

Lab 4: Extending Geometric Shapes

Este lab deve ser feito individualmente.

Submeta o seu código ao mooshak <http://deei-mooshak.ualg.pt/~jvo/> usando o seu login da ualg (sem @ualg.pt). Ex: a123456

Uma submissão ao problema permanecerá *pending* até que seja validada pelo professor durante a aula prática. Só as submissões *final* serão consideradas para avaliação.

Todas as submissões deverão ser feitas até

22 de março 2024

A validação poderá ser feita posteriormente, se necessário, até 12 de abril de 2024

NB: Nos problemas seguintes, sempre que necessário, considera-se que dois double d e g são iguais se $|d-g| < 1e-9$

Neste laboratório consideram-se os casos particulares de polígonos: triângulos, retângulos e quadrados, mantendo-se as seguintes condições dos exercícios anteriores:

- os polígonos são definidos por uma lista ordenada de pontos (vértices), em número igual ou superior a 3;
- as coordenadas dos vértices são números inteiros pertencentes ao primeiro quadrante;
- 3 pontos seguidos não podem ser colineares
- a última aresta do polígono é composta pela ligação entre o último e o primeiro vértice
- nenhum par de arestas se pode intersestar
- os polígonos são imutáveis

Restrições complementares

Os polígonos são objetos imutáveis.

Em todos os problemas deve seguir os princípios da programação orientada por objetos, estruturando o código segundo um conjunto de classes apropriadas. Para cada um dos problemas seguintes, antes de escrever qualquer código, defina um conjunto de testes unitários que reflitam o comportamento pretendido.

Comece por esboçar um diagrama de classes inicial (de análise) UML com as classes necessárias para implementar os problemas descritos.

Adicionalmente, deve comentar o código, indicando

- i) Uma linha com a descrição da responsabilidade da classe ou do que o método faz
- ii) @author (apenas para classes)
- iii) @version (apenas para classes; inclua uma data)

- iv) `@inv` (apenas para classes; inclua uma descrição da invariante usada)
- v) `@param` (apenas para métodos e construtores)
- vi) `@return` (apenas para métodos)
- vii) `@see` (qualquer referência bibliográfica ou sitio web consultado para o desenvolvimento do código respetivo)

Em todos os exercícios, o programa deve seguir a estrutura da seguinte classe Main (deve adaptar aos requisitos dos exercícios e submeter sua versão no Mooshak).

```
public class Main {  
  
    public static String capital(String s) {  
        if (s == null || s.isEmpty())  
            return s;  
        return s.substring(0, 1).toUpperCase() + s.substring(1).toLowerCase();  
    }  
  
    public static void main(String[] args) throws Exception {  
        Scanner sc = new Scanner(System.in);  
        Constructor<?> constructor;  
        Class<?> cl;  
        Poligono p;  
        String s;  
        String[] aos;  
        while (sc.hasNextLine()) {  
            s = sc.nextLine();  
            if (s.isEmpty())  
                break;  
            aos = s.split(" ", 2);  
            try {  
                cl = Class.forName(capital(aos[0]));  
                constructor = cl.getConstructor(String.class);  
                p = (Poligono) constructor.newInstance(aos[1]);  
  
                } catch (ClassNotFoundException cnfe) {  
                    System.out.println("Não foi encontrada a classe: " + cnfe.get-  
Message());  
                } catch (Exception e) {  
                    e.printStackTrace();  
                }  
            }  
            sc.close();  
        }  
    }  
}
```

Problema H: toString()

Pretende-se detalhar a representação em *string* de cada polígono através do método *toString()*. Assim, dependendo do tipo de polígono, o método *toString()* deve devolver as seguintes *strings*:

Poligono: N vértices [(x₁,y₁),... (x_N,y_N)]

Retangulo: [(x₁,y₁), (x₂,y₂), (x₃,y₃), (x₄,y₄)]

Quadrado: [(x₁,y₁), (x₂,y₂), (x₃,y₃), (x₄,y₄)]

Triangulo: [(x₁,y₁), (x₂,y₂), (x₃,y₃)]

Desenvolva um programa que recebe um conjunto de polígonos, armazena-os numa única estrutura de dados, garanta que não existem polígonos duplicados e imprime-os usando o método *toString()* acima especificado.

Entrada

A entrada do programa corresponde a um conjunto de linhas. Cada linha tem o nome da classe que representa a figura geométrica (Polígono, Retângulo, Quadrado ou Triângulo). Se for um polígono genérico, a linha tem o número natural que indica o número de vértices, seguida das coordenadas *x* e *y* de cada vértice, sempre separados por espaços.

Saída

Uma linha com uma das opções seguintes:

- i) Ponto:vi: indicação de que os pontos não pertencem ao primeiro quadrante;
- ii) “Reta:vi”; indicação de que dois pontos não formam uma reta;
- iii) “Polígono:vi”; indicação de que os *n* pontos não formam um polígono;
- iv) “Triângulo:vi”; indicação de que os pontos não formam um triângulo;
- v) “Retângulo:vi”; indicação de que os pontos não formam um retângulo;
- vi) “Quadrado:vi”; indicação de que os pontos não formam um quadrado;
- vii) “Duplicado”; indicação de que foram introduzidos polígonos duplicados;
- viii) Impressão do conjunto de polígonos, um por linha, conforme a definição indicada na especificação do problema.

Exemplo de Entrada 1

```
Poligono 4 5 5 8 6 8 7 5 7
Triangulo 7 1 9 1 9 3
```

Exemplo de Saída 1

```
Poligono de 4 vértices: [(5,5), (8,6), (8,7), (5,7)]
Triangulo: [(7,1), (9,1), (9,3)]
```

Exemplo de Entrada 2

```
Quadrado 2 2 4 6 7 6 10 4
Triangulo 5 5 8 5 8 7
```

Exemplo de Saída 2

```
Retangulo:vi
```

Exemplo de Entrada 3

```
Poligono 3 9 3 7 1 9 1
Triangulo 7 1 9 1 9 3
```

Exemplo de Saída 3

```
Duplicado
```

Exemplo de Entrada 3

Retangulo 3 0 4 2 3 4 2 2

Exemplo de Saída 3

Retangulo:vi

Problema I: Movendo os polígonos: rotação

Pretende-se aplicar movimentos de rotação a cada polígono. Para esse efeito, desenvolva um método que tem como parâmetros, o ângulo de rotação (definido em graus, i.e., uma rotação completa corresponde a 360 graus) e um ponto de rotação. Deve ser possível executar uma rotação sem definir o ponto de rotação. Neste caso, a rotação deve ser feita em torno do *centroide* (centro geométrico) do polígono.

Entrada

Tal como no exercício anterior, a entrada começa com uma linha que tem o nome da classe que representa a figura geométrica (Poligono, Retangulo, Quadrado ou Triangulo). Se for um polígono genérico, a linha tem o número natural que indica o número de vértices, seguida das coordenadas x e y de cada vértice, sempre separados por espaços.

A linha seguinte indica o ângulo de rotação em graus e, possivelmente, as coordenadas x e y do ponto de rotação (ponto pivot).

Saída

Uma linha com uma das opções seguintes:

- i) Ponto:vi: indicação de que os pontos não pertencem ao primeiro quadrante;
- ii) "Reta:vi"; indicação de que dois pontos não formam uma reta;
- iii) "Poligono:vi"; indicação de que os n pontos não formam um polígono;
- iv) "Triangulo:vi"; indicação de que os pontos não formam um triângulo;
- v) "Retangulo:vi"; indicação de que os pontos não formam um retângulo;
- vi) "Quadrado:vi"; indicação de que os pontos não formam um quadrado;
- vii) "Duplicado"; indicação de que foram introduzidos polígonos duplicados;
- viii) Impressão do polígono, após aplicada a rotação definida.

Exemplo de Entrada 1

Retangulo 1 1 3 1 3 5 1 5
90

Exemplo de Saída 1

Retangulo: [(4,2), (4,4), (0,4), (0,2)]

Exemplo de Entrada 2

Retangulo 1 1 3 1 3 5 1 5
-90 3 1

Exemplo de Saída 2

Retangulo: [(3,3), (3,1), (7,1), (7,3)]

Exemplo de Entrada 3

Triangulo 2 1 4 1 3 4
180

Exemplo de Saída 3

Triangulo: [(4,3), (2,3), (3,0)]

Problema J: Movendo os polígonos: translação

Pretende-se aplicar movimentos de translação a cada polígono. Para esse efeito, desenvolva um método que tem como parâmetros, o deslocamento x d_x e o deslocamento y d_y a aplicar ao polígono, significando que o polígono se deve movimentar em x e em y pelas quantidades definidas em d_x e em d_y .

Entrada

Tal como nos exercícios anteriores, a entrada começa com uma linha que tem o nome da classe que representa a figura geométrica (Poligono, Retangulo, Quadrado ou Triangulo). Se for um polígono genérico, a linha tem o número natural que indica o número de vértices, seguida das coordenadas x e y de cada vértice, sempre separados por espaços.

A linha seguinte tem dois valores inteiros que correspondem aos deslocamentos d_x e d_y .

Saída

Uma linha com uma das opções seguintes:

- i) Ponto:vi: indicação de que os pontos não pertencem ao primeiro quadrante;
- ii) “Reta:vi”; indicação de que dois pontos não formam uma reta;
- iii) “Poligono:vi”; indicação de que os n pontos não formam um polígono;
- iv) “Triangulo:vi”; indicação de que os pontos não formam um triângulo;
- v) “Retangulo:vi”; indicação de que os pontos não formam um retângulo;
- vi) “Quadrado:vi”; indicação de que os pontos não formam um quadrado;
- vii) “Duplicado”; indicação de que foram introduzidos polígonos duplicados;
- viii) Impressão do polígono, após aplicada a translação definida.

Exemplo de Entrada 1

Poligono 4 1 2 5 6 8 7 12 14
-1 3

Exemplo de Saída 1

Poligono de 4 vértices: [(0,5), (4,9), (7,10), (11,17)]

Exemplo de Entrada 2

Poligono 4 1 2 5 6 8 7 12 14
-5 1

Exemplo de Saída 2

Ponto:vi

Problema K: Movendo os polígonos: translação baseada no centroide

Pretende-se aplicar movimentos de translação a cada polígono. Para esse efeito, desenvolva um método que tem como parâmetros, as coordenadas de um ponto que deve ser o novo centroide do polígono, i.e., o polígono deve mover-se de forma a que essas sejam as coordenadas do novo centroide de polígono.

Entrada

Tal como nos exercícios anteriores, a entrada começa com uma linha que tem o nome da classe que representa a figura geométrica (Poligono, Retangulo, Quadrado ou Triangulo). Se for um polígono genérico, a linha tem o número natural que indica o número de vértices, seguida das coordenadas x e y de cada vértice, sempre separados por espaços.

A linha seguinte tem dois valores inteiros que correspondem às coordenadas x e y do centroide.

Saída

Uma linha com uma das opções seguintes:

- i) Ponto:vi: indicação de que os pontos não pertencem ao primeiro quadrante;
- ii) “Reta:vi”; indicação de que dois pontos não formam uma reta;
- iii) “Poligono:vi”; indicação de que os n pontos não formam um polígono;
- iv) “Triangulo:vi”; indicação de que os pontos não formam um triângulo;
- v) “Retangulo:vi”; indicação de que os pontos não formam um retângulo;
- vi) “Quadrado:vi”; indicação de que os pontos não formam um quadrado;
- vii) “Duplicado”; indicação de que foram introduzidos polígonos duplicados;
- viii) Impressão do polígono, após aplicada a translação definida.

Exemplo de Entrada 1

```
Retangulo 1 1 5 1 5 3 1 3
3 3
```

Exemplo de Saída 1

```
Retangulo: [(1,2), (5,2), (5,4), (1,4)]
```

Exemplo de Entrada 2

```
Triangulo 1 0 3 0 2 3
2 3
```

Exemplo de Saída 2

```
Triangulo: [(1,2), (3,2), (2,5)]
```