

Objective of the session :

Learn how to manipulate HMI in text mode.

1 Moving a symbol with the keyboard

The first program to be written requires the use of the Crt and Keyboard units to display a symbol at a given position in the output terminal and to recognize events from the keyboard, respectively.

To use these units, you must rely on their online documentation¹ and you can use the examples seen during the course.

Exercices

1. Create a new file `deplace.pas` that contains the main program below

```
begin
  InitKeyBoard()
  deplacement()
  DoneKeyBoard()
end.
```

First, write in the body of the `deplacement` procedure the instructions allowing to retrieve in a variable an event from the keyboard and to write on the screen the result of its conversion to string (`KeyEventToString` function). Keyboard events are displayed in a loop until the 'q' key is pressed. Find the text corresponding to the arrow keys on the keyboard.

2. Modify the `deplacement` procedure so that it uses 2 variables `posX` and `posY` indicating the coordinates of a symbol (use the character 'o' for that) to be displayed. Using the Crt unit's `GotoXY` procedure, display this symbol at an initial coordinate position (2,2) at the beginning of the `deplacement` procedure.
3. Modify the `deplacement` procedure to no longer display the string that corresponds to a keyboard event but to react to the arrow keys on the keyboard. The expected reaction is to move the symbol in the direction corresponding to the pressed arrow. To do this, you must first delete the symbol, calculate its new position and then display it again.
4. Make one last modification to the `deplacement` procedure to impose a minimum and a maximum limits on the displacement. The symbol may not go below coordinates 2 and above coordinates 5 on the abscissa and on the ordinate.

1. <http://www.freepascal.org/docs-html/rtl/>

2 Moving through a labyrinth

The second step of the TP is to display a labyrinth in which the symbol will move. Download from moodle the `labyrinth.pas` and `laby1.txt` files. The pascal program must be completed and the text file contains the description of a given labyrinth.

Exercices

1. Read the contents of these files carefully to understand how they work.
 2. Complete the `affichage` procedure so that it displays on the screen the labyrinth stored in the array `l`. This array represents a map of the labyrinth seen from above. The true value in a box represents a wall (to be displayed with the `#` symbol) and the false value a path (to be displayed with a space).
 3. Copy the previous `deplacement` procedure into this program. Call this procedure after you have called the `affichage` procedure. For the moment, moving the symbol will "erase" the boxes of the labyrinth.
 4. Modify the input parameters of the `deplacement` procedure so that the minimum and the maximum displacement limits are no longer 2 and 5 but are the dimensions of the labyrinth.
 5. Add the array representing the labyrinth as an input parameter of the `deplacement` procedure. Make sure that the movement of the symbol respects the configuration of the labyrinth, i.e. it is not possible to move through a wall.
-

3 Optional exercises

Implement the following improvements :

Exercices

1. Display symbols around the labyrinth (for example, the character "I" on the left and on the right and "-" at the top and at the bottom), to better define its boundaries.
 2. Write other text files representing labyrinths that can be used by your program.
 3. Count and display the number of movements made above or below the labyrinth.
 4. Insert the symbol 'S' in the text file of the labyrinth to represent the output box. The program will now stop when the exit box has been reached.
-