Marmara University – Faculty of Engineering – Department of Computer Engineering

Spring 2020 – CSE1142 Computer Programming II Project

(Due: 10.05.2020 – 23:00)

In the term project, you will implement the following game by using JavaFX framework. This game aims moving a ball from a start point to an end point if there is an appropriate path. The user is supposed to form a path using pipes given as sliding tiles. Initial state of the game can be seen in the figure below:

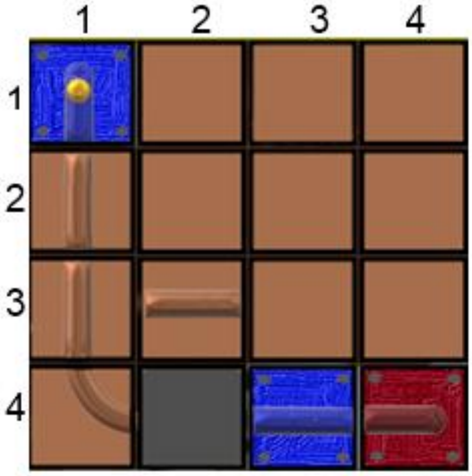


Figure 1: Game Board

Each tile can be identified by using [rowIndex, columnIndex]. A sample game board is shown in Figure 1. The user will use the mouse to drag the tile located at [3, 2] to [4, 2] to form the path. When the complete path is formed, the ball will roll from tile [1, 1] to [4, 4] as in Figure 2.

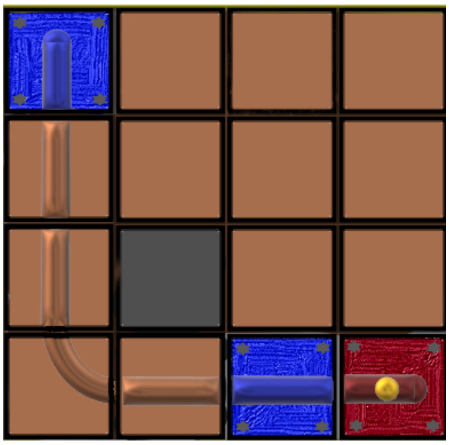


Figure 2: Game Board After Rolling the Ball

For the project, you will implement this game with following properties:

- There will be several levels; the user will not be able to play next level without completing a previous one.
- Level aspects will be constructed according to the input files those are attached to this document.
- The ball will start from the blue tile and the path will end at the red tile. (Positions of start and end tiles may be different for each level.)

- The blue and red tiles cannot move.
- The user will use mouse to move the tiles.
- Grey tiles indicate movement free spaces.
- Tiles can move only if the desired position is movement free.
- The tiles can move 1 tile distance, vertically and horizontally.
- Tiles cannot move diagonally.
- Number of moves must be recorded and printed to the screen.
- While you are constructing the game board, you may use images for each tile.

The details of game board creation, level properties, and submission instructions are described in detail in subsequent sections of this document.

GAME BOARD CREATION

Level appearances will be prepared using input files. The appearance of first level (Level 1) is illustrated in Figure 3. There are totally 16 tiles on the game board, each with different characteristics. The tiles in Figure 3 are named from 1 to 16 starting from the upper left one to the last one in bottom right. Game board must be created according to the data in input file.

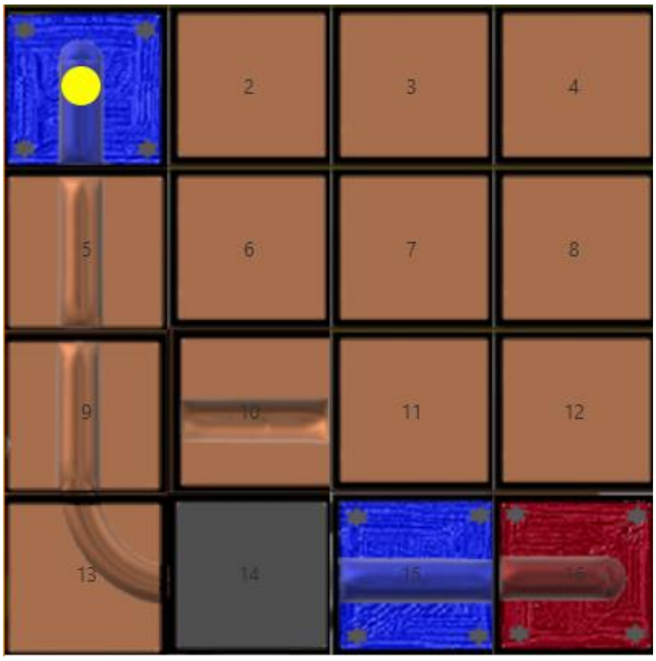


Figure 3: Game Board Level1

Level 1 Input File:

```
id,type,property
1,Starter,Vertical
2,Empty,none
3,Empty,none
4,Empty,none
5,Pipe,Vertical
6,Empty,none
7,Empty,none
8,Empty,none
9,Pipe,Vertical
10,Pipe,Horizontal
11,Empty,none
12,Empty,none
13,Pipe,01
14,Empty,free
15,PipeStatic,Horizontal
16,End,Horizontal
```

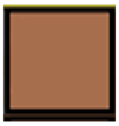
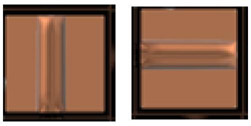
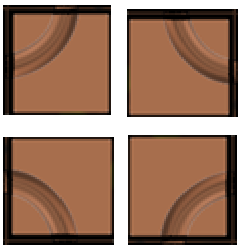
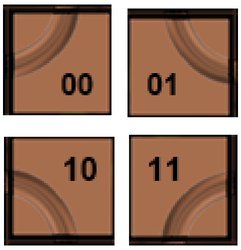
Input File

Input file contains the **id**, **type** and **property** attributes about 16 tiles on the game board; each separated by commas. First line of input file corresponds to the captions for each data separated by commas. **Id** attribute varies between 1 and 16. Each tile with that specific id attribute has a **type** and **property**.

Type attribute may be equal to **Starter**, **End**, **Empty**, **Pipe** or **PipeStatic**. Please check the next section for type details.

Property attribute may be equal to **Vertical**, **Horizontal**, **none**, **Free** or **(00, 01, 10, 11)**. Please check the next section for property details.

Tile Types and Properties

 Figure 4: Starter Tile	**Starter** There is only one starter tile in a game board. Starter tiles cannot move; their locations are static. Starter tiles may have horizontal or vertical pipes inside.	
 Figure 5: End Tile	**End** There is only one end tile in a game board. End tiles cannot move; their locations are static. End tiles may have horizontal or vertical pipes inside.	
 Figure 6: Empty Tiles (Free)	**Empty – Free** Empty tiles are tiles without pipe. There are two types of empty tiles. One presented in Figure 6 stands for an empty space. Other movable tiles can move to the location of this empty tile. Their property is specified as "free" in input file.	
 **Figure 7: Empty Tiles**	**Empty** The empty tile presented in Figure 7 stands for an empty space. Tiles cannot move to the location of this empty tile. To give an example; tile 10 in Figure 3 cannot move to tile 11. However, tile 10 can move to the place of tile number 14. Their property is specified as "**none**" in input file. Empty tiles may move to empty free tiles.	
 Figure 8: PipeStatic Tiles	**PipeStatic** Static pipe property is specified as "**PipeStatic**" in input file. Static pipes may be horizontal or vertical. **PipeStatic** tiles cannot move; their locations are static.	
 **Figure 9: Pipe Tiles**	**Pipe** There may be many pipes in a game board. Pipes may be horizontal or vertical. Pipes may move to empty free tiles.	
 **Figure 10: Curved Pipe Tiles**	**Curved Pipe** Curved pipes may move to empty free tiles. The type of a curved pipe is represented as "**Pipe**" in input file. Curved pipe property may take values of "**00, 01, 10, 11**", each representing the curve type. There may be four different curved pipe alignments. The exact representation of each alignment is illustrated in Figure 10. If the property value of a tile is equal to **00**, then it corresponds to the top left tile in Figure 10. As an example, pipe type is **01** is used for the tile number 13 in the sample game board given in Figure 3. It should be noted that, the numbers (**00, 01, 10, 11**) are just used for curve type representations, they do not represent tile IDs.	 **Figure 11: Curved Pipe Tiles with properties**

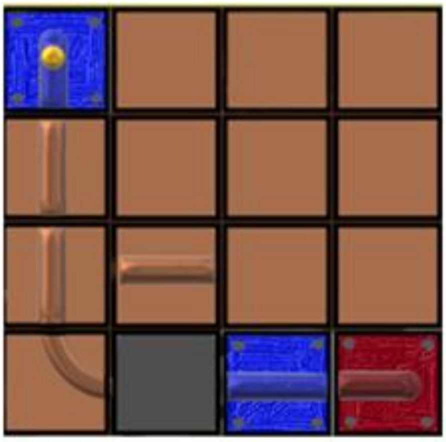
It should be noted that the directions of tiles do not change once the game board is created. You can only drag the tiles to place them correct positions.

LEVELS

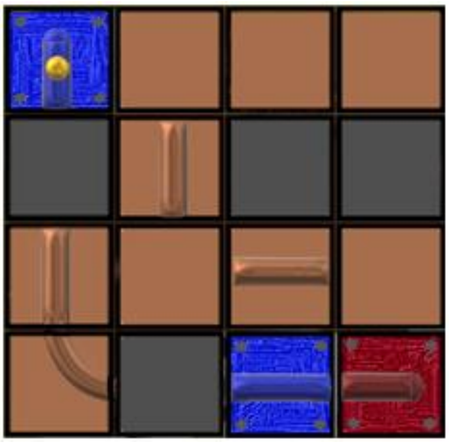
You should use the input files those are generated for each level to construct the game board. It should be noted that, each level can be given with the same type of tiles locating at different positions.

Level appearances may be similar as follows:

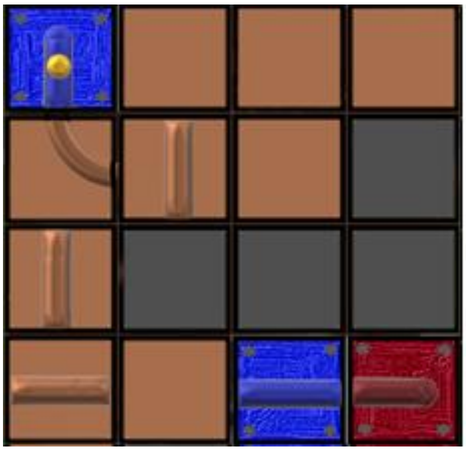
Level 1



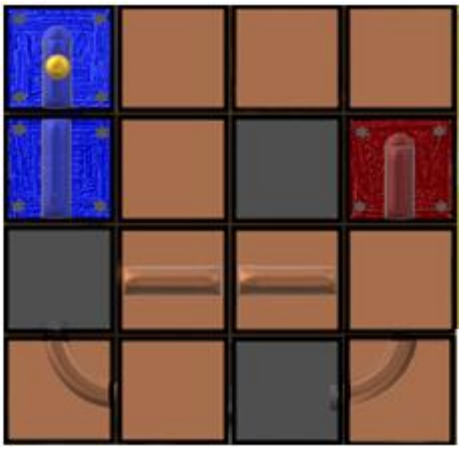
Level 2



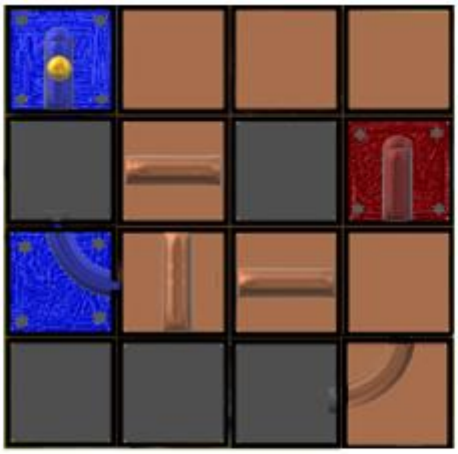
Level 3



Level 4



Level 5



SUBMISSION INSTRUCTIONS

1) The due date for the project is 17/05/2020. You are supposed to work in groups of 2 people.

2) Please zip and submit your files using filename Student1Number_Student2Number_Project.zip (ex: 150118123_150717015_Project.zip) to Canvas system (under Assignments tab). Your zip file should contain the following files:

   a) The commented source code of your project.

   b) A 5-10 pages long report that contains UML diagram of your project, implementation details, and screenshots.

DEMO SESSIONS

You will have demo sessions in the week of 18-24/05/2020. The exact time and date will be announced later. You should demonstrate what you have done in 20 minutes. You should also answer some questions about your implementation. Each group member will be assessed separately.

DETAILS ABOUT PROJECT REPORT

1) Your project reports are to be typed with normal sizes (Ex: Times New Roman 12pt.).

2) Your report must have a cover page with the following information:

   a) Title

   b) Project Name

   c) Authors (IDs, Names, and Surnames)

   d) CSE1142 Computer Programming II, Spring 2020

   e) Date Submitted: May 17, 2020

3) Firstly, you should provide a section named as “Problem Definition” and briefly describe the problem or the game in your project in 1-2 paragraphs with your own words.

4) Then, you should add a section named as “Implementation Details” and provide the UML diagrams of your project. In this section, you should also describe how you design and implement the project in more detail.

   a) Additionally, you may provide information about

      i) which parts are complete/incomplete in your project?

      ii) what are the difficulties you have encountered during the implementation?

      iii) what are the additional functionalities of your project added by your team?

5) Then, you should add a section named as “Test Cases” and this section should contain the results of your testing phase. You should provide the screenshots of your project execution for the given test cases and add explanations about them. Filling this part only with screenshots is not a feasible solution.

6) These are the minimum requirements for your project report. You can add more.

7) Be careful to have correct spelling and proper English grammar.

8) The most important part about writing a project report is using your own words without copying-pasting from the Internet or the project document. Please show your own work.

NOTES

1) Write a comment at the beginning of each program to explain the purpose of the program.

2) Write your name and student ID as a comment.

3) Include detailed comments to explain your actions. Since this is the term project, lack of comments will have an effect on your final grade.

4) Select meaningful names for your variables and class names.

5) In case of any form of **copying and cheating** on solutions, you will get **FF** grade from the course! You should submit your own work. In case of any forms of cheating or copying, both giver and receiver are equally culpable and suffer equal penalties. **All types of plagiarism will result in FF grade from the course.**

6) No late submission will be accepted.