

Part 1

Your functions will ask first the user for a "difficulty" and a board size, then for a coordinate on that board. Your program must check for validity within certain criteria.

Your program must also print out the board.

[The board is created by the background program and populated with 0's and 1's in a way you do not direct]

1	1	1	0	0
1	0	1	1	0
0	1	0	1	1
1	0	1	1	1
1	0	0	0	0

Example board where size 5 was entered.

The coordinate you give is a row and a column where each can be zero to the size of the board less 1. So for the example shown, the row and column values must be in the range 0 to 4.

The aim of the game is to find a line of 1's whose length is the same or exceeds the difficulty level you chose.

Part 2

Here you produce two more functions:

Function 1 Takes as input a cell location and changes the cell location according to two inputs - forward (boolean) and direction (integer 0-3) according to the following (light blue columns):

forward	direction	change to row	change to column
true	0	-1	0
true	1	-1	+1
true	2	0	+1
true	3	-1	-1
false	0	+1	0
false	1	+1	-1
false	2	0	-1
false	3	+1	+1

Function 2 Calculates the score (number of consecutive 1s) in a give direction from some starting point from that point in the directions given above (0-3).

For example, if the location selected was row=2, column=2 for the board at right, the score would be 0 for all directions since the first element is not a 1. If the location selected was row 2, column=3 (shown in light red in the figure) then the scores in each direction would be

Direction 0 - Score 3<-- this is shown in light blue.

1 - Score 2

2 - Score 2

3 - Score 4

Direction 0 can be thought of as N-S of a compass (no change left to right). Direction 1 can be thought of as NE-SW, direction 2 as E-W and direction 3 as NW-SE.

1	1	1	0	0
1	0	1	1	0
0	1	0	1	1
1	0	1	1	1
1	0	0	0	0

Example board where size 5 was entered.

Part 3

You check a user's cell coordinate input (row, column) and see if there is a direction where the score equals or exceeds the difficulty level chosen

Part 4

There may not be a sequence of 1's that allows a "win" from part 3. We say these boards are not valid.

In this part you construct a board that guarantees a win in part 3, that is a board that is valid.