

**HACETTEPE UNIVERSITY**  
**DEPARTMENT OF COMPUTER ENGINEERING BBM203**



**Prepared by OKAN ALAN**

**Number : 21526638**

**E-Mail : [okanalan97.14@gmail.com](mailto:okanalan97.14@gmail.com)**

**Subject : Assignment 1**

## Aim

Learning data structures especially stack and queue, recursive function, file operations. Using pointers. Escape the maze.

## Problem

Problem is solve maze. There is rat and we give exit path to it. Path mustn't be shortest path. It is only truth path. Our output is the path. There are few doors and keys in the maze. If you don't have a key, you won't open the door. We can move North, South, West and East direction.

## Algorithm

I used stack and recursive function to figure out the assignment. Firstly we don't know  $n \times n$  matrix, I read first line from input file. Then I created  $n \times n$  matrix. I filled around the maze with '1' because '1' is wall. Everytime I don't want to check current place which is in maze. After I read the inputs I send maze and starting point to my solution function.

My function doing something. Primarily check the place is it can move. After I push that place my stack and I go its down, right, up and left. If that place is dead end, I do '1' it. Then adding wallStack because sometimes dead ends are made by doors. I delete from stack because it is not wrong place for now. When I found the key I originalize the maze. So I delete 'X' and '1' but my created '1'. When I was 'E', I print my found path. My algorithm does these steps again and again until arrive exit point.

What did I learn? I learned stack, BFS and DFS implementation in c. I learned backtracking algorithm.

Firstly I solved maze without door and key. After that I added keys. Then I added doors.

## Sample Results

```
gcc -o snc main.c
```

```
./snc input.txt
```

```
Input>>input.txt
```

```
b21526638@
```

```
S001001  
1101010  
0001010  
a011100  
000A000  
1101110  
0b000B
```

```
Output>>path.txt
```

```
b21526638@rdev:~/OdevDeneme/2YIL/1Odev1
```

```
Start  E E S S W S S E S S W b E E E E B E  Exit  
~  
~
```