

## Programação Imperativa – Folha das aulas práticas nº 5

**Temas:** EXERCÍCIOS SIMPLES COM VECTORES.

Nome \_\_\_\_\_ Ncd \_\_\_\_\_

1. Atrás e à frente...

Num vector  $v$  dado com  $m$  elementos (como se sabe, os índices vão de 0 a  $m-1$ ) substituir cada elemento, excepto o primeiro e o último, pelo soma do elemento anterior com o elemento seguinte.

**Exemplo.**

no início:  $v=\{3,2, 1,8,6\}$  ( $m=5$ )  
no fim:  $v=\{3,4,10,7,8\}$

**Nota.** Para testar o seu programa pode usar o seguinte esquema

**Nota.** Não pode usar outros vectores.

```
//-- Imprime um vector
void imprime(int n,int a[]){
    int i;
    for(i=0;i<n;i++){
        printf("%3d",a[i]);
        printf("    (%2d elementos) \n",n);
    }
}

main(){
    int m=5, v[]={3,2,1,8,6},...
    imprime(m,v);
    ...
    imprime(m,v);
}
```

2. Frio?

Um vector `temp[]` contém 24 medições da temperatura do ar num determinado dia: às 0 horas,..., e às 23 horas. Escreva um programa que determine sucessivamente: (i) a média das temperaturas entre as 8 da manhã e as 8 da noite (extremos incluídos), (ii) a temperatura mínima e a(s) hora(s) correspondente(s).

**Resultados.**

Temperatura media = 21.9 graus

Temperatura mínima = 10.2 graus

'as 3 horas

'as 4 horas

'as 6 horas

**Possível estrutura do programa.** Note a inicialização de `ten` e os comentários indicando as horas.

```
#define HORAS 24

main(){
//      0   1   2   3   4   5
double ten[]={9.2,9.5, 12.4,15.2,11.5,12.3,
//      6   7   8   9   10  11
           9.8,15.2,9.9, 12.1,15.2,11.5,
...
... cálculo da soma ...
printf("Temperatura media = %lf\n\n",s/HORAS);

... cálculo do mínimo ...
printf("Temperatura mínima = %lf\n",minimo);
... cálculo das horas em que ocorreu o máximo ...
printf("      'as %2d horas\n",i);
}
```

### 3. Repetidos?

Seja  $w$  um vector com  $m$  elementos (não ordenado, em geral). Pretende-se implementar uma função

```
int distintos(int m,int w[])
```

que tem como objectivo eliminar de  $w$  os elementos que ocorrem mais que uma vez (os valores de  $w$  são “chegados à esquerda”). O valor retornado pela função é o número de elementos do vector resultante (o qual, obviamente, não excede  $m$ ).

**Exemplo.**

```
chamada de "unicos": v={6,5,1,5,5,8,4,2,8,2}, n=10
no final:           v={6,5,1,8,4,2},           (porquê?)
valor retornado:    6                          (porquê?)
```

**Nota.** Para testar a função que escreveu pode utilizar o seguinte programa

```
//-- Elimina de v[] os valores duplicados
int distintos(int m,int w[])...

main(){
  int m=10, v[]={1,2,2,4,5,5,5,6,8,8};
  imprime(m,v);
  m=unicos(m,v);
  imprime(m,v);
}
```

onde `imprime` é a função dada no primeiro problema.

**Possível método.** Para cada índice  $i=1, 2, \dots, n-1$  procure  $v[i]$  para trás; se já ocorreu, “elimine”  $v[i]$ , chegando os elementos seguintes à esquerda.

---