

## Ponteiros + Memória Dinâmica + Estruturas

Considere a seguinte definição de um tipo estrutura associada aos dados de uma pessoa:

```
typedef struct {  
    int    numAluno;    // Número de aluno  
    int    Notas[4];    // Notas em Aprendizagem, Ex. Normal, Ex. Recurso e Ex. Especial  
    int    NotaFinal;    // melhor nota das 4 anteriores  
} ALUNO;
```

a) Implementar uma função que, dado um vetor de inteiros com N elementos, devolva o maior deles. Para tal, use o seguinte cabeçalho:

```
int maiorVetor (int V[], int N)
```

b) Implementar uma função que leia do ficheiro "Alunos.txt" e construa um vetor do tipo ALUNO. Cada linha do ficheiro contém 5 valores inteiros (associados a um aluno):

- numero de aluno,
- nota obtida em Aprendizagem,
- nota obtida em Exame Normal,
- nota obtida em Exame Recurso, e
- nota obtida em Exame Especial.

Para tal, use o seguinte cabeçalho:

```
ALUNO *lerVetor (int *N)
```

c) Construa um programa em C que realize as seguintes ações pela ordem apresentada:

- construir um vetor do tipo ALUNO com os dados do ficheiro "Alunos.txt",
- complete os dados daquele vetor, atribuindo o valor adequado ao campo NotaFinal,
- guarde o número de estudante (**numAluno**) e a respetiva nota final (**NotaFinal**) de cada um dos alunos no ficheiro "AlunosFinal.txt" (construir uma função para o efeito),
- determine o aluno (número) com melhor nota final (construir uma função para o efeito).

d) Implementar uma segunda versão da função **lerVetor**. Para tal, use o seguinte cabeçalho:

```
void lerVetor2 (ALUNO **V, int *N)
```

e teste no programa implementado em c)