Sample test plan

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Test | Expected | Actual | Date | Notes |
| Expected cases | *Inputting coordinate*  *[1, a]* | *Cell [1, a] is flagged.* |  | *15/05/24* | *Coordinate for y axis is string but it has to be an integer in order to locate a particular cell e.g. board[3][3]* |
| *Inputting coordinate*  *[1, a]* | *Cell [1, a] is flagged.* | *Cell [1, a] is flagged.* | 15/05/24 |  |
| When open a cell that is empty, it should expand out. |  | \*\*stack overflow error. | 1/06/24 | The recursive function that check the cell around one cell checks each other back and forth causes infinite loop. |
| When open a cell that is empty, it should expand out. |  |  | 2/06/24 | If the cell is already check and empty change that cell to “0” then if the cell is “0” return because its already been check. |
| When all the flag is use and the flag is at the correct location. | Tell the player that they won. |  | 5/06/24 |  |
|  |  |  |  |  |
| Invalid cases | *e.g. -1* | *“You must enter a number >= 0”* |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
| Boundary cases | *1/1000000000000000* | *0* |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

Sample test plan

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Test | Expected | Actual | Date | Notes |
| Expected cases | *Inputting coordinate*  *[1, a]* | *Cell [1, a] is flagged.* |  | *15/05/24* | *Coordinate for y axis is string but it has to be an integer in order to locate a particular cell e.g. board[3][3]* |
| *Inputting coordinate*  *[1, a]* | *Cell [1, a] is flagged.* | *Cell [1, a] is flagged.* | 15/05/24 |  |
| When open a cell that is empty, it should expand out. |  | \*\*stack overflow error. | 1/06/24 | The recursive function that check the cell around one cell checks each other back and forth causes infinite loop. |
| When open a cell that is empty, it should expand out. |  |  | 2/06/24 | If the cell is already check and empty change that cell to “0” then if the cell is “0” return because its already been check. |
| When all the flag is use and the flag is at the correct location. | Tell the player that they won. |  | 5/06/24 |  |
|  |  |  |  |  |
| Invalid cases | *e.g. -1* | *“You must enter a number >= 0”* |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
| Boundary cases | *1/1000000000000000* | *0* |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

Sample test plan

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Test | Expected | Actual | Date | Notes |
| Expected cases | *Inputting coordinate*  *[1, a]* | *Cell [1, a] is flagged.* |  | *15/05/24* | *Coordinate for y axis is string but it has to be an integer in order to locate a particular cell e.g. board[3][3]* |
| *Inputting coordinate*  *[1, a]* | *Cell [1, a] is flagged.* | *Cell [1, a] is flagged.* | 15/05/24 |  |
| When open a cell that is empty, it should expand out. |  | \*\*stack overflow error. | 1/06/24 | The recursive function that check the cell around one cell checks each other back and forth causes infinite loop. |
| When open a cell that is empty, it should expand out. |  |  | 2/06/24 | If the cell is already check and empty change that cell to “0” then if the cell is “0” return because its already been check. |
| When all the flag is use and the flag is at the correct location. | Tell the player that they won. |  | 5/06/24 |  |
|  |  |  |  |  |
| Invalid cases | *e.g. -1* | *“You must enter a number >= 0”* |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
| Boundary cases | *1/1000000000000000* | *0* |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |