

Estruturas (struct)

1. Construa o tipo **struct PONTO** para armazenar um ponto de \mathbf{R}^3 e resolva as questões que a seguir se apresentam.

a) Escreva um subprograma para ler um ponto de \mathbf{R}^3 (inicializar uma variável do tipo PONTO).

Duas alternativas:

```
PONTO lerPONTO ();  
void lerPONTO (PONTO*);
```

b) Escreva um subprograma para escrever um ponto de \mathbf{R}^3 (variável do tipo PONTO).

```
void escreverPONTO (PONTO, PONTO);
```

c) Escreva um subprograma para determinar a distância entre dois pontos de \mathbf{R}^3 .

```
float distanciaEntreDoisPontos (PONTO, PONTO);
```

2. Usando o tipo **COMPLEXO** abaixo transcrito, escreva um programa que retorne a soma de dois números complexos. Lembre-se que um número complexo é um número que pode ser escrito na forma **a+bi**, em que a e b são números reais, sendo a designado por parte real e b por parte imaginária dum número complexo.

```
typedef struct {  
    float a; // parte real  
    float b; // parte imaginaria  
} COMPLEXO;
```

Utilize subprogramas para:

a) ler um número complexo.

b) escrever um número complexo.

c) calcular a soma de dois números complexos.

3. Considere a estrutura ALUNO com a seguinte informação: número de aluno e nota final obtida numa UC (por esta ordem). Definir um tipo de dados com o nome ALUNO da seguinte forma:

```
typedef struct{  
    int numero;  
    int notaFinal;  
} ALUNO;
```

Implementar um programa em C para determinar a nota média obtida na UC. Para tal, construir e usar os seguintes subprogramas:

a) um subprograma para ler um array 1D com valores dados pelo utilizador

b) um subprograma para calcular a soma das notas obtidas pelos alunos (usar o array)

c) um subprograma para calcular a média das notas dos alunos aprovados (usar o array).

4. Considere a estrutura ALUNO com a seguinte informação: número de aluno, nota dos trabalhos práticos (TP) e nota do teste escrito (T) (por esta ordem).

Definir um tipo de dados com o nome ALUNO da seguinte forma:

```
typedef struct{  
    int    numero;  
    int    notaTP;  
    int    notaT;  
    int    notaFinal;  
} ALUNO;
```

Implementar um programa para determinar as notas finais de todos os alunos e o número de alunos aprovados numa UC. Para tal, construir e usar os seguintes subprogramas:

- a)** um subprograma para ler um array 1D com dados fornecidos pelo utilizador
- b)** um subprograma para atualizar o array anterior determinado a nota final
- c)** um subprograma para calcular o número de alunos aprovados (usar o array).