



Universidade Federal do Pará
Faculdade de Computação – ICEN
Curso: Bacharelado em Sistemas de Informação
EN05170 - Programação de Computadores II
Prof. Lidio Campos 2023.2
LISTA DE EXERCÍCIOS 3

1-Enunciado: Definir uma classe **Circulo** com os atributos **raio**, **coordenada x** e **y** do centro do **Circulo** , criar uma classe **Principal.java** que instancie a classe **circulo** imprimir as coordenadas **Raio**, **x** e **y** do **circulo**, após isso deslocar a coordenada **x** de 17 unidades e imprimir novamente as coordenadas. Fazer o **Diagrama de Classes UML**, executar o código.

2-Altere a classe **circulo** do exercício anterior criando um método **inicializa publico** e que seja **void** cuja finalidade e **inicializa** as variáveis de instancia **x** , **y** e **raio**, adicionar um segundo método **altera_raio publico** e sem valor de retorno que altera o **raio** do **circulo** e um terceiro método **retorna_raio** que retorna o **raio** do **circulo**. Criar uma classe **Principal** que instancie a classe **Circulo** e **inicializa** as coordenadas do **circulo** **inicializa(0,0,10)** mostra as coordenadas **x,y** e **raio** do **circulo** criado, move o **circulo** para a coordenada **(1,1)**, altera a coordenada **x** do **circulo** para **100**, mostra as novas coordenadas depois altera o **raio** do **circulo** para **12**.

3)Crie uma classe **Trava** que contenha dois atributos “**quem**” do tipo **String** e “**travado**” do tipo **boolean**, a classe deve conter um método **trave** que recebe um argumento do tipo **String** e altera o status da variável de instancia **travado** para **true** e a variável de instancia **quem** com o valor passado no método, esse método deve ser público e não retorna nenhum valor. Além disso, a classe deve conter um método **destrave publico** e sem valor de retorno que recebe um parâmetro do tipo **String** e altera e altera o status da variável de instancia **travado** para **false** e a variável de instancia **quem** com o valor passado no método. Um terceiro método público que retorna um **boolean** deve retornar o estado do objeto **travado**

4-Implementar uma classe denominada **Histograma** que fornece um método para ler uma série de caracteres (**A,B,C,D,E,F**) fornecidos por um usuário e conta o número de letras fornecidas. Também criar um método que imprima o **histograma** coma distribuição das letras. Podem ser utilizadas na implementação da classe declarações **switch** e **for**.

5-Enunciado: O Programa a seguir exemplifica chamadas a métodos, para tal define um objeto que serve como contador, a implementação representa a contagem no atributo **num** que é um número inteiro. Os métodos são simples: **incrementa** adiciona um ao contador em qualquer estado e começa **inicializa** a contagem em zero. **Decrementa** faz o oposto de **incrementa**.



Universidade Federal do Pará
Faculdade de Computação – ICEN
Curso: Bacharelado em Sistemas de Informação
EN05170 - Programação de Computadores II
Prof. Lidio Campos 2023.2
LISTA DE EXERCÍCIOS 3

6-Escreva em Java a classe NumeroComplexo que represente um número complexo. A classe deverá ter os seguintes métodos:

- **inicializaNumero**, que recebe dois valores como argumentos para inicializar os campos da classe (parte real e imaginária);
- **imprimeNumero**, que deve imprimir o número complexo encapsulado usando a notação $a+bi$ onde a é a parte real e b a imaginária;
- **eIgual**, que recebe outra instância da classe NumeroComplexo e retorna true se os valores dos campos encapsulados forem iguais aos da instância passada como argumento;
- **soma**, que recebe outra instância da classe NumeroComplexo e soma este número complexo com o encapsulado usando a fórmula $(a+bi)+(c+di) = (a+c)+(b+d)i$;
- **subtrai**, que recebe outra instância da classe NumeroComplexo e subtrai o argumento do número complexo encapsulado usando a fórmula $(a+bi)-(c+di) = (a-c)+(b-d)i$;
- **multiplica**, que recebe outra instância da classe NumeroComplexo e multiplica este número complexo com o encapsulado usando a fórmula $(a+bi)*(c+di) = (ac-bd)+(ad+bc)i$;
- **divide**, que recebe outra instância da classe NumeroComplexo e divide o número encapsulado pelo passado como argumento usando a fórmula $(a+bi)/(c+di) = (ac+bd)/(c^2+d^2) + ((bc-ad)/(c^2+d^2)) * i$;

7)Crie uma classe Ponto com os atributos x e y do tipo float, a classe deve possuir um método construtor que inicializa as variáveis de instância, criar um método move que recebe deslocamentos dx e dy do tipo float e desloca o ponto no plano cartesiano, além disso, criar o método mostra que imprime as coordenadas do ponto. Criar outra classe principal instanciar e desloca-lo no plano cartesiano.

8)Crie uma classe Reta que contenha dois atributos a e b do tipo Ponto, o construtor da classe Reta deve ter o seguinte protótipo public Reta(float ax,float ay,float bx,float by) esse método deve instanciar dois objetos Ponto $a=new Ponto(ax,ay)$, $b=new Ponto(bx,by)$, a classe Reta dever possuir um método mostra que chama o método mostra da classe Ponto e mostra as coordenadas da reta.



Universidade Federal do Pará
Faculdade de Computação – ICEN
Curso: Bacharelado em Sistemas de Informação
EN05170 - Programação de Computadores II
Prof. Lidio Campos 2023.2
LISTA DE EXERCÍCIOS 3

9) Considere o seguinte código:

```
for (int i=0 ; i<2 ; i++)  
    for(int j=0 ; j<3; j++)  
    {  
        if (i== j)  
        {  
            continue;  
        }  
        System.out.println("i="+i+"j="+j);  
    }
```

Quais as saídas apresentadas pelo mesmo?

10-Estudo de Caso

- **Evolução do Estudo de Caso 3:**
- **Altere a classe Banco para trabalhar com ArrayList ao invés de Arrays e armazene instâncias de Contas;**
- **A classe Banco deverá ter os seguintes métodos:**
 - **adicionarConta(ContaCorrente c) // Adicionar uma nova conta**
 - **ContaCorrente obterConta(String numConta) // Recupera uma conta**
 - **removerConta(ContaCorrente c) // Remove uma conta**
 - **exibirContas() // Lista as contas na tela**
- **Teste-a declarando um método main que recebe dados do teclado e realiza a operação correspondente.**