

Lab. 16

Tipos Estruturados

ECT2303 - T02 - 19.1

Em todas as questões, defina um tipo estruturado para representar um aluno de Linguagem de Programação. Os campos armazenados são: o nome (máximo de 50 caracteres), um inteiro para armazenar a matrícula do aluno, um inteiro para armazenar o número de faltas e um vetor de ponto flutuante com 3 elementos para armazenar as notas da disciplina.

16.1 Exercícios de Fixação

1. Crie uma função em C++ que recebe um aluno e retorna um booleano indicando se o aluno foi aprovado (caso verdadeiro) ou reprovado (caso falso) por nota. Assuma, para isto, as seguintes condições:
 - Se a média das notas do aluno for maior ou igual que 7,0, ele está aprovado;
 - Se a média das notas do aluno for menor que 7,0, porém maior ou igual que 5,0, e todas as notas forem maiores ou iguais a 3,0, ele também é considerado aprovado;
 - Em qualquer outro caso, ele é considerado reprovado.

Na função principal (main), o usuário deve fornecer os dados do aluno e, a seguir, verificar se ele está aprovado por nota.

Exemplo de execução:

```
//Exemplo 01:
Digite o nome do aluno: Joao da Silva
Digite a matricula do aluno: 1234
Digite o numero de faltas do aluno: 2
Digite as três notas do aluno: 7.6 6.3 8.4
O aluno Joao da Silva foi aprovado!
//Exemplo 02:
```

```

Digite o nome do aluno: Joao da Silva
Digite a matricula do aluno: 1234
Digite o numero de faltas do aluno: 2
Digite as três notas do aluno: 4.4 5.6 5.7
O aluno Joao da Silva foi aprovado!
//Exemplo 03:
Digite o nome do aluno: Joao da Silva
Digite a matricula do aluno: 1234
Digite o numero de faltas do aluno: 2
Digite as três notas do aluno: 8.0 1.0 1.5
O aluno Joao da Silva foi reprovado!

```

2. Crie uma função em C++ que recebe um vetor de alunos, bem como a quantidade de alunos, e um valor inteiro informando uma matrícula como parâmetros de entrada. A função deve retornar o índice para o aluno com a respectiva matrícula ou -1, caso não exista.

Na função principal (main), o usuário deve fornecer os dados de vários alunos, receber um valor de matrícula, chamar a função e, em seguida, exibir todos os dados do aluno obtido ou uma mensagem informando que não foi encontrado nenhum aluno com a matrícula informada.

Exemplo de execução:

```

Digite o numero de alunos: 3
Digite o nome do aluno: Joao da Silva
Digite a matricula do aluno: 1234
Digite o numero de faltas do aluno: 2
Digite as três notas do aluno: 8.0 8.0 7.0
Digite o nome do aluno: Jose da Costa
Digite a matricula do aluno: 4321
Digite o numero de faltas do aluno: 10
Digite as três notas do aluno: 5.0 3.0 3.0
Digite o nome do aluno: Maria do Socorro
Digite a matricula do aluno: 2222
Digite o numero de faltas do aluno: 18
Digite as três notas do aluno: 1.0 1.0 2.0
Digite a matricula: 4321
Dados do aluno:
Nome: Jose da Costa
Matricula: 4321
Faltas: 10
Notas: 5, 3, 3

```

3. Crie uma função em C++ que recebe um vetor de alunos, bem como a quantidade de alunos, e um número inteiro informando o número máximo de faltas da disciplina como parâmetros de entrada. Também, ela deve receber dois vetores de alunos como parâmetros de saída. A sua função deve armazenar em um vetor os alunos reprovados por falta (ou seja, aqueles cuja quantidade de faltas é maior que o número máximo fornecido) e no outro vetor, os alunos que não estão reprovados por falta.

Como o número de elementos de cada um destes vetores não é conhecido, estes devem ser fornecidos como parâmetro de saída também, sendo calculados durante a execução da função.

Na função principal (main), o usuário deve fornecer os dados de vários alunos, receber o número máximo de faltas, chamar a função e, em seguida, exibir o nome e a matrícula dos alunos reprovados por falta e dos alunos não-reprovados por falta.

Exemplo de execução:

```
Digite o numero de alunos: 4
Digite o nome do aluno: Joao da Silva
Digite a matricula do aluno: 1234
Digite o numero de faltas do aluno: 2
Digite as três notas do aluno: 10.0 8.0 8.0
Digite o nome do aluno: Gasparzinho
Digite a matricula do aluno: 2222
Digite o numero de faltas do aluno: 48
Digite as três notas do aluno: 3.0 3.0 3.0
Digite o nome do aluno: Maria do Socorro
Digite a matricula do aluno: 5555
Digite o numero de faltas do aluno: 10
Digite as três notas do aluno: 4.0 5.0 6.0
Digite o nome do aluno: Penaldinho
Digite a matricula do aluno: 8888
Digite o numero de faltas do aluno: 56
Digite as três notas do aluno: 1.0 1.0 1.0
Digite o numero maximo de faltas: 18
Alunos reprovado por falta:
2222 - Gasparzinho
8888 - Penaldinho
Alunos nao reprovados por falta:
1234 - Joao da Silva
5555 - Maria do Socorro
```

16.2 Referências Bibliográficas

1. MANZANO, J.A.; OLIVEIRA, J.F.; **Algoritmos - Lógica para Desenvolvimento de Programação**. Editora Erica.
2. ASCENCIO, A.F.G.; CAMPOS, E.A.V. **Fundamentos da Programação de Computadores - Algoritmos, Pascal e C/C++**. 3ed. Editora Pearson.
3. DEITEL, H. M.; DEITEL, P. J. **C++ Como Programar**. 3ed. Editora Bookman.