Design Project Name: *Soliton*

Semester Term: *Spring 2014*

Team Members: *Joshua Trahan*

Project Overview:

Soliton is a simple, text-based, turn-based, top-down perspective stealth game inspired by Metal Gear Solid. Levels are loaded directly from text files. The guards move one tile every time the player moves one tile, and have a view extending four tiles diagonally 90 degrees from each other, including the tiles in between. If the player crosses into that range, the game ends and the player loses. If the player gets to the end of the level without being spotted, he wins.

Project Scope: *Solition is a single player only game.*

Project Specifics: *The goal of the game is for the player to get from the beginning of the level to the end of the level without being spotted by any guards. Guards move one tile in their pre-set route when the player moves one tile. If the player ends up up in the view of the guards, they lose and the game ends. Players and guards can NOT move diagonally. If there is a wall between the guard and the player, the guard can NOT see the player.*

Levels will be built 1:1 from text files containing the characters that represent the level (outlined in the User View section). A line containing 'G' will designate the end of the level drawing, and user-defined guard paths will start on the line after that. To move the guard in a direction, the user will enter 'm' followed by the direction for the guard to move (ex. "M^" for move up). To make a guard simply face a direction, the user will enter 'f' followed by the direction for the guard to move. To make a guard stay, the user will simply enter 's'.

User View: *The user sees a top-down view of the entire level, including current position and heading of the guards, represented by ASCII characters. Floor tiles will be represented by the '.' character, wall tiles will be represented by '|', '-', and '+' characters, the player will be represented by an '@' character, and guards will be represented according to their heading: A guard facing left will be '<', right will be '>', up will be '^', and down will be 'v'. The end of the level will be represented by a 'E' tile.*

Project Details: *The board will be drawn in the beginning. Everything will be represented on the screen except the paths guards WILL take.*

Project Flow (flowchart or pseudo-code):

1. *Load level and guard paths from text file.*
2. *Draw level into console, placing player and guards into positions designated by the level file.*
3. *Ask user to input direction he will move.*
4. *Move user in their specified direction.*
5. *Move all guards in the direction specified by the paths loaded from level file.*
6. *If the tile the player is on is the same tile the end of the level is on, continue to 7. Else, go to 3.*
7. *Draw a cool congratulatory screen. Ask the user if they want to load a new level, repeat that level, or end the game.*
8. *If they want to load a new level, go to 1. If they want to repeat the level, go to 2. If they want to end the game, continue to 9.*
9. *Close the program.*