Given a 2-dimensional array containing positive integers (not overlapping) simulating a maze, a robot is placed at position (x,y). The robot can only go in 4 directions (top, bottom, left, right). The robot will choose the direction (cell) with a large value to go, the cells that go away will not go back. Score is calculated by the total value of the cells the robot passes through. Example robot is located at (0,0) -> 2 3 16 56 87 100 101 66 543 200 150 154 178 76 54 43 27 4

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 2 | 1 | 14 | 12 | 17 |
| 3 | 16 | 22 | 91 | 23 |
| 4 | 56 | 87 | 31 | 65 |
| 27 | 43 | 90 | 100 | 101 |
| 76 | 54 | 32 | 99 | 66 |
| 178 | 154 | 150 | 200 | 543 |

File Input.txt:

6 5

2 1 14 12 17

3 16 22 91 23

4 56 87 31 65

27 43 90 100 101

76 54 32 99 66

178 154 150 200 543

File output.txt:

19

2 3 16 56 87 90 100 101 66 543 200 150 154 178 76 54 43 27 4

Let's build an application that ensures the following basic features:

1. Apply the following knowledge: recursion/struct/dynamic allocation/reading/writing files/functions,...

2. Apply graphic knowledge: use SFML libraries,... to visualize robot path, simulate step by step, graphics for matrix,...

3. Install the output algorithm.

4. When entering the game will appear the game interface (with graphics for the interface) with buttons including:

- Mode 1: Automatic PvP. Details of this mode are as follows:

+ Generate a random matrix with positive integers (with graphics for the matrix)

+ Display initial position options for robot 1 and robot 2 so that the player can choose any starting position on the matrix but cannot select invalid cells such as selecting cells

outside the matrix, selecting cells that match other robots,... (with graphics for both robots)

+ Robot 1 will move first, the cells that robot 1 has passed will draw a red path for the player to know that the cell has gone, similar to robot 2, it will draw a blue path with the cells already gone. That is, wherever the two robots go, they will draw the path of the two robots there to visualize the robot path (Apply graphics for two robots, moving effects of two robots)

+ There is a table to display the current total score on the screen for players to easily track the score. (with graphics for the table showing the current total score)

+ Only 1 robot can move each turn. If that cell has gone, the other robot can't walk, can't go out of the maze. If the user reluctantly mistypes, output the message and ask to re-enter the direction.

+ The program will stop if the 4 cells around the robot can't be moved anymore, this will calculate the sum based on the values ​​in the matrix the player has passed. The winning robot is the robot with the largest total value of the passed cells.

+ Export the results of the total score of two players and export the cells both robots have passed. Also output the cells where the two robots overlap.

- Mode 2: Adjust PvP

+ Same as Mode 1 but instead of letting the Robot move automatically, the user will enter (W/A/D/S) (top, left, right, bottom) with Robot 1, and Robot 2 will enter (I) /J/L/K) (top, left, right, bottom) to move. Same requirement as above.

- Exit: Exit the software

+ Before exiting, confirm if the player wants to exit? If yes, exit, if not, then quit.