# KAMRAN BALAYEV 17011904

The purpose of this project is to find same number pairs and fill their locations with the  gap.  Then the elements of matrix will slip down and after the slipping process, if there is no same number pairs, program will fill the empty spaces with random numbers.The program will repeat this process until it can not find the same number pairs(only 4 neighborhood connection).After this process program will requires from the user to enter 1 in order to continue the game or enter 0 for finishing the game. If user enter 0 the game will be finished  and the score will be printed (score equals the square of number of exploded elements). If user enter 1 he/she should replace the elements of matrix in order to explode the same number pairs.

USED DATA STRUCTURES

- 2 dimensional character array

Purpose of this matrix is to store the numbers pairs.

USED DATA TYPES

- Integer
- Character

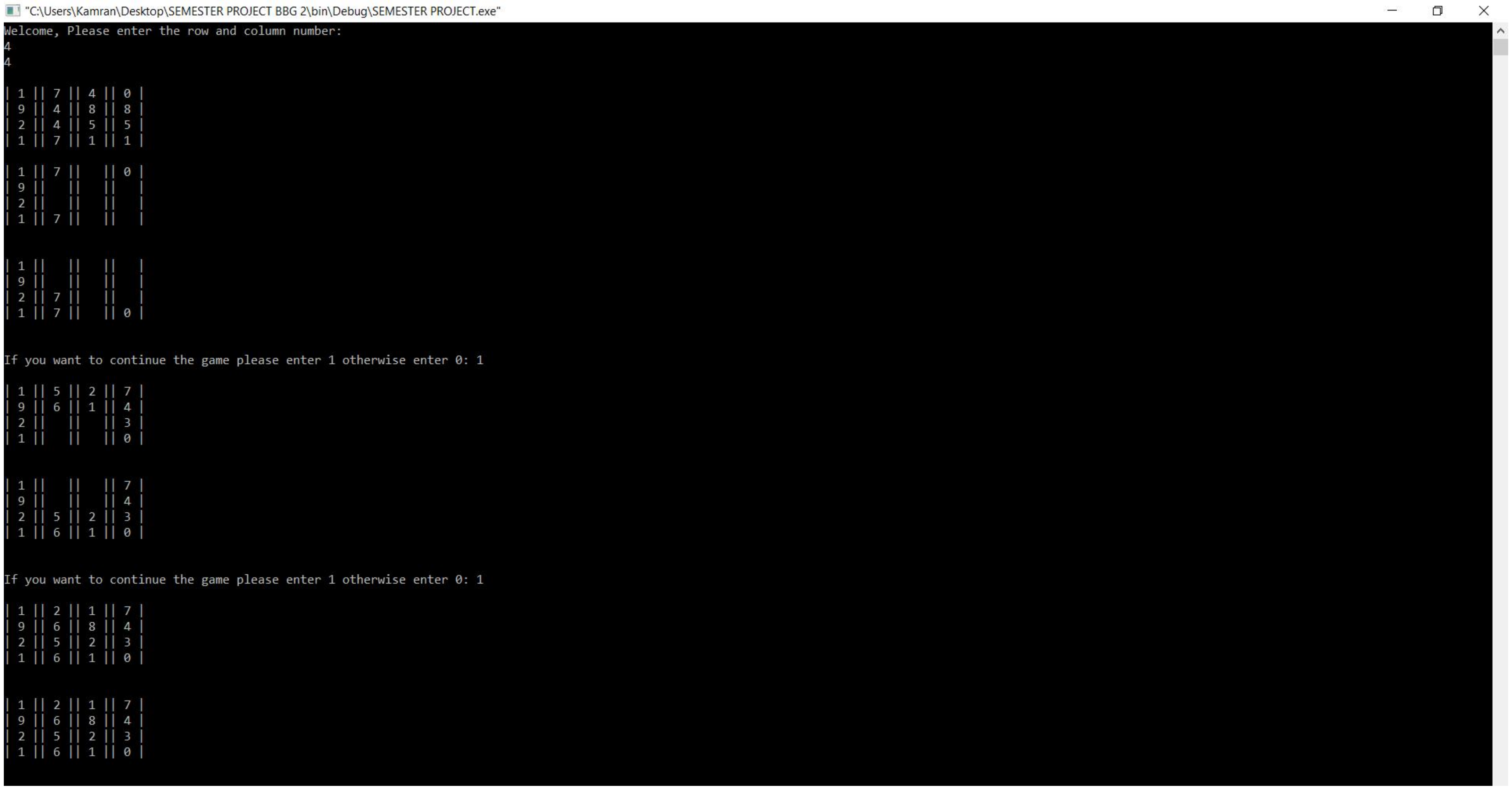
Total number of functions: 6

- The aim of first function is delaying the printing process in order to reach the qualified result.
- The purpose of second function is printing matrix.
- The third function is recursive function which is the most crucial part of the program.This function requires 3 properties from main function.

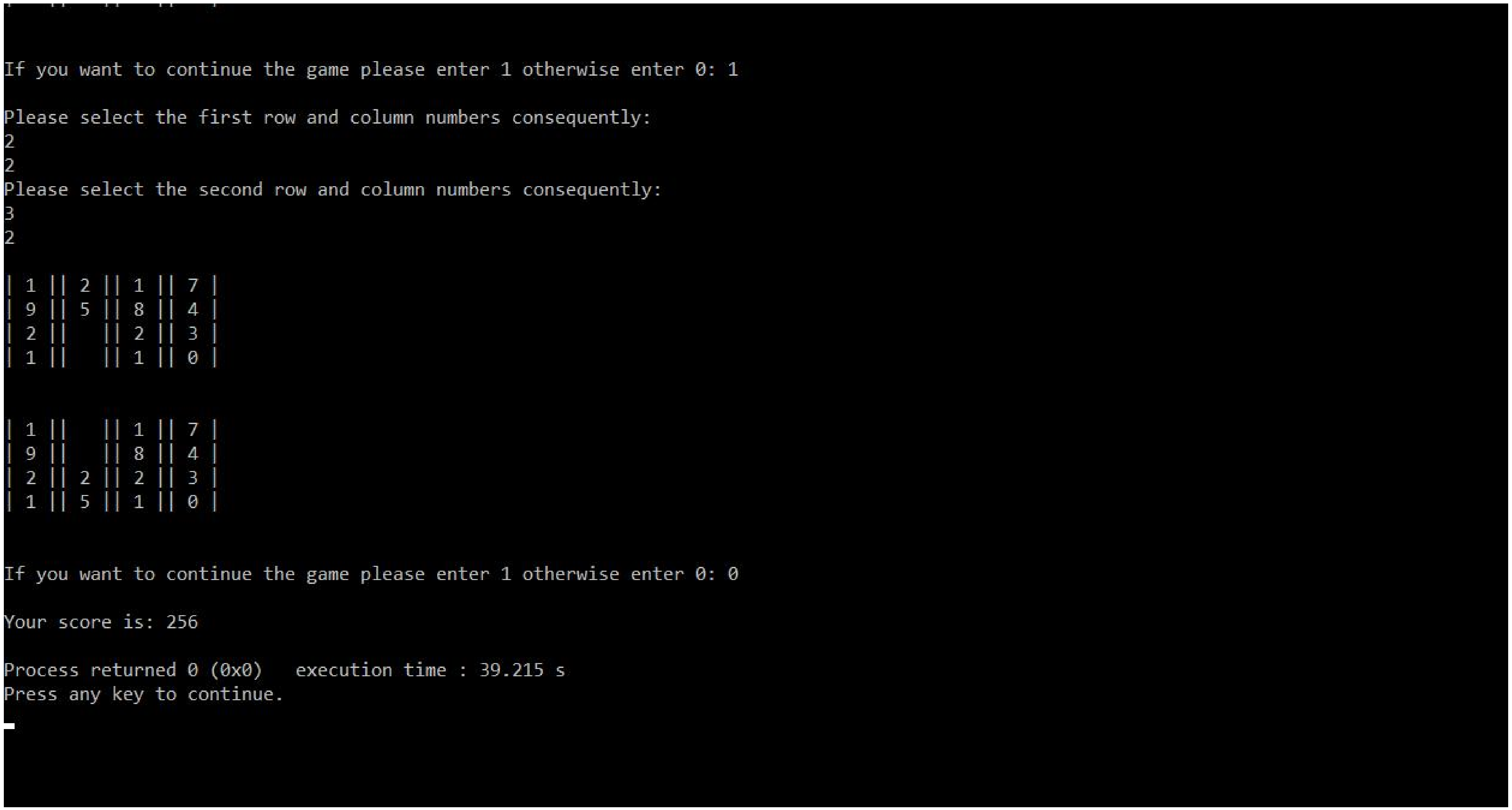
  First one is gameboard matrix which stores the number pairs and the other ones are row and column numbers.The final one is score. I declared the score variable in the main function. If i declared it in recursive function, when the function will calls itself the value of score will reset.

  Firstly, i control the number pairs (not all of them only 4 neighborhood connection), then the same number pairs will explode and new version of matrix will be printed.After the printing process the slipping one starts.The final part of this function is nested for loops in order to determine that the function will call itself or not.

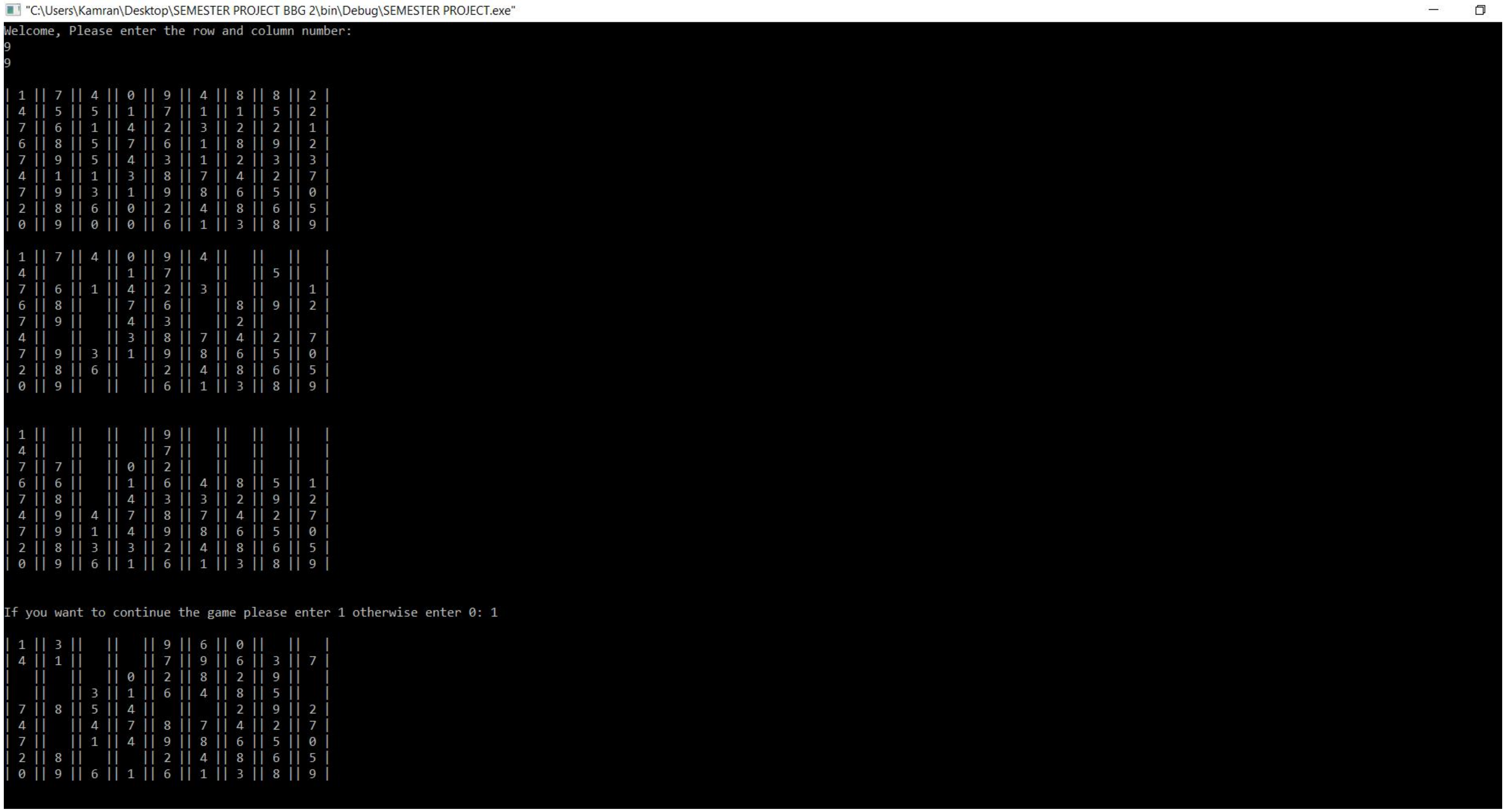
- The fourth function is random element creator  which will start after the slipping process.
- The fifth function asks user to enter row and column numbers in order to replace elements of matrix via the usage of their indices.
- The final function is main function which includes 2 nested for loops, which create random elements in order to fill the matrix. And finally, the recursion function starts.


"C:\Users\Kamran\Desktop\SEMESTER PROJECT BBG 2\bin\Debug\SEMESTER PROJECT.exe"

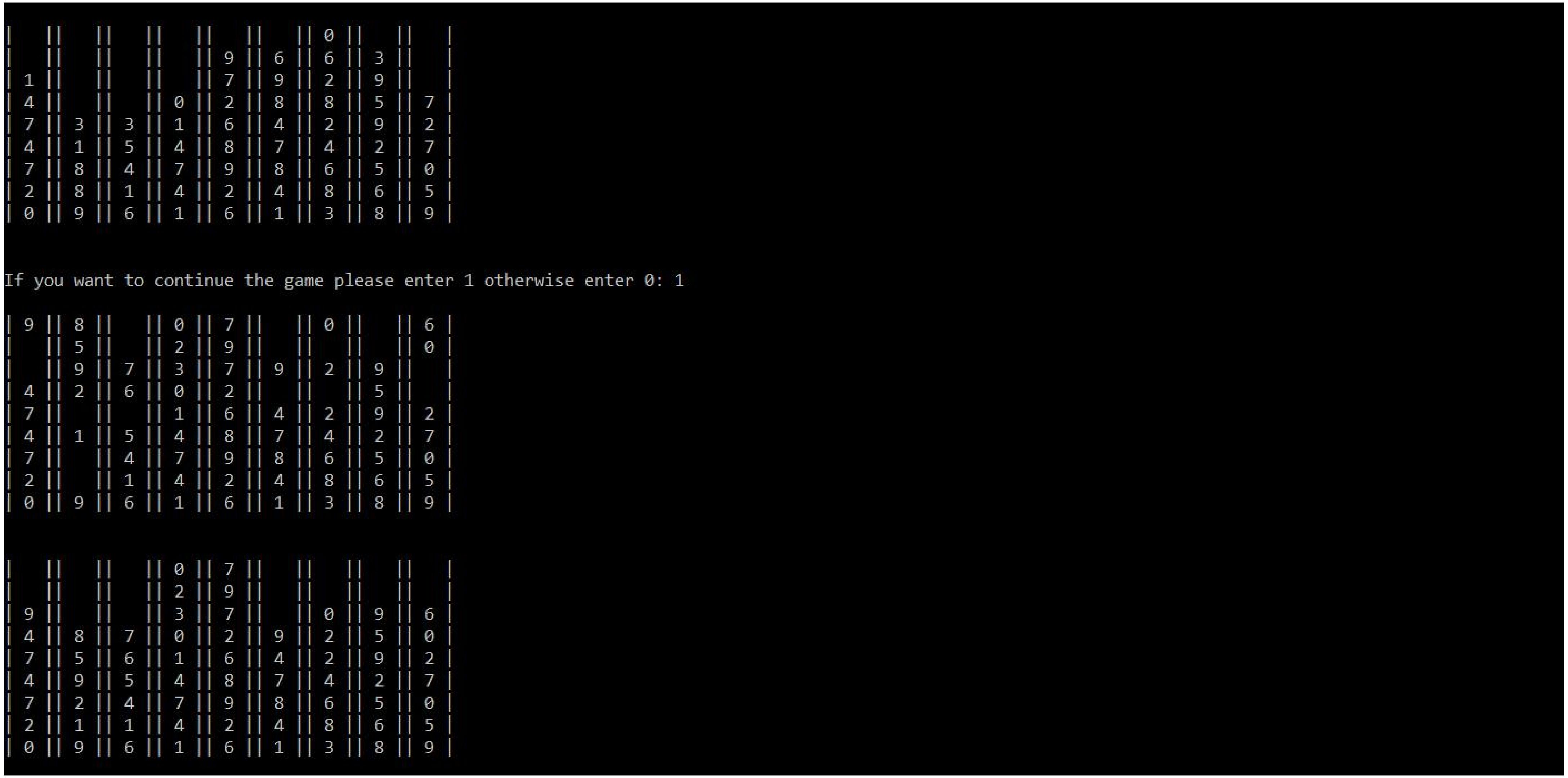
```
Welcome, Please enter the row and column number:
4
4

| 1 || 7 || 4 || 0 |
| 9 || 4 || 8 || 8 |
| 2 || 4 || 5 || 5 |
| 1 || 7 || 1 || 1 |

| 1 || 7 ||   || 0 |
| 9 ||   ||   ||   |
| 2 ||   ||   ||   |
| 1 || 7 ||   ||   |


| 1 ||   ||   ||   |
| 9 ||   ||   ||   |
| 2 || 7 ||   ||   |
| 1 || 7 ||   || 0 |


If you want to continue the game please enter 1 otherwise enter 0: 1

| 1 || 5 || 2 || 7 |
| 9 || 6 || 1 || 4 |
| 2 ||   ||   || 3 |
| 1 ||   ||   || 0 |


| 1 ||   ||   || 7 |
| 9 ||   ||   || 4 |
| 2 || 5 || 2 || 3 |
| 1 || 6 || 1 || 0 |


If you want to continue the game please enter 1 otherwise enter 0: 1

| 1 || 2 || 1 || 7 |
| 9 || 6 || 8 || 4 |
| 2 || 5 || 2 || 3 |
| 1 || 6 || 1 || 0 |


| 1 || 2 || 1 || 7 |
| 9 || 6 || 8 || 4 |
| 2 || 5 || 2 || 3 |
| 1 || 6 || 1 || 0 |
```



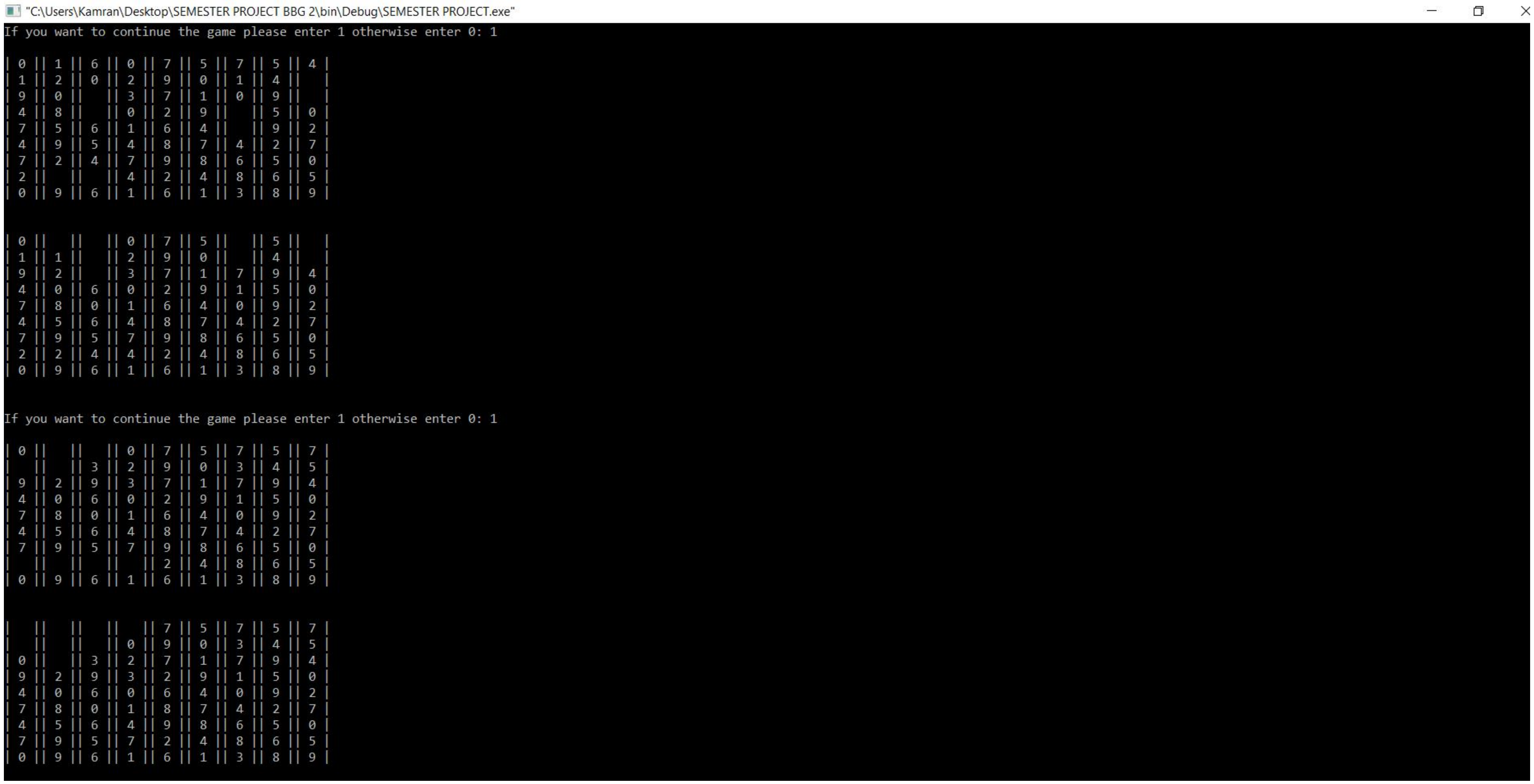
```
If you want to continue the game please enter 1 otherwise enter 0: 1

Please select the first row and column numbers consequently:
2
2
Please select the second row and column numbers consequently:
3
2

| 1 || 2 || 1 || 7 |
| 9 || 5 || 8 || 4 |
| 2 ||   || 2 || 3 |
| 1 ||   || 1 || 0 |


| 1 ||   || 1 || 7 |
| 9 ||   || 8 || 4 |
| 2 || 2 || 2 || 3 |
| 1 || 5 || 1 || 0 |


If you want to continue the game please enter 1 otherwise enter 0: 0

Your score is: 256

Process returned 0 (0x0)   execution time : 39.215 s
Press any key to continue.
```


"C:\Users\Kamran\Desktop\SEMESTER PROJECT BBG 2\bin\Debug\SEMESTER PROJECT.exe"

```
Welcome, Please enter the row and column number:
9
9

| 1 || 7 || 4 || 0 || 9 || 4 || 8 || 8 || 2 |
| 4 || 5 || 5 || 1 || 7 || 1 || 1 || 5 || 2 |
| 7 || 6 || 1 || 4 || 2 || 3 || 2 || 2 || 1 |
| 6 || 8 || 5 || 7 || 6 || 1 || 8 || 9 || 2 |
| 7 || 9 || 5 || 4 || 3 || 1 || 2 || 3 || 3 |
| 4 || 1 || 1 || 3 || 8 || 7 || 4 || 2 || 7 |
| 7 || 9 || 3 || 1 || 9 || 8 || 6 || 5 || 0 |
| 2 || 8 || 6 || 0 || 2 || 4 || 8 || 6 || 5 |
| 0 || 9 || 0 || 0 || 6 || 1 || 3 || 8 || 9 |

| 1 || 7 || 4 || 0 || 9 || 4 ||   ||   ||   |
| 4 ||   ||   || 1 || 7 ||   ||   || 5 ||   |
| 7 || 6 || 1 || 4 || 2 || 3 ||   ||   || 1 |
| 6 || 8 ||   || 7 || 6 ||   || 8 || 9 || 2 |
| 7 || 9 ||   || 4 || 3 ||   || 2 ||   ||   |
| 4 ||   ||   || 3 || 8 || 7 || 4 || 2 || 7 |
| 7 || 9 || 3 || 1 || 9 || 8 || 6 || 5 || 0 |
| 2 || 8 || 6 ||   || 2 || 4 || 8 || 6 || 5 |
| 0 || 9 ||   ||   || 6 || 1 || 3 || 8 || 9 |


| 1 ||   ||   ||   || 9 ||   ||   ||   ||   |
| 4 ||   ||   ||   || 7 ||   ||   ||   ||   |
| 7 || 7 ||   || 0 || 2 ||   ||   ||   ||   |
| 6 || 6 ||   || 1 || 6 || 4 || 8 || 5 || 1 |
| 7 || 8 ||   || 4 || 3 || 3 || 2 || 9 || 2 |
| 4 || 9 || 4 || 7 || 8 || 7 || 4 || 2 || 7 |
| 7 || 9 || 1 || 4 || 9 || 8 || 6 || 5 || 0 |
| 2 || 8 || 3 || 3 || 2 || 4 || 8 || 6 || 5 |
| 0 || 9 || 6 || 1 || 6 || 1 || 3 || 8 || 9 |


If you want to continue the game please enter 1 otherwise enter 0: 1

| 1 || 3 ||   ||   || 9 || 6 || 0 ||   ||   |
| 4 || 1 ||   ||   || 7 || 9 || 6 || 3 || 7 |
|   ||   ||   || 0 || 2 || 8 || 2 || 9 ||   |
|   ||   || 3 || 1 || 6 || 4 || 8 || 5 ||   |
| 7 || 8 || 5 || 4 ||   ||   || 2 || 9 || 2 |
| 4 ||   || 4 || 7 || 8 || 7 || 4 || 2 || 7 |
| 7 ||   || 1 || 4 || 9 || 8 || 6 || 5 || 0 |
| 2 || 8 ||   ||   || 2 || 4 || 8 || 6 || 5 |
| 0 || 9 || 6 || 1 || 6 || 1 || 3 || 8 || 9 |
```

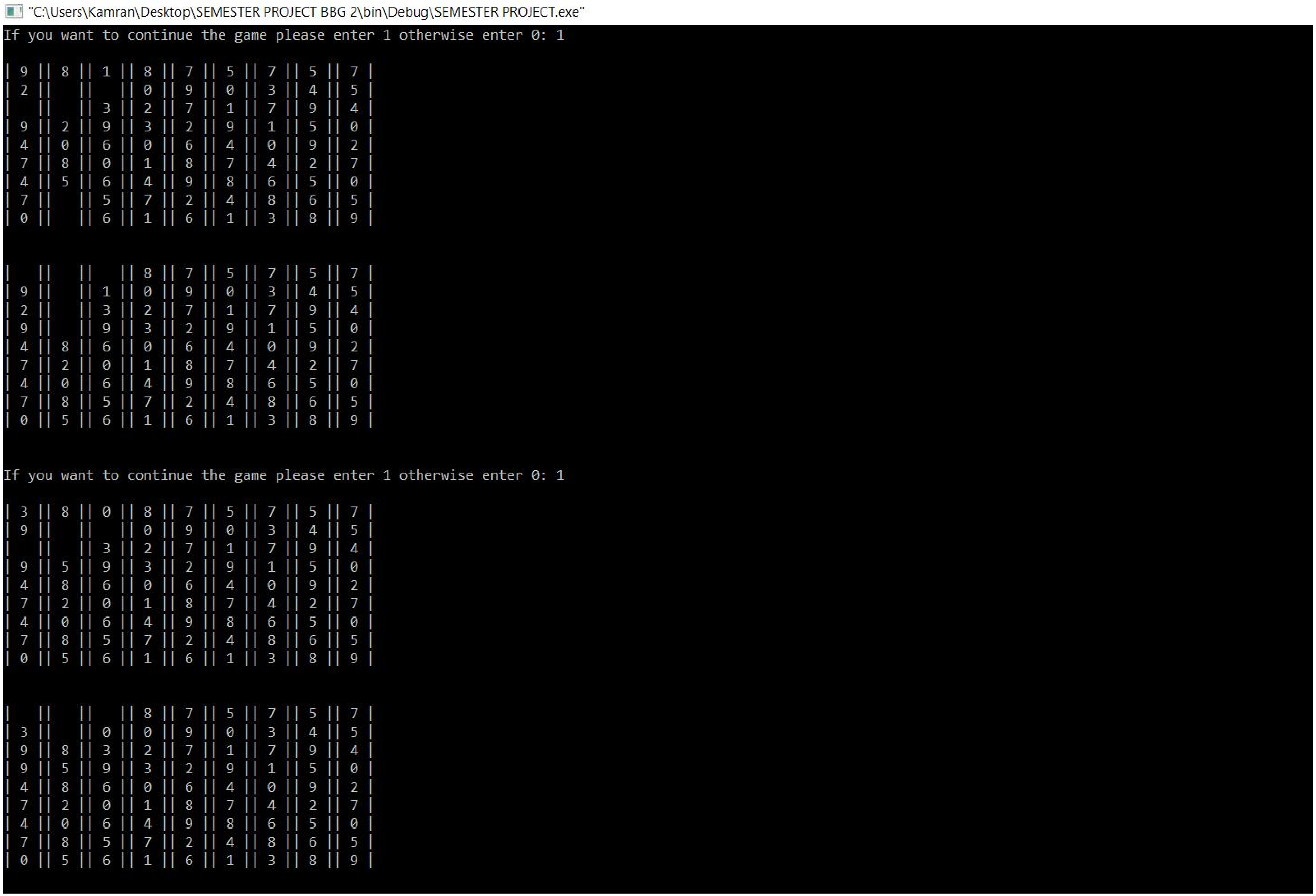


```
|   ||   ||   ||   ||   ||   || 0 ||   ||   |
|   ||   ||   ||   || 9 || 6 || 6 || 3 ||   |
| 1 ||   ||   ||   || 7 || 9 || 2 || 9 ||   |
| 4 ||   ||   || 0 || 2 || 8 || 8 || 5 || 7 |
| 7 || 3 || 3 || 1 || 6 || 4 || 2 || 9 || 2 |
| 4 || 1 || 5 || 4 || 8 || 7 || 4 || 2 || 7 |
| 7 || 8 || 4 || 7 || 9 || 8 || 6 || 5 || 0 |
| 2 || 8 || 1 || 4 || 2 || 4 || 8 || 6 || 5 |
| 0 || 9 || 6 || 1 || 6 || 1 || 3 || 8 || 9 |


If you want to continue the game please enter 1 otherwise enter 0: 1

| 9 || 8 ||   || 0 || 7 ||   || 0 ||   || 6 |
|   || 5 ||   || 2 || 9 ||   ||   ||   || 0 |
|   || 9 || 7 || 3 || 7 || 9 || 2 || 9 ||   |
| 4 || 2 || 6 || 0 || 2 ||   ||   || 5 ||   |
| 7 ||   ||   || 1 || 6 || 4 || 2 || 9 || 2 |
| 4 || 1 || 5 || 4 || 8 || 7 || 4 || 2 || 7 |
| 7 ||   || 4 || 7 || 9 || 8 || 6 || 5 || 0 |
| 2 ||   || 1 || 4 || 2 || 4 || 8 || 6 || 5 |
| 0 || 9 || 6 || 1 || 6 || 1 || 3 || 8 || 9 |


|   ||   ||   || 0 || 7 ||   ||   ||   ||   |
|   ||   ||   || 2 || 9 ||   ||   ||   ||   |
| 9 ||   ||   || 3 || 7 ||   || 0 || 9 || 6 |
| 4 || 8 || 7 || 0 || 2 || 9 || 2 || 5 || 0 |
| 7 || 5 || 6 || 1 || 6 || 4 || 2 || 9 || 2 |
| 4 || 9 || 5 || 4 || 8 || 7 || 4 || 2 || 7 |
| 7 || 2 || 4 || 7 || 9 || 8 || 6 || 5 || 0 |
| 2 || 1 || 1 || 4 || 2 || 4 || 8 || 6 || 5 |
| 0 || 9 || 6 || 1 || 6 || 1 || 3 || 8 || 9 |
```



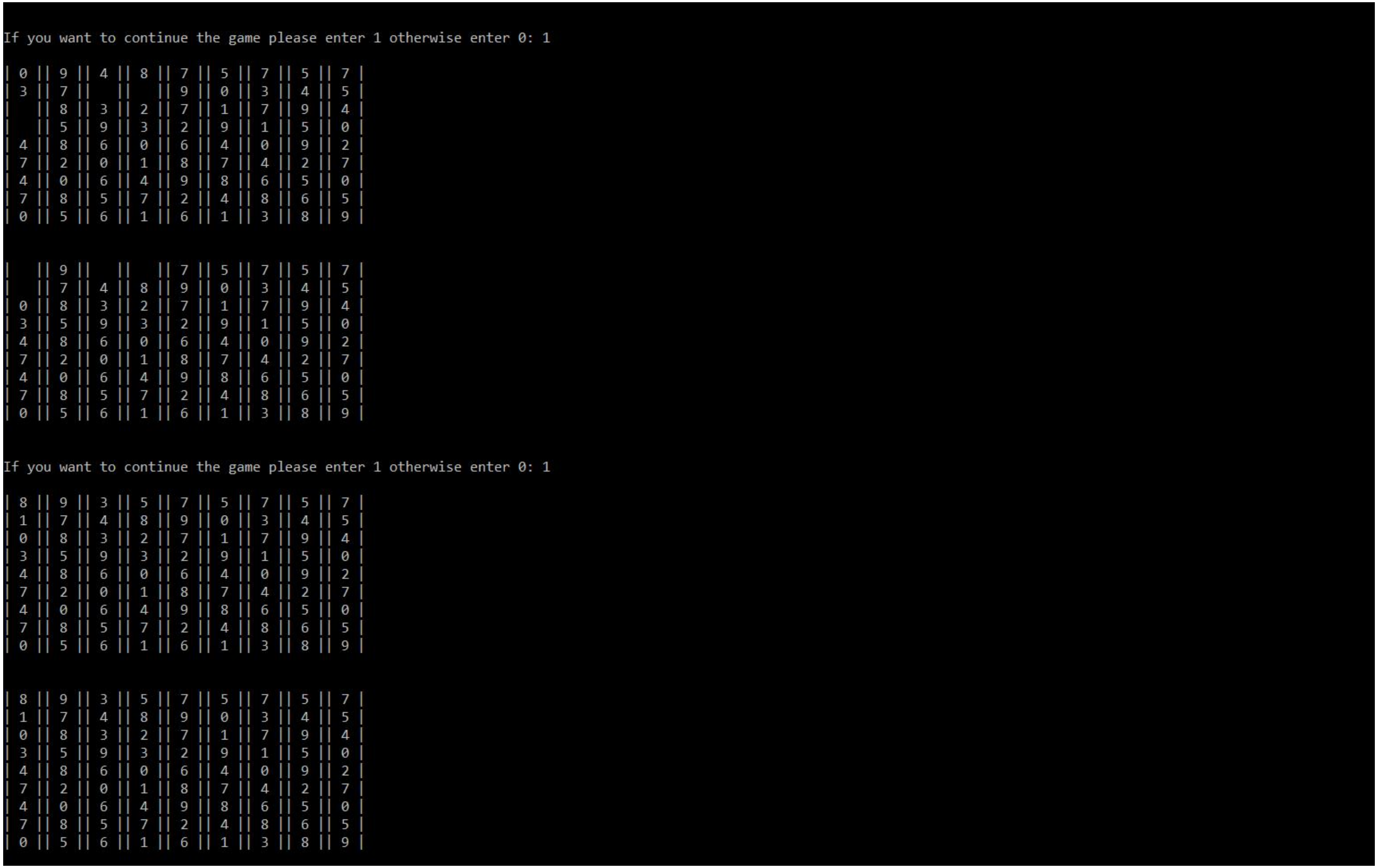
"C:\Users\Kamran\Desktop\SEMESTER PROJECT BBG 2\bin\Debug\SEMESTER PROJECT.exe"

```
If you want to continue the game please enter 1 otherwise enter 0: 1

| 0 || 1 || 6 || 0 || 7 || 5 || 7 || 5 || 4 |
| 1 || 2 || 0 || 2 || 9 || 0 || 1 || 4 ||   |
| 9 || 0 ||   || 3 || 7 || 1 || 0 || 9 ||   |
| 4 || 8 ||   || 0 || 2 || 9 ||   || 5 || 0 |
| 7 || 5 || 6 || 1 || 6 || 4 ||   || 9 || 2 |
| 4 || 9 || 5 || 4 || 8 || 7 || 4 || 2 || 7 |
| 7 || 2 || 4 || 7 || 9 || 8 || 6 || 5 || 0 |
| 2 ||   ||   || 4 || 2 || 4 || 8 || 6 || 5 |
| 0 || 9 || 6 || 1 || 6 || 1 || 3 || 8 || 9 |


| 0 ||   ||   || 0 || 7 || 5 ||   || 5 ||   |
| 1 || 1 ||   || 2 || 9 || 0 ||   || 4 ||   |
| 9 || 2 ||   || 3 || 7 || 1 || 7 || 9 || 4 |
| 4 || 0 || 6 || 0 || 2 || 9 || 1 || 5 || 0 |
| 7 || 8 || 0 || 1 || 6 || 4 || 0 || 9 || 2 |
| 4 || 5 || 6 || 4 || 8 || 7 || 4 || 2 || 7 |
| 7 || 9 || 5 || 7 || 9 || 8 || 6 || 5 || 0 |
| 2 || 2 || 4 || 4 || 2 || 4 || 8 || 6 || 5 |
| 0 || 9 || 6 || 1 || 6 || 1 || 3 || 8 || 9 |


If you want to continue the game please enter 1 otherwise enter 0: 1

| 0 ||   ||   || 0 || 7 || 5 || 7 || 5 || 7 |
|   ||   || 3 || 2 || 9 || 0 || 3 || 4 || 5 |
| 9 || 2 || 9 || 3 || 7 || 1 || 7 || 9 || 4 |
| 4 || 0 || 6 || 0 || 2 || 9 || 1 || 5 || 0 |
| 7 || 8 || 0 || 1 || 6 || 4 || 0 || 9 || 2 |
| 4 || 5 || 6 || 4 || 8 || 7 || 4 || 2 || 7 |
| 7 || 9 || 5 || 7 || 9 || 8 || 6 || 5 || 0 |
|   ||   ||   ||   || 2 || 4 || 8 || 6 || 5 |
| 0 || 9 || 6 || 1 || 6 || 1 || 3 || 8 || 9 |


|   ||   ||   ||   || 7 || 5 || 7 || 5 || 7 |
|   ||   ||   || 0 || 9 || 0 || 3 || 4 || 5 |
| 0 ||   || 3 || 2 || 7 || 1 || 7 || 9 || 4 |
| 9 || 2 || 9 || 3 || 2 || 9 || 1 || 5 || 0 |
| 4 || 0 || 6 || 0 || 6 || 4 || 0 || 9 || 2 |
| 7 || 8 || 0 || 1 || 8 || 7 || 4 || 2 || 7 |
| 4 || 5 || 6 || 4 || 9 || 8 || 6 || 5 || 0 |
| 7 || 9 || 5 || 7 || 2 || 4 || 8 || 6 || 5 |
| 0 || 9 || 6 || 1 || 6 || 1 || 3 || 8 || 9 |
```

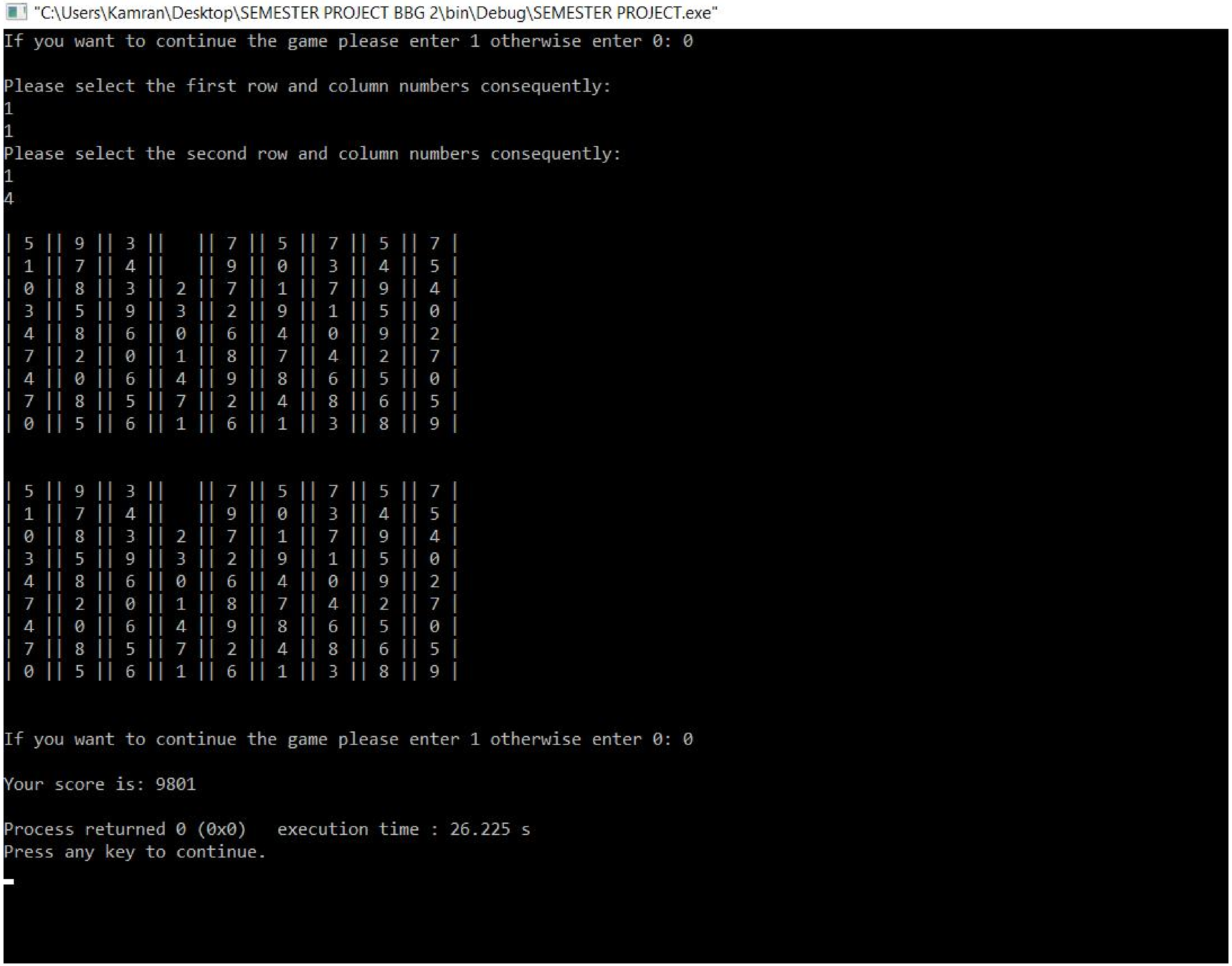


"C:\Users\Kamran\Desktop\SEMESTER PROJECT BBG 2\bin\Debug\SEMESTER PROJECT.exe"

```
If you want to continue the game please enter 1 otherwise enter 0: 1

| 9 || 8 || 1 || 8 || 7 || 5 || 7 || 5 || 7 |
| 2 ||   ||   || 0 || 9 || 0 || 3 || 4 || 5 |
|   ||   || 3 || 2 || 7 || 1 || 7 || 9 || 4 |
| 9 || 2 || 9 || 3 || 2 || 9 || 1 || 5 || 0 |
| 4 || 0 || 6 || 0 || 6 || 4 || 0 || 9 || 2 |
| 7 || 8 || 0 || 1 || 8 || 7 || 4 || 2 || 7 |
| 4 || 5 || 6 || 4 || 9 || 8 || 6 || 5 || 0 |
| 7 ||   || 5 || 7 || 2 || 4 || 8 || 6 || 5 |
| 0 ||   || 6 || 1 || 6 || 1 || 3 || 8 || 9 |


|   ||   ||   || 8 || 7 || 5 || 7 || 5 || 7 |
| 9 ||   || 1 || 0 || 9 || 0 || 3 || 4 || 5 |
| 2 ||   || 3 || 2 || 7 || 1 || 7 || 9 || 4 |
| 9 ||   || 9 || 3 || 2 || 9 || 1 || 5 || 0 |
| 4 || 8 || 6 || 0 || 6 || 4 || 0 || 9 || 2 |
| 7 || 2 || 0 || 1 || 8 || 7 || 4 || 2 || 7 |
| 4 || 0 || 6 || 4 || 9 || 8 || 6 || 5 || 0 |
| 7 || 8 || 5 || 7 || 2 || 4 || 8 || 6 || 5 |
| 0 || 5 || 6 || 1 || 6 || 1 || 3 || 8 || 9 |


If you want to continue the game please enter 1 otherwise enter 0: 1

| 3 || 8 || 0 || 8 || 7 || 5 || 7 || 5 || 7 |
| 9 ||   ||   || 0 || 9 || 0 || 3 || 4 || 5 |
|   ||   || 3 || 2 || 7 || 1 || 7 || 9 || 4 |
| 9 || 5 || 9 || 3 || 2 || 9 || 1 || 5 || 0 |
| 4 || 8 || 6 || 0 || 6 || 4 || 0 || 9 || 2 |
| 7 || 2 || 0 || 1 || 8 || 7 || 4 || 2 || 7 |
| 4 || 0 || 6 || 4 || 9 || 8 || 6 || 5 || 0 |
| 7 || 8 || 5 || 7 || 2 || 4 || 8 || 6 || 5 |
| 0 || 5 || 6 || 1 || 6 || 1 || 3 || 8 || 9 |


|   ||   ||   || 8 || 7 || 5 || 7 || 5 || 7 |
| 3 ||   || 0 || 0 || 9 || 0 || 3 || 4 || 5 |
| 9 || 8 || 3 || 2 || 7 || 1 || 7 || 9 || 4 |
| 9 || 5 || 9 || 3 || 2 || 9 || 1 || 5 || 0 |
| 4 || 8 || 6 || 0 || 6 || 4 || 0 || 9 || 2 |
| 7 || 2 || 0 || 1 || 8 || 7 || 4 || 2 || 7 |
| 4 || 0 || 6 || 4 || 9 || 8 || 6 || 5 || 0 |
| 7 || 8 || 5 || 7 || 2 || 4 || 8 || 6 || 5 |
| 0 || 5 || 6 || 1 || 6 || 1 || 3 || 8 || 9 |
```



```
If you want to continue the game please enter 1 otherwise enter 0: 1

| 0 || 9 || 4 || 8 || 7 || 5 || 7 || 5 || 7 |
| 3 || 7 ||   ||   || 9 || 0 || 3 || 4 || 5 |
|   || 8 || 3 || 2 || 7 || 1 || 7 || 9 || 4 |
|   || 5 || 9 || 3 || 2 || 9 || 1 || 5 || 0 |
| 4 || 8 || 6 || 0 || 6 || 4 || 0 || 9 || 2 |
| 7 || 2 || 0 || 1 || 8 || 7 || 4 || 2 || 7 |
| 4 || 0 || 6 || 4 || 9 || 8 || 6 || 5 || 0 |
| 7 || 8 || 5 || 7 || 2 || 4 || 8 || 6 || 5 |
| 0 || 5 || 6 || 1 || 6 || 1 || 3 || 8 || 9 |


|   || 9 ||   ||   || 7 || 5 || 7 || 5 || 7 |
|   || 7 || 4 || 8 || 9 || 0 || 3 || 4 || 5 |
| 0 || 8 || 3 || 2 || 7 || 1 || 7 || 9 || 4 |
| 3 || 5 || 9 || 3 || 2 || 9 || 1 || 5 || 0 |
| 4 || 8 || 6 || 0 || 6 || 4 || 0 || 9 || 2 |
| 7 || 2 || 0 || 1 || 8 || 7 || 4 || 2 || 7 |
| 4 || 0 || 6 || 4 || 9 || 8 || 6 || 5 || 0 |
| 7 || 8 || 5 || 7 || 2 || 4 || 8 || 6 || 5 |
| 0 || 5 || 6 || 1 || 6 || 1 || 3 || 8 || 9 |


If you want to continue the game please enter 1 otherwise enter 0: 1

| 8 || 9 || 3 || 5 || 7 || 5 || 7 || 5 || 7 |
| 1 || 7 || 4 || 8 || 9 || 0 || 3 || 4 || 5 |
| 0 || 8 || 3 || 2 || 7 || 1 || 7 || 9 || 4 |
| 3 || 5 || 9 || 3 || 2 || 9 || 1 || 5 || 0 |
| 4 || 8 || 6 || 0 || 6 || 4 || 0 || 9 || 2 |
| 7 || 2 || 0 || 1 || 8 || 7 || 4 || 2 || 7 |
| 4 || 0 || 6 || 4 || 9 || 8 || 6 || 5 || 0 |
| 7 || 8 || 5 || 7 || 2 || 4 || 8 || 6 || 5 |
| 0 || 5 || 6 || 1 || 6 || 1 || 3 || 8 || 9 |


| 8 || 9 || 3 || 5 || 7 || 5 || 7 || 5 || 7 |
| 1 || 7 || 4 || 8 || 9 || 0 || 3 || 4 || 5 |
| 0 || 8 || 3 || 2 || 7 || 1 || 7 || 9 || 4 |
| 3 || 5 || 9 || 3 || 2 || 9 || 1 || 5 || 0 |
| 4 || 8 || 6 || 0 || 6 || 4 || 0 || 9 || 2 |
| 7 || 2 || 0 || 1 || 8 || 7 || 4 || 2 || 7 |
| 4 || 0 || 6 || 4 || 9 || 8 || 6 || 5 || 0 |
| 7 || 8 || 5 || 7 || 2 || 4 || 8 || 6 || 5 |
| 0 || 5 || 6 || 1 || 6 || 1 || 3 || 8 || 9 |
```



```
"C:\Users\Kamran\Desktop\SEMESTER PROJECT BBG 2\bin\Debug\SEMESTER PROJECT.exe"
If you want to continue the game please enter 1 otherwise enter 0: 0

Please select the first row and column numbers consequently:
1
1
Please select the second row and column numbers consequently:
1
4

| 5 || 9 || 3 ||   || 7 || 5 || 7 || 5 || 7 |
| 1 || 7 || 4 ||   || 9 || 0 || 3 || 4 || 5 |
| 0 || 8 || 3 || 2 || 7 || 1 || 7 || 9 || 4 |
| 3 || 5 || 9 || 3 || 2 || 9 || 1 || 5 || 0 |
| 4 || 8 || 6 || 0 || 6 || 4 || 0 || 9 || 2 |
| 7 || 2 || 0 || 1 || 8 || 7 || 4 || 2 || 7 |
| 4 || 0 || 6 || 4 || 9 || 8 || 6 || 5 || 0 |
| 7 || 8 || 5 || 7 || 2 || 4 || 8 || 6 || 5 |
| 0 || 5 || 6 || 1 || 6 || 1 || 3 || 8 || 9 |


| 5 || 9 || 3 ||   || 7 || 5 || 7 || 5 || 7 |
| 1 || 7 || 4 ||   || 9 || 0 || 3 || 4 || 5 |
| 0 || 8 || 3 || 2 || 7 || 1 || 7 || 9 || 4 |
| 3 || 5 || 9 || 3 || 2 || 9 || 1 || 5 || 0 |
| 4 || 8 || 6 || 0 || 6 || 4 || 0 || 9 || 2 |
| 7 || 2 || 0 || 1 || 8 || 7 || 4 || 2 || 7 |
| 4 || 0 || 6 || 4 || 9 || 8 || 6 || 5 || 0 |
| 7 || 8 || 5 || 7 || 2 || 4 || 8 || 6 || 5 |
| 0 || 5 || 6 || 1 || 6 || 1 || 3 || 8 || 9 |


If you want to continue the game please enter 1 otherwise enter 0: 0

Your score is: 9801

Process returned 0 (0x0)   execution time : 26.225 s
Press any key to continue.
```

```
//KAMRAN BALAYEV 17011904
#include <stdlib.h>
#include <stdio.h>
#include <time.h>
#include <conio.h>
void delay(int number_of_seconds)
{
   // Converting time into milli_seconds
   int milli_seconds = 1000 * number_of_seconds;

   // Stroing start time
   clock_t start_time = clock();

   // looping till required time is not acheived
   while (clock() < start_time + milli_seconds)
      ;
}
int printMatrix(char gameBoard[100][100],int row,int column){
   int i,j;
   for(i=0;i<row;i++){
      for(j=0;j<column;j++){
         printf("| %c |" ,gameBoard[i][j]);
      };
      printf("\n");
   };
}
//recursive function: purpose of this function is to find the same number pairs and make their places empty
//and slide down elements
void recursion(char gameBoard[100][100],int row,int column,int score){
   int i,j,exit,count=0,end,endFlag;
```

```
char flag;
endFlag=0;
for(i=0;i<row;i++){
  for(j=0;j<column;j++){
    if(gameBoard[i][j]==gameBoard[i+1][j]){
      gameBoard[i][j]=' ';
      score++;
      if(gameBoard[i+1][j]==gameBoard[i+2][j]){
        if(gameBoard[i+2][j]==gameBoard[i+3][j]){
          gameBoard[i+3][j]=' ';
          score++;
        }
        gameBoard[i+2][j]=' ';
        score++;
      }
      else if(gameBoard[i+1][j]==gameBoard[i+1][j+1]){
        if(gameBoard[i+1][j+1]==gameBoard[i+1][j+2]){
          gameBoard[i+1][j+2]=' ';
          score++;
        }
        else if(gameBoard[i+1][j+1]==gameBoard[i+2][j+1]){
          gameBoard[i+2][j+1]=' ';
          score++;
        }
        gameBoard[i+1][j+1]=' ';
        score++;
      }
      else if(gameBoard[i+1][j]==gameBoard[i+1][j-1]){
        if(gameBoard[i+1][j-1]==gameBoard[i+1][j-2]){
          gameBoard[i+1][j-2]=' ';
          score++;
        }
        else if(gameBoard[i+1][j-1]==gameBoard[i+2][j-1]){
            gameBoard[i+2][j-1]=' ';
            score++;
        }
        gameBoard[i+1][j-1]=' ';
        score++;
    }
    else if(gameBoard[i+1][j]==gameBoard[i-1][j+1]){
        if(gameBoard[i+1][j]==gameBoard[i-1][j+2]){
            gameBoard[i-1][j+2]=' ';
            score++;
        }
        gameBoard[i-1][j+1]=' ';
        score++;
    }
    gameBoard[i+1][j]=' ';
    score++;
}

else if(gameBoard[i][j]==gameBoard[i][j+1]){
    gameBoard[i][j]=' ';
    score++;
    if(gameBoard[i][j+1]==gameBoard[i][j+2]){
        if(gameBoard[i][j+2]==gameBoard[i][j+3]){
            gameBoard[i][j+3]=' ';
            score++;
        }
        gameBoard[i][j+2]=' ';
        score++;
    }
```

```
         else if(gameBoard[i][j+1]==gameBoard[i+1][j+1]){
            if(gameBoard[i+1][j+1]==gameBoard[i+2][j+1]){
               gameBoard[i+2][j+1]=' ';
               score++;
            }
            else if(gameBoard[i+1][j+1]==gameBoard[i+2][j+1]){
               gameBoard[i+2][j+1]=' ';
               score++;
            }
            else if(gameBoard[i+1][j+1]==gameBoard[i+1][j+2]){
              gameBoard[i+1][j+2]=' ';
              score++;
            }
            gameBoard[i+1][j+1]=' ';
            score++;
         }
         gameBoard[i][j+1]=' ';
         score++;
      }
   }
}
delay(1);
printf("\n");
printMatrix(gameBoard,row,column);printf("\n");
//slide down elements of matrix in order to add new random values
for(j=0;j<column;j++){
   for(i=0;i<row;i++){
      count=i;
      if(gameBoard[i+1][j]==' '){
         count++;
         while(count>0){
            if(gameBoard[count-1][j]!=' '){
               gameBoard[count][j]=gameBoard[count-1][j];
               gameBoard[count-1][j]=' ';
            }
            count--;
         };
      };
   };
};
delay(1);
printf("\n");
printMatrix(gameBoard,row,column);printf("\n");
//continue or exit
printf("\nIf you want to continue the game please enter 1 otherwise enter 0: ");
scanf("%d",&end);
//control the matrix in order to find same numbers and call function
for(i=0;i<row;i++){
   for(j=0;j<column;j++){
      if((gameBoard[i][j]==gameBoard[i+1][j]) || (gameBoard[i][j]==gameBoard[i][j+1]) ){
         if((gameBoard[i][j]!=' ') && (gameBoard[i+1][j]!=' ')&& (gameBoard[i][j+1]!=' ')){
             return recursion(gameBoard,row,column,score);
         }
         else{
            if (end==1){
               return newRandomElements(gameBoard,row,column,score);
            }
            else if(end==0){
               return printf("\nYour score is: %d\n",score*score);
            }


         }
```

```
            }
        }
    }
    score--;
    selectLocations(gameBoard,row,column,score);
}
void newRandomElements (char gameBoard[100][100],int row,int column,int score){
    int i,j;
    for(i=0;i<row;i++){
        for(j=0;j<column;j++){
            if(gameBoard[i][j]==' '){
                gameBoard[i][j]=(char)('0'+rand()%('9'-'0'+1));
            }
        }
    }
    return recursion(gameBoard,row,column,score);
}
void  selectLocations(char gameBoard[100][100],int row,int column,int score){
    int i,j,flag,rowChange,columnChange,secondRow,secondColumn;
    printf("\n");
    printf("Please select the first row and column numbers consequently:\n");
    scanf("%d %d",&rowChange,&columnChange);
    printf("Please select the second row and column numbers consequently:\n");
    scanf("%d %d",&secondRow,&secondColumn);
    //change elements of matrix
    flag=gameBoard[rowChange-1][columnChange-1];
    gameBoard[rowChange-1][columnChange-1]=gameBoard[secondRow-1][secondColumn-1];
    gameBoard[secondRow-1][secondColumn-1]=flag;
    //call recursion function
    return recursion(gameBoard,row,column,score);
```

```
}
int main()
{
    int i,j,k,row,column,flag,score=0;
    char gameBoard[100][100];
    printf("Welcome, Please enter the row and column number: \n");
    scanf("%d\n%d",&row,&column);printf("\n");
    //assign random values to the matrix
    for(i=0;i<row;i++){
        for(j=0;j<column;j++){
            gameBoard[i][j]=(char)('0'+rand()%('9'-'0'+1));;
            printf("| %c |" ,gameBoard[i][j]);
        };
        printf("\n");
    };
    recursion(gameBoard,row,column,score);
}
```