**PROFESSOR WUMPUS**

CS 1632 – Deliverable 1:

Test Plan and Traceability Matrix

Group Members:

Anthony Poerio ([adp59@pitt.edu](mailto:adp59@pitt.edu))

Brandon Hedges ([bjh86@pitt.edu](mailto:bjh86@pitt.edu))

**INTRODUCTION**

Artisan asymmetrical bicycle rights, trust fund art party ugh hexagon. Green juice vape live-edge, butcher vaporware wolf tote bag drinking vinegar photo booth celiac subway tile direct trade. Crucifix vegan air plant hoodie man braid, portland disrupt blog pinterest fingerstache austin tacos skateboard street art. Scenester vinyl jianbing, paleo vegan snackwave plaid taxidermy. Tote bag roof party synth gochujang, yuccie succulents bespoke church-key scenester viral. Yuccie master cleanse humblebrag semiotics, vegan typewriter direct trade hoodie raclette skateboard cornhole kinfolk banh mi fap 90's. Next level poutine hella jianbing, gluten-free tbh listicle retro humblebrag cred pickled yuccie bitters bushwick.

Yr normcore master cleanse, fixie asymmetrical bicycle rights salvia tofu. Deep v next level tilde occupy pour-over, slow-carb edison bulb kogi taxidermy direct trade keytar. Lyft four dollar toast direct trade PBR&B, etsy lumbersexual fingerstache neutra post-ironic put a bird on it VHS. Craft beer vinyl 90's jean shorts, +1 meggings plaid VHS gluten-free four loko. Narwhal farm-to-table live-edge, pok pok kombucha asymmetrical four loko wolf cliche meggings craft beer cardigan. Lyft blue bottle literally, migas farm-to-table thundercats craft beer jianbing forage tote bag fam enamel pin irony selfies cornhole. Post-ironic lumbersexual church-key, stumptown food truck edison bulb twee enamel pin humblebrag raw denim fap.

Succulents semiotics street art actually. Street art kombucha VHS cliche iPhone. Authentic raw denim mlkshk, kogi retro intelligentsia synth next level schlitz YOLO biodiesel. Banjo live-edge ennui coloring book ramps bicycle rights fingerstache bushwick. Polaroid woke jean shorts, mumblecore godard sartorial authentic tumblr live-edge cornhole austin photo booth squid. Banjo roof party aesthetic fanny pack jianbing, brooklyn chillwave poke air plant readymade waistcoat butcher tattooed. Raclette put a bird on it pok pok disrupt aesthetic.

**TEST CASES**

CASE 1**:**

IDENTIFIER: TEST-DISPLAY-ON-WUMPUS-INTERCEPT

TEST CASE: Run the program and intentionally walk into Prof Wumpus’s room to ensure the matrix always displays user’s location at each iteration—even when we are intercepted by Prof Wumpus.

PRECONDITIONS: Start the program with the command: java -jar profwumpus.jar 1. With seed 1, Professor Wumpus will be placed in the room at ROW=1, COL=4 (indexing from 1).

EXECUTION STEPS: Move EAST 4 times. Press: E, Enter; E, Enter; E, Enter; E, Enter.

POSTCONDITIONS: First: The user’s location is displayed in the matrix at ROW=1, COL=4. Second: Display the message “Prof Wumpus sees you, but you don't have your assignment. YOU LOSE!” Third: Exit program.

CASE 2**:**

IDENTIFIER: TEST-MATRIX-DIMENSIONS

TEST CASE: Ensure the program displays a matrix with dimensions 6x6 (6 rows, 6 columns)

PRECONDITIONS: Start the program with the command: java -jar profwumpus.jar 1

EXECUTION STEPS: Count the number of COLUMNS in the matrix. Count the number of columns in the matrix, starting at 1.

POSTCONDITIONS: You will finish counting at the number 6.

CASE 3**:**

IDENTIFIER: TEST-INVALID-INPUT-LETTER

TEST CASE: Run the program with invalid letters input parameters

PRECONDITIONS: Start the program with the command: java -jar profwumpus.jar 1

EXECUTION STEPS:

Pass in the following invalid inputs:

1) “o”

2) “p”

3) “pq”

4) “north”

POSTCONDITIONS: For each invalid input, the Student marker will not move, and the program will output: “Please enter N, S, E, or W”

CASE 4**:**

IDENTIFIER: TEST-INVALID-INPUT-NUMBER

TEST CASE: Run the program with invalid numbers as input parameters

PRECONDITIONS: Start the program with the command: java -jar profwumpus.jar 1

EXECUTION STEPS:

Pass in the following invalid inputs:

1) “1”

2) “-2”

3) “10”

4) “1000000000”

POSTCONDITIONS: For each invalid input, the Student marker will not move, and the program will output: “Please enter N, S, E, or W”

CASE 5**:**

IDENTIFIER: TEST-VALID-LOWERCASE-INPUT

TEST CASE: Run the program and pass in N,E,S,and W in lowercase form

PRECONDITIONS: Start the program with the command: java -jar profwumpus.jar 1

EXECUTION STEPS:

Pass in the following invalid inputs:

1) “e”

2) “s”

3) “w”

4) “n”

POSTCONDITIONS: For each input, the student marker will move in the direction specified.

CASE 6**:**

IDENTIFIER: TEST-VALID-LOWERCASE-INPUT-AT-WALLS

TEST CASE: Test that valid lowercase input is accepted when the move itself is invalid because the user would be walking into a wall

PRECONDITIONS: Start the program with the command: java -jar profwumpus.jar 1

EXECUTION STEPS:

Pass in the following inputs:

1) “n” 🡪 1x

2) “s” 🡪 5x (until bottom left wall is reached)

3) “w” 🡪 1x

4) “e” 🡪 5x (until bottom right wall is reached)

POSTCONDITIONS:

Each time the user attempts to walk into a wall—the program will accept the lowercase input, and respond with the following error message: “That’s a wall there, buddy!”

CASE 7**:**

IDENTIFIER:

TEST CASE:

PRECONDITIONS:

EXECUTION STEPS:

POSTCONDITIONS:

CASE 8**:**

IDENTIFIER:

TEST CASE:

PRECONDITIONS:

EXECUTION STEPS:

POSTCONDITIONS:

CASE 9**:**

IDENTIFIER:

TEST CASE:

PRECONDITIONS:

EXECUTION STEPS:

POSTCONDITIONS:

CASE 10**:**

IDENTIFIER:

TEST CASE:

PRECONDITIONS:

EXECUTION STEPS:

POSTCONDITIONS:

CASE 11**:**

IDENTIFIER:

TEST CASE:

PRECONDITIONS:

EXECUTION STEPS:

POSTCONDITIONS:

CASE 12**:**

IDENTIFIER:

TEST CASE:

PRECONDITIONS:

EXECUTION STEPS:

POSTCONDITIONS:

CASE 13**:**

IDENTIFIER:

TEST CASE:

PRECONDITIONS:

EXECUTION STEPS:

POSTCONDITIONS:

CASE 14**:**

IDENTIFIER:

TEST CASE:

PRECONDITIONS:

EXECUTION STEPS:

POSTCONDITIONS:

CASE 15**:**

IDENTIFIER:

TEST CASE:

PRECONDITIONS:

EXECUTION STEPS:

POSTCONDITIONS:

CASE 16**:**

IDENTIFIER:

TEST CASE:

PRECONDITIONS:

EXECUTION STEPS:

POSTCONDITIONS:

CASE 17**:**

IDENTIFIER:

TEST CASE:

PRECONDITIONS:

EXECUTION STEPS:

POSTCONDITIONS:

CASE 18**:**

IDENTIFIER:

TEST CASE:

PRECONDITIONS:

EXECUTION STEPS:

POSTCONDITIONS:

CASE 19**:**

IDENTIFIER:

TEST CASE:

PRECONDITIONS:

EXECUTION STEPS:

POSTCONDITIONS:

CASE 20**:**

IDENTIFIER:

TEST CASE:

PRECONDITIONS:

EXECUTION STEPS:

POSTCONDITIONS:

CASE 21**:**

IDENTIFIER:

TEST CASE:

PRECONDITIONS:

EXECUTION STEPS:

POSTCONDITIONS:

CASE 22**:**

IDENTIFIER:

TEST CASE:

PRECONDITIONS:

EXECUTION STEPS:

POSTCONDITIONS:

**TRACEABILITY MATRIX**

|  |  |  |
| --- | --- | --- |
| TEST CASE | REQUIREMENT | PASS/FAIL |
| Case 1: TEST-DISPLAY-ON-WUMPUS-INTERCEPT | 1. The game shall consist of a 6 by 6 matrix of rooms, which shall be displayed to the player at each iteration, along with the location of the Student (indicated as an S). | **FAIL. This is an EDGE CASE.** |
| Case 2: TEST-MATRIX-DIMSIONS | 1. The game shall consist of a 6 by 6 matrix of rooms, which shall be displayed to the player at each iteration, along with the location of the Student (indicated as an S). | **FAIL** |
| Case 3: TEST-INVALID-INPUT-LETTER | 2. At each iteration, the player shall be able to input the direction they wish the Student to move (N for North, S for South, E for East or W for West). There are no other options. If a player enters any command other than N, S, E, or W, or their lowercase equivalents, the game shall display "Please enter N, S, E, or W". | **PASS** |
| Case 4: TEST-INVALID-INPUT-NUMBER | 2. At each iteration, the player shall be able to input the direction they wish the Student to move (N for North, S for South, E for East or W for West). There are no other options. If a player enters any command other than N, S, E, or W, or their lowercase equivalents, the game shall display "Please enter N, S, E, or W". | **PASS** |
| Case 5: TEST-VALID-LOWERCASE-INPUT | **3. All user input shall be case-insensitive. For example, typing either "N" or "n" shall take the Student to the room to the North (if it exists).** | **PASS** |
| Case 6: TEST-VALID-LOWERCASE-AT-WALLS | **3. All user input shall be case-insensitive. For example, typing either "N" or "n" shall take the Student to the room to the North (if it exists).** | **FAIL** |
| 7 |  |  |
| 8 |  |  |
| 9 |  |  |
| 10 |  |  |
| 11 |  |  |
| 12 |  |  |
| 13 |  |  |
| 14 |  |  |
| 15 |  |  |
| 16 |  |  |
| 17 |  |  |
| 18 |  |  |
| 19 |  |  |
| 20 |  |  |
| 21 |  |  |
| 22 |  |  |

**DEFECTS FOUND**

DEFECT 1**:**

SUMMARY: Matrix Display NOT updated when Student is intercepted by Prof Wumpus

DESCRIPTION: When a user walks into Professor Wumpus, the location is not updated before ending the game. -- MINOR

REPRODUCTION STEPS: 1) Start the program with the command: java -jar profwumpus.jar 1. 2) Move EAST 4 times. Press: E, Enter; E, Enter; E, Enter; E, Enter.

EXPECTED BEHAVIOR: First: The user’s location is displayed in the matrix at ROW=1, COL=4. Second: Display the message “Prof Wumpus sees you, but you don't have your assignment. YOU LOSE!” Third: Exit program.

OBSERVED BEHAVIOR: The user’s location is NOT updated, and Student remains in the matrix at ROW=1, COL=3—even though s/he has moved into room [1,4]. The correct message displays, and program exits.

DEFECT 2**:**

SUMMARY: Matrix is not 6x6

DESCRIPTION: The Matrix has dimensions 5x5, but the requirements specify that dimensions must be 6x6 -- MAJOR

REPRODUCTION STEPS: 1) Start the program with the command: java -jar profwumpus.jar 1. 2) Count the number of rows and columns that appear on the screen, starting at 1.

EXPECTED BEHAVIOR: There will be 6 rows and 6 columns.

OBSERVED BEHAVIOR: There are 5 rows and 5 columns

DEFECT 3**:**

SUMMARY: Program crashes when trying to walk into bottom right wall

DESCRIPTION: The program crashes when the user tries to walk into the bottom right wall of the game-space matrix, using the lowercase input character “e”.

REPRODUCTION STEPS:

Start the program with the command: java -jar profwumpus.jar 1.

Pass in the following inputs:

1) “n” 🡪 1x

2) “s” 🡪 5x (until bottom left wall is reached)

3) “w” 🡪 1x

4) “e” 🡪 5x (until bottom right wall is reached)

EXPECTED BEHAVIOR: Each time the user attempts to walk into a wall—the program will accept the lowercase input, and respond with the following error message: “That’s a wall there, buddy!”

OBSERVED BEHAVIOR: When the user attempts to walk into the bottom right wall, using the input “e”, the program crashes and outputs the following error message:

Exception in thread "main" java.lang.ArrayIndexOutOfBoundsException: 5

at ProfWumpus.moveStudent(ProfWumpus.java:51)

at ProfWumpus.playGame(ProfWumpus.java:335)

at ProfWumpus.main(ProfWumpus.java:362)