

SME0230 - Introdução à Programação de Computadores

Primeiro semestre de 2020

Professoras: Marina Andretta (andretta@icmc.usp.br)

Estagiários PAE: Filomen Incahuanaco (fincahuanaco@usp.br) e
Germain García Zanabria (germaingarcia@usp.br)

Monitores: Gabriel Dalorno Silvestre (gdalorno7@usp.br)

Exercício 18 - Área de um Polígono

1 Descrição

Um polígono é uma figura geométrica bidimensional fechada e delimitada por segmentos de reta (Fig.1). Uma forma de representar polígonos é listar as coordenadas x e y de seus vértices em sentido anti-horário.

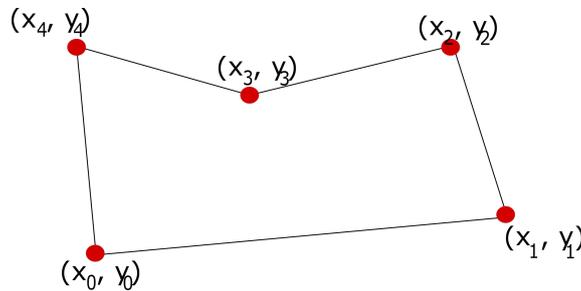


Figure 1: Exemplo de polígono.

Para calcular a área A de um polígono de vértices $(x_0, y_0), (x_1, y_1), \dots, (x_{N-1}, y_{N-1})$, existe a seguinte fórmula:

$$A = 1/2(x_{N-1}y_0 - x_0y_{N-1}) + 1/2 \sum_{i=0}^{N-2} (x_i y_{i+1} - x_{i+1} y_i). \quad (1)$$

Para armazenar os dados de um polígono no computador, é possível usar um registro do tipo *poligono*, que contém seu número N de vértices, bem como a lista de vértices. Cada vértice pode ser representado por um registro do tipo *ponto*. Esses registros são dados por:

```
//struct ponto e poligono sao obrigatorias
struct ponto{
    float x; //coordenada x
    float y; //coordenada y
};

struct poligono{
    int N; //numero de vertices
    struct ponto *vertices; //ponteiro para o vetor de vertices
};
```

Escreva um programa em C que leia os dados de um polígono e determine sua área. Para representar o polígono, use os registros *poligono* e *ponto* definidos acima. O campo *vertices* do registro *poligono*, que deve receber um vetor com as coordenadas dos vértices do polígono no sentido anti-horário, deve ser alocado dinamicamente, exatamente do tamanho necessário.

2 Entrada

Na primeira linha haverá um inteiro N representando o número de vértices do polígono. Nas N linhas seguintes haverá as coordenadas x e y dos vértices do polígono no sentido anti-horário.

3 Saída

O programa deverá mostrar a área do polígono de entrada, usando duas casas decimais.

4 Exemplos

Entrada

```
4
5 0
5 3
1 3
1 0
```

Saída

```
12.00
```

Entrada

```
5
6 5
-5 2
-3 -1
4 -3
6 5
```

Saída

```
49.50
```

5 Observações

- **Formato da saída:** Se atente para o formato da saída! O Run Codes só considerará correta a saída do seu programa se estiver **idêntica** à saída esperada.
- **Forma de entrega:** Os exercícios deverão ser entregues pelo Run Codes (<https://run.codes>). Código de matrícula da disciplina: **4Z51**
- **Plágio:** Esse é um exercício individual. Códigos iguais (ou **muito** parecidos) receberão nota 0.
- **Nota do Run Codes:** Essa nota corresponde à quantidade de casos de teste que seu programa foi capaz de responder corretamente, e não à sua nota final neste exercício! Esta será calculada com base na estruturação do código, inclusão de comentários, abordagem para resolução do exercício, etc.