

LISTA DE EXERCÍCIO

Registro

1 - Crie um programa que possa armazenar dados de alunos (nome, rg, matrícula, semestre) e possa ainda armazenar notas (nota1, nota2, nota3). Além disso, o programa deve ser capaz de exibir os dados dos alunos cadastrados.

Para isso siga os passos:

A) Crie um registro do tipo Aluno que possua os seguintes atributos: nome, rg e matrícula (todos strings com tamanho 50, 20 e 5 respectivamente) e semestre (inteiro).

B) Crie um registro do tipo Notas que possua os seguintes atributos: nota1, nota2 e nota3 (todos float).

C) Adicione mais um atributo em Aluno, notas do tipo Notas.

D) Crie uma função chamada cadastroAluno que deve solicitar o nome, rg, matricula e semestre. Você deve armazenar em um registro e armazenar isso no array chamado listaDeAlunos (*dica: listaDeAlunos deve ser passado como parâmetro*).

E) Crie um função chamada cadastroNota que deve solicitar o número de matricula do aluno e as notas. Você deve armazenar as notas em uma registro do tipo Nota e atribuir este registro ao respectivo Aluno (Dica: faça uma busca no array listaDeAlunos em busca da matricula).

F) Crie uma função exibeLista que deve exibir os dados dos alunos no seguinte formato (atenção para o cálculo da média ponderada: peso2, peso3 e peso5). A nota para aprovação é 6)

```
Aluno : Luis Gustavo
Matricula : 1010
RG : 10101010
NOTAS : N1 = 10.00 N2 = 5.00, N3 = 7.00
MEDIA : 7.00
APROVADO
-----
Process exited after 11.73 seconds with return value 9
Pressione qualquer tecla para continuar. . .
```

G) Na main, crie um menu que peça a ação ao usuário: 1 - Cadastrar alunos 2, cadastrar Notas, 3 Exibir Lista. Para cada opção faça a chamada da função respectiva criada anteriormente. Ao fim de todas as funções, deve-se voltar ao menu.

```
Ola Usuario! O que deseja:
```

```
1) Cadastrar Alunos.  
2) Cadastrar Notas.  
3) Exibir Alunos.  
4) Sair.
```

USE O TEMPLATE

```
#include<stdio.h>  
#include<stdlib.h>  
#include<string.h>
```

```
/*  
Crie seus registros  
*/
```

```
int main(){  
    //inicializando o nosso array como todos elementos 0 (é um valor invalido)  
    int j;  
    Aluno listaDeAlunos[10];  
    for(j=0;j<10;++j)  
        listaDeAlunos[j]=(Aluno){0,0,0};
```

```
    //crie o menu com o while para cada opcao chame as funcoes
```

```
}
```

```
/*  
Crie suas funções registros  
*/
```

DICAS:

Passando array por parâmetro (aula de ponteiros) no cadastro Aluno.

```
void cadastroAluno(Aluno* listaDeAlunos)
```

Buscando uma posição vazia no array de alunos

Realize um for do 0 até o limite, ou use um while com a seguinte condicional. Use o break para interromper o loop quando achar o aluno.

```
strcmp(listaDeAlunos[indice].matricula, "") == 0
```

Ou, com ponteiros

```
Aluno a = *(listaDeAlunos + indice );  
while( strcmp( a.matricula, "") != 0){
```

Obs: isso só é válido pois inicializamos todas as matriculas com "" (siga o template dado).

Pesquisando um registro específico no array

Realize um for do 0 até o limite, ou use um while com a seguinte condicional. Use o break para interromper o loop quando achar o aluno.

```
strcmp(listaDeAlunos[indice].matricula, matriculaProcurada) == 0
```

Ou, com ponteiros

```
Aluno a = *(listaDeAlunos + indice );  
while( strcmp( a.matricula, matriculaProcurada) != 0){
```

Obs: matriculaProcurada é um exemplo da variável que está armazenada a matricula. Você pode usar qualquer nome.