HACETTEPE UNIVERSITY

DEPARTMENT OF COMPUTER ENGINEERING

NAME-SURNAME: DOĞUKAN BERAT KARATAŞ

NUMBER: 21527142

LECTURE: SOFTWARE LABORATORY I (BBM 203)

1) **Problem:** Labyrinth Find Path Problem

2) **Aim:**

The labyrinth we are given consists of a square matrix that can be changed. In this labyrinth, "0" represents the paths that can be traveled and "1" represents the walls. He has been given a labyrinth that starts at the "S" (start) point, ends at the "E" (exit) point, and is located on the door and keys. Well, how do we solve this labyrinth?

3) **Answer:** Actually the solution is simple;

- ✓ Determine the starting point (because the starting point may vary).
- ✓ In the algorithm written in memory depending on this point, look to the right, left, up and down point of this point, and if there is "0" on that side, go,
- ✓ If the door comes out against you, look for the key,
- ✓ If you do not have a key, try to find the key by going back from where you came from,
- ✓ Open the door by continuing from the point where you find the key,
- ✓ Proceed to find the "E" (exit) point,
- ✓ Finally, print the path from point "S" to point "E" to a file.

4) **So how did I solve it or how I could not solve it:**

- ✧ Normally I needed to read data from the file and create a matrix with the "malloc" function. I read the file twice because I can not do that. On the first read, it keeps the total number of characters in the file. Then he takes the root of this number and rounds it to the nearest integer. Thus, the size of the matrix has emerged. In the second reading, the data contained in the file inserts the matrix.
- ✧ Then, with the "find_path" function, you can look at the four sides of the location, east, west, north and south, and if there is a road on that side, it So it progresses to "Exit". But I could not do it here.