OLA#4A EVALUATION FORM

Name: _____ **C# :** _____

Handin (Penalty)	
Didn't turn in the soft copy of the program	**-50**
Didn't turn in the hard copy of the program	**-5**
Documentation	
Each method of MazeClass and CreatureClass is documented with function description, pre-condition, and post-condition in the header file.	__/6
Comments at the beginning of each source file	____/2
Comments for each function definition and function prototype	____/2
Comments for each loop statement	____/2
Comments for each branch of conditional statements	____/2
Comments for all the constants and local variables	____/2
Programming Style	
Meaningful names for constants and variables.	____/2
Use indentation and white space to make program easier to read.	____/2
Assignment Specific Requirements	
#ifndef/#define/#endif is used properly in CreatureClass.h and MazeClass.h	/2
A struct type "Coordinate" is used to represent the coordinates of the creature	/2
Command line arguments is used correctly	/3
A enum type is defined and used properly to represent the squares in the maze	/3
Size of the arrays defined as constants	/2
The member functions of "MazeClass" are implemented **correctly** according to the description given in the assignment (2 pts each): ReadMaze, Display, GetEntrance, GetExit, MarkVisited, MarkPath, IsWall, IsClear, IsPath, IsVisited, IsExit, IsInMaze	____/24
The member functions of "CreatureClass" are implemented **correctly** according to the description given in the assignment (2pts each): MoveOneStepLeft, MoveOneStepRight, MoveOneStepUp, MoveOneStepDown, AssignLocation, ReportLocation	__ / 12
const modifier is used correctly for class member functions	____/2
Program output:	
Program run4A1:	
Maze is read and displayed correctly	____/10
The program reports the correct entrance and exit locations	___ / 5
The program produces the correct output for the 3 locations selected	____/ 5
Program run4A2: The program produces the correct location of the creature before and after the moves	____/10
Total	___ / 100