**Student Name/Grade: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Rubric:**

\_\_\_\_\_\_ / 10 Board is at least 5x5, has obstacles and warps

\_\_\_\_\_\_ / 10 Board prints nicely before every move

\_\_\_\_\_\_ / 30 Allows proper movement up, down, left, and right

\_\_\_\_\_\_ / 10 Randomly generates start and goal positions

\_\_\_\_\_\_ / 5 Congratulates user and ends game when goal reached

\_\_\_\_\_\_ / 10 Assignment submitted correctly and on time

\_\_\_\_\_\_\_ Total

**Description:**

Write a maze game that’s at least 5 by 5 spaces that randomly selects a starting position and a goal position and lets the player move through it.

Your maze needs to have at least 2 Pac-Man-like breaks in the walls that allow warping to the other side: at least one along the top and bottom and at least one along the sides. There should also be obstacles in the maze that can’t be passed through. For example, my empty maze board looks like this when it’s printed:

---------- ----

X

| X X X |

| X |

X X X

| X |

---------- ----

So the player is allowed to go from the right edge over to the left (or vice versa) in the top and 4th rows, and can go from the bottom to the top (or vice versa) in the 4th column over.

Once your board works, you need to be able to move around it, and that’s the hardest part. I used the W, A, S, and D keys for Up, Left, Down, and Right, but you can do that however you want. Depending on what the player enters, either move the player that direction if it’s a legal move or else do nothing and let them try again.

Once THAT works, randomly generate a starting point and a goal to reach, and neither one can be on an obstacle. I indicated the player by putting a “P” on the board and the goal is indicated by “G”. Then run the game until the user reaches the goal and congratulate them. I’ll put a run-through of my game on the next page.

Once you’re done, play it a bunch of times and then submit it to me. Name it maze\_lastname and email it with **[ICS] Week 19 LastName**

**Here is one complete run of my game:**

---------- ----

X G

| X X X |

| P X |

X X X

| X |

---------- ----

Move choice: w, a, s, d: w

---------- ----

X G

| X P X X |

| X |

X X X

| X |

---------- ----

Move choice: w, a, s, d: w

---------- ----

P X G

| X X X |

| X |

X X X

| X |

---------- ----

Move choice: w, a, s, d: a

---------- ----

P X G

| X X X |

| X |

X X X

| X |

---------- ----

Move choice: w, a, s, d: a

---------- ----

X G P

| X X X |

| X |

X X X

| X |

---------- ----

Move choice: w, a, s, d: a

---------- ----

X P

| X X X |

| X |

X X X

| X |

---------- ----

You win!

Methods my program uses:

**print\_board(board)** #prints entire board’s current state

**move(r, c, direction)** #checks if a move is legal, updates the board, and returns [row, col]

**start()** #generates random starting point that isn’t an obstacle

**winner()** #generates random goal that isn’t an obstacle OR the starting point