SESSION 5: TWO-DIMENSIONAL ARRAYS

GOAL:

- Practice with two-dimensional arrays.

EXERCISE:

Management of the purchase of tickets for a bus with **several seats per row**.

Write a program that reads the number of rows on a bus (R) and number of seats per row (C) and generates a two-dimensional array of RxC boolean values (*true/false*), where the value *true* for a position in the array represents that this seat is available and the value *false* represents that it is taken. Initially, all seats are available (all values of the array must be initialized to *true*).

Next, the ticket sale process is carried out, which works as follows:

- The user will be asked to enter the number of tickets to be purchased (M), giving the option that, if he/she does not want to continue with the purchase process, they can enter the value 0 and the program will end.
- If M <= C, the program will check if there are M available seats **in the same row** of the bus (M elements in a same row in the array with value *true*). If so, it will show in which one.
- Otherwise (M > C or no row with M available seats), it will check if there are M available seats on the bus (**in different rows**).
- In both cases, the program will display the entire contents of the vector, then set the M seats to taken (*false*), and finally display the contents of the vector again.
- If there are not M available seats, the program inform the user and display the content of the bus.
- This process shall be repeated as long as the number of available seats is greater than 0.

In addition, the program must be properly documented, including representative names for the defined variables, and the interaction with the user must be properly explained.

EXAMPLE OF EXECUTION:

Assuming that the number of rows entered by the user is 2; the number of seats per row is 4; and the number of tickets requested is 3, 2 and 3; the execution of the program would be as follows:

(SEE NEXT PAGE) =>

```
Introduce the number of rows for the bus
2
Introduce the number of seats per row
4
Introduce the number of tickets to buy (between 1 and 8). Introduce 0 to finish the program 3
The row 0 has enoguh seats available for the tickets requested
The availability in the bus before the sale is as follows:
  true   true   true   true
  true   true   true   true

The availability in the bus after the sale is as follows:
 false  false  false   true
  true   true   true   true

Introduce the number of tickets to buy (between 1 and 5). Introduce 0 to finish the program 2
The row 1 has enoguh seats available for the tickets requested
The availability in the bus before the sale is as follows:
 false  false  false   true
  true   true   true   true

The availability in the bus after the sale is as follows:
 false  false  false   true
 false  false   true   true

Introduce the number of tickets to buy (between 1 and 3). Introduce 0 to finish the program 3
There is not an available row for the requested tickets
There are enough available seats for the 3 tickets requested
The availability in the bus before the sale is as follows:
 false  false  false   true
 false  false   true   true

The availability in the bus after the sale is as follows:
 false  false  false  false
 false  false  false  false

All seats are taken. The ticket sale process will end.

***** End of the program... ******
```