graus, minutos, segundos e graus decimais
G_AZ = int((Azimute_Graus));
M_AZ = int ((Azimute_Graus - G_AZ) * 100);
S_AZ = (((Azimute_Graus - G_AZ) * 100) - M_AZ)*100;

```

```

AZO = G_AZ + (M_AZ/60) + (S_AZ/3600);
//Converter GrausDecimais em Radianos
AZO_RAD = (AZO*pi)/180;

//Convertendo (AH)de Graus, Minutos e Segundos para GrausDecimais
//separando em graus, minutos, segundos e graus decimais
G_AH = int(strtod(Ângulo_Horizontal));
M_AH = int ((strtod(Ângulo_Horizontal) - G_AH)* 100);
S_AH = (((strtod(Ângulo_Horizontal) - G_AH)* 100) - M_AH)*100;
AH_poligonal_Deg = G_AH + (M_AH/60) + (S_AH/3600);
//Converter GrausDecimais em Radianos
AH_poligonal_rad = (AH_poligonal_Deg*pi)/180;

//Convertendo (AV)de Graus, Minutos e Segundos para GrausDecimais
//separando em graus, minutos, segundos e graus decimais
G_AV = int(strtod(Ângulo_Vertical));
M_AV = int ((strtod(Ângulo_Vertical) - G_AV)* 100);
S_AV = (((strtod(Ângulo_Vertical) - G_AV)* 100) - M_AV)*100;
AV_poligonal_Deg = G_AV + (M_AV/60) + (S_AV/3600);
//Converter GrausDecimais em Radianos
AV_poligonal_rad = (AV_poligonal_Deg*pi)/180;

//Convertendo (TA)de Graus, Minutos e Segundos para GrausDecimais
//separando em graus, minutos, segundos e graus decimais
G_TA = int((TA));
M_TA = int (((TA) - G_TA)* 100);
S_TA = (((((TA) - G_TA)* 100) - M_TA)*100);
TA_poligonal_Deg = G_TA + (M_TA/60) + (S_TA/3600);
//Converter GrausDecimais em Radianos
TA_poligonal_rad = (TA_poligonal_Deg*pi)/180;

//-----1) Verificação do erro angular da Poligonal
Fechada-----
//Verificar os Ângulos Horizontais da Poligonal Fechada do arquivo txt
AH_poli = find (Tipo == 'P' | Tipo == 'p');
AH_polinal = AH_poligonal_Deg(AH_poli);

//Somatório dos Ângulos Horizontais da Poligonal
ΣAH_AH_poligonal_Deg = sum(AH_polinal);

//Índice dos Elementos dos Ângulos Horizontais que são maiores que 0
graus decimais
indices_AH_poligonal_deg = find(AH_polinal > 0);

//Contar o número de pontos da Poligonal Fechada
n_pontos = size(indices_AH_poligonal_deg, "*.");

//Cálculo do Erro Angular
//Verificar se os Ângulos medidos são (Externo ou Interno)
if (Ang_Medidos == 1) then
    EA = (ΣAH_AH_poligonal_Deg - (n_pontos+2)*180);
elseif(Ang_Medidos==2) then
    EA = (ΣAH_AH_poligonal_Deg - (n_pontos-2)*180);
end

//Calculo da Tolerância Angular
TA_poligonal_Deg_C = TA_poligonal_Deg * sqrt (n_pontos);
//Verificação se há o Erro Angular
if (abs (EA) <= abs (TA_poligonal_Deg_C)) then
    resultado11 = "Não ha o Erro Angular";

```

```

elseif(abs(EA) > abs(TA_poligonal_Deg_C)) then
    resultadol1 = "Erro Angular - Volte a Campo";
end

//Classificação da poligonal segundo o EA na NBR 13.1333
if (abs(EA) <= 0.001666667) then
    classe_EA = "Classe I P - Adensamento da rede geodésica -
(transporte de coordenadas).";
elseif (abs(EA) > 0.001666667) & (abs(EA) <= 0.004166667) then
    classe_EA = "Classe II P - Apoio topográfico para projetos
básicos, executivos, como executado e obras de engenharia."
elseif (abs(EA) > 0.004166667) & (abs(EA) <= 0.005555556) then
    classe_EA = "Classe III P - Adensamento do apoio topográfico
para projetos básicos, executivos, como executado, e obras de engenharia."
elseif (abs(EA) > 0.005555556) & (abs(EA) <= 0.011111111) then
    classe_EA = "Classe IV P - Adensamento do apoio topográfico
para poligonais IIIP. Levantamentos topográficos para estudos de
viabilidade em projetos de engenharia."
elseif (abs(EA) > 0.011111111) & (abs(EA) <= 0.05) then
    classe_EA = "Classe V P - Levantamentos topográficos para
estudos expeditos."
end

//-----2) Correção do Erro
Angular-----
correção_ang = -(EA/n_pontos);
//Verificar se os angulos horizontais são verdadeiros ou não
AH1 = AH_polinal > 0;
AH2 = AH1 * correção_ang;
//Ângulos Horizontais Corrigidos
AH_poligonal_Deg_Corri2 = [AH_polinal + AH2]';
//Posicao do ponto de partida para verificar o erro linear
AH_OPP = [AH_poligonal_Deg_Corri2(1)]';
AH_poligonal_Deg_Corri = [AH_poligonal_Deg_Corri2
AH_OPP]';//=====
//Somatório dos Angulos Horizontais corrigidos
ΣAH_poligonal_Deg_Corri = sum(AH_poligonal_Deg_Corri);
//Verificando a correção dos angulos corrigidos
if (Ang_Medidos==1) then
    ΣAH_C_AH_poligonal_Deg = (ΣAH_poligonal_Deg_Corri -
(n_pontos+2)*180);
elseif(Ang_Medidos==2) then
    ΣAH_C_AH_poligonal_Deg = (ΣAH_poligonal_Deg_Corri - (n_pontos-
2)*180);
elseif (Ang_Medidos==0 && Ang_Medidos=="")
    //messagebox ("Digite 1 ou 2");
end

//-----3) CÁLCULO DOS AZIMUTES DA POLIGONAL
indeces_AH_poligonal_deg2 = find(AH_poligonal_Deg_Corri>0);
n_pontos2 = size(indeces_AH_poligonal_deg2, "*");
//Azimute
azimute_p0 = AZ0;
azimute_p(1) = azimute_p0;
n = n_pontos2;
//Cálculo dos Azimutes do Sentido Horário
if(Sent_da_Poligonal == 1) then
for i=2:n
    if ((azimute_p(i-1)+AH_poligonal_Deg_Corri(i)-180)>0) && ((azimute_p(i-
1)+AH_poligonal_Deg_Corri(i)-180)<=360) then

```

```

        azimute_p(i)=azimute_p(i-1)+AH_poligonal_Deg_Corri(i)-180;
elseif ((azimute_p(i-1)+AH_poligonal_Deg_Corri(i)-180)>360) then
    azimute_p(i)=azimute_p(i-1)+AH_poligonal_Deg_Corri(i)-180-360;
else ((azimute_p(i-1)+AH_poligonal_Deg_Corri(i)-180)<0)
    azimute_p(i)=azimute_p(i-1)+AH_poligonal_Deg_Corri(i)-180+360;
end
end
end

//Cálculo dos Azimutes do Sentido Anti-Horário
if(Sent_da_Poligonal == 2) then
for i=2:n
    if ((azimute_p(i-1)-AH_poligonal_Deg_Corri(i)+180)>0) && ((azimute_p(i-1)-AH_poligonal_Deg_Corri(i)+180)<=360) then
        azimute_p(i)=azimute_p(i-1)-AH_poligonal_Deg_Corri(i)+180;
elseif ((azimute_p(i-1)-AH_poligonal_Deg_Corri(i)+180)>360) then
        azimute_p(i)=azimute_p(i-1)-AH_poligonal_Deg_Corri(i)+180-360;
else ((azimute_p(i-1)-AH_poligonal_Deg_Corri(i)+180)<0)
        azimute_p(i)=azimute_p(i-1)-AH_poligonal_Deg_Corri(i)+180+360;
end
end
end

//-----4) Cálculo das coordenadas provisórias
azimute_p_rad = (azimute_p*pi)/180;
//Verificar as Distâncias Horizontais da Poligonal Fechada do arquivo
txt
DH_pol = find (Tipo == 'P' | Tipo == 'p');
dh_float = strtod (dh);
DH_pol2 = dh_float(DH_pol);
//Índice dos Elementos das Distâncias Horizontais que são maiores que 0
graus decimais
DH_pol_deg1 = find(DH_pol2 > 0);
DH_pol_deg = DH_pol2(DH_pol_deg1);

Xi0 = coord_x0;
Xi(1)= Xi0;
Yi0 = coord_y0;
Yi(1)=Yi0;
for i=2:n
    Xi(i)= Xi(i-1)+ DH_pol_deg(i-1) .* sin (azimute_p_rad(i-1));
    Yi(i)= Yi(i-1)+ DH_pol_deg(i-1) .* cos (azimute_p_rad(i-1));
end

//-----5) Verificação do erro linear
Xopp = Xi($);
Yopp = Yi($);
ex = Xopp - coord_x0;
ey = Yopp - coord_y0;
EL = sqrt((ex)^2 + (ey)^2);
Perimetro = sum (DH_pol2);
Z_Escala = Perimetro / EL;
//Verificando se há o Erro Linear
if (Z_Escala >= Tol_Line_Escala) then
    resultado12 = "Erro Linear - OK";
elseif(Z_Escala < Tol_Line_Escala) then
    resultado12 = "Houve Erro Linear, Volte a campo";
end

//Classificação da poligonal segundo o EL na NBR 13.1333
if (EL <= 0.10) then

```

```

        classe_EL = "Classe I P - Adensamento da rede geodésica -
(transporte de coordenadas).";
        elseif (EL > 0.10) & (EL <= 0.30)then
            classe_EL = "Classe II P - Apoio topográfico para projetos
básicos, executivos, como executado e obras de engenharia."
        elseif (EL > 0.30) & (EL <= 0.42)then
            classe_EL = "Classe III P - Adensamento do apoio topográfico
para projetos básicos, executivos, como executado, e obras de engenharia."
        elseif (EL > 0.42) & (EL <= 0.56)then
            classe_EL = "Classe IV P - Adensamento do apoio topográfico
para poligonais IIIP. Levantamentos topográficos para estudos de
viabilidade em projetos de engenharia."
        elseif (EL > 0.56) & (EL <= 2.20)then
            classe_EL = "Classe V P - Levantamentos topográficos para
estudos expeditos."
        end

//-----6) Cálculo das coordenadas
corrigidas-----
//Correção do Erro Linear
Cxi = -ex * (DH_pol_deg/Perimetro);
Cyi = -ey * (DH_pol_deg/Perimetro);
//Coordenadas Corrigidas
Xc0 = coord_x0;
Xc(1)= Xc0;
Yc0 = coord_y0;
Yc(1)=Yc0;
for i=2:n
    Xc(i)= Xc(i-1)+ DH_pol_deg(i-1) .* sin (azimute_p_rad(i-1)) +
Cxi(i-1);
    Yc(i)= Yc(i-1)+ DH_pol_deg(i-1) .* cos (azimute_p_rad(i-1)) +
Cyi(i-1);
end

//-----7)Cálculo de Área da
Poligonal-----
for i=2:n
    Σarea1(i)=Xc(i) * Yc(i-1);
    Σarea2(i)=Yc(i) * Xc(i-1);
end

Area_da_poligonal_m2 = abs((sum(Σarea1)-sum(Σarea2)))/2;

//-----Tabela com os resultados dos
calculos-----
parametros2 = [" " "Estacao" "Vante" "Tipo" "ÂH" "DH" "AH_Corrig" "Azimute"
"Xi" "Yi" "Xfinal" "Yfinal"];
ntabela = n - 1;
for i = 1:ntabela
Pontos_cont02(i) = [string(i)];
end
AH00 = string(AH_poligonal_Deg);
DH00 = string(dh_float);
azimute_p00 = string(azimute_p(1:ntabela));
AH_poligonal_Deg_Corri1 = string(AH_poligonal_Deg_Corri(1:ntabela));
Xi1 = string(Xi(1:ntabela));
Yi1 = string(Yi(1:ntabela));
Xc1 = string(Xc(1:ntabela));
Yc1 = string(Yc(1:ntabela));

```

```

//Classificação da poligonal segundo o EL na NBR 13.1333 - So a Classe
if (EL <= 0.10) then
    classe_EL = "Classe I P"
elseif (EL > 0.10) & (EL <= 0.30)then
    classe_EL = "Classe II P"
elseif (EL > 0.30) & (EL <= 0.42)then
    classe_EL = "Classe III P"
elseif (EL > 0.42) & (EL <= 0.56)then
    classe_EL = "Classe IV P"
elseif (EL > 0.56) & (EL <= 2.20)then
    classe_EL = "Classe V P"
elseif EL > 2.20 then
    classe_EL = "Sem classificação"
end

//table2 = [parametros2; [Pontos_cont02 Estacao Tipo Ângulo_Horizontal
Ângulo_Vertical dh AH_poligonal_Deg_Corri1 azimuth_p1 X11 Y11 Xc1 Yc1]];
table2 = [parametros2; [Pontos_cont02 Estacao Vante Tipo AH00 DH00
AH_poligonal_Deg_Corri1 azimuth_p00 X11 Y11 Xc1 Yc1]];

tamanho_figura2 = get(f, "axes_size");

configuracao_tabela2 = uicontrol(f, "style", "table",...
    "string", table2,..
    "position", [10 (tamanho_figura2(2) - 667) 900 120],...
    "tooltipstring", "Data from majors towns");

//-----Tabela com os Resultados dos Erros, Tolerâncias, Perímetro e
Área
params = [" " "Nome" "Resultado"];
cont = ["1" "2" "3" "4" "5" "6" "7" "8" "9" "10"]';
Nome = ["Erro Angular" "Tolerância Angular" "Houve erro Angular" "Erro
x(m)" "Erro y(Escala)" "Erro Linear(m)" "Houve erro Linear" "Perímetro(m)"
"Classe - NBR131333" "Área(m²)"]';

//convertendo para string
EA1 = string(EA)
TA1 = string(TA)
resultad0111 = string(resultad011)
ex1 = string(ex)
ey1 = string(ey)
Z_Escala_int = int(Z_Escala)
EL1 = string(Z_Escala_int)
resultad0121 = string(resultad012)
Perimetrol = string(Perimetro)
classe_EL1 = string(classe_EL)
Area_da_poligonal_m21 = string(Area_da_poligonal_m2)
valores = string([EA1 TA1 resultad0111 ex1 ey1 EL1 resultad0121 Perimetrol
classe_EL1 Area_da_poligonal_m21]');
table1 = [params; [ cont Nome valores ]]

as = f.axes_size; // [Altura Comprimento]

confi_table1 = uicontrol("style","table",...
    "string",table1,..
    "position",[420 as(2)-540 438 188],...
    "tooltipstring","Data from majors towns")

//=====Croqui da Poligonal
Fechada=====
set("auto_clear","on")// Fechar as figuras abertas e limpar

```

```

plot(Xc,Yc,'r--d',Xc(1),Yc(1),'o','markersize',15,'markbackgro','red')
legend('Pontos da Poligonal','Ponto de Partida',-6);
xlabel("X(m)", "fontsize", 3, "font_style",4, "color", "black")
ylabel("Y(m)", "fontsize", 3, "font_style",4, "color", "black")
title("Croqui da Poligonal Fechada", "fontsize", 5, "color", "black")

//Elaborando arquivo de saida resultados em txt
Exportar=get(handles.menu102,'Value')
if Exportar == 1 then
pol_sai = mopen("resultados_da_poligonal_fechada.txt","w");
mfprintf(pol_sai,"Resultados_da_Poligonal_Fechada \n")
mfprintf(pol_sai,"\n\nErro_Ângular = ");mfprintf(pol_sai,"%s", EA1)
mfprintf(pol_sai,"\nTolerância_Ângular = ");mfprintf(pol_sai,"%s", TA1)
mfprintf(pol_sai,"\nErro_Linear = ");mfprintf(pol_sai,"%s", EL1)
mfprintf(pol_sai,"\nTolerância_Linear = ");mfprintf(pol_sai,"%f",
Tol_Line_Escala)
mfprintf(pol_sai,"\nPerimetro = ");mfprintf(pol_sai,"%s", Perimetrol)
mfprintf(pol_sai,"\nÁrea_da_Poligonal_Fechada(m²) =
");mfprintf(pol_sai,"%s\n\n", Area_da_poligonal_m21)
mfprintf(pol_sai,"\nunidade unidade GrausDecimais metros
GrausDecimais GrausDecimais metros metros metros metros")
mfprintf(pol_sai,"\nEstação Vante Tipo Âng_Ho Distância
Âng_Ho_Corrig Azimute xi yi X Y")

for i=1:1:ntabela
mfprintf(pol_sai,"\n%s %s %s %s %s %s %s %s %s %s %s %s",
Estacao(i),Vante(i),Tipo(i),AH00(i),DH00(i),AH_poligonal_Deg_Corrig(i),azim
ute_p00(i),Xi1(i),Yi1(i),Xc1(i),Yc1(i));
end

fclose(pol_sai)

elseif Exportar == 2 then
0
end
endfunction

//Cálculo do Nivelamento Geometrico Composto
function butt4_NG(handles)
handles.text300=uicontrol(f,'unit','normalized','BackgroundColor',
[1,1,0],'Enable','on','FontAngle','normal','FontName','Tahoma','FontSize',
[12],'FontUnits','points','FontWeight','bold','ForegroundColor',[-1,-1,-
1],'HorizontalAlignment','center','ListboxTop',[],'Max',[1],'Min',
[0],'Position',[0.325,0.495,0.30,0.03],'Relief','default','SliderStep',
[0.01,0.1],'String','RESULTADOS DO CÁLCULO DO
NIVELAMENTO:', 'Style','text','Value',
[0],'VerticalAlignment','middle','Visible','on','Tag','text30','Callback','
')
Poligonal_irrad_entrada = uigetfile("*.txt",pwd(),"Selecione o arquivo da
Poligonal Fechada");
//Abrindo e Lendo o arquivo txt de entrada
Nivelamento_entrada = uigetfile("*.txt",pwd(),"Selecione o arquivo do
Nivelamento Geometrico");
nivela_entrada = mopen(Nivelamento_entrada,"r");
i = 1;
while ~feof(nivela_entrada)
[num, Estacao(i),...
Tipo(i),...
Re_FS(i),...
Re_FI(i),..

```

```

                Vante_FS(i), ...
                Vante_FI(i)] = mfscanf(nivela_entrada, "%s %s %s %s %s %s")
            i = i + 1;
        end
        n = size(Re_FS, "*");
//-----Lendo as variaveis inseridas pelo usuario em formato de
string---
        T_Alti = handles.edit11.string
        coord_z = handles.edit10.string

//-----Converter string para
Float-----
Re_FS0 = strtod(Re_FS);
Re_FI0 = strtod(Re_FI);
Vante_FS0 = strtod(Vante_FS);
Vante_FI0 = strtod(Vante_FI);
T_Alti0 = strtod(T_Alti);
coord_z0 = strtod(coord_z);

//-----Separando o Nivelamento e o
Contranivelamento-----
        pts_N = find (Tipo == 'N' | Tipo == 'n');
        pts_C = find (Tipo == 'C' | Tipo == 'c');
        Estacao_N = Estacao(pts_N);
        Tipo_N = Tipo(pts_N);
        Re_FS_N = Re_FS0(pts_N);
        Re_FI_N = Re_FI0(pts_N);
        Vante_FS_N = Vante_FS0(pts_N);
        Vante_FI_N = Vante_FI0(pts_N);

        Re_FS_C = Re_FS0(pts_C);
        Re_FI_C = Re_FI0(pts_C);
        Vante_FS_C = Vante_FS0(pts_C);
        Vante_FI_C = Vante_FI0(pts_C);

//-----Cálculo do Fio
Nivelador-----
        Fio_re_N = (Re_FS_N + Re_FI_N)/2
        Fio_Vante_N = (Vante_FS_N + Vante_FI_N)/2

        Fio_re_C = (Re_FS_C + Re_FI_C)/2
        Fio_Vante_C = (Vante_FS_C + Vante_FI_C)/2

//-----Cálculo da Distância Ré e
Vante-----
        Dist_re_N = (Re_FS_N - Re_FI_N) * 100;
        Dist_Vante_N = (Vante_FS_N - Vante_FI_N) * 100;

        Dist_re_C = (Re_FS_C - Re_FI_C) * 100;
        Dist_Vante_C = (Vante_FS_C - Vante_FI_C) * 100;

//-----Cálculo da Distância
Nivelada-----
        Distancia_nive_N = (Dist_re_N + Dist_Vante_N);
        Distancia_nive_C = (Dist_re_C + Dist_Vante_C);

//-----Cálculo
desnível-----
        Desniv_N = (Fio_re_N - Fio_Vante_N);
        Desniv_C = (Fio_re_C - Fio_Vante_C);

```



```

//-----Cálculo da Tolerância e
Erro-----
Soma_desnivel_N = (sum(Desniv_N));
Soma_desnivel_N1 = abs(Soma_desnivel_N);
Soma_desnivel_C = (sum(Desniv_C));
Soma_desnivel_C1 = abs(Soma_desnivel_C);
Dist_nivelada_N = sum(Dist_re_N) + sum(Dist_Vante_N);
Dist_nivelada_C = sum(Dist_re_C) + sum(Dist_Vante_C);

//Cálculo da Tolerância
//Distância média nivelada
Dm = ((Dist_nivelada_N) + (Dist_nivelada_C))/2;//em metros
Dm_km = Dm / 1000;//em Km
t = T_Alti0 * sqrt(Dm_km)/1000;//em metros
//Cálculo do Erro cometido
ec = ((Soma_desnivel_N1) - (Soma_desnivel_C1))//em mm;
ec1 = abs(ec);
if abs(ec1) <= abs(t) then
informacao = "Não a Erro Altimétrico";
else
informacao = "Houve erro, Volte a Campo";
end

//-----Cálculo da Altitude do
Nivelamento-----
n = size(Dist_re_N, "*.");
Z = coord_z0;
for i=2:n
Z(i) = Z(i-1) + Desniv_N(i-1);
end

//-----Cálculo da Altitude do Contra-
Nivelamento-----
Z01 = Z($);
n_contranive = size(Dist_re_C, "*.");
for i=2:n_contranive
Z01(i) = Z01(i-1) + Desniv_C(i-1);
end

//-----Tabela com os resultados do
Nivelamento-----
parametros7 = [" " "Estacao" "Tipo" "Fio Nivelada-RE" "Fio Nivelada-Vante"
"Distância-Ré" "Distância-Vante" "Dist_nivelada" "Desnível" "Altitude"];
for i = 1:n
Pontos_cont07(i) = [string(i)];
end
Fio_re_N00 = string(Fio_re_N); Fio_Vante_N00 = string(Fio_Vante_N);
Dist_re_N0 = string(Dist_re_N);
Dist_Vante_N0 = string(Dist_Vante_N);
Distancia_nive_N0 = string(Distancia_nive_N);
Desniv_N0 = string(Desniv_N);
Z0 = string(Z);
table7 = [parametros7; [Pontos_cont07 Estacao_N Tipo_N Fio_re_N00
Fio_Vante_N00 Dist_re_N0 Dist_Vante_N0 Distancia_nive_N0 Desniv_N0 Z0]];
tamanho_figura7 = get(f, "axes_size");
configuracao_tabela7 = uicontrol(f, "style", "table",...
"string", table7,...
"position", [10 (tamanho_figura7(2) - 667) 900 120],...

```

```

        "tooltipstring", "Data from majors towns");

//-----Tabela de Erros do
Nivelamento-----
parametros8 = [" " "Nome" "Resultado"];
cont8 = ["1" "2" "3" "4" "5" "6" "7" "8" "9"];
//Nome = ["Numero de Pontos" "Σ-Distâncias Nivelamento" "Σ-Distâncias
Contra-Nivelamento" "Tolerância Altimetrica(m)" "Erro Altimetrico(m)"
"Resultado do Erro"];
Nome = ["Numero de Pontos" "Desnível do nivelamento (m)" "Desnível do
contranivelamento (m)" "Distância nivelada - nivelamento (m)" "Distância
nivelada-contranivela(m)" "Distância média nivelada (m)" "Tolerância
altimétrica (m)" "Erro Cometido (m)" "Resultado do Erro"];
//convertendo para string
n0 = string(n)
Soma_desnivel_N0 = string(Soma_desnivel_N)
Soma_desnivel_C0 = string(Soma_desnivel_C)
Dist_nivelada_N0 = string(Dist_nivelada_N);
Dist_nivelada_C0 = string(Dist_nivelada_C);
Dm0 = string(Dm);
t0 = string(t)
ec0 = string(ec)
informacao0 = string(informacao)

valores8 = string([n0 Soma_desnivel_N0 Soma_desnivel_C0 Dist_nivelada_N0
Dist_nivelada_C0 Dm0 t0 ec0 informacao0]);

table8 = [parametros8; [ cont8 Nome valores8]]

as = f.axes_size; // [Altura height]
confi_table8 = uicontrol("style","table",...
    "string",table8,...
    "position",[420 as(2)-540 438 188],...
    "tooltipstring","Data from majors towns")

//-----Gráfico de Perfil das
Altitudes-----
set("auto_clear","on")
plot(Z,'linest','.', 'marker','s','markeredge','cyan','markerFace','blue','ma
rkersize',7)
title("Altitudes calculadas dos Pontos", "fontsize", 5, "color", "black")
xlabel('Pontos');
ylabel('Altitudes (m) ')

//-----Elaborando arquivo de saida resultados em
txt-----
Fio_nive_Re = string([Fio_re_N Fio_re_C]);
Fio_nive_Vante = string([Fio_Vante_N Fio_Vante_C]);
Dista_re = string([Dist_re_N Dist_re_C]);
Dista_Vante = string([Dist_Vante_N Dist_Vante_C]);
Distancia_Nivelada = string([Distancia_nive_N Distancia_nive_C]);
Desnivel = string([Desniv_N Desniv_C]);
Z00 = string([Z Z01]);

n_nivel = n + n_contranive;

Exportar2=get(handles.menu105,'Value')

if Exportar2 == 1 then
nivela_entrada = mopen("resultados_do_nivelamento.txt","w");

```

```

mfprintf(nivela_entrada, "Resultados_do_Nivelamento_Geométrico_Composto_Visa
das_Iguais\n");
mfprintf(nivela_entrada, "\nNumero_de_Pontos=
");mfprintf(nivela_entrada, "%s", n0)
mfprintf(nivela_entrada, "\nSoma_Distância_Nivelamento_(m)=
");mfprintf(nivela_entrada, "%s", Soma_desnivel_N0);
mfprintf(nivela_entrada, "\nSoma_Distância-ContraNivelamento_(m)=
");mfprintf(nivela_entrada, "%s", Soma_desnivel_C0);
mfprintf(nivela_entrada, "\nErro_Altimétrico(mm)=
");mfprintf(nivela_entrada, "%s", t0);
mfprintf(nivela_entrada, "\nTolerância_Altimétrica(mm)=
");mfprintf(nivela_entrada, "%s", ec0);
mfprintf(nivela_entrada, "\n\n uni metros metros metros metros metros
metros metros");
mfprintf(nivela_entrada, "\nEst Tipo FN_Ré FN_Vante Dist_Ré
Dist_Vant Dist_nivelada Desnivel Altitude");

for i=1:1:n_nivel
mfprintf(nivela_entrada, "\n%s %s %s %s %s %s
%s %s %s %s", Estacao(i),
Tipo(i), Fio_nive_Re(i),
Fio_nive_Vante(i), Dista_re(i), Dista_Vante(i), Distancia_Nivelada(i), Desnivel
(i), Z00(i));
end

mclose(nivela_entrada)
elseif Exportar2 == 2 then
0
end
endfunction

```

### APÊNDICE B - RESULTADOS DA POLIGONAL FECHADA

Resultados da Poligonal Fechada										
Erro Ângular	=	-0,0044444								
Tolerância Ângular	=	0,001								
Erro Linear	=	2570								
Tolerância Linear	=	2000								
Perimetro	=	513,37								
Área da Poligonal Fechada(m <sup>2</sup> )	=	12941,555								
unidade	unidade	unidade	Graus Decimais	metros	Graus Decimais	Graus Decimais	metros	metros	metros	metros
Estação	Vante	Tipo	Âng_Horiz	Distância	Âng_Horiz_Corrig	Azimute	xi	yi	X	Y
OPP	1	P	301,48417	100,18	301,48506	106,86861	224,19	589,25	224,19	589,25
1	2	P	246,79028	115,8	246,79117	173,65978	320,05952	560,17997	320,04918	560,21755
2	3	P	261,49278	116,68	261,49367	255,15344	332,84756	445,08824	332,82526	445,16925
3	4	P	301,75306	91,65	301,75394	16,907389	220,06286	415,19118	220,02852	415,31596
4	OPP	P	148,47528	89,06	148,47617	345,38356	246,71702	502,87971	246,67322	503,03886

### APÊNDICE C - RESULTADOS DO NIVELAMENTO

Resultados_do_Nivelamento_Geométrico_Composto_Visadas_Iguais								
Numero de Pontos=	36							
Soma Distância Nivelamento (m)=	-16,326							
Soma Distância- ContraNivelamento(m)=	16,3346							
Erro Altimétrico(mm)=	0,0101599							
Tolerância Altimétrica(mm)=	-0,0086							
	uni	metros	metros	metros	metros	metros	metros	metros
Est	Tipo	FN_Ré	FN_Vante	Dist_Ré	Dist_Vant	Dist_nivelada	Desnivel	Altitude
1	N	1,2721	1,8055	10,16	10	20,16	0,5334	100
2	N	1,2334	1,85015	9,9	10,23	20,13	-0,61675	99,4666
3	N	1,23355	1,9038	9,85	10,3	20,15	-0,67025	98,84985
4	N	1,27165	1,92915	9,87	10,77	20,64	-0,6575	98,1796
5	N	1,1284	1,7855	10,28	9,7	19,98	-0,6571	97,5221
6	N	0,8935	2,14475	10,1	9,83	19,93	-1,25125	96,865
7	N	0,67475	2,27015	9,77	10,37	20,14	-1,5954	95,61375
8	N	0,77425	2,3036	9,93	10	19,93	-1,52935	94,01835
9	N	0,85675	2,3803	10,63	9,54	20,17	-1,52355	92,489
10	N	0,652	2,23	10	10	20	-1,578	90,96545
11	N	0,57945	2,21965	9,89	10,47	20,36	-1,6402	89,38745
12	N	0,7443	2,2285	9,66	10,5	20,16	-1,4842	87,74725
13	N	0,7898	2,14045	10	10,03	20,03	-1,35065	86,26305
14	N	0,5495	2,1645	9,5	9,1	18,6	-1,615	84,9124
15	N	0,801	2,52195	9,6	10,41	20,01	-1,72095	83,2974
16	N	0,645	2,4083	10,12	9,94	20,06	-1,7633	81,57645
17	N	0,71625	2,4238	9,85	10,04	19,89	-1,70755	79,81315
18	N	0,6984	2,501	9,88	10,2	20,08	-1,8026	78,1056
19	N	0,6093	2,36215	10	9,97	19,97	-1,75285	76,303
20	N	0,71925	1,97075	10,13	9,97	20,1	-1,2515	74,55015
21	N	0,95695	1,7918	10,33	9,64	19,97	-0,83485	73,29865

22	N	1,3601	1,7361	10,24	9,9	20,14	-0,376	72,4638
23	N	1,1993	1,7454	8,4	8,68	17,08	-0,5461	72,0878
24	N	1,20075	1,49045	9,91	9,97	19,88	-0,2897	71,5417
25	N	1,4968	1,3175	10,2	9,7	19,9	0,1793	71,252
26	N	1,84745	1,29895	10,23	9,81	20,04	0,5485	71,4313
27	N	1,95695	1,13045	10,81	9,97	20,78	0,8265	71,9798
28	N	2,13415	0,7283	10,07	9,8	19,87	1,40585	72,8063
29	N	2,3591	0,79145	10,04	10,09	20,1	1,56765	74,21215
30	N	2,021	0,9765	10	10,9	20,9	1,0445	75,7798
31	N	2,2539	0,89105	10	9,49	19,49	1,36285	76,8243
32	N	2,37615	1,04025	10,17	10,07	20,24	1,3359	78,18715
33	N	2,2906	0,9312	10,4	9,48	19,88	1,3594	79,52305
34	N	1,9658	1,09575	10	9,97	19,97	0,87005	80,88245
35	N	2,0973	1,03775	9,94	10,05	19,99	1,05955	81,7525
36	N	2,0555	1,19355	8,5	9,27	17,77	0,86195	82,81205
1	C	1,309	2,17245	9,4	8,51	17,91	-0,86345	82,81205
2	C	1,0441	2,1031	10,02	9,9	19,92	-1,059	81,9486
3	C	0,968	1,8383	9,8	10,14	19,94	-0,8703	80,8896
4	C	0,93045	2,28665	9,83	10,13	19,96	-1,3562	80,0193
5	C	0,9904	2,3263	10,68	9,66	20,34	-1,3359	78,6631
6	C	0,8983	2,2598	9,86	9,56	19,42	-1,3615	77,3272
7	C	1,01875	2,06275	10,23	9,85	20,08	-1,044	75,9657
8	C	0,8134	2,3806	10,08	10	20,08	-1,5672	74,9217
9	C	0,77755	2,1846	9,73	10	19,73	-1,40705	73,3545
10	C	0,96545	1,7933	9,83	10,26	20,09	-0,82785	71,94745
11	C	1,5935	2,1415	10,5	9,7	20,2	-0,548	71,1196
12	C	1,38615	1,56695	9,97	9,87	19,84	-0,1808	70,5716
13	C	1,5583	1,27565	9,94	10,13	20,07	0,28265	70,3908
14	C	1,8883	1,3419	9,34	8,58	17,92	0,5464	70,67345
15	C	1,8926	1,5139	10	10,02	20,02	0,3787	71,21985
16	C	1,9599	1,12155	10	10,01	20,01	0,83835	71,59855
17	C	2,0273	0,7713	9,54	10,94	20,48	1,256	72,4369
18	C	2,4994	0,74695	10,1	10,13	20,23	1,75245	73,6929
19	C	2,4228	0,6218	10,04	10,04	20,08	1,801	75,44535
20	C	2,3679	0,6528	10,08	9,84	19,92	1,7151	77,24635
21	C	2,33075	0,56495	10,15	10,39	20,54	1,7658	78,96145
22	C	2,4151	0,69405	10,5	9,91	20,41	1,72105	80,72725

23	C	2,3283	0,7108	9,66	8,96	18,62	1,6175	82,4483
24	C	2,1845	0,83545	9,9	10,29	20,19	1,34905	84,0658
25	C	2,2483	0,7656	10,14	9,66	19,8	1,4827	85,41485
26	C	2,3023	0,6615	10,14	9,9	20,04	1,6408	86,89755
27	C	2,2165	0,64245	9,7	10,29	19,99	1,57405	88,53835
28	C	2,3079	0,7859	9,98	10,42	20,4	1,522	90,1124
29	C	2,4015	0,8682	10,3	10,12	20,42	1,5333	91,6344
30	C	2,11665	0,5254	9,87	10,18	20,05	1,59125	93,1677
31	C	2,127	0,8731	10,2	9,9	20,1	1,2539	94,75895
32	C	1,768	1,1125	10	9,9	19,9	0,6555	96,01285
33	C	1,78075	1,12425	10,15	10,05	20,2	0,6565	96,66835
34	C	1,9281	1,2575	10,36	10,1	20,46	0,6706	97,32485
35	C	1,79445	1,1782	10,03	10,06	20,09	0,61625	97,99545
36	C	1,72945	1,1945	9,89	9,8	19,69	0,53495	98,6117