



Fundação CECIERJ - Vice Presidência de Educação Superior a Distância

**Curso de Tecnologia em Sistemas de Computação**  
**AD2 de Programação OO**  
**1º semestre de 2019**

**EXERCÍCIO (ENTREGAR OS ARQUIVOS EM MÍDIA, PARA FINS DE TESTE, JUNTAMENTE COM A AD IMPRESSA):**

Implemente um programa em Java, utilizando os conceitos de orientação a objetos, que gerencie a frequência dos alunos de uma turma em um dado período, composto de 4 meses. Uma turma contém um nome, um código e o período. Cada pauta de frequência deve incluir a descrição da turma, o mês e uma lista de alunos, onde cada aluno é identificado pelo seu nome e sua matrícula. O programa deve permitir a inclusão e remoção de alunos da pauta, além da marcação da presença ou falta, em cada um dos dias de aula. O sistema deve poder informar o total de faltas de cada aluno e se ele está reprovado por falta, isto é, se o total de presenças for menor que 75% do total de aulas no período. Considere, para simplificação, que os meses possuem 30 dias. Utilize a seguinte classe de teste para auxiliar na implementação desse programa:

```
public class AD2_P00_2019_1{
    public static void main(String[] args){
        Turma turma = new Turma("Est. de dados", "A1", "20191");
        turma.inserirAluno(new Aluno("Anselmo", 111));
        turma.inserirAluno(new Aluno("Pedro", 222));
        turma.inserirAluno(new Aluno("Joao", 333));
        turma.imprimir();
        System.out.println("-----");
        Pauta pauta = new Pauta(3,6,turma);
        pauta.marcarFalta(111, 3, 5);
        pauta.marcarFalta(111, 3, 12);
        pauta.marcarFalta(111, 3, 21);
        pauta.marcarFalta(111, 3, 26);
        pauta.marcarFalta(111, 4, 2);
        pauta.marcarFalta(111, 4, 4);
        pauta.marcarFalta(111, 4, 11);
        pauta.marcarFalta(111, 4, 23);
        pauta.marcarFalta(111, 4, 25);
        pauta.marcarFalta(111, 4, 30);
        pauta.marcarFalta(111, 5, 7);
        pauta.marcarFalta(111, 5, 23);
        pauta.marcarFalta(222, 3, 12);
        pauta.marcarFalta(222, 3, 21);
        pauta.marcarFalta(222, 4, 9);
        pauta.marcarFalta(222, 4, 16);
        pauta.marcarFalta(222, 4, 25);
        pauta.marcarFalta(222, 4, 30);
        pauta.imprimir();
    }
}
```

```

        for(Aluno aluno:turma.getAlunos()){
            int totalFaltas = pauta.totalFaltasAluno(aluno.getMatricula());
            System.out.print(String.format("%-44s", aluno.toString()) + " "
+ "#faltas = " + String.format("%2d",totalFaltas));
            if (pauta.verificarReprovacaoPorFalta(aluno.getMatricula())){
                System.out.println(" Presença: reprovado");
            }
            else{
                System.out.println(" Presença: normal");
            }
        }
    }
}

```

RESPOSTA:

```
import java.util.*;
```

```

class Aluno implements Comparable{
    private String nome;
    private int matricula;

    public Aluno(String nome, int matricula) {
        this.nome = nome;
        this.matricula = matricula;
    }

    public String getNome() {
        return nome;
    }

    public void setNome(String nome) {
        this.nome = nome;
    }

    public int getMatricula() {
        return matricula;
    }

    public void setMatricula(int matricula) {
        this.matricula = matricula;
    }

    public String toString() {
        return "Aluno{" + "nome=" + nome + ", matricula=" + matricula +
    '}'';
    }

    public int compareTo(Object o) {
        Aluno aluno = (Aluno)o;
        return this.getNome().compareTo(aluno.getNome());
    }
}

```

```

class Turma {
    String nome;
    String codigo;
    String periodo;
    SortedSet<Aluno> alunos;

    public Turma(String nome, String codigo, String periodo) {
        this.nome = nome;
        this.codigo = codigo;
        this.periodo = periodo;
        alunos = new TreeSet<Aluno>();
    }

    public String getNome() {
        return nome;
    }

    public void setNome(String nome) {
        this.nome = nome;
    }

    public String getCodigo() {
        return codigo;
    }

    public void setCodigo(String codigo) {
        this.codigo = codigo;
    }

    public String getPeriodo() {
        return periodo;
    }

    public void setPeriodo(String periodo) {
        this.periodo = periodo;
    }

    public SortedSet<Aluno> getAlunos() {
        return alunos;
    }

    public void setAlunos(SortedSet<Aluno> alunos) {
        this.alunos = alunos;
    }

    Aluno getAluno(int matricula){
        for (Aluno aluno:alunos){
            if (aluno.getMatricula()==matricula){
                return aluno;
            }
        }
        return null;
    }
}

```

```

void inserirAluno(Aluno aluno){
    alunos.add(aluno);
}

void removerAluno(int matricula){
    Aluno aluno = getAluno(matricula);
    alunos.remove(aluno);
}

void imprimir(){
    System.out.println("Turma: " + nome + " ,Codigo:" + codigo +
",Período:" + periodo);
    for (Aluno aluno:alunos){
        System.out.println(aluno.toString());
    }
}
}

```

```

class Frequencia {
    private String nomealuno;
    private boolean[] dias;

    public Frequencia(String nome){
        this.nomealuno = nome;
        dias = new boolean[30];
        for (int i=0;i<30;i++){
            dias[i] = true;
        }
    }

    void marcarFrequencia(int i){
        dias[i-1] = true;
    }

    void desmarcarFrequencia(int i){
        dias[i-1] = false;
    }

    public String getNome() {
        return nomealuno;
    }

    public void setNome(String nome) {
        this.nomealuno = nome;
    }

    public boolean[] getDias() {
        return dias;
    }

    public void setDias(boolean[] dias) {
        this.dias = dias;
    }
}

```

```

    public String toString() {
        String s = String.format("%-40s", nomealuno) + " ";
        for (int i=0;i<30;i++){
            if (dias[i]==true){
                s+='P';
            }else{
                s+='F';
            }
        }
        return s;
    }
}

class PautaMes {
    private int mes;
    private Map<Integer,Frequencia> mapaFrequencias;

    public PautaMes(int mes, Turma turma) {
        this.mes = mes;
        mapaFrequencias = new LinkedHashMap<Integer,Frequencia>();
        for (Aluno aluno:turma.getAlunos()){
            mapaFrequencias.put(aluno.getMatricula(), new
Frequencia(aluno.getNome()));
        }

    }

    public Map<Integer, Frequencia> getMapaFrequencias() {
        return mapaFrequencias;
    }

    public void marcarPresenca(Aluno aluno, int dia){
        Frequencia f = mapaFrequencias.get(aluno);
        f.marcarFrequencia(dia);
    }

    public void marcarFalta(Aluno aluno, int dia){
        Frequencia f = mapaFrequencias.get(aluno);
        f.desmarcarFrequencia(dia);
    }

    public void imprimir(){
        System.out.println("-----"
-----"
+ "-----");
        System.out.println("Mes:"+mes);
        Set<Map.Entry<Integer,Frequencia>>alunos=mapaFrequencias.entrySet
());
        for (Map.Entry<Integer,Frequencia> entry:alunos){
            System.out.print(entry.getKey() + " ");
            Frequencia frequencia = entry.getValue();
            System.out.println(frequencia.toString());
        }
    }
}

```

```

        System.out.println("-----"
        + "-----");
    }
}

```

```

class Pauta {
    static final int NUM_AULAS_PERIODO = 32;
    private int mesInicial;
    private int mesFinal;
    private int meses;
    private Turma turma;
    private PautaMes[] pautas;

    public Pauta(int mesInicial, int mesFinal, Turma turma){
        this.turma = turma;
        this.mesInicial = mesInicial;
        this.mesFinal = mesFinal;
        this.meses = mesFinal-mesInicial+1;
        this.pautas = new PautaMes[meses];
        for (int i = mesInicial;i<=mesFinal;i++){
            this.pautas[i-mesInicial] = new PautaMes(i,turma);
        }
    }

    public void marcarPresenca(int matricula, int mes, int dia){
        Frequencia f = pautas[mes-
mesInicial].getMapaFrequencias().get(matricula);
        f.marcarFrequencia(dia);
    }

    public void marcarFalta(int matricula, int mes, int dia){
        Frequencia f = pautas[mes-
mesInicial].getMapaFrequencias().get(matricula);
        f.desmarcarFrequencia(dia);
    }

    public int totalFaltasAluno(int matricula){
        int totalFaltas = 0;
        for (int mes=mesInicial;mes<=mesFinal;mes++){
            Frequencia freq=pautas[mes-
mesInicial].getMapaFrequencias().get(matricula);
            boolean[] dias = freq.getDias();
            for (int j=0;j<dias.length;j++){
                if (!dias[j]){
                    totalFaltas ++;
                }
            }
        }
        return totalFaltas;
    }
}

```

```
public boolean verificarReprovacaoPorFalta(int matricula){
    if (totalFaltasAluno(matricula)>=0.35*NUM_AULAS_PERIODO){
        return true;
    }else{
        return false;
    }
}

public void imprimir(){
    System.out.println("Turma: " + turma.getNome() + " ,Codigo:" +
turma.getCodigo() + ",Periodo:" + turma.getPeriodo());
    for (int i=0;i<meses;i++){
        pautas[i].imprimir();
    }
}
}
```