

## Algoritmos e Estrutura de Dados - Laboratório 2

Tópicos: Estruturas, Ponteiros, Funções (valor/referência) e Alocação dinâmica

**Exercício 1.** Crie uma estrutura para armazenar os dados de alunos de um curso de graduação. A estrutura deve conter a matrícula do aluno, nome, nota da primeira prova, nota da segunda prova, nota da terceira prova, média, número de faltas e situação (aprovado, reprovado por média ou reprovado por faltas).

- (a) Cadastre os dados de 3 alunos no seu código, exceto os campos média e situação que devem ser processados. Não pedir para digitar – cadastre no próprio código fonte, utilizando comandos de atribuição.
- (b) Calcule a média de cada aluno, preenchendo o campo correspondente na estrutura.
- (c) Calcule a situação final de cada aluno, preenchendo essa informação no campo da estrutura. Para ser aprovado o aluno precisa ter nota média final maior ou igual a 60 e ter presença maior ou igual a 75% (considere um total de 72 aulas). No caso de reprovado por falta (prioritário), mostrar “Reprovado por falta”. Caso tenha sido reprovado por nota, mostrar “Reprovado por nota”.
- (d) Mostre na tela os dados de cada aluno, incluindo a média e a situação final.

Os itens (a) – (d) devem ser implementados como funções.

**Exercício 2.** Complemente o código, incluindo:

- (e) Estatísticas da turma: Encontre e mostre na tela o aluno com a maior nota em cada prova. Encontre e mostre o aluno com maior média geral. Encontre e mostre o aluno com menor média geral. Encontre e mostre os nomes dos alunos com média acima (ou igual) da média da turma. O item (e) devem ser implementados como função e executado após (d).

**Exercício 3.** Complemente o código, incluindo:

- (f) Incluir uma opção após executar (e) que pergunta ao usuário: “Editar aluno?” Caso o usuário digite ‘N’, o programa é encerrado. Caso o usuário digite ‘Y’, deve ser solicitado o número de matrícula que deve ser editado. Após a digitação do número de matrícula, o vetor de alunos deve ser percorrido até o cadastro correspondente ser encontrado. O usuário deve editar as notas das 3 provas do aluno por scanf(). Após a digitação, os itens (b) a (e) devem ser processados novamente. Ao final, perguntar novamente “Editar aluno?” e seguir passos anteriores.

Obs: A edição das notas do aluno selecionado deve ser implementado como função e apenas o cadastro do aluno selecionado deve ser passado por referência.

**Exercício 4.** Reescreva o código do exercício 3, usando alocação dinâmica de memória. Iniciar perguntando ao usuário o número de alunos que deseja incluir. O cadastro dos alunos deve ser implementado por uma função que utilize o scanf(). Após o usuário ter cadastrado todos alunos, executar os itens (a) – (f) dos exercícios anteriores.