CSE102 – Computer Programming

Homework #3

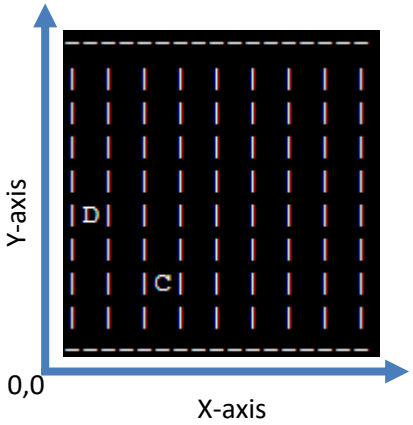
Functions and Selection

Due Date: 05/04/2023

Hand in: A student with number 20180000001 should hand in a zip file named 20180000001.zip for this homework.

Homework Description: You have been asked to provide a single C file that contains a main function along with several other functions. This homework will give you hands-on experience with variables, input/output operations, selections, loops, and functions.

The main task of this homework is to develop a 2D puzzle game. The game consists of a dark room of size NxN, as shown below. The objective of the game is to free the character by leading her/him to the door. The game is over when the chracter reaches the door. The locations of the character and the door will randomly appear at each run of the game. The details of the game are explained explicitly in the following part descriptions, so please read them carefully. Each part must be written as a function.



Part 1. [30 pts] Drawing the room

The room's size is variable and inputted by the user. However, the user cannot specify a size smaller than 5 or greater than 10. Each (x, y) coordinate in the room represents a location. For example, there are 25 different locations when the size of the room is 5. Please see the given figure for a better understanding on the coordinates and locations. The character's current position is indicated by a capital C, drawn at that location. The current position of the door is indicated by a capital D at its location.

Part 2. [30 pts] Gameplay

The character is able to move one space in any of the four cardinal directions: up, down, left, and right. Diagonal moves are not allowed. The user will input their desired move using the following keys: 'a' for left, 'd' for right, 'w' for up, and 's' for down. These moves should be defined at the beginning of the code with direction names as the keys. Any control or selection with [a, d, w, s] characters will result in a missgrading. The game will prompt the user for a new move after each move is made until the game is over.

The current state of the room should be printed after each move, except when the character reaches the door. If the character attempts to move through a wall, a warning message should be displayed. The game ends when the character reaches the door, and a message should be displayed to notify the user that the game is over and how many moves were made during the game.

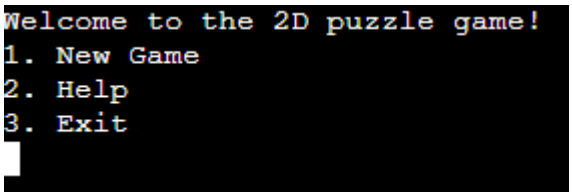
Part 3. [10 pts] Control of Status

After each move, the game's status should be checked to determine whether it is over or not. A control function should be used to perform this check. If the character's location and the door's location are the same, the function should return 1, indicating that the game is over. Otherwise, it should return 0.

Part 4. [30 pts] The Game

The game will start automatically when the program is executed, and the menu should be displayed to the user when the game is active. The user can select the "New Game" option (1) to start a new game. It should be noted that the locations of the character and door will be randomly selected each time a new game is started. The size of the room must be specified at first to start the new game.

After the game is over, the user will return to the menu. If the user selects the "Help" option (2), the game's instructions and rules should be displayed (you can use the description in part 2). The menu will then prompt the user again after help is printed. If the user selects the "Exit" option (3), the game and program should terminate with a goodbye message.



Notes:

***** The homework must be done without using arrays, otherwise you will get 0 from this homework.**

***** To avoid receiving a lower grade, it is important to use if-else and switch-case appropriately according to their respective needs, rather than using them interchangeably.**

****Attach the screenshots of the gameplay.**

****Do not forget to prepare a makefile (-50 points)**

General Rules:

1. We do not give you any function prototypes. We expect that you are experienced enough to understand when to use methods and name them. These will also be graded.
2. Make sure to include appropriate comments and variable names in your code to make it easy to understand.
3. The program must be developed on given version of OS and must be compiled with GCC compiler, any problem which rises due to using another OS or compiler won't be tolerated.
4. Note that if any part of your program is not working as expected, then you can get zero from the related part, even it is working partially.
5. Zip your homework files before uploading them to MS Teams. The zip file must contain the C file with your solution and screenshots of the valid outputs of the program.
6. You can ask any question about the homework by sending an email to b.koca@gtu.edu.tr or by using the homework channel on MS Teams page of the course.