

[PT] arrays

A - Classificações de APROG

Faça um programa que leia as classificações de exame de APROG de uma turma com N alunos (para um array). O número de alunos (N) é inserido pelo utilizador.

De seguida, calcule e mostre a média da turma. Finalmente, mostre quantos alunos reprovaram à disciplina (nota <10).

O programa deve mostrar a média da turma (formatada com 1 casa decimal) e quantos alunos reprovaram à disciplina, em linhas separadas e com o seguinte formato:

```
average=<media>
failures=<nº de reprovações>
```

[EN] arrays

A - APROG Ratings

Write a program to read the APROG exam scores for a class with N students (for an array). The number of students (N) is entered by the user.

Then calculate and show the class average.

Finally, display how many students failed the course (grade <10).

The program should display the class average (formatted to 1 decimal place) and how many students failed the course on separate lines with the following format:

```
average=<average>
failures=<number of failures>
```

Example1:

Input	Output
5 12 8 17 19 5	average=12.2 failures=2

Example2:

Input	Output
10 10 11 9	average=9.6 failures=5

12 8 13 7 14 6 6	
------------------------------------	--

Example3:

Input	Output
10 1 2 3 4 5 6 7 8 9 9	average=5.4 failures=10

Example4:

Input	Output
15 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11	average=11.0 failures=0

[PT] arrays

B - Menores elementos

Faça um programa que leia um conjunto de números inteiros positivos (para um array). A leitura de números termina quando for inserido um número negativo.

De seguida, determine qual o menor elemento do conjunto e quantas vezes

foi inserido.

O programa deve mostrar o menor número inserido e quantas vezes foi inserido, em linhas separadas e com o seguinte formato:

```
min=<numero>
occurrences=<nº de ocorrencias>
```

[EN] arrays

B - Minimum

Write a program to read a set of positive integers (for an array). Number reading ends when a negative number is entered.

Then, determine which is the element with minimum value in the set and how many times it has been inserted. The last number (negative) should be avoided.

The program should display the number with minimum value and how many times it was inserted, on separate lines and with the following format:

```
min=<lowest number>
occurrences=<number of occurrences>
```

Example1:

Input	Output
5 12 2 17 19 -5	min=2 occurrences=1

Example2:

Input	Output
12 5 12 8 5 17 5 -51	min=5 occurrences=3

Example3:

Input	Output
-1	

[PT] arrays

C - Sequência crescente

Faça um programa que verifique se a sequência de números inseridos pelo utilizador é sempre crescente.

O programa deve pedir ao utilizador a quantidade de números a inserir e, de seguida, proceder à sua leitura.

No final, o programa deve mostrar uma das seguintes mensagens: **"always ascending = true"** ou **"always ascending = false"** caso a sequência dos números inseridos seja sempre crescente ou não, respetivamente.

Para este efeito construa:

- Um método para ler e guardar num array os números inseridos pelo utilizador.
- Um método para verificar se a sequência do array é crescente. O método deve retornar true ou false, caso a sequência seja sempre crescente ou não, respetivamente.

[EN] arrays

C - Ascending Sequence

Write a program to verify if a sequence of numbers entered by the user is always ascending.

The program should prompt the user for the amount of numbers to enter and then read them.

At the end, the program should display one of the following messages: **"always ascending = true"** or **"always ascending = false"** if the sequence of numbers entered is always ascending or not, respectively.

For this purpose implement:

- A method to read and store in an array the numbers entered by the user.
- A method to check if the array sequence is ascending. The method must return true or false if the sequence is always ascending or not, respectively.

Example1:

Input	Output
5 11 12 13 14 15	always ascending = true

Example2:

Input	Output
6 11 12 1 10 2 15	always ascending = false

Example3:

Input	Output
10 1 2 3 3 4 5 6 7 8 9	always ascending = false

[PT] arrays

D - Frequências das notas

Faça um programa que leia as classificações de um teste de avaliação resolvido por N alunos e calcule as frequências absolutas para as notas obtidas. Pretende-se saber para cada nota, de zero (0), um (1), ..., até vinte (20), quantos alunos obtiveram essa nota.

As classificações são valores inteiros e podem variar entre zero (0) e vinte (20), inclusive.

O programa deve ler o valor de N, seguindo-se a leitura das N notas dos alunos.

O programa deve mostrar a escala zero (0) a vinte (20) e, para cada valor da escala, a quantidade de alunos que obtiveram essa nota, separado por um espaço. Cada nota da escala (de zero a vinte) e respetiva frequência deve aparecer em linhas separadas, no seguinte formato:

```
0 <n. alunos com nota 0>
1 <n. alunos com nota 1>
...
20 <n. alunos com nota 20>
```

Implemente as seguintes funcionalidades usando modularização:

a) Ler e armazenar as notas dos alunos.

- b) Determinar as frequências das notas, de zero a vinte.
- c) Mostrar os valores das frequências

[EN] arrays

D - Grades frequency

Write a program to read the grades that N students obtained on a test, and calculate the absolute frequencies for the grades obtained. It is intended to know for each grade, from zero (0), one (1), ..., up to twenty (20), how many students obtained this grade. Grades are integer values and can range from zero (0) to twenty (20) inclusive. The program should start by reading the value of N, followed by reading the grade of each student.

The program must display the scale from zero (0) to twenty (20) and, for each value of the scale, the number of students who obtained this grade, separated by a space. Each scale grade (from zero to twenty) and its frequency should appear on separate lines in the following format:

```
0 <n. students with grade 0>
1 <n. students with grade 1>
...
20 <n. students with grade 20>
```

Implement the following functionality using modularization:

- a) Read and store student grades.
- b) Determine the frequencies of grades from zero to twenty.
- c) Display frequency values

Example1:

Input	Output
7	0 0
17	1 0
9	2 0
13	3 0
17	4 0
9	5 0
15	6 0
17	7 0
	8 0
	9 2
	10 0
	11 0
	12 0
	13 1
	14 0
	15 1
	16 0
	17 3
	18 0
	19 0

20 0

Example2:

Input	Output
8	0 0
12	1 0
13	2 0
10	3 0
5	4 0
12	5 1
10	6 0
12	7 0
10	8 0
	9 0
	10 3
	11 0
	12 3
	13 1
	14 0
	15 0
	16 0
	17 0
	18 0
	19 0
	20 0

Example3:

Input	Output
42	0 2
0	1 2
1	2 2
2	3 2
3	4 2
4	5 2
5	6 2
6	7 2
7	8 2
8	9 2
9	10 2
10	11 2
11	12 2
12	13 2
13	14 2
14	15 2
15	16 2
16	17 2
17	18 2
18	19 2
19	20 2
20	
0	
1	
2	

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

Example4:

Input	Output
9	0 5
0	1 0
20	2 0
0	3 0
10	4 0
0	5 0
10	6 0
0	7 0
10	8 0
0	9 0
	10 3
	11 0
	12 0
	13 0
	14 0
	15 0
	16 0
	17 0
	18 0
	19 0
	20 1

[PT] arrays

E - Estatísticas de vencimentos

Faça um programa para determinar algumas estatísticas sobre vencimentos de funcionários de uma empresa. O número de funcionários pode variar ao longo do tempo, será pelos menos um, mas nunca será superior a 20. O programa deve mostrar o valor da média dos vencimentos assim como o

nome dos funcionários com vencimentos inferior à média.

Usando modularização, implemente as seguintes funcionalidades:

- a) Ler os nomes e vencimentos dos funcionários da empresa. Para cada funcionário deve ser lido o seu nome e respetivo vencimento. A leitura termina quando for inserido o nome **"end"**.
- b) Calcular a média dos vencimentos.
- c) Mostrar resultados. Mostrar em linhas separadas, o valor da média (1 casa decimal) e os nomes dos funcionários com vencimento inferior a essa média.

[EN] arrays

E - Salary Statistics

Write a program to determine some statistics on employee salaries of a company. The number of employees may vary over time, will be at least one, but will never exceed 20.

The program must read to each employee their name and salary. Reading ends when the name **"end"** is entered.

The program must display in separate lines the average salary (with 1 decimal place) as well as the name of each employee earning less than the average salary.

Example1:

Input	Output
ana 1000 berta 500 carla 1200 daniela 5000 end	1925.0 ana berta carla

Example2:

Input	Output
ana 150.5 berta 500.7 end	325.6 ana

Example3:

Input	Output
alberto 100 bernardo 100 carlos 100 daniel 100 ernesto 100 end	100.0

Example4:

Input	Output
alberto 100 end	100.0

[PT] arrays

F - Aplicação financeira

Faça um programa que realize a simulação de uma aplicação financeira ao fim de 6 meses. A aplicação consiste num depósito bancário inicial com juros capitalizáveis, isto é, no final de cada mês o montante em depósito será acrescido do juro respetivo desse mês.

O funcionamento do programa deve obedecer à seguinte sequência de instruções:

1. Ler as taxas de juro para cada um dos meses (6).
2. Ler o valor do depósito inicial.
3. Mostrar o montante final após os 6 meses de depósito, usando duas casas decimais, no seguinte formato:

"final value=<valor>"

[EN] arrays

F - Financial investment

Write a program to simulate a financial investment after 6 months. The investment consists of an initial bank deposit with capitalizable bank rate for each month, ie, at the end of each month the deposit amount will be increased by the respective bank rate of that month.

First, the program must read the bank rates for each of the six months. Second, it must read the amount of the initial deposit. Finally, it must

display the amount of the deposit after 6 months, using two decimal places in the following format:

"final value=<value>"

Example1:

month		1	2	3	4	5	6
rate		0.15	0.20	0.11	0.01	0.02	0.05
deposit	10000.00	11500.00	13800.00	15318.00	15471.18	15780.60	16569.63

Input	Output
0.15 0.20 0.11 0.01 0.02 0.05 10000	final value=16569.63

Example2:

Input	Output
0 0 0 0 0 0 1000	final value=1000.00

Example3:

Input	Output
0 -0.1 0 -0.1 0.1 0 1000	final value=891.00

[PT] arrays

G - Inverter sequência

Faça um programa modular que leia uma sequência de N números inteiros e os mostre pela ordem inversa da inserção. O valor de N é inserido,

inicialmente, pelo utilizador.

Os números devem ser visualizado em linhas separadas.

[EN] arrays

G - Invert sequence

Write a modular program to read a sequence of N integers and display them in the reverse order of insertion. The value of N is initially entered by the user.

Each number must be displayed in a separate lines.

Example1:

Input	Output
5	50
10	40
20	30
30	20
40	10
50	

Example2:

Input	Output
10	-1
1	9
2	8
3	7
4	6
5	5
6	4
7	3
8	2
9	1
-1	

Example3:

Input	Output
0	

[PT] arrays

H - Rodar sequência

Faça um programa modular que leia uma sequência de N números inteiros. O valor de N é inserido, inicialmente, pelo utilizador.

De seguida, o programa deve ler os N números inteiros.

Finalmente, o programa deve aceitar 3 comandos possíveis- **"right"**, **"left"** e **"exit"** cujas funcionalidades são:

"right" - rodar 1 posição a sequência de números para a direita (ex: 1 2 3 4 → 4 1 2 3)

"left" - rodar 1 posição a sequência de números para a esquerda (ex: 1 2 3 4 → 2 3 4 1)

"exit" - termina o programa

O programa deve aceitar comandos **"right"** ou **"left"** até ser inserido o comando **"exit"** que termina o programa. Sempre que for executado o comando **"right"** ou **"left"** deve ser mostrado o resultado da sequência.

Cada sequência deve ser visualizada numa linha única e cada um dos seus elementos deve ser mostrado entre parêntesis retos, no seguinte formato:

`"[a][b]...[n]"`

[EN] arrays

H - Rotate sequence

Write a modular program to read a sequence of N integers.

The value of N is initially entered by the user.

Then the program should read the N integers.

Finally, the program must accept 3 possible commands: **"right"**, **"left"** and **"exit"** whose meaning is:

"right" - rotate the sequence 1 position to the right (eg 1 2 3 4 → 4 1 2 3)

"left" - rotate the sequence 1 position to the left (eg 1 2 3 4 → 2 3 4 1)

"exit" - end the program

The program must accept **"right"** or **"left"** commands until the **"exit"** command that terminates the program.

Each time the **"right"** or **"left"** command is executed, the result of the sequence should be displayed.

Each sequence should be displayed on a single line and each element should be displayed in square brackets in the following format:

`"[a][b]...[n]"`

Example1:

Input	Output
5	[50][10][20][30][40]
10	[40][50][10][20][30]
20	[50][10][20][30][40]
30	
40	
50	
right	

right left exit	
-----------------------	--

Example2:

Input	Output
5	[50] [10] [20] [30] [40]
10	[40] [50] [10] [20] [30]
20	[30] [40] [50] [10] [20]
30	[20] [30] [40] [50] [10]
40	[10] [20] [30] [40] [50]
50	
right	
right	
right	
right	
right	
exit	

Example3:

Input	Output
7	[20] [30] [40] [50] [60] [70] [10]
10	[30] [40] [50] [60] [70] [10] [20]
20	[40] [50] [60] [70] [10] [20] [30]
30	[30] [40] [50] [60] [70] [10] [20]
40	
50	
60	
70	
left	
left	
left	
right	
aaa	
exit	

Example4:

Input	Output
3	
11	
12	
13	
exit	

[PT] arrays

I - Condomínio (v1)

Considere um prédio de apartamentos com 3 entradas e 4 pisos, completamente ocupados.

Faça um programa modular que leia o nome do proprietário de cada um dos apartamentos. A leitura deve ser feita por ordem crescente de entrada e piso, isto é, para cada entrada devem ser lidos os proprietários dos respectivos pisos.

De seguida, o programa deve ler um dado nome inserido pelo utilizador, procurar e mostrar para esse nome qual é a respetiva entrada e piso, no seguinte formato:

```
"name=<nome>"
```

```
"entrance=<n° da entrada>"
```

```
"floor=<n° do piso>"
```

Se existirem múltiplas soluções (vários proprietários com o mesmo nome), deve indicar todas as possibilidades, por ordem ascendente de entrada e piso.

Caso não exista um proprietário com esse nome no prédio, deve imprimir a mensagem **"Do not live in the building"**.

[EN] arrays

I - Condominium (v1)

Consider a completely occupied apartment building with 3 entrances and 4 floors.

Write a modular program to:

-First, read the name of the owner of each apartment. The reading should be done in order of entrance and floor, ie, for each entrance must be read the owners of each of its floors;

-Second, read a particular name and display which entrance and floor it owns, in the following format:

```
"name=<name>"
```

```
"entrance=<entrance number>"
```

```
"floor=<floor number> "
```

If there are multiple solutions (multiple owners with the same name), all possibilities should be displayed, in ascending order of entrance and floor.

If there is no owner with that name in the building, the message **"Do not live in the building"** should be displayed.

Example:

floor

3	Luisa Lima	Hercilia Hora	Maria Mota
2	Carla Costa	Gloria Gomes	Luisa Lima
1	Berta Barata	Francisca Fe	Joana Jales
0	Ana Anacleto	Elsa Eira	Ines Iris

0

1

2

entrance

Example1: (search for Luisa Lima)

Input	Output
Ana Anacleto Berta Barata Carla Costa Luisa Lima Elsa Eira Francisca Fé Gloria Gomes Hercilia Hora Ines Iris Joana Jales Luisa Lima Maria Mota Luisa Lima	name=Luisa Lima entrance=0 floor=3 name=Luisa Lima entrance=2 floor=2

Example 2: (search for Zita Zulmira)

Input	Output
Ana Anacleto Berta Barata Carla Costa Luisa Lima Elsa Eira Francisca Fe Gloria Gomes Hercilia Hora Ines Iris Joana Jales Luisa Lima Maria Mota Zita Zulmira	Do not live in the building

Example 3: (search for Luisa)

Input	Output
Ana Berta Barata Carla Luisa Lima Elsa Francisca Fe Gloria Hercilia Hora Ines Joana Jales Luisa Maria Mota Luisa	name=Luisa entrance=2 floor=2

Example 4: (search for Mario M Mota)

Input	Output
Mario M Mota	name=Mario M Mota
Maria M Mata	entrance=0
Maria M Mata	floor=0
Maria M Mata	name=Mario M Mota
Maria M Mata	entrance=1
Mario M Mota	floor=1
Maria M Mata	name=Mario M Mota
Maria M Mata	entrance=2
Maria M Mata	floor=3=Mario M Mota
Maria M Mata	entrance=0
Maria M Mata	floor=0
Mario M Mota	
Mario M Mota	

Example 5: (search for Mario S Mota)

Input	Output
Mario M Mota	Do not live in the building
Maria M Mata	
Maria M Mata	
Maria M Mata	
Maria M Mata	
Mario M Mota	
Maria M Mata	
Maria M Mata	
Maria M Mata	
Maria M Mata	
Maria M Mata	
Mario M Mota	
Mario S Mota	