



Fundação CECIERJ - Vice Presidência de Educação Superior a Distância

Curso de Tecnologia em Sistemas de Computação

Disciplina: Programação III

AP1 1º semestre de 2009.

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) (2.5 pontos)

Implemente um programa que receba, como parâmetro de entrada, um número **n** e retorne, num vetor, os **n** primeiros números primos existentes.

RESPOSTA:

```
class primo{
```

```
    public static void main (String[] args){
        int n = Integer.parseInt(args[0]);
        int vet[] = new int[n];
        int i = 1, primo;

        vet[0] = 2;
        while (i < n){
            primo = vet[i - 1];
            primo++;
            while (!Testa_Primeiro (primo)) primo++;
            vet[i] = primo;
            i++;
        }

        for(i = 0; i < n; i++)
            System.out.println(vet[i]);
    }
}
```

```

public static boolean Testa_Primeiro (int primo){
    for(int i = 2; i <= Math.sqrt (primo); i++)
        if (primo % i == 0) return false;

    return true;
}
}

```

Questão 2) (2.5 pontos)

Implemente um programa que permita a avaliação de polinômios. Cada polinômio é definido por um vetor que contém seus coeficientes. Por exemplo, o polinômio de grau dois $3x^2 + 2x + 12$ terá um vetor de coeficientes $v = \{12, 2, 3\}$. O programa deve receber o valor de x e deve retornar o valor da avaliação do polinômio que for definido no programa principal.

RESPOSTA:

```

class polinomio{

    public static void main (String[] args){
        float x = Float.parseFloat(args[0]);
        float vet[] = {10, -9, 8, -7, 6, -5, 4, -3, 2, -1};
        System.out.println(Avalia(x,vet));

    }

    public static float Avalia (float x, float vet[]){
        float result = vet[0], mult = 1;

        for(int i = 1; i < vet.length; i++){
            mult *= x;
            result += vet[i] * mult;
        }

        return result;
    }
}

```

Questão 3) (2.5 pontos)

Considere o código abaixo que modela objetos de um jogo de guerra:

```

class ObjetoVisual {

    float vida; // Varia entre 0, .25, .5, 1

    int pontuacao; // + 1 para cada acerto

    int pos_x; // Posicao no eixo x
}

```

```

    int pos_y; // Posicao no eixo y

    float largura; // Largura do objeto

    float altura; // Altura do objeto

    void desenha();
}

class Aviao extends ObjetoVisual {

    void movimenta (int x, int y) {

        pos_x = x;

        pos_y = y;

    }

}

```

a) (0.5 pontos) Que mudança(s) precisamos fazer no código para que a classe *Avião* seja obrigada a redefinir o método *desenha()* da classe *ObjetoVisual*?

RESPOSTA: Precisamos definir que a classe *ObjetoVisual* é abstrata (*abstract*), assim como o método *desenha()*. Dessa forma, dado que a classe *Aviao* herda da classe *ObjetoVisual*, esta é obrigada a apresentar uma implementação do método abstrato *desenha()*.

b) (0.5 pontos) Para as 2 classes listadas, defina construtores que recebam parâmetros que inicializem os objetos das respectivas classes.

RESPOSTA:

```

abstract class ObjetoVisual {
    ...
    public ObjetoVisual(float v,
                        int pont,
                        int x,
                        int y,
                        float l,
                        float a) {

        vida = v;
        pontuacao = pont;
        pos_x = x;
        pos_y = y;
        largura = l;
        altura = a;

    }
    ...
}

```

```

class Aviao extends ObjetoVisual {
    public Aviao(float v, int pont, int x, int y, float l, float a) {
        super(v, pont, x, y, l, a);
    }
    ...
}

```

c) (0.75 ponto) Defina uma classe *Canhão* que, além das características de um objeto visual, esta também possua um atributo decimal chamado *angulo*, o qual armazenará o ângulo de inclinação do canhão. Note que um canhão também precisará ser desenhado.

RESPOSTA:

```

class Canhao extends ObjetoVisual {
    double inclinacao;
    public Canhao(float v, int pont, int x, int y, float l, float a,
double incl) {
        super(v, pont, x, y, l, a);
        inclinacao = incl;
    }
    void desenha() {
    }
}

```

d) (0.75 ponto) Crie uma classe chamada *Bomba*, cujos objetos poderão ser criados à partir de um avião. Cada bomba terá uma posição (x, y) e suas altura e largura são desprezíveis (valem 0). Além disso, possui um atributo chamado alcance, que fornece o raio de destruição da bomba. Uma bomba não possui dados como vida ou pontuação.

RESPOSTA:

```

abstract class Objeto2D {
    int pos_x; // Posicao no eixo x
    int pos_y; // Posicao no eixo y
    float largura; // Largura do objeto
    float altura; // Altura do objeto

    public abstract void desenha();
}

abstract class ObjetoVisual extends Objeto2D {
    float vida; // Varia entre 0, .25, .5, 1
    int pontuacao; // + 1 para cada acerto

    ...
}

class Aviao extends ObjetoVisual {
    ...
}

class Canhao extends ObjetoVisual {
    ...
}

```

```

class Bomba extends Objeto2D {
    int pos_x; // Posicao no eixo x
    int pos_y; // Posicao no eixo y
    float alcance;

    public Bomba(int x, int y, float a) {
        pos_x = x;
        pos_y = y;
        alcance = a;
    }
    public void desenha() {
        // Comandos para o desenho de uma bomba
    }
}

```

Questão 4) (2.5 pontos)

Endereços WWW usualmente se referem a algum domínio virtual e possuem o seguinte formato (exemplo): <http://www.cederj.edu.br> Eventualmente estes endereços também podem apontar para recursos específicos dentro deste domínio: <http://www.cederj.edu.br/vestibular/> , onde /vestibular é o recurso específico desejado. A comunicação entre servidores e navegadores web normalmente é feita através da porta padrão 80. Sem entrar em detalhes sobre o que isso representa, os endereços <http://www.cederj.edu.br:80> e <http://www.cederj.edu.br> são equivalentes. Entretanto, podem existir domínios que respondam por uma porta não padrão, por exemplo: <http://www.cederj.edu.br:8080>

Crie uma classe que permite a manipulação de um endereço WWW, ou seja:

- Armazene os campos básicos de um endereço WWW: o domínio, a porta (80, caso não seja fornecida) e o recurso apontado
- Esta classe deve conter construtores que recebam as 3 informações em separado ou uma string completa
- Esta classe deve conter métodos para retornar cada um dos campos

RESPOSTA:

```

class EnderecoWWW {
    private String dominio;
    private int porta;
    private String recurso;

    // Construtores pedidos no item 2
    public EnderecoWWW(String dom,
                        int port,
                        String rec) {
        dominio = dom;
        porta = port;
        recurso = rec;
    }

    public EnderecoWWW(String endereco) {
        // Para criação deste objetos foram implementados métodos
        //estáticos
        //que recuperam cada campo de uma string www em separado
    }
}

```

```

        dominio = EnderecoWWW.obtemDominio(endereco);
        porta = EnderecoWWW.obtemPorta(endereco);
        recurso = EnderecoWWW.obtemRecurso(endereco);
    }

    public static String obtemDominio(String endereco) {
        String inicioPadrao = "http://", dominio;
        if (endereco.startsWith(inicioPadrao)) // Remote inicio
padrão
            endereco = endereco.substring(inicioPadrao.length(),
endereco.length());

        int fimDominio = endereco.indexOf(':');
        if (fimDominio == -1) { // testa se existe porta não padrão
            fimDominio = endereco.indexOf('/');
            if (fimDominio == -1)
                fimDominio = endereco.length();
        }
        dominio = endereco.substring(0, fimDominio);
        return dominio;
    }

    public static int obtemPorta(String endereco) {
        int fimDominio, fimPorta, porta;
        String inicioPadrao = "http://", sporta;
        if (endereco.startsWith(inicioPadrao)) // Remote inicio
padrão
            endereco = endereco.substring(inicioPadrao.length(),
endereco.length());

        fimDominio = endereco.indexOf(':');
        if (fimDominio == -1)
            porta = 80;
        else {
            fimPorta = endereco.indexOf('/', fimDominio);
            if (fimPorta == -1)
                fimPorta = endereco.length();
            sporta = endereco.substring(fimDominio + 1, fimPorta);
            porta = Integer.parseInt(sporta);
        }

        return porta;
    }

    public static String obtemRecurso(String endereco) {
        int inicioRecurso;
        String inicioPadrao = "http://", recurso;
        if (endereco.startsWith(inicioPadrao)) // Remote inicio
padrão
            endereco = endereco.substring(inicioPadrao.length(),
endereco.length());

        inicioRecurso = endereco.indexOf('/');
        if (inicioRecurso != -1)
            recurso = endereco.substring(inicioRecurso+1,
endereco.length());
        else

```

```

        recurso = null;
        return recurso;
    }

    // Método exemplo de exibição (não solicitado na questão)
    public void exhibe() {
        String saida = "URL criada: " + "http://" + dominio;
        if (porta != 80) {
            saida = saida + ":" + porta;
        }
        if (recurso != null) {
            saida = saida + "/" + recurso;
        }
        System.out.println(saida);
    }

    // Métodos de obtenção e alteração de campos solicitados no item
c)

    public String getDominio() {
        return dominio;
    }

    public void setDominio(String dominio) {
        this.dominio = dominio;
    }

    public int getPorta() {
        return porta;
    }

    public void setPorta(int porta) {
        this.porta = porta;
    }

    public String getRecurso() {
        return recurso;
    }

    public void setRecurso(String recurso) {
        this.recurso = recurso;
    }
}

// Classe para teste da classe solicitada na questão (não pedido)
public class AP1_2009_1_Q4 {
    public static void main(String[] args) {
        String www = "http://www.cederj.edu.br/vest";
        EnderecoWWW cederj = new EnderecoWWW(www);
        cederj.exibe();
    }
}

```