



Fundação CECIERJ - Vice Presidência de Educação Superior a Distância

Curso de Tecnologia em Sistemas de Computação

Disciplina: Programação III

AP1 1º semestre de 2011.

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) (4.0 pontos)

Considere que você esteja num supermercado comprando algumas “guloseimas”, carregando uma mochila vazia. Cada produto tem um código único, um peso e um preço. Você pode comprar quantos produtos quiser, desde que você não ultrapasse o seu limite de gastos, o limite de volumes da mochila, e nem o seu limite de peso da mochila, já que você vai carregar a sua mochila para casa. Assim:

- (a) Represente os atributos da classe produto e da classe mochila;
- (b) Implemente métodos que criem objetos das duas classes. Todos os limites da mochila devem ser passados neste método de criação, sendo todos estes limites números reais, exceto o limite de volumes da mochila;
- (c) Implemente um método que insira produtos na mochila, se for permitido. A permissão deverá testar TODOS os limites pré-estabelecidos da mochila;
- (d) Implemente um método que retire produtos da mochila, modificando informações a respeito da mochila; e
- (e) Implemente um método que informe o número de volumes da mochila, o peso da mochila, o gasto até agora e todos os produtos existentes na mochila.

RESPOSTA:

```
class produto{ //(a)

    private static int prox_cod = 1;
    int codigo;
    float preco, peso;
```

```

    produto(float preco, float peso){ //(b)

        codigo = prox_cod++;

        this.preco = preco;

        this.peso = peso;
    }

    float getPreco() { return preco;
}

    float getPeso()  {
return peso;
}

    float getCodigo(){
return codigo;
}

    public String toString(){ //(e)

        String resp = "codigo: " + codigo + " preco: " + preco + "
peso: " + peso;

        return resp;
    }
}

class mochila{ //(a)

    private float peso_atual, gasto_atual;
    float peso, gasto;
    produto[] vetor;
    int num_prod;

    mochila(float p, float g, int l){ //(b)

        peso = p;

        gasto = g;

        peso_atual = gasto_atual = 0;

        vetor = new produto[l];

        num_prod = 0;

    }

    void insere(produto p){ //(c)

```

```

        if(((gasto_atual + p.getPreco()) <= gasto) && ((peso_atual +
p.getPeso()) <= peso) && (num_prod != vetor.length)){

            gasto_atual += p.getPreco();

            peso_atual += p.getPeso();

            num_prod++;

            for(int i = 0; i < vetor.length; i++)

                if(vetor[i] != null){

                    vetor[i] = p;

                    break;

                }
            }
        }

void retira(int cod){ //(d)

    int i;

    for(i = 0; i < vetor.length; i++) if((vetor[i] != null) &&
(vetor[i].getCodigo() == cod)) break;

    if(i == vetor.length) return;

    produto p = vetor[i];

    vetor[i] = null;

    num_prod--;

    peso_atual -= p.getPeso();

    gasto_atual -= p.getPreco();
}

public String toString(){ //(e)

    String resp = "Mochila de " + num_prod + " de " + peso + " kg
e total de " + gasto + ": restantes " + peso_atual + " kg e R$ "
+ gasto_atual + "\n";

    for(int i = 0; i < vetor.length; i++)
        if(vetor[i] != null) resp += vetor[i].toString();

    return resp;

```

```
}  
}
```

Questão 2) (3.0 pontos)

Defina uma classe de objetos que representam uma data, contendo os seguintes métodos:

- i) um método *toString*, que retorna uma cadeia de caracteres correspondente à data representada pelo objeto alvo da chamada, no formato “dd/mm/aaaa”, onde dd, mm e aaaa correspondem, respectivamente, ao dia, mês e ano desta data;
- ii) um método *compara*, que compara a data representada pelo objeto alvo da chamada com uma data passada como argumento para o método; o valor retornado deve ser 0 se essas datas são iguais, ou um número negativo se a primeira data é anterior à última, ou um número positivo se a primeira é posterior à última.

RESPOSTA:

```
class Data {  
    int dia;  
    int mes;  
    int ano;  
  
    public Data(int dia, int mes, int ano) {  
        super();  
        this.dia = dia;  
        this.mes = mes;  
        this.ano = ano;  
    }  
  
    public String toString() {  
        return dia + "/" + mes + "/" + ano;  
    }  
  
    public int compara (Data d) {  
        if (dia == d.getDia() && mes == d.getMes() && ano == d.getAno())  
            return 0;  
        if (ano < d.getAno() ||  
            (ano == d.getAno() && mes < d.getMes()) ||  
            (ano == d.getAno() && mes == d.getMes() && dia <  
d.getDia()))  
            return -1;  
        return 1;  
    }  
  
    public int getDia() {  
        return dia;  
    }  
  
    public int getMes() {  
        return mes;  
    }  
  
    public int getAno() {  
        return ano;  
    }  
}  
  
// Classe de teste para a classe Telefone acima  
public class AP1_2011_1_Q2 {  
    public static void main(String[] args) {
```

```

        Data d = new Data(28, 4, 2011);
        Data d2 = new Data(4, 4, 2011);
        String comparacao = d.compara(d2) == 0 ? "iguais" : (d.compara(d2) == -1 ?
"menor" : "maior");
        System.out.println("Comparação de datas " + d.toString() + " e " +
d2.toString() + ": " + comparacao);
    }
}

```

Questão 3) (3.0 pontos)

Crie uma classe Telefone para representar números de telefone. Para dar suporte à generalidade e à expansão futuras, códigos internacionais (o do Brasil, por exemplo, é 55), códigos de área (o do Rio, também por exemplo, é 21) e o número do telefone devem ser armazenados separadamente. Crie 3 construtores: 1o.) inicializando os 3 campos; 2o.) inicializando área e o número, deixando 55 como padrão para o código internacional; 3o.) inicializando o número, deixando 55 e 21 como padrão para os códigos internacional e de área, respectivamente. Crie também um método *toString()* que retorne uma string no seguinte formato: “+55 21 78684527”, onde 78684527 é o número do telefone.

RESPOSTA:

```

class Telefone {
    int codInternacional;
    int codArea;
    int numero;

    public Telefone(int codInternacional, int codArea, int numero) {
        this.codInternacional = codInternacional;
        this.codArea = codArea;
        this.numero = numero;
    }

    public Telefone(int codArea, int numero) {
        this(55, codArea, numero);
    }

    public Telefone(int numero) {
        this(55, 21, numero);
    }

    public String toString() {
        return "+" + codInternacional + " " + codArea + " " + numero;
    }
}

// Classe de teste para a classe Telefone acima
public class AP1_2011_1_Q2 {
    public static void main(String[] args) {
        System.out.println(new Telefone(78684527));
    }
}

```