Project Assignment 1
50 Points
References: Referenced Textbook and Week 2, 3, & 4 handouts

Skills Required:

1. Writing code that controls a user interface
2. Primitive data types
3. The use flow controls and control loops
4. Iterations: For, While, Do .. While statements
5. Controls: IF … Else and /or Switch statements
6. Java Enumerator
7. Simple message dialogs
8. Calling predefined object methods
9. Displaying messages in a text are box
10. Formatting output

Description:

You will be given a number of files to setup a maze program in Eclipse IDE. The goal of this program is to add some code to move an image around the maze according to a set of rules. If you follow the rules precisely, your code will find the Java logo, if not; your code is not implemented correctly with the respect to the specified rules.

You must follow the exact rules to receive full credit on this program, as the code will be tested against different mazes

Task Specifications:

1. Complete the program setup in the Part I document
2. Once, the program is correctly instantiated, then proceed to the next step
3. Complete the program instructions in Part II document to solve the mazes by finding the Java logo

You should only add/modify code in the specified class and method as shown by Part II document

Evaluation Criteria

1. All tasks must be completed correctly in order to receive full credit for this project
2. The maze program should solve the given mazes (1 & 2) using the provided algorithm
3. The program will be tested against other solvable mazes by the provided algorithm

Submission:

Copy the .java source files from the *src* folder in your *work space* to another folder that should be named following the provided naming format in this course, then zip and upload the file under this assignment answer in Canvas. You should read the project demo document in Canvas before the review time.

File Name: *FLLLLPP1.zip (F = first letter in your first name and LLLL = your last name)*

Grading Rubric - Project 1 (PP01)

Student Names: _____

Evaluation Criteria	Comments	Max	Points
Program compilation and running (Program should run on your machine during the video call/demonstration): Proper use of the directory structure and the class templates provided (3) Successful compilation and running, meaning no unexpected runtime error of the program during the demonstration (3) Use of good variable names, proper use of data types, use of flow controls and control loops, iterations, conditional logic, enumerations, dialog boxes, message display (4)		**10**	
Part I: You will not be tested on part I during the demonstration, but the part I setup needs to be done in order to part II. We will expect that the student has tried part I, understood part I, in order to be able to proceed to part II. As the instruction suggests, you will comment the code that you wrote in part I. During the video demonstration, you will show the commented out code and orally explain how they worked. We will honor your oral presentation and grade you on part I based on that. We will not expect the		**10**	

students to demonstrate (run) the part I during the demonstration. Walk-through the commented code for part I, proper implementation of the method signature, the return type and the method body (4) Orally explain (no running/demonstration is necessary for part I) how you have tested your program so that the student image could be moved up, down, left, right (2) Orally explain (no running/demonstration is necessary for part I) how you have tested your program to display the success or failure (2) Orally explain (no running/demonstration is necessary for part I) how you have tested to display the current student image row (2)			
Part II (You will run Part II on your machine during the video call/demonstration): Proper implementation of the expected methods, their signatures and the return type (3) Ability to move the student icon from top to bottom (3) Ability to move the icon on the even rows from left to right, and on the odd rows from right to left (3) Ability to move in the reverse direction when boundaries are reached (3) Ability to move when an obstacle is encountered (3)		**30**	

Not allowing diagonal move, or outside the maze (3) Ability to find the Java logo (9) Displaying the proper message when Java logo is found (3)			
Total		**50**	

Total= ____/50