



Fundação CECIERJ - Vice Presidência de Educação Superior a Distância
Curso de Tecnologia em Sistemas de Computação
Disciplina: EAD05032 - Programação Orientada a Objetos
AP2 – 1º semestre de 2023.

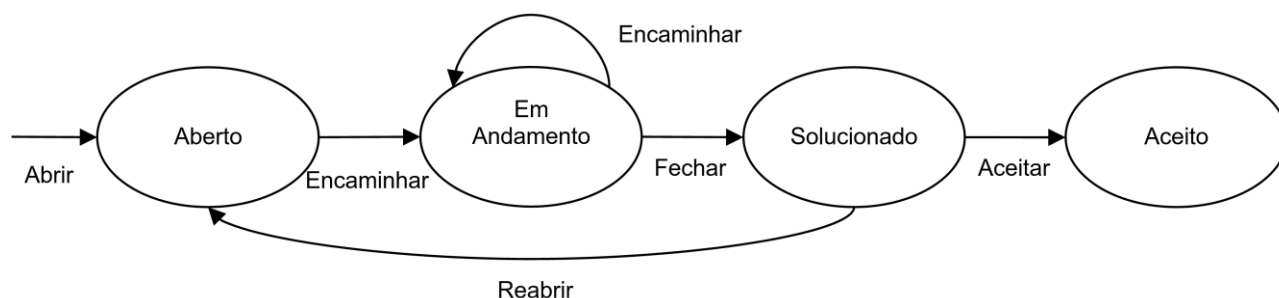
Observações:

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)

A Central de Atendimento ao Cliente das Casas Resolve-se Tudo trabalha com tickets de atendimento. Quando um cliente liga para a Central, um ticket de atendimento é criado e este assume o estado “Aberto”. Depois de aberto, este pode ser preenchido com a solicitação do cliente e, após preenchimento, o ticket é encaminhado para solução. A ação de encaminhar tem como parâmetro o nome do(a) responsável pela solução do ticket. Após encaminhado, dizemos que o estado do ticket é “Em Andamento”. O ticket pode ser reencaminhado para solução quantas vezes for necessário, pois para resolver o problema às vezes é preciso passar por várias pessoas. Quando o problema parece ter sido solucionado, o ticket é fechado, assumindo o estado “Solucionado”. Neste momento, o cliente avalia se o problema relatado foi de fato resolvido. Caso não tenha sido resolvido, o ticket é reaberto, voltando para o estado “Aberto”. Mas, se o problema foi resolvido, a solução é aceita e o ticket assume o estado “Aceito”.

O diagrama abaixo ilustra as transições e os estados de um ticket:



Sempre que uma ação inválida é efetuada em um ticket (por exemplo, um ticket somente pode ser aceito se ele já tiver sido solucionado), uma exceção é lançada indicando qual ação inválida foi efetuada.

Projete e construa as classes para essa questão. Implemente também um programa de teste que crie um ticket e execute diversas ações sobre ele, mostrando como está o ticket entre cada ação solicitada. Um possível caso de teste poderia ser:

```
try {
    Ticket t = new Ticket();
    t.definirDescricao("Minha geladeira pifou");
    System.out.println(t);
    t.encaminhar("Leandro");
    System.out.println(t);
    t.encaminhar("Dante");
    System.out.println(t);
    t.fechar();
    System.out.println(t);
    t.reabrir();
    System.out.println(t);
    t.encaminhar("Isabel");
    System.out.println(t);
    t.fechar();
    System.out.println(t);
    t.aceitar();
    System.out.println(t);
    t.fechar();
    System.out.println(t);
}
catch (AcaoInvalidaException e) {
    System.out.println(e);
}
```

Questão 2)

Considerando as seguintes declarações abaixo citadas de uma lista encadeada, e usando **SOMENTE UMA VARREDURA** nesta lista, retorne a posição do elemento do meio dessa lista. **NÃO É PERMITIDO:**

- (1) alterar o método *main*. Só é possível retirar o comentário da linha `System.out.println("O elemento do meio da lista esta na posicao " + descobreElementoMeio(1));` e
- (2) copiar os dados para outras estruturas de dados já implementadas na linguagem JAVA como, por exemplo, vetores estáticos, *List*, *ArrayList*, entre outras.

O método *main* recebe um arquivo de dados contendo matrícula, nota e nome, separados por vírgula, um por linha, lê esse arquivo e insere os elementos na lista *ListaUsandoVetor*. A seguir, o método *descobreElementoMeio* retorna a posição do elemento do meio dessa lista.

```
import java.util.Random;
import java.io.*;

class No{
    int mat;
    float nota;
    String nome;
    int prox_no;

    No(int mat, float nota, String nome){
        this.mat = mat;
        this.nota = nota;
        this.nome = nome;
        prox_no = -1;
    }
}

class ListaUsandoVetor{
    int tam_vet_elem, prim, num_elem;
    No vet_elem[];

    ListaUsandoVetor(int tam_vet_elem){
        prim = -1;
        num_elem = 0;
        this.tam_vet_elem = tam_vet_elem;
        vet_elem = new No[tam_vet_elem];
        int i;
        for(i = 0; i < tam_vet_elem; i++) vet_elem[i] = null;
    }

    void InsereInicio(int mat, float nota, String nome){
        if(num_elem == tam_vet_elem) DobraTamanhoCopiandoElementos();
        Random gerador = new Random();
        int n;
        do{
            n = gerador.nextInt() % tam_vet_elem;
            if((n >= 0) && (vet_elem[n] == null)) break;
        }while(true);
        vet_elem[n] = new No(mat, nota, nome);
        vet_elem[n].prox_no = prim;
        prim = n;
        num_elem++;
    }
}
```

```

    public String toString(){
        int ind = prim;
        String resp = "";
        while(ind != -1){
            resp = resp + vet_elem[ind].mat + " " + vet_elem[ind].nota + " " +
vet_elem[ind].nome + "\n";
            ind = vet_elem[ind].prox_no;
        }
        return resp;
    }

    void DobraTamanhoCopiandoElementos(){
        No copia[] = new No[tam_vet_elem];
        int i;
        for(i = 0; i < num_elem; i++){
            copia[i] = vet_elem[i];
            vet_elem[i] = null;
        }
        tam_vet_elem *= 2;
        vet_elem = new No[tam_vet_elem];
        for(i = num_elem; i < tam_vet_elem; i++) vet_elem[i] = null;
        for(i = 0; i < num_elem; i++){
            vet_elem[i] = copia[i];
            copia[i] = null;
        }
    }
}

public class Q2_AP2_2023_1{
    public static void main (String[] args) throws Exception{
        System.out.print("No arquivo de entrada, a(s) matricula(s), a(s) nota(s)
e o(s) nome(s) estao separados por ;\n");
        BufferedReader in = new BufferedReader(new FileReader(args[0]));
        try {
            ListaUsandoVetor l = new ListaUsandoVetor(1);
            String s, vs[];
            s = in.readLine();
            while(s != null){
                vs = s.split(";");
                l.InsereInicio(Integer.parseInt(vs[0]),          Float.parseFloat(vs[1]),
vs[2]);
                s = in.readLine();
            }
            in.close();
            System.out.println("Lista: ");
            System.out.print(l);

            //IMPLEMENTAR O METODO A SEGUIR:
            //RETORNAR A POSICAO DO ELEMENTO DO MEIO NA LISTA ListaUsandoVetor
            //System.out.println("O elemento do meio da lista esta na posicao " +
descobreElementoMeio(l));
        }
        catch (Exception e){
            System.out.println("Excecao\n");
        }
    }

    //IMPLEMENTAR O METODO A SEGUIR
    /*
    public static int descobreElementoMeio(ListaUsandoVetor l){
    }
    */
}

```

