

Name: Tom Owen

Student Number: 2520277

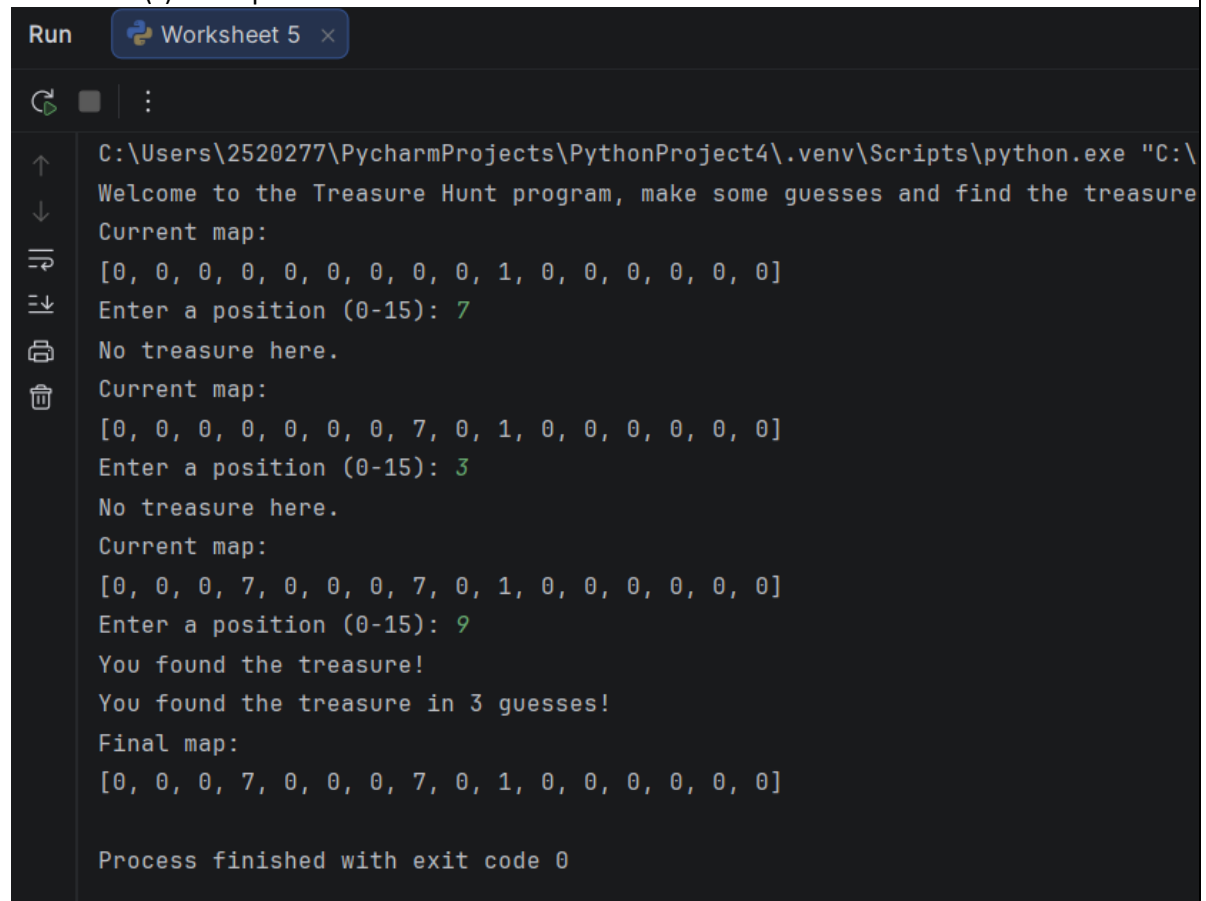
WORKSHEET 5 – TESTING (Hunting for Treasure)

For this worksheet, the program can be implemented in a number of different ways and demonstrate different features.

Create your own test set to demonstrate strengths and/or limitations of your solution. Use as many of the template boxes below as you need.

Test 1 - Normal game: wrong guesses then treasure found

Screenshot(s) of output:



```
Run Worksheet 5 x
C:\Users\2520277\PycharmProjects\PythonProject4\.venv\Scripts\python.exe "C:\
Welcome to the Treasure Hunt program, make some guesses and find the treasure
Current map:
[0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0]
Enter a position (0-15): 7
No treasure here.
Current map:
[0, 0, 0, 0, 0, 0, 0, 7, 0, 1, 0, 0, 0, 0, 0, 0]
Enter a position (0-15): 3
No treasure here.
Current map:
[0, 0, 0, 7, 0, 0, 0, 7, 0, 1, 0, 0, 0, 0, 0, 0]
Enter a position (0-15): 9
You found the treasure!
You found the treasure in 3 guesses!
Final map:
[0, 0, 0, 7, 0, 0, 0, 7, 0, 1, 0, 0, 0, 0, 0, 0]

Process finished with exit code 0
```

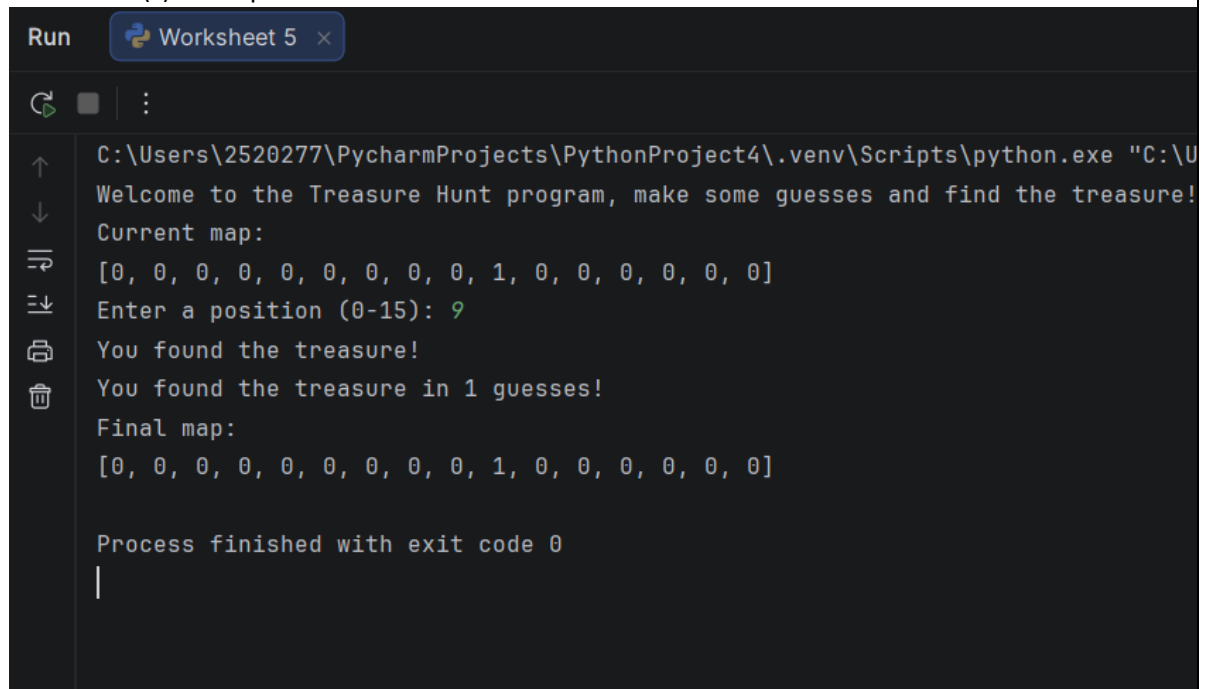
Result: Did the programme behave as expected ie syntactically and logically correct?

YES

The program behaved correctly. The user guessed positions 7 and 3 first, which were incorrect, so the program marked those positions with 7. The user then guessed position 9, where the treasure was stored, and the program displayed that the treasure had been found. It also correctly counted that the treasure was found in 3 guesses.

Test 2 - Treasure found straight away

Screenshot(s) of output:



```
Run Worksheet 5 x
C:\Users\2520277\PycharmProjects\PythonProject4\.venv\Scripts\python.exe "C:\U
Welcome to the Treasure Hunt program, make some guesses and find the treasure!
Current map:
[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0]
Enter a position (0-15): 9
You found the treasure!
You found the treasure in 1 guesses!
Final map:
[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0]

Process finished with exit code 0
|
```

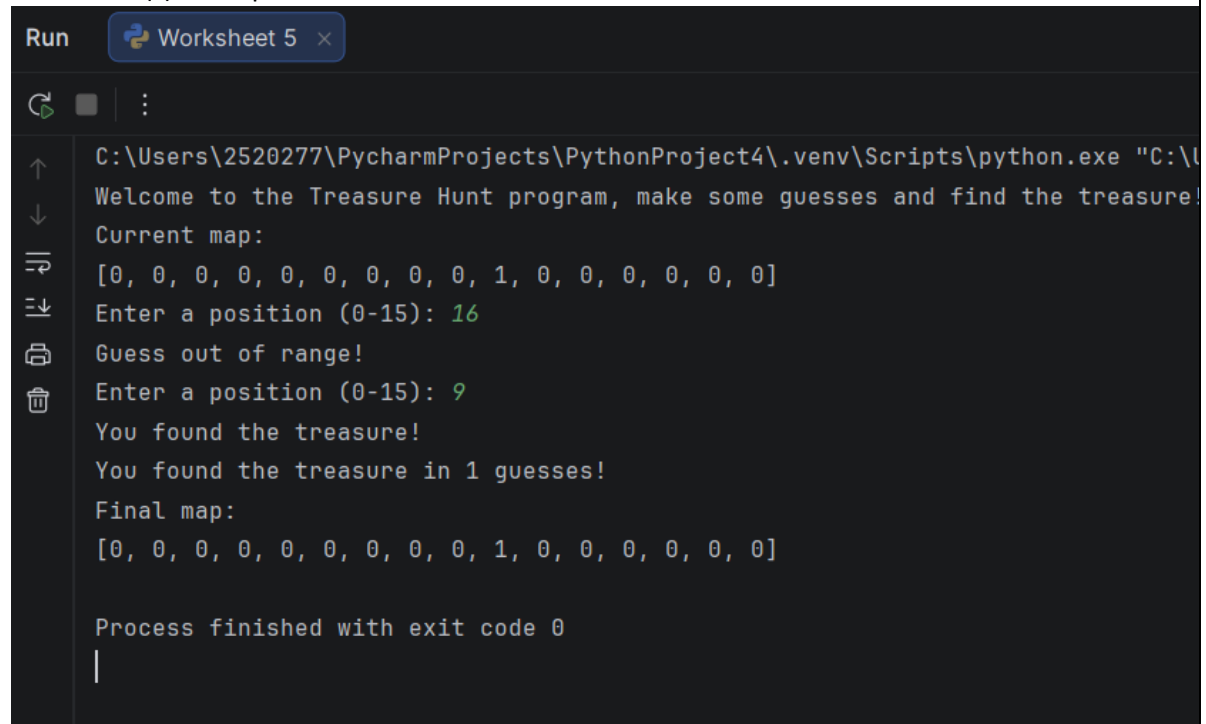
Result: Did the programme behave as expected ie syntactically and logically correct?

YES

The program behaved correctly. The user entered position 9, which is where the treasure is stored. The program displayed that the treasure was found and correctly counted that it was found in 1 guess. The final map was then displayed.

Test 3 - out-of-range high number

Screenshot(s) of output:



```
Run Worksheet 5 x
C:\Users\2520277\PycharmProjects\PythonProject4\.venv\Scripts\python.exe "C:\l
Welcome to the Treasure Hunt program, make some guesses and find the treasure!
Current map:
[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0]
Enter a position (0-15): 16
Guess out of range!
Enter a position (0-15): 9
You found the treasure!
You found the treasure in 1 guesses!
Final map:
[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0]

Process finished with exit code 0
|
```

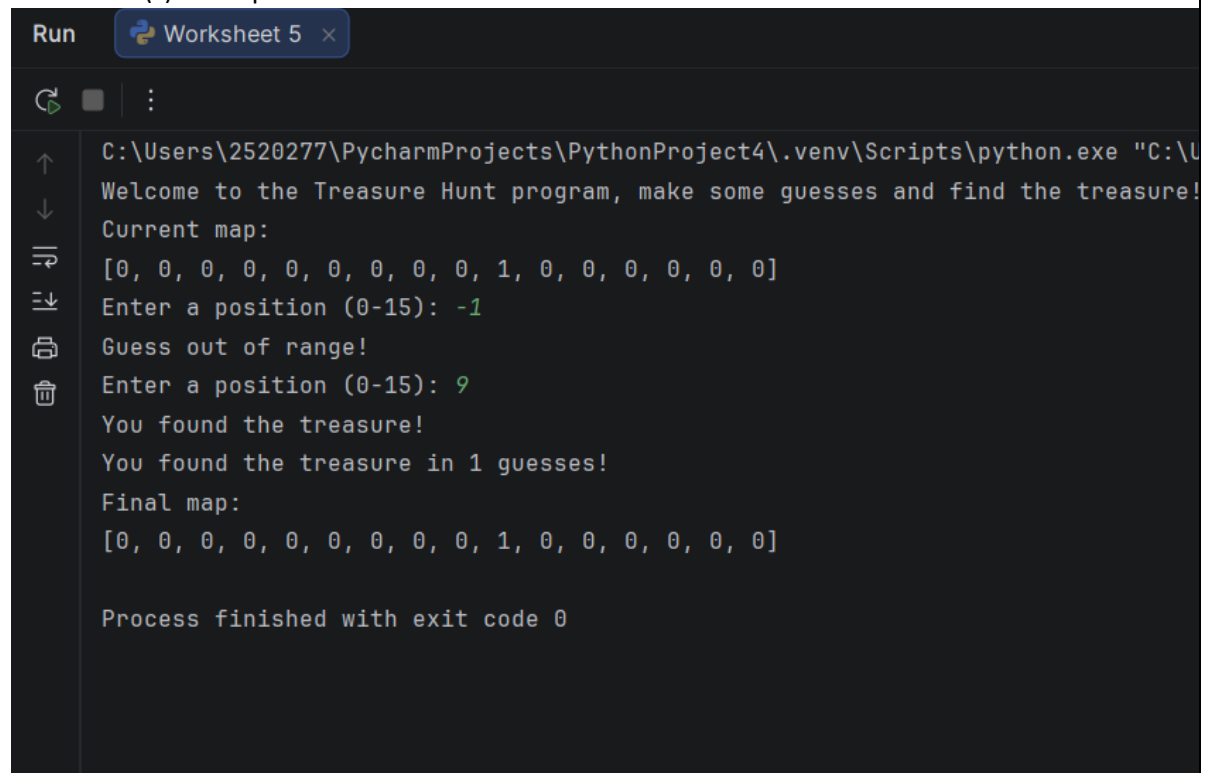
Result: Did the programme behave as expected ie syntactically and logically correct?

YES

The program behaved correctly. The user entered 16, which is outside the valid range of 0 to 15, so the program displayed "Guess out of range!" and asked for another position. The user then entered 9, where the treasure was stored, and the program found the treasure. This shows that the range validation works for numbers above 15.

Test 4 - out-of-range negative number.

Screenshot(s) of output:



```
Run Worksheet 5 x
C:\Users\2520277\PycharmProjects\PythonProject4\.venv\Scripts\python.exe "C:\U
Welcome to the Treasure Hunt program, make some guesses and find the treasure!
Current map:
[0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0]
Enter a position (0-15): -1
Guess out of range!
Enter a position (0-15): 9
You found the treasure!
You found the treasure in 1 guesses!
Final map:
[0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0]

Process finished with exit code 0
```

Result: Did the programme behave as expected ie syntactically and logically correct?

YES

The program behaved correctly. The user entered -1, which is outside the valid range of 0 to 15, so the program displayed "Guess out of range!" and asked for another position. The user then entered 9, where the treasure was stored, and the program found the treasure. This shows that the range validation works for negative numbers.

Final code:

[Copy and paste your code here – note, do not use a screenshot.

Note that there is no “before” and “after” code for this Worksheet because it is “Open” – not done in class. Therefore just post your final code below]

```
treasureMap = [0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0]
```

```
found = False
```

```
guesses = 0
```

```
print("Welcome to the Treasure Hunt program, make some guesses and find the treasure!")
```

```
while found == False:
```

```
    print("Current map:")
```

```
    print(treasureMap)
```

```
    guess = int(input("Enter a position (0-15): "))
```

```
    while guess < 0 or guess > 15:
```

```
        print("Guess out of range!")
```

```
        guess = int(input("Enter a position (0-15): "))
```

```
    guesses = guesses + 1
```

```
    if treasureMap[guess] == 1:
```

```
        print("You found the treasure!")
```

```
        print("You found the treasure in", guesses, "guesses!")
```

```
        found = True
```

```
    else:
```

```
        print("No treasure here.")
```

```
        treasureMap[guess] = 7
```

```
print("Final map:")
```

```
print(treasureMap)
```