

# Exercícios (Ponteiros)

1) Implementar um algoritmo que armazene um **vetor de floats**. Este vetor deve ser criado dinamicamente através da função **malloc** ou **calloc**. O programador deve questionar o usuário sobre quantos valores **floats** ele deseja armazenar. De posse do número de valores, o espaço de memória adequado deve ser alocado e posteriormente a leitura dos valores deve ser realizada.

Após a leitura de todos os valores, deve ser chamada uma função **lista\_floats** para mostrar na tela todos os valores.

2)

Dada a estrutura abaixo para armazenar dados de um aluno:

```
struct aluno {  
    int matricula;  
    char *nome;  
    float media;  
};
```

Implementar um algoritmo que armazene um **vetor de struct aluno**. Este vetor deve ser criado dinamicamente através da função **malloc** ou **calloc**. O programador deve questionar o usuário sobre quantos **alunos** ele deseja armazenar. De posse do número de alunos, o espaço de memória adequado deve ser alocado e posteriormente a leitura dos dados dos alunos deve ser realizada.

Após a leitura de todos os alunos, deve ser chamada uma função **lista\_alunos** para mostrar na tela todos os valores relacionados aos alunos.

3) Implementar um algoritmo que armazene um **vetor de strings**. Este vetor deve ser criado dinamicamente através da função **malloc** ou **calloc**. O programador deve questionar o usuário sobre quantas **strings** ele deseja armazenar. De posse do número de valores, o espaço de memória adequado deve ser alocado e posteriormente a leitura dos valores deve ser realizada.

Após a leitura de todos os valores, deve ser chamada uma função **lista\_strings** para mostrar na tela todos os valores cadastrados.