

ALP

EXERCISE SHEET
ACADEMIC ACTIVITY

Algorithms and Data Structures

CURRICULAR UNIT

Sheet 06 – Lists

1. Create the **aboveAverage** function that receives a list of 10 integers (numbers entered by the user) and returns how many of these numbers are above the average.
2. Create a **generateNumbers** function that allows you to generate a random Euromillions key: 5 numbers between 1 and 50, as well as the stars (two stars between 1 and 12).

The generateNumbers function must have 3 input arguments: lower limit, upper limit, and number of numbers to generate: **generateNumbers(lowerLimit, upperLimit, numbers)**.

For example, to generate a random Euromillions key, you must invoke the function with the following arguments:

generateNumbers(1,50, 5) -> generates 5 numbers between 1 and 50

generateNumbers(1,12,2) -> generates 2 numbers between 1 and 12 (stars)

The function must return a list with the generated numbers. Obviously, the function cannot return repeated numbers or repeated stars!

At the end, print the generated Euromillions key and ask the user if he wants to generate a new key (Y/N).

Example:

C:\WINDOWS\py.exe

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

Deseja gerar nova Chave(S/N)? ☐

3. Create a program that reads the scores of 10 participants in a programming contest (the scores must be validated between 0 and 20, using a try-except structure).

The program must invoke a function, **positiveList**, that receives the list of 10 scores and returns a new list with only the positive scores (≥ 10).

Example:

```
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. Change the previous program to also include the input of the names of the contest participants.

The **positiveList** function should now return two lists: one with scores and another with names of participants who obtained positive scores (≥ 10).

Example:

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

5. Create a program that allows you to read the sales volume over the 12 months of the year (from January to December).

The program must include the calling of **3 functions** which return, respectively:

a) the month with the highest volume sales

b) the month with the lowest sales

c) the average sales value

```
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:
```