

Exercício de Programação: Gerador de Polígono

Uma forma muito básica de representar um polígono irregular é armazenar os vértices que ele tem como pontos (x,y) no plano cartesiano e, em seguida, usar um vetor desses pontos para representar as arestas entre eles que formam desse modo o polígono em questão.

Com isso em mente, você foi contratado por uma empresa de software de design gráfico para desenvolver uma funcionalidade que permita aos usuários criar polígonos personalizados. Para isso, você precisa escrever um programa em C/C++ que faça o seguinte:

1. Solicite ao usuário que insira o número de vértices do polígono.
2. Para cada vértice, solicite ao usuário que insira as coordenadas x e y no plano cartesiano.
3. Após todos os vértices serem inseridos, o programa deve imprimir o polígono gerado, listando as coordenadas de cada vértice.

Para representar um polígono, você deve usar duas estruturas: Ponto e Poligono. A estrutura Ponto deve ter dois campos para armazenar as coordenadas x e y . A estrutura Poligono deve ter um campo para armazenar o número de vértices e um ponteiro para um vetor de Ponto que será alocado dinamicamente.

Requisitos:

- O programa deve ser escrito em C/C++ como visto em sala, seguindo a estrutura em anexo:
 - <https://github.com/diogots/Curso-AlgoritmosEstruturasdeDados/blob/main/Atividade/template.cpp>
- O programa deve usar as estruturas Ponto e Poligono para representar computacionalmente o polígono irregular.
- O programa deve solicitar ao usuário que insira o número de vértices e as coordenadas de cada vértice.
- O programa deve imprimir o polígono gerado.

Dicas:

- Lembre-se de alocar dinamicamente o vetor de pontos e caso queira, de liberar a memória quando ela não for mais necessária.
- Considere que o usuário entra o dados de forma correta ("usuário atípico")
- Um vídeo com a execução de uma possível implementação está exibido abaixo:

Solução:

```
#include <iostream>
#include <locale.h>
using namespace std;

struct ponto{
    double x;
    double y;
};

typedef struct ponto Ponto;

struct poligono{
    int nVertices;
    Ponto *vertices;
};

typedef struct poligono Poligono;

int main()
{
    //setando o locale para pt-br
    setlocale(LC_ALL, "");
    cout << "Gerador de Polígono" << endl;

    Poligono p;
    //lendo o número de vértices
    cout << "Digite o número de vértices do polígono:" <<
endl;
    cin >> p.nVertices;
    //lendo os vértices do polígono
    cout << "\nRealizando a leitura de um polígono de " <<
p.nVertices << " vertices:" << endl;
    //alocando o vetor de pontos
    p.vertices = new Ponto[p.nVertices];
    for (int i=0; i<p.nVertices; i++){
        cout << "Lendo dados do vertice " << (i+1) << ":" <<
endl;
        cout << "X: " ;
        cin >> p.vertices[i].x;
        cout << "Y: " ;
        cin >> p.vertices[i].y;
    }

    //imprimindo o polígono:
    cout << "\nPolígono gerado:" << endl;
    cout << "Polígono de " << p.nVertices << " vértices:" <<
endl;
    for (int i=0; i<p.nVertices; i++){
```

```
        cout << "(X,Y):(" << p.vertices[i].x << ",";
        cout << p.vertices[i].y << ")" << endl;
    }

    return 0;
}
```