Submission Date: 23th December 2021 till 11:59 PM

FINAL PROJECT

PROGRAMMING FUNDAMENTALS (CS1002) – FALL2021

INSTRUCTIONS

1. **Plagiarism in course project will result in F grade in the course**
2. This is not a group project and each person will be working on the project individually.
3. Make sure you submit your project before the submission time. Late submissions won't be accepted even if they are late by just one minute.
4. You can earn bonus marks by implementing extra features in the project.
5. Use good programming practices (well commented and indented code; meaningful variable names, readable code etc.).
6. Each file that you submit must contain your name, student-id, and assignment # on top of the file in comments.
7. Combine all your work in one folder and compress it into a zip file. The folder must contain .cpp files (no binaries, no exe files etc.).
8. Submit the solutions via google classroom. Submissions via email will not be accepted.
9. Use proper naming convention to name the file containing source code. E.g**.** ***i21xxxx_project.cpp*** , replace i21xxxx with your roll number.
10. **Follow the given instructions to the letter, failing to do to so will result in a zero.**

Rush Hour GAME

Rush Hour is a game for one player, in which the player takes up the role of a taxi driver who must collect money by dropping passengers to their destinations.

In this project, you have to make the Rush Hour game of 1 player with the following features:

Game Start Menu

a) When the game starts, a user should be given the choice in the menu to view the leaderboard (displaying top 10 scores along with player names). The high-scores and player names should be read from the file (named highscores.txt) on the hard-disk and stored in the arrays of size 10. If the score of the current game makes to the list of high-scores, the lowest score is removed from the array and the new contents of the array overwrite the file.

If the game is being played for the first time the leaderboard will be empty.

b) The game can be played using one of the following, a yellow taxi and a red taxi. An option will be shown to the user on screen to either assigning the taxi color randomly or users can select the color themselves.

c) Once the player has selected a taxi he/she will enter their name.

The Board

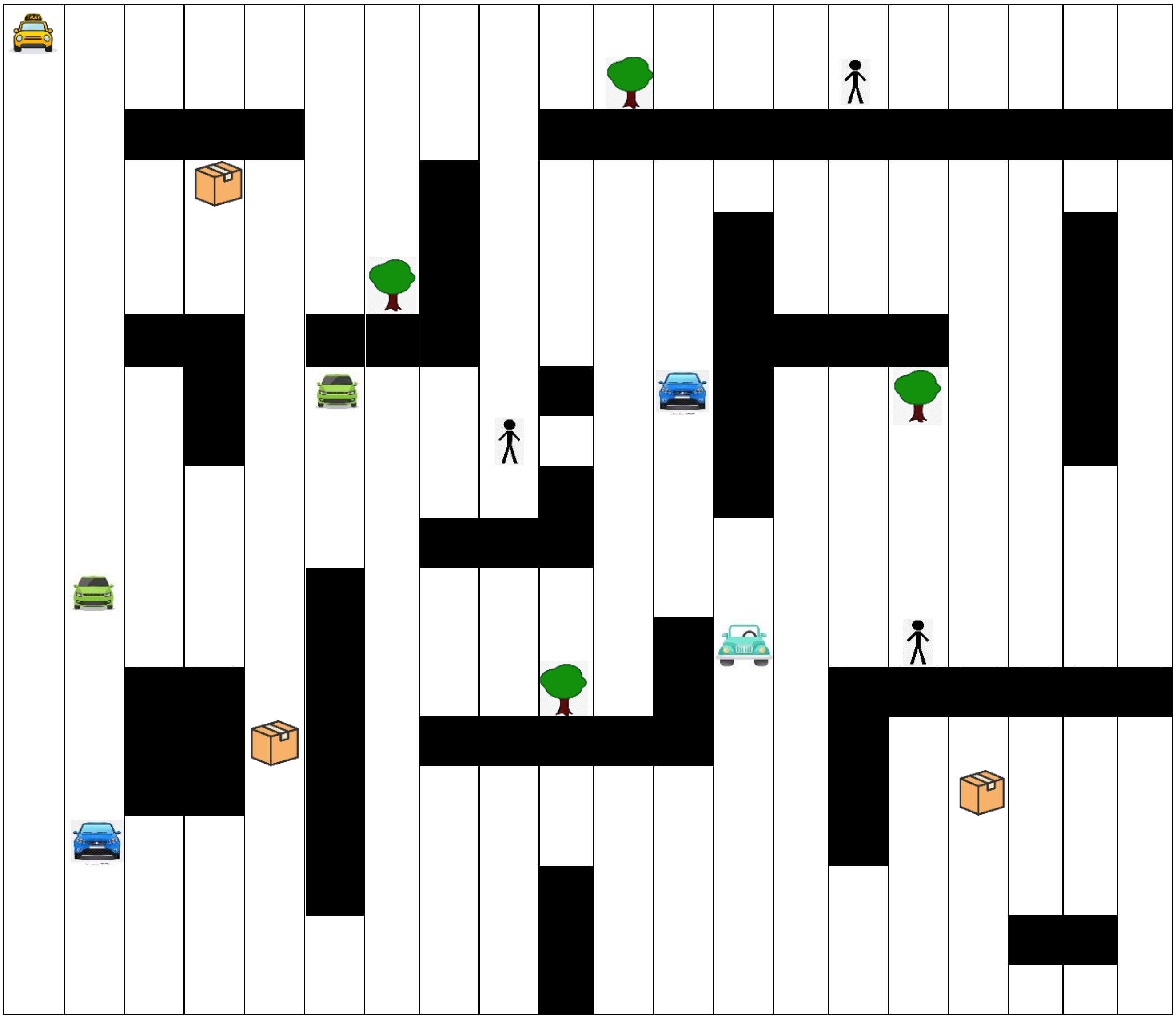
d) Draw a 20x20 Board on the screen. A sample board is attached below but you are free to be more creative and make as beautiful board as possible. Projects will be evaluated relatively, so you can earn more marks by designing more attractive board.

e) The board has two areas, roads(shown in white) and buildings (shown in black), cars can be driven anywhere on the roads.

f) The board consists of the player's taxi and other cars, obstacles and passengers.

g) At the start of the game, the player's taxi will be at the top left corner of the board as shown in the picture above. Other cars, obstacles and passengers will be randomly spread out on the board.

h) The board shall always have at least 3 and at most 5 passengers waiting to be picked up. The passengers can be present at any location on the board except the position where other cars or obstacles are present.



Game play

i) The player's taxi can move in all four directions (up, down, right and left using the four arrow keys → ←⬇⬆

j) The taxi has to avoid obstacles and other cars and get to the people waiting for a taxi. Once the taxi is next to a person, the player will press the ***Spacebar*** to pick the passenger.

Passenger's Desired Destination

k) Each passenger would have a desired destination on the board, that would be highlighted in **GREEN** once the taxi picks up the passenger. The taxi has to get the passenger to the desired location as fast as possible.

l) The desired destinations of the passenger will be a randomly generated position on the board. The destination cannot be a place occupied by an obstacle.

m) The desired destination of the passenger cannot be the same as the pickup location. To drop a passenger the player will press the ***Spacebar*** after getting to the location.

n) Once the taxi has picked up a passenger, they cannot drop them anywhere except for their desired destination.

o) Every time the taxi picks up a passenger and drops them to their desired location the player gets **10 points**.

p) All passengers are in a rush and pay extra fare for overtaking another car, if the taxi overtakes another car on the board the player gets **1 extra point** for each overtake.

Negative Points

q) However, there are some negative points as well,
If the taxi hits a person **-5 points**

If the taxi hits other obstacles the points deducted depend on the type of taxi:
Red Taxi: Hits an obstacle (tree or box) **-2 points**
Red Taxi: Hits another car **-3 points**
Yellow Taxi: Hits an obstacle (tree or box) **-4 points**
Yellow Taxi: Hits another car **-2 points**

Other features

r) Cars other than the taxi cannot be red or yellow.

s) All other cars on the board are also moving, they can move up, down, right and left. The other cars cannot stop anywhere during the game.

t) Each time the taxi drops two passenger successfully at their desired destination the number of cars on the board will increase by 1.

u) Each time the taxi drops two passenger successfully at their desired destination the speed of the cars on the board will increase.

v) The two taxis can move at different speeds, the yellow taxi is faster than the red taxi.

Winning Criteria

w) The game would have a total running time of 3 minutes.

x) If the player is able to score 100 points or more in 3 minutes, the player wins the game.

Graphics

You cab create a stick figure to represent passengers and a rectangle with two circles to represent a car.

Bonus Features:

1. The taxi can drive in AUTO mode

a) Find the shortest path on the board from the pickup location to the passenger's destination and maneuver the taxi automatically to that location while avoiding obstacles and collisions with other cars on the road.

2. A new board is generated every time a new game is started, however every corner in the board must be reachable. 35% of the board should be buildings and the remaining 65% should be roads.

You may think of other interesting (and programmatically challenging) features to implement to get bonus marks. The decision to give the bonus marks for that feature will be with the teacher so you should discuss any idea with your respective teachers to know whether it qualifies as bonus or not.

Starter Code:

We have provided you the starter code in C++ to draw circle, square and triangles of different colors and sizes. You are allowed to change it and add functionality according to the project statement.

To execute the starter code, you need to do the following:

a) Extract the attached zip file.

b) Open the terminal and navigate to the path of extracted directory

c) Install the required libraries by executing the command below:

```
bash install-libraries.sh
```

d) Compile the project by writing the command

```
make
```

e) Run the main file

```
./game
```

Important Note:

You must use the concepts of functions, pointers, dynamic memory allocation and file handling that have been taught to you in the course.

Happy CODING ☺