ASSIGNMENT 2 REPORT - MUHAMMET FATİH ALBAYIN

CODE FOR THE ASSIGNMENT

```c
#include <stdio.h>
#include <stdlib.h>
#include <time.h>

int main(){

    int width, height;                      /* Variable for width and the height of the board */
    int doorX, doorY;                       /* Variable for the locations of the door */
    int playerX, playerY;                   /* Variable for the locations of the player */
    char move;                              /* Variable for keeping the move input */
    srand(time(0));                         /* For generating random numbers that are different from each other */

    FILE *fptr;

    fptr = fopen("game_state.txt", "r+");   /* Opens the file in r+ mode to be able to edit it */

    if(fptr == NULL){
    fptr = fopen("game_state.txt", "w");    /* Opens the file in writable mode to generate it if doesn't exist */

    printf("Generating a new board...\n");

    printf("Enter the width and the height of the board:\n");   /* Gets the size of the board from the user */
    scanf("%d %d", &width, &height);

    playerX = (rand() % (width - 0 + 1));                       /* Generates the coordinates of the player randomly */

    playerY = (rand() % (height - 0 + 1));

    doorX = (rand() % (width - 0 + 1));                         /* Generates the coordinates of the door randomly */

    doorY = (rand() % (height - 0 + 1));

    printf("Width: %d, Height: %d, Player: (%d,%d), Door: (%d,%d)\n", width, height, playerX, playerY, doorX, doorY);   /* Notifies the user about the parameters */

    fprintf(fptr, "%d %d %d %d %d %d", width, height, playerX, playerY, doorX, doorY);        /* Prints the parameters to the file */

    if(playerX == doorX && playerY == doorY){                  /* Checks if the door is reached */
    printf("Congratulations! You escaped!\n");
    printf("Generating a new board...\n");
    remove("game_state.txt");                                  /* Deletes the file if the game is over */
    return 0;
    }

    getchar();                                                 /* For taking the backspace left from the scanf */
    }
    else                                                       /* In case the file exists */
    {
    if(playerX == doorX && playerY == doorY){                  /* Checks if the door is reached */
    printf("Congratulations! You escaped!\n");
    printf("Generating a new board...\n");
    remove("game_state.txt");                                  /* Deletes the file if the game is over */
    return 0;
    }

    printf("Enter the move(WASD): ");                          /* Gets the move from the user */
    scanf("%c", &move);


    fscanf(fptr, "%d %d %d %d %d %d", &width, &height, &playerX, &playerY, &doorX, &doorY);  /* Reads the parameters from the file */


    if(move == 'W' || move == 'w'){                            /* Checks the move */
    if(playerY < height){
    printf("Player moves up to (%d,%d)\n", playerX, playerY + 1);  /* Updates the coordinates of the player if the move is not out of borders */
    playerY += 1;
    }
    else                                                       /* Warns the user to not get out of borders */
    printf("You can't get out of borders!\n");
    if(playerX != doorX || playerY != doorY){                  /* Gives this message if the door is not reached */
    printf("Game continues...\n");
    }
    }
    else if(move == 'A' || move == 'a'){
    if(playerX > 0){
    printf("Player moves left to (%d,%d)\n", playerX - 1, playerY);
    playerX -= 1;
    }
    else
    printf("You can't get out of borders!\n");
    if(playerX != doorX || playerY != doorY){
    printf("Game continues...\n");
    }
    }
    else if(move == 'S' || move == 's'){
    if(playerY > 0){
    printf("Player moves down to (%d,%d)\n", playerX, playerY - 1);
    playerY -= 1;
    }
    else
    printf("You can't get out of borders!\n");
    if(playerX != doorX || playerY != doorY){
    printf("Game continues...\n");
    }
```

```
 93  }
 94  else if(move == 'D' || move == 'd'){
 95  if(playerX < width){
 96  printf("Player moves right to (%d,%d)\n", playerX + 1, playerY);
 97  playerX += 1;
 98  }
 99  else
100  printf("You can't get out of borders!\n");
101  if(playerX != doorX || playerY != doorY){
102  printf("Game continues...\n");
103  }
104  }
105  else
106  printf("Please enter a valid input!\n");              /* Warns the user if the input is not valid */
107
108  rewind(fptr);                                         /* Gets the file pointer to the beginning */
109  fprintf(fptr, "%d %d %d %d %d %d", width, height, playerX, playerY, doorX, doorY);    /* Updates the coordinates */
110
111  if(playerX == doorX && playerY == doorY){             /* Checks if the door is reached */
112  printf("Congratulations! You escaped!\n");
113  printf("Generating a new board...\n");
114  remove("game_state.txt");
115  return 0;
116  }
117  }
118  fclose(fptr);                                         /* Closes the file */
119  return 0;
120  }
```

THE GENERATED OUTPUT

```
albay@albay-VirtualBox:~/Desktop$ gcc -ansi 240104004064.c  -o  m
albay@albay-VirtualBox:~/Desktop$ ./m
Generating a new board...
Enter the width and the height of the board:
5 5
Width: 5, Height: 5, Player: (1,3), Door: (3,4)
albay@albay-VirtualBox:~/Desktop$ ./m
Enter the move(WASD): d
Player moves right to (2,3)
Game continues...
albay@albay-VirtualBox:~/Desktop$ ./m
Enter the move(WASD): d
Player moves right to (3,3)
Game continues...
albay@albay-VirtualBox:~/Desktop$ ./m
Enter the move(WASD): w
Player moves up to (3,4)
Congratulations! You escaped!
Generating a new board...
albay@albay-VirtualBox:~/Desktop$
```