

## Folha 4

Começar por rever os apontamentos sobre **Variáveis Indexadas** (em <https://www.dcc.fc.up.pt/~pbv/aulas/progimp/teoricas/teorica13.html>), sem o exemplo de escrita em binário.

1. Pretendemos um programa para ler  $m$  **sequências de  $n$  inteiros não negativos** e escrever a sequência que tem **maior média arredondada** para o inteiro mais próximo. Os valores são lidos do *standard input* (entrada padrão) e escritos no *standard output* (saída padrão). Em caso de empate, deve indicar a primeira que encontrou. Admita que  $n \leq 1000$ .

Para resolver o problema deve implementar as funções seguintes:

a) `void ler_seq(int x[], int n)` para ler uma sequência  $n$  inteiros e a colocar no *array* dado por  $x$ . Na posição de índice  $i$  fica o  $(i + 1)$ -ésimo elemento da sequência, para  $0 \leq i \leq n - 1$ .

b) `void escrever_seq(int x[], int n)` para escrever os  $n$  primeiros elementos do *array* dado por  $x$ . Deve apresentá-los numa linha separados por um espaço. (*Sugestão*: comece por escrever o primeiro elemento (sem espaço) e, depois, cada um dos restantes precedido de um espaço).

c) `copiar(int y[], int x[], int n)` para copiar os primeiros  $n$  elementos do *array* dado por  $x$  para o *array* dado por  $y$ .

d) `int media_arred(int x[], int n)` que retorna a média dos  $n$  primeiros elementos do *array* dado por  $x$ , arredondada para o inteiro mais próximo (pode somar 0.5 e truncar).

e) `int main()`, que deverá usar as funções anteriores com funções auxiliares.

## Exemplo

## Input

```
4 10
2 3 4 6 7 7 2 3 4 9
5 6 7 14 5 8 9 8 10 3
8 8 8 8 8 8 8 8 9 10
4 5 6 7 8 8 9 6 5 4
```

## Output

```
5 6 7 14 5 8 9 8 10 3
Media = 8
```

2. Dispomos de uma caixa com moedas de valores 200, 100, 50, 20, 10, 5, 2, 1 cêntimos e pretendemos formar uma quantia  $Q$  (em cêntimos). Queremos formar  $Q$  aplicando o método seguinte:

- usamos  $Q/v$  moedas de valor  $v$ , sendo  $v$  o valor mais alto não superior a  $Q$ ; se não existirem  $Q/v$  moedas de valor  $v$ , usamos todas as que a caixa contém de valor  $v$ ;
- sendo  $k$ , com  $k \geq 0$ , o número de moedas de valor  $v$  que usámos, aplicamos a mesma estratégia para formar a quantia  $Q - kv$ , se  $v > 1$  e  $Q \neq 0$ .

a) Implemente uma função `int troca(int vmoeda[], int dmoeda[], int n, int q)` que aplica o método indicado e retorna a quantia que sobra (que é 0 só se conseguiu trocar, o que tal método não garante, mesmo se o conteúdo da caixa o permitia). Em `dmoeda[i]` tem o número de moedas disponíveis de valor `vmoeda[i]`, para  $0 \leq i \leq n - 1$ . Admita que o *array* `vmoeda` está ordenado por ordem decrescente de valor.

b) Use a função `troca` para resolver o problema enunciado. A função `main` deve ler o número de moedas existentes dos oito tipos referidos, por ordem decrescente de valor, e a seguir o valor de  $Q$ . Imprime a quantia que sobraria.

#### Exemplo 1

##### Input

```
5 3 2 3 0 0 1 1
60
```

##### Output

```
7
```

#### Exemplo 2

##### Input

```
0 2 2 3 0 10 1 1
230
```

##### Output

```
0
```