

**Questão 1 (0,2 pt)**

Crie a classe Grau, que possui uma data de começo e uma data de fim (ambas do tipo String). Crie 2 construtores: um que recebe parâmetros para inicializar todos os atributos e outro que não recebe parâmetros, inicializado as datas com null. Crie os métodos de acesso e o método toString.

**Questão 2 (0,3 pt)**

Crie a classe GrauA, que é um tipo de Grau. O Grau A possui 2 notas, uma do trabalho e uma da prova. Crie 2 construtores: um que recebe parâmetros para inicializar todos os atributos e outro que não recebe parâmetros (este último inicializa as notas em 0). Crie os métodos de acesso e sobrescreva o método toString.

**Questão 3 (0,3 pt)**

Crie a classe GrauB, que é um tipo de Grau. O Grau B possui 3 notas, uma de atividades, uma de seminário e uma de participação. Crie 2 construtores: um que recebe parâmetros para inicializar todos os atributos e outro que não recebe parâmetros (este último inicializa as notas em 0). Crie os métodos de acesso e sobrescreva o método toString.

**Questão 4 (0,5 pt)**

Nas classes GrauA e GrauB, crie um método chamado calculaNotaFinalGrau, que retorna a nota do grau correspondente, sabendo que:

- *Nota final do Grau A:* nota do trabalho vale 30% e a nota da prova vale 70%
- *Nota final do Grau B:* nota de atividades vale 30%, a nota do seminário vale 60% e a nota de participação vale 10%.

**Questão 5 (1,0 pt)**

Crie a classe Aluno, que possui um nome e duas referências do tipo Grau, chamadas ga e gb. No construtor de Aluno, receba apenas o nome do aluno por parâmetro e diga que ga é um objeto do tipo GrauA e que gb é um objeto do tipo GrauB. Crie os métodos de acesso dos atributos e o método toString.

**Questão 6 (0,5 pt)** Na classe Aluno, crie o método calculaNotaFinal, que retorna a nota final do aluno, sabendo que o Grau A vale 33% e o Grau B vale 67% da nota.

**Questão 7 (3,0 pt)**

(1,0 pt) Crie uma classe chamada Turma. Esta classe possui como atributos um código (tipo String) e um array de Aluno. No construtor, receba o código da turma e a quantidade máxima de alunos por parâmetro (Inicialize o array de Aluno com esta quantidade). Ainda na classe Turma, faça o que se pede:

- (1,0 pt) crie um método `insereAluno`, que recebe um objeto do tipo Aluno por parâmetro e o insere na primeira posição livre do array, retornando verdadeiro caso seja possível inserir e falso caso contrário
- (1,0 pt) crie um método chamado `mediaNotasTurma`, que retorna a média de nota (nota final) de todos os alunos presentes no array (ou -1 caso não existam alunos)

**Questão 8 (1,2 pt)**

Na classe Turma, crie o método `alteraNotasGrauA`, que recebe por parâmetro o nome do aluno que deve ter as notas do GA alteradas e duas notas, que se referem à nota do trabalho e da prova do Grau A. Altere as notas do Grau A do aluno com o nome especificado. Crie, na mesma classe, o método `alteraNotasGrauB`, que recebe o nome do aluno e três notas, que se referem à nota de atividades, nota do seminário e nota de participação do Grau B. Este método deve alterar as respectivas notas do Grau B do aluno especificado.

**Questão 9 (3,0 pt)**

Crie a classe Principal com o método `main`. Neste método, faça o que se pede:

- (0,5 pt) crie uma Turma, com código sendo solicitado pelo teclado ao usuário e capacidade máxima definida por sorteio, entre 1 e 50 alunos
- (1,0 pt) nesta turma, tente inserir X alunos, onde X deve ser um valor aleatório entre 1 e 100. A cada tentativa, informe sucesso ou insucesso na inserção. Os nomes dos alunos devem ser obtidos através de sorteios no código ou de valores pré-definidos (como "Aluno 1", "Aluno 2" etc.); as notas dos alunos devem ser inseridas em cada objeto utilizando os métodos de acesso, sendo os valores obtidos através de sorteios entre 0 e 100.
- (0,5 pt) imprima a média de notas da turma
- (1,0 pt) sabendo que a nota final para aprovação deve ser maior ou igual é 6,0, imprima a mensagem "Aluno NOME\_DO\_ALUNO: Passou"

por média!" ou "Aluno NOME\_DO\_ALUNO: Ficou em Grau C.", para cada um dos alunos da turma