

18/12/2016

**BLM1551**  
**Introduction to Computer Science - I**  
**SEMESTER PROJECT**  
**(Due 12/01/2016)**

**Important Note:** This document might be updated to answer your FAQ, so please frequently check website for updates and check the version number of document to see if it is updated.

**SUBJECT**

You will implement a SYMBOL game

**SYMBOL GAME**

This game aims at reaching the end of a given board and then writing the symbols given in random order as fast as possible.

First, the player would move on a board consisting of holes and stones. On this board, the player should reach the finish line as soon as possible. In each round the player would try to predict one of the stones in the next step on a given board. After reaching the final destination, the system would produce N symbols consisting of numbers and small/big letters display these symbols on the screen until the player press any button. The player should memorize the order and write the symbols in given order. If he/she makes a mistake then the system would give an alert and the system would redisplay the correct order of symbols. When the player finds the correct order of the symbols the system would stop the time and display it on the screen.

You should also display the ranking list of the player based on their finishing time in descending order.

0	1	1	0	2	0	0	0	1	0
0	0	2	1	0	0	2	0	0	1
0	0	3	2	0	0	0	0	0	0
0	0	0	0	3	1	0	0	0	0
1	0	0	0	0	0	0	2	0	0
2	0	0	0	1	0	1	1	0	0

Fig.1 An example of a board consisting of stones (0) and holes (1,2,3,...)

## 2. System Details

This section describes the inputs and the outputs of your implementation. You must design your implementation according to the rules mentioned in this section.

### 2.1 System Parameters (Inputs and Outputs)

The program should ask the following parameter at the beginning of the program

- The length of the board (the number of columns) ( $5 \leq \text{length} \leq 10$ )
- The width of the raceway (the number of rows) ( $8 \leq \text{width} \leq 12$ )
- The ratio of stones and holes (stones  $\geq 70\%$  and holes  $\leq 30\%$ )
- The number of symbols
- The name of the player

The program would give the following outputs:

- The elapsed time
- The ranking list
- 

### Details

- The board (matrix) would consist of  $N$  rows and  $M$  columns
- Rows and columns represent the steps and the possible stones/holes, respectively.
- Each row would consist of 70% stones and 30%holes.

- According to the number of holes, each hole would be represented by 1,2,3... . Thus, if the player selects the hole with number N, the player would automatically move on N step back.
- The name of the player should be given to the system.
- The system should have a menu (some example items could be given as follows: game parameters, play, and the ranking list).
- In each new game the symbols and thus the order of symbols should change. However, in each round during the game the order of symbols should remain.

## 2.2 Interface

The board should be displayed by using underscore character " \_ ", hyphen " - " or vertical strip " | ".

0	1	1	0	2	0	0	0	1	0
0	0	2	1	0	0	2	0	0	1
0	0	3	2	0	0	0	0	0	0
0	0	0	0	3	1	0	0	0	0
1	0	0	0	0	0	0	2	0	0
2	0	0	0	1	0	1	1	0	0

Fig.2 The randomly generated board

*	*	*	*	*	*	*	*	*	*
*	*	*	*	*	*	*	*	*	*
*	*	*	*	*	*	*	*	*	*
*	*	*	*	*	*	*	*	*	*
*	*	*	*	*	*	*	*	*	*
*	*	*	*	*	*	*	*	*	*

Fig. 3 The initial state of the board

*	*	*	*	*	*	*	*	*	*
*	*	*	*	*	*	*	*	*	*
*	*	*	*	*	*	*	*	*	*
p	*	*	*	*	*	*	*	*	*
*	*	*	*	*	*	*	*	*	*
*	p	*	*	*	*	*	*	*	*

Fig. 4 The second round look of the board. P stands for "Player"

*	*	*	*	*	*	p	p	*	p
*	*	*	*	p	p	*	*	*	*
*	*	*	*	*	*	*	*	*	*
p	*	*	p	*	*	*	*	*	*
*	*	p	*	*	*	*	*	*	*
*	p	*	*	*	*	*	*	p	*

Fig. 5 A fully completed board

p	1	*	*	*	*	*	*	*	*
*	*	*	*	*	*	*	*	*	*
*	*	*	*	*	*	*	*	*	*
*	*	*	*	*	*	*	*	*	*
*	*	*	*	*	*	*	*	*	*
*	*	*	*	*	*	*	*	*	*

Fig. 5 The player would move "1" column backward, since he/she fell into the hole with number 1

## 2.3 Symbol Generation

After completing the board, the system would display the randomly generated symbols on the screen.

A 1 B 2 c 3 4 1 2 5 k y m † 1 9

Then, after the user presses a button, the symbols would disappear and the player would try to predict the symbols in its order. This will last until the player writes the same order of generated symbols.