



Fundação CECIERJ - Vice Presidência de Educação Superior a Distância

Curso de Tecnologia em Sistemas de Computação

Disciplina: Programação III

AP2 2º semestre de 2012.

Nome –

Assinatura –

Observações:

1. Prova sem consulta e sem uso de máquina de calcular.
 2. Use caneta para preencher o seu nome e assinar nas folhas de questões e nas folhas de respostas.
 3. Você pode usar lápis para responder as questões.
 4. Ao final da prova devolva as folhas de questões e as de respostas.
 5. Todas as respostas devem ser transcritas nas folhas de respostas. As respostas nas folhas de questões não serão corrigidas.
-

Questão 1) (3 pontos)

Escreva um programa que receba o nome de um arquivo e um número N , leia o arquivo de entrada **SOMENTE UMA VEZ** (este arquivo é composto de números inteiros) e escreva, num arquivo de saída, chamado de “saida” acrescido do nome do arquivo de entrada, todos os inteiros na mesma ordem do arquivo de entrada, trocando todas as ocorrências de N por $N - 1$ e $N + 1$. Um exemplo desta execução seria (SEU PROGRAMA DEVE FUNCIONAR PARA TODOS OS ARQUIVOS PASSADOS COMO PARÂMETRO DE ENTRADA):

a) se o arquivo de entrada **valores.txt** for:

```
1
2
1
3
1
```

b) se o comando for **java Q1_AP2_2012_2 valores.txt 1**, o arquivo de saída **saida-valores.txt** será composto de:

```
0
2
2
0
2
3
0
2
```

RESPOSTA:

```
import java.io.*;

class no{
    int info;
    no prox;

    no(int info){
        this.info = info;
        prox = null;
    }

    public String toString(){ return info + "\n"; }
}

class lista{
    no prim;

    lista(){ prim = null; }

    void insere(int info){
        no novo = new no(info);
        if(prim == null)
            prim = novo;
        else{
            no p = prim;
            while(p.prox != null) p = p.prox;
            p.prox = novo;
        }
    }

    public String toString(){
        String resp = "";
        no p = prim;
        while(p != null){
            resp += p.toString();
            p = p.prox;
        }
        return resp;
    }

    void altera(int elem){
        no p = prim;
        while(p != null){
            while((p != null) && (p.info != elem)) p = p.prox;
            if(p != null){
                p.info = elem - 1;
                no novo = new no(elem + 1);
                novo.prox = p.prox;
                p.prox = novo;
                p = p.prox;
            }
        }
    }
}
```

```

public class Q1_AP2_2012_2{
    public static void main(String[] args) throws IOException {
        BufferedReader in = new BufferedReader(new FileReader(args[0]));
        String s;
        lista l = new lista();

        try{
            while ((s = in.readLine()) != null)
l.insere(Integer.parseInt(s));
        }catch (Exception e){
            System.out.println("Excecao leitura\n");
        }finally{
            in.close();
        }

        BufferedWriter out = new BufferedWriter(new FileWriter("saida-" +
args[0]));
        int elem = Integer.parseInt(args[1]);

        l.altera(elem);
        try{
            out.write(l.toString());
        }catch (Exception e){
            System.out.println("Excecao de escrita\n");
        }finally{
            out.close();
        }
    }
}

```

Questão 2) (3 pontos)

Escreva um programa que receba, como parâmetro de entrada, um arquivo, contendo as dimensões de uma matriz e os valores de cada uma das posições. Seu programa deve ler esta matriz uma única vez e verificar se a matriz é simétrica. Se a matriz for simétrica, ele deve imprimir, num arquivo de saída, chamado de “saida” acrescido do nome do arquivo de entrada, a matriz triangular inferior. Senão, a matriz do arquivo de saída será igual a matriz do arquivo de entrada. Um exemplo desta execução seria (SEU PROGRAMA DEVE FUNCIONAR PARA TODOS OS ARQUIVOS PASSADOS COMO PARÂMETRO DE ENTRADA):

a) se o arquivo de entrada **matriz.txt** for:

```

2
2
1
2
2
1

```

b) este arquivo representa a matriz simétrica:

```

1 2
2 1

```

b) se o comando for `java Q2_AP2_2012_2 matriz.txt`, o arquivo de saída `saida-matriz.txt` será composto de:

```
1
1
2
```

RESPOSTA:

```
import java.io.*;

public class Q2_AP2_2012_2{
    public static void main(String[] args) throws IOException {
        BufferedReader in = new BufferedReader(new FileReader(args[0]));
        BufferedWriter out = new BufferedWriter(new FileWriter("saida-" +
args[0]));
        String s;
        int l, c, i, j, sim = 1;
        int mat[][];

        try{
            s = in.readLine();
            l = Integer.parseInt(s);
            s = in.readLine();
            c = Integer.parseInt(s);

            mat = new int[l][c];
            for(i = 0; i < l; i++)
                for(j = 0; j < c; j++){
                    s = in.readLine();
                    mat[i][j] = Integer.parseInt(s);
                }

            if(l != c) sim = 0;
            else{
                for(i = 0; i < l; i++)
                    for(j = 0; j <= i; j++){
                        if(mat[i][j] != mat[j][i]){
                            sim = 0;
                            break;
                        }
                    }
            }

            out.write(imprimeMatriz(mat, l, c, sim));
        }catch (Exception e){
            System.out.println("Excecao\n");
        }finally{
            in.close();
            out.close();
        }
    }

    public static String imprimeMatriz(int mat[][], int l, int c, int
sim){
        int i,j;
        String resp = "";
```

```

        if(sim == 1){
            for(i = 0; i < l; i++)
                for(j = 0; j <= i; j++)
                    resp = resp + mat[i][j] + "\n";
        }
        else{
            for(i = 0; i < l; i++)
                for(j = 0; j < c; j++)
                    resp = resp + mat[i][j] + "\n";
        }
        return resp;
    }
}

```

Questão 3) (4 pontos)

Considere as classes abaixo, as quais modelam figuras geométricas.

```

class Quadrilatero {
    private double lado1, lado2, lado3, lado4;
    public Quadrilatero(double l1, double l2, double l3, double l4) {
        lado1 = l1; lado2 = l2; lado3 = l3; lado4 = l4;
    }
}

class Retangulo extends Quadrilatero {
    public Retangulo(double b, double h) {
        super(b, h, b, h);
    }
    public void exibe() {
        System.out.println("Retangulo com lados " + lado1 + " e " +
lado2);
    }
}

```

- Crie classes para modelar quadrados e círculos.
- Modifique as classes acima de forma que possamos, para cada objeto de uma classe criado, calcular seu perímetro (recorde que o perímetro de um quadrilátero, quadrado, retângulo, etc., é dado pela soma de seus lados, enquanto que o perímetro de um círculo é o dobro do seu raio multiplicado por PI – aproximadamente 3,14). Dica: Utilize a constante `java.lang.Math.PI`
- Forneça uma maneira de obrigarmos que novas classes a serem inseridas na hierarquia (novos objetos geométricos) sejam obrigadas a definir uma forma de calcular o perímetro de seus objetos.
- Declare uma classe que contenha um método `main()` (sintaxe aproximada, apenas), a qual declare e instancie uma coleção de objetos dentro deste método. Insira nesta coleção 1 instância exemplo, com qualquer valor para os seus atributos, **para cada classe criada**, percorra esta coleção somando os perímetros de todos os objetos desta coleção e exiba esta soma.

Obs: 1) Sempre que possível, utilize os conceitos de orientação a objetos vistos; 2) A solução da questão pode ser apresentada como um programa completo

RESPOSTA:

```
import java.util.ArrayList;
import java.util.List;

interface IObjetoGeom {
    double perimetro();
}

class Quadrad extends Retangul {
    public Quadrad(double l) {
        super(l, l);
    }
}

class Circul implements IObjetoGeom {
    protected double raio;
    public Circul(double raio) {
        this.raio = raio;
    }
    public double perimetro() {
        return 2*raio*Math.PI;
    }
}

public class AP3_2012_2_Q3 {
    public static void main(String[] args) {
        List<IObjetoGeom> figuras = new ArrayList<IObjetoGeom>();
        figuras.add(new Quadrilater(2, 2, 4, 4));
        figuras.add(new Retangul(3, 5));
        figuras.add(new Quadrad(7));
        figuras.add(new Circul(6));
        double soma = 0;
        for (IObjetoGeom f : figuras) {
            soma = soma + f.perimetro();
        }
        System.out.println("A soma dos perimetros dos objetos e: " + soma);
    }
}
```