

Aula Prática 9

Desenvolver os seguintes programas em Linguagem “C”, **em equipes de até no máximo 3 integrantes** (“duplas” de até 3 alunos):

1. Escreva um programa capaz de ler “todos” os registros armazenados em um arquivo binário, listando, em ordem de código, todos os nomes dos acadêmicos presentes no arquivo que tenham ficado em exame final (média maior ou igual a 4.0, e menor que 7.0), sendo que no arquivo estão armazenadas as notas das provas (np1 e np2), e dos trabalhos(nt1 e nt2), mas a média ainda não está calculada. A média é calculada pela fórmula: $media = (np1*0.7+nt1*0.3) + (np2*0.7+nt2*0.3) / 2.0$. A estrutura utilizada para gerar o arquivo binário é:

```
struct tipo_aluno
{
    int codigo;
    char nome[80];
    float np1, nt1, np2, nt2;
    float media;
};
```

O programa deve utilizar o comando “typedef struct tipo_aluno ACADEMICO;” para definir um novo tipo de dados, e criar um **vetor de estruturas, utilizando um ponteiro**:

```
ACADEMICO *vetor_academicos;
vetor_academicos = (ACADEMICO *) malloc (num_regs_no_arq * sizeof(struct tipo_aluno);
```

2. No programa do exercício anterior, com vetor de estruturas, é necessário primeiro calcular a quantidade de registros presentes no arquivo para poder realizar a alocação do vetor de ponteiros. Para isto, é necessário abrir o arquivo, e utilizar o comando “fread” tantas vezes quantas forem necessárias até encontrar o final do arquivo, contando quantos registros estão presentes no arquivo. O desafio apresentado por este exercício é evitar esta contagem, através da implementação do armazenamento dos registros em memória através do uso de uma “Lista Encadeada”, utilizando para isto a estrutura abaixo: (O programa deve utilizar técnicas de “Alocação Dinâmica” dos registros.)

```
struct tipo_lista_encadeada
{
    struct tipo_aluno aluno;
    struct tipo_lista_encadeada *prox;
} *Lista;
```