

ALP

FICHA DE EXERCÍCIOS
ATIVIDADE LETIVA

Algoritmia e Estruturas de Dados

UNIDADE CURRICULAR

Ficha 04 – Listas

1. Elabore a função **aboveAverage** que receba uma lista com 10 números inteiros (número inseridos pelo utilizador) e que devolva quantos desses números estão acima da média.
2. Elabora uma função **generateNumbers** que permita gerar uma chave do Euromilhões de forma aleatória (5 números inteiros entre 1 e 50), assim como as estrelas (duas estrelas entre 1 e 12).

A função **generateNumbers** deve ter 3 argumentos de entrada: limite inferior, limite superior, e quantidade de números a gerar. Por exemplo, para gerar chaves do Euromilhões, deve invocar a função com os seguintes argumentos:

generateNumbers(1,50, 5) -> gera 5 números entre 1 e 50

generateNumbers(1,12,2) -> gera 2 números entre 1 e 12 (estrelas)

A função deve devolver uma lista com os números gerados. Obviamente que a função não pode devolver números nem estrelas repetidas!

No final imprima a chave gerada e pergunte ao utilizador se pretende gerar nova chave (S/N).

Exemplo:

C:\WINDOWS\py.exe

Chave do Euromilhões: [26, 36, 11, 27, 47] Estrelas: [3, 6]

Deseja gerar nova Chave(S/N)? _

3. Crie um programa que leia a pontuação de 10 participantes num concurso de programação (a pontuação deve **estar validada** entre 0 a 20, por uma estrutura try-except).

O programa deve invocar uma função, **positiveList** que receba a lista das 10 pontuações e devolva uma nova lista apenas com as pontuações positivas (≥ 10).

Exemplo:

```
C:\WINDOWS\py.exe
Pontuação: 12
Pontuação: 9
Pontuação: 6
Pontuação: 7
Pontuação: 16
Pontuação: 17
Pontuação: 14
Pontuação: 2
Pontuação: 3
Pontuação: 1
Pontuações positivas: [12, 16, 17, 14]
```

4. Altere o programa anterior de modo a incluir também a leitura dos nomes dos participantes no concurso.

A função **positiveList** deve agora devolver duas listas: uma com pontuações e outra com nomes dos participantes que obtiveram pontuações positivas (≥ 10).

Exemplo:

```
C:\WINDOWS\py.exe
Participantes com pontuações positivas
Nomes      : ['António', 'Carla', 'Maria']
Pontuações: [12, 15, 18]
```

5. Elabore um programa que permita ler a faturação mensal de uma empresa ao longo dos 12 meses do ano (de janeiro a dezembro).

O programa deve incluir a chamada de 3 funções que devolvam:

- a) o mês de maior faturação
- b) o mês de menor faturação
- c) a média mensal de faturação

```
C:\WINDOWS\py.exe
Faturação do mês Janeiro : 800
Faturação do mês Fevereiro : 900
Faturação do mês Março : 1100
Faturação do mês Abril : 1210
Faturação do mês Maio : 1320
Faturação do mês Junho : 1570
Faturação do mês Julho : 1400
Faturação do mês Agosto : 1300
Faturação do mês Setembro : 750
Faturação do mês Outubro : 900
Faturação do mês Novembro : 1400
Faturação do mês Dezembro : 1350

Mês de maior faturação : Junho
Mês de menor faturação : Setembro
Valor médio de faturação 1166.67:
```

6. Elabore um programa que leia uma lista de 10 números inteiros.

Em seguida, dado um determinado valor de pesquisa, invoque a função **searchNumber**(*lista, pesquisa*) que deve devolver as posições em que encontra o valor de pesquisa, na lista.

Caso o valor de pesquisa não exista na lista, deve surgir uma mensagem adequada.

7. O Instituto de meteorologia pretende registar o valor total de pluviosidade ocorrida em cada mês, ao longo de um ano (de janeiro a dezembro).

Implemente a função **pluviosidade** que receba a lista com a pluviosidade de cada mês, e imprima esses mesmos dados (lista de pluviosidade), mas ordenados por ordem decrescente de pluviosidade.

Upgrade versão 2.0: imprimir também os nomes dos respetivos meses, como na imagem!

```

C:\WINDOWS\py.exe
Pluviosidade no mês de Janeiro : 150
Pluviosidade no mês de Fevereiro : 190
Pluviosidade no mês de Março : 225
Pluviosidade no mês de Abril : 270
Pluviosidade no mês de Maio : 145
Pluviosidade no mês de Junho : 110
Pluviosidade no mês de Julho : 90
Pluviosidade no mês de Agosto : 60
Pluviosidade no mês de Setembro : 80
Pluviosidade no mês de Outubro : 120
Pluviosidade no mês de Novembro : 175
Pluviosidade no mês de Dezembro : 185

Abril 270
Março 225
Fevereiro 190
Dezembro 185
Novembro 175
Janeiro 150
Maio 145
Outubro 120
Junho 110
Julho 90
Setembro 80
Agosto 60

```

8. Dado uma lista de N elementos (N deve ser pedido ao utilizador), crie uma função que ordene a lista e que permita gerar uma outra lista sem valores duplicados.

Exemplo lista inicial:

2	4	6	6	10	12	12	16	18	20
---	---	---	---	----	----	----	----	----	----

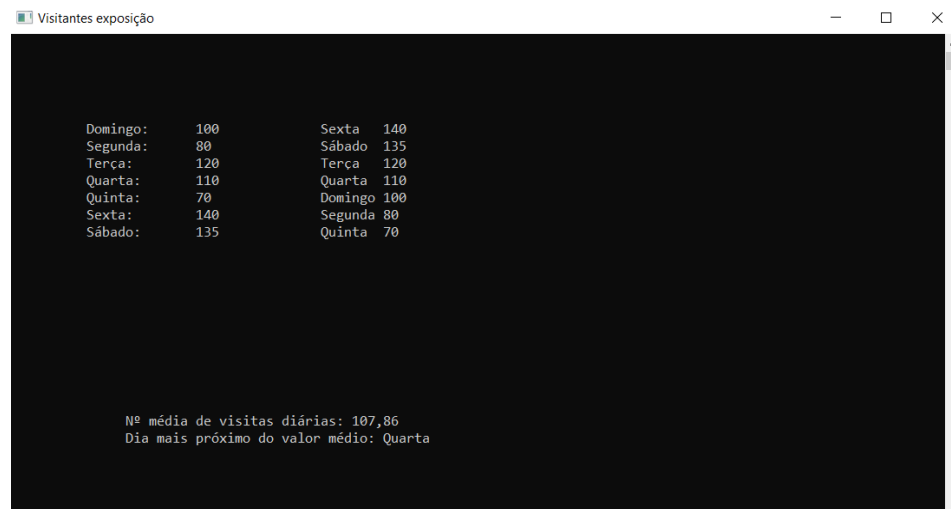
Lista gerada:

2	4	6	10	12	16	18	20
---	---	---	----	----	----	----	----

9. Implemente um programa que permita ler o número de visitantes numa exposição, que decorre de Domingo a Sábado.

O seu Em seguida elabore uma função que permita listar o número de visitantes diário por ordem decrescente, tal como na imagem abaixo apresentada.

Indique ainda, no final, o número médio de visitantes por dia (com 2 casas decimais) e o dia que mais se aproximou do número médio de visitantes.



10. Elabore um programa que permita gerir uma fila de espera (como num supermercado, ou num serviço de atendimento) com capacidade máxima para 20 lugares.

Quando o programa se inicia, todos os lugares da fila devem estar livres.

Layout da fila de espera (lista com 20 posições):

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
---	---	---	---	---	---	---	---	---	----	----	----	----	----	----	----	----	----	----	----

O seu programa deve conter um menu com as seguintes opções:

MENU

1 - Tirar Ticket

2 - Atendimento

3 - Estado da fila de espera

0 - Sair

- Opção Tirar Ticket

Deve chamar uma função que permita ocupar o primeiro lugar que estiver livre na fila de espera, indicando na consola a posição ocupada (ou seja, o número do ticket).

Se todos os lugares estiverem ocupados deverá surgir a mensagem de “Fila completa”.

- Opção Atendimento

Deve invocar uma função que permita atender sempre o elemento que está na primeira posição da fila de espera, num dado momento (imprime o nº da sua senha na consola).

Neste caso, todos os restantes elementos da fila de espera devem avançar uma posição à frente.

- Opção Estado da fila de espera

Deve invocar uma função que imprima, na consola, o número de lugares ocupados e o número lugares livres na fila de espera.

- Opção Sair

Termina a execução do programa