

Sumário

1. Introdução:	2
2. Implementação:	2
3. Testes	2
4. Conclusão	5
Referências	5
Anexos	5
ponto.h	5
ponto.c	5
principal.c	5
poligono.txt	5

1. Introdução:

O objetivo deste trabalho é ler as coordenadas de um polígono que estão em um arquivo do tipo .txt, calcular a área deste polígono e imprimir o resultado no console/terminal.

GitHub:

<https://github.com/arthursleite/Estrutura-Dados-UCB/tree/main/tp03>

2. Implementação:

Foi utilizada a linguagem C com assistência da IDE Visual Studio Code e compilador GCC.

A implementação foi feita utilizando um arquivo .h com as definições que serão utilizadas, um arquivo .c contendo as funções auxiliares do programa e outro arquivo .c contendo o programa principal.

3. Testes

Este é o arquivo ponto.h com as declarações da TAD e das funções:

```
1  #ifndef PONTO_H
2  #define PONTO_H
3
4  typedef struct
5  {
6      float x;
7      float y;
8  } Ponto;
9
10
11 Ponto criarPonto(float x, float y);
12 float calcularAreaTriangulo(Ponto p1, Ponto p2, Ponto p3);
13 float calcularAreaPoligono(Ponto coordenadas[], int numVertices);
14
15 #endif
```

Este é o arquivo ponto.c com as funções auxiliares que serão utilizadas no programa principal.

```
1  #include <stdio.h>
2  #include "ponto.h"
3
4  Ponto criarPonto(float x, float y)
5  {
6      Ponto p;
7      p.x = x;
8      p.y = y;
9      return p;
10 }
11
12 float calcularAreaTriangulo(Ponto a, Ponto b, Ponto c)
13 {
14     float area = (a.x * b.y * 1.0 +
15                  a.y * 1.0 * c.x +
16                  1.0 * b.x * c.y -
17                  a.y * b.x * 1.0 -
18                  a.x * 1.0 * c.y -
19                  1.0 * b.y * c.x) *
20     0.5;
21
22     return area;
23 }
24
25 float calcularAreaPoligono(Ponto coordenadas[], int numVertices)
26 {
27     float area = 0.0;
28
29     if (numVertices < 3)
30     {
31         printf("Um poligono deve ter pelo menos 3 vertices para calcular a area.\n");
32         return area;
33     }
34
35     for (int i = 1; i < numVertices - 1; i++)
36     {
37         area += calcularAreaTriangulo(coordenadas[0], coordenadas[i], coordenadas[i + 1]);
38     }
39
40     if (area < 0)
41     {
42         area *= -1;
43     }
44     return area;
45 }
46
47
```

Este é o código com a função principal que lerá o arquivo com as coordenadas, chamará as funções auxiliares para realizar o cálculo da área do polígono e imprimirá o resultado no terminal.

```
1  #include <stdio.h>
2  #include "ponto.h"
3  #include "ponto.c"
4
5  int main()
6  {
7      int numVertices;
8      float area;
9      float x, y;
10
11     FILE *arquivo;
12
13     arquivo = fopen("../poligono.txt", "r");
14     if (arquivo == NULL)
15     {
16         printf("Erro ao abrir o arquivo.\n");
17         return 1;
18     }
19
20     if (fscanf(arquivo, "%d", &numVertices) != 1)
21     {
22         printf("Nao foi possivel ler a quantidade de vertices.\n");
23         fclose(arquivo);
24         return 1;
25     }
26
27     Ponto coordenadas[numVertices];
28
29     for (int i = 0; i < numVertices; i++)
30     {
31         if (fscanf(arquivo, "%f %f", &x, &y) != 2)
32         {
33             printf("Nao foi possivel ler as coordenadas %d.\n", i + 1);
34             fclose(arquivo);
35             return 1;
36         }
37         coordenadas[i] = criarPonto(x, y);
38     }
39
40     area = calcularAreaPoligono(coordenadas, numVertices);
41     printf("A area do poligono eh: %.2f\n", area);
42
43     fclose(arquivo);
44
45     return 0;
46 }
```

4. Conclusão

O programa funcionou perfeitamente de acordo com o que foi pedido no enunciado.

Referências

<https://www.todamateria.com.br/matrizes-e-determinantes/>

<https://www.infoescola.com/matematica/area-de-poligonos-irregulares/>

<https://www.omnicalculator.com/math/irregular-polygon-area>

Anexos

ponto.h

ponto.c

principal.c

poligono.txt