



Universitário Maurício de Nassau
Curso de Sistemas de Informação
Curso de Superior de Tecnologia em Análise e Desenvolvimento de Sistemas
Prof. Dr. Diogo Rodrigues
010117368@prof.uninassau.edu.br

2º Atividade Programação Orientada a Objetos

Instruções:

O arquivo deve ser nomeado da seguinte forma: **ATV2_nome completo** ex.: **ATV2_Diogo Francisco Borba Rodrigues.zip** ou **ATV1_Diogo Francisco Borba Rodrigues** no github.

Os pontos acima devem ser atendidos para que a atividade seja avaliada sem nenhum risco de penalização.

OBSERVAÇÃO: as classes abaixo não apresentam métodos iguais, porém vocês precisam defini-los em todas elas, considerando que para resolver as questões vocês precisarão usá-los.

- 1) Crie uma classe chamada MyPoint para representar um ponto e se apresenta na seguinte forma (em UML):

MyPoint
-x:int = 0 -y:int = 0
+MyPoint() +MyPoint(x:int, y:int) +getX():int +setX(x:int):void +getY():int +setY(y:int):void +setXY(x:int, y:int):void +toString():String +distance(x:int, y:int):double +distance(another:MyPoint):double

Obs.: o método toString retorna o formato do "MyPoint (x, y)" a) Defina uma classe TestMyPoint que contenha um método main e teste todos os métodos e construtores da classe definida. Não precisa receber valores do teclado.

- 2) Crie uma classe chamada MyCircle seguindo a descrição em UML e reusando a definição da classe MyPoint:

MyCircle
-center:MyPoint -radius:int = 1
+MyCircle(x:int, y:int, radius:int) +MyCircle(center:MyPoint, radius:int) +getRadius():int +setRadius(radius:int):void +getCenter():MyPoint +setCenter(center:MyPoint):void +getCenterX():int +getCenterY():int +setCenterXY(x:int, y:int):void +toString():String +getArea():double

Obs.: o método toString retorna o formato do "Circle @ (x, y) radius=r; area=a" a) Defina uma classe TestMyCircle que contenha um método main e teste todos os métodos da classe MyCircle definida. Não precisa receber valores do teclado