

Name: Tom Owen

Student Number: 2520277

WORKSHEET 4 – TESTING (Dice Duel)

1. Original code:

[Copy and paste your original code from Moodle into this box. Do not change it.]

```
import random
playerRoll = []
comRoll = []

print("Do you want to role your dice? (y/n)")
choice = input()

if choice == 'y':
    for counter in range(0, 3):
        playerRoll.append (random.randint(1, 6))
        print("You rolled: ", playerRoll)

    for counter in range(0, 3):
        comRoll.append (random.randint(1, 6))
        print("Now the computer will roll: ", comRoll)

total = sum(playerRoll)
print("You rolled: ",total)

total = sum(comRoll)
print("The computer rolled: ",total)

""""if choice == 'n':
    print("Okay ")""""
```

2. Code Testing

For each test below (i – ix), run your code and show the results of the scenarios below. Present a screenshot of what happened when the programme ran. Did the programme behave correctly or did it do something unexpected, and if so, why?

Part a)

- i. Computer wins round
- ii. User wins round
- iii. Draw
- iv. "Go again" – enter *y*
- v. "Go again" – enter *n*
- vi. "Go again" – enter *Y*
- vii. "Go again" – enter *b*

Part b)

- viii. Show score/score tracking (depending on how you have implemented your program)
- ix. Show an overall win (however this is achieved eg first to 5, best of 12 etc)

Part c)

- x. Analyse your code and write a short paragraph commenting on the efficiency of the original code. Could efficiency be improved and if so comment on what is currently inefficient?

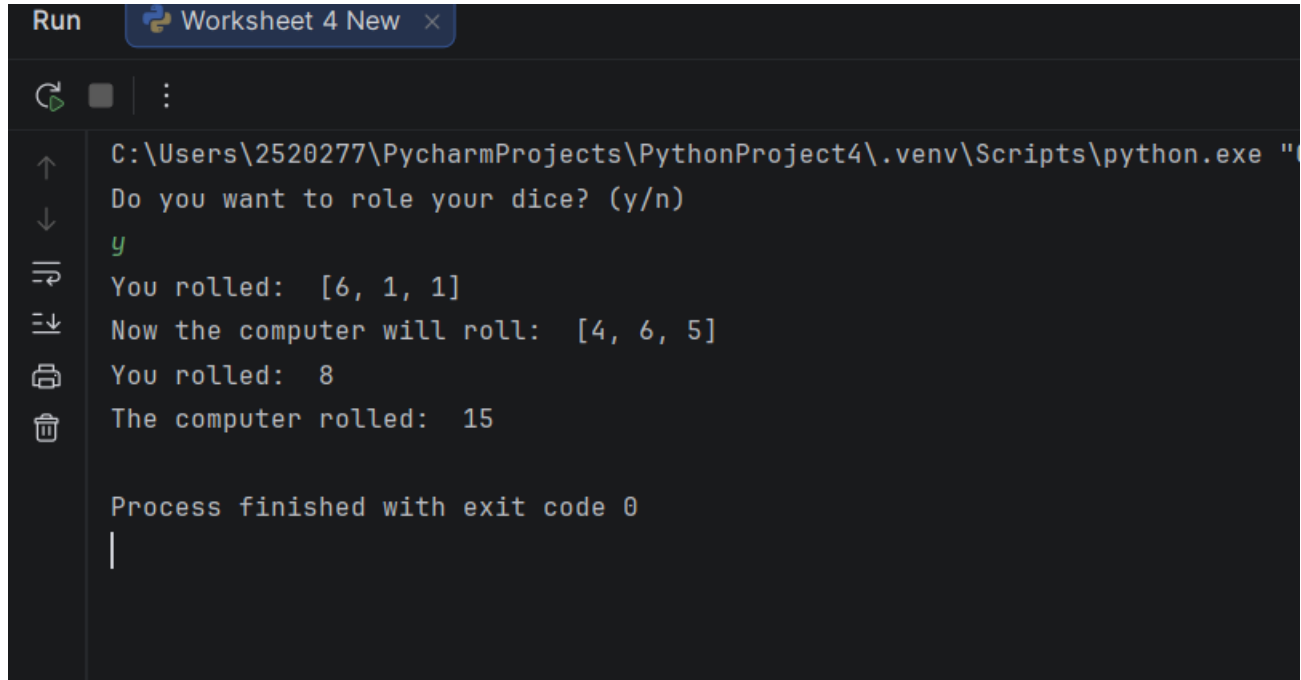
Template for tests:

Test:
Screenshot(s) of output:
Result: Did the programme behave as expected ie syntactically and logically correct? YES/NO (delete as appropriate) If YES, comment on what happened. If NO, what went wrong and why? Suggest how things could be improved

(Copy and paste box above for each of the rest of the tests)

Test: i. Computer wins round

Screenshot(s) of output:



```
Run Worksheet 4 New x
C:\Users\2520277\PycharmProjects\PythonProject4\.venv\Scripts\python.exe "
Do you want to role your dice? (y/n)
y
You rolled: [6, 1, 1]
Now the computer will roll: [4, 6, 5]
You rolled: 8
The computer rolled: 15

Process finished with exit code 0
|
```

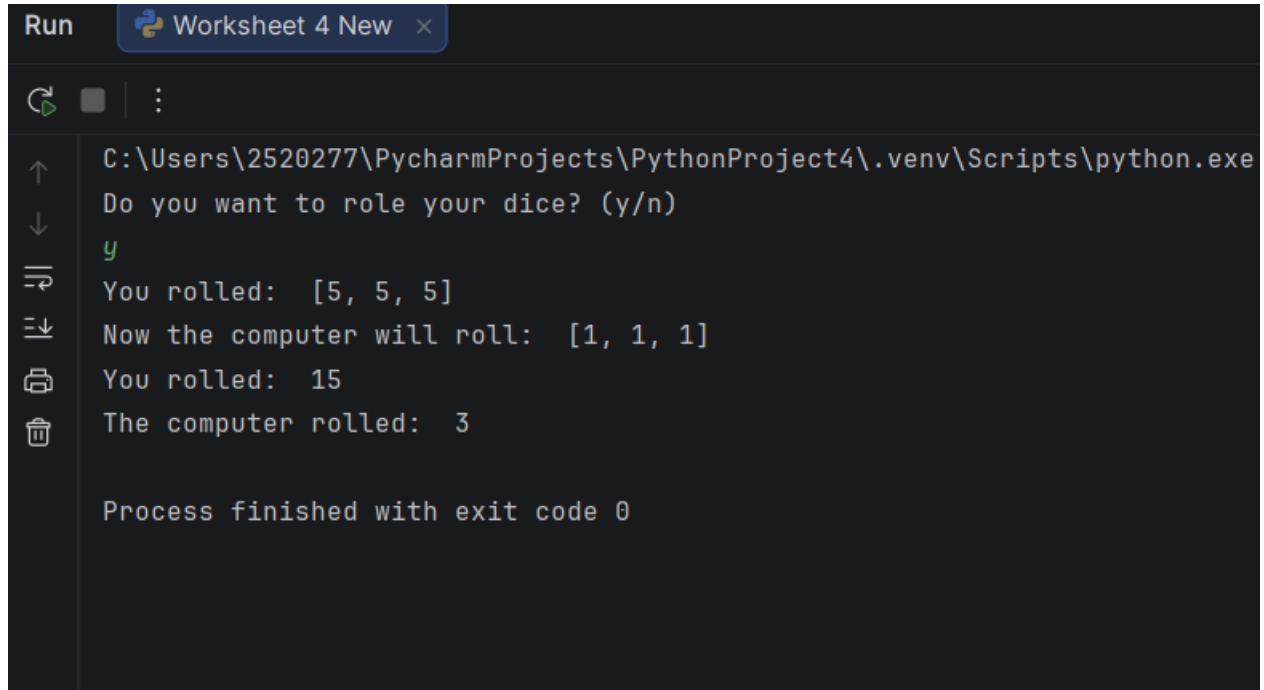
Result: Did the programme behave as expected ie syntactically and logically correct?

PARTLY

The program rolled three dice for the player and three dice for the computer. The player rolled a total of 8 and the computer rolled a total of 15, so the computer won the round. The totals were calculated correctly. However, the program did not actually display a message saying that the computer won, so this part is only partly complete in the original code.

Test: ii. User wins round

Screenshot(s) of output:



```
Run Worksheet 4 New x
C:\Users\2520277\PycharmProjects\PythonProject4\.venv\Scripts\python.exe
Do you want to role your dice? (y/n)
y
You rolled:  [5, 5, 5]
Now the computer will roll:  [1, 1, 1]
You rolled:  15
The computer rolled:  3

Process finished with exit code 0
```

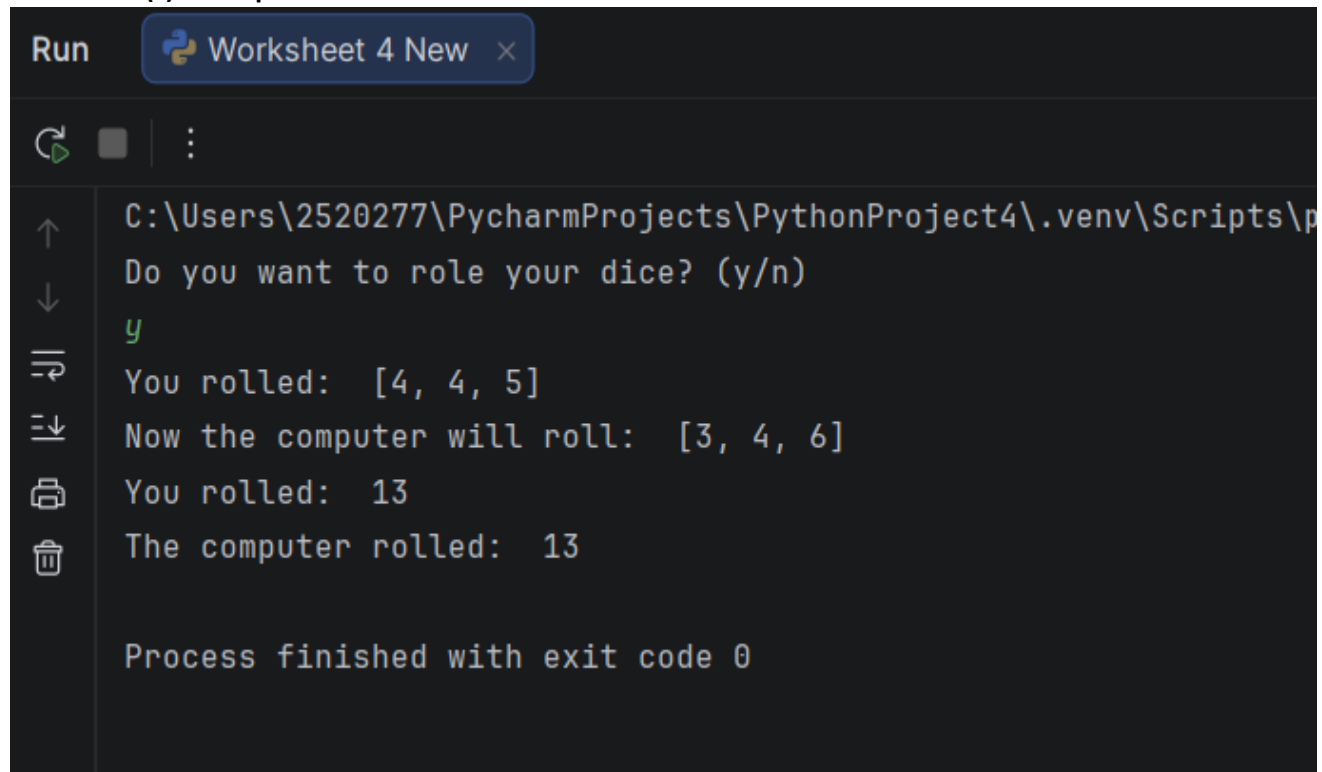
Result: Did the programme behave as expected ie syntactically and logically correct?

PARTLY

The program rolled three dice for the player and three dice for the computer. The player rolled a total of 15 and the computer rolled a total of 3, so the user won the round. The totals were calculated correctly. However, the program did not display a message saying that the user won, so this part is only partly complete in the original code.

Test: iii. Draw

Screenshot(s) of output:



```
Run Worksheet 4 New x
C:\Users\2520277\PycharmProjects\PythonProject4\.venv\Scripts\p
Do you want to role your dice? (y/n)
y
You rolled: [4, 4, 5]
Now the computer will roll: [3, 4, 6]
You rolled: 13
The computer rolled: 13

Process finished with exit code 0
```

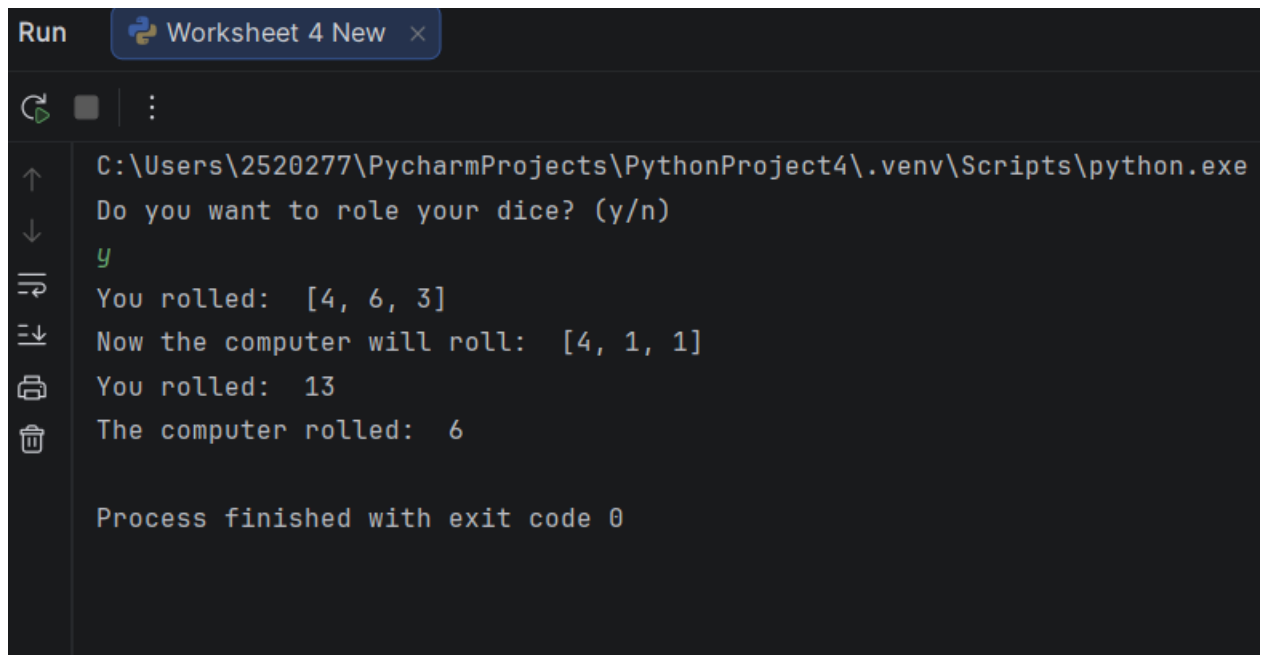
Result: Did the programme behave as expected ie syntactically and logically correct?

PARTLY

The program rolled three dice for the player and three dice for the computer. Both the player and the computer rolled a total of 13, so the round should be a draw. The totals were calculated correctly. However, the program did not display a message saying that the round was a draw, so this part is only partly complete in the original code.

Test: iv. "Go again" – enter y

Screenshot(s) of output:



```
Run Worksheet 4 New x
C:\Users\2520277\PycharmProjects\PythonProject4\.venv\Scripts\python.exe
Do you want to role your dice? (y/n)
y
You rolled: [4, 6, 3]
Now the computer will roll: [4, 1, 1]
You rolled: 13
The computer rolled: 6

Process finished with exit code 0
```

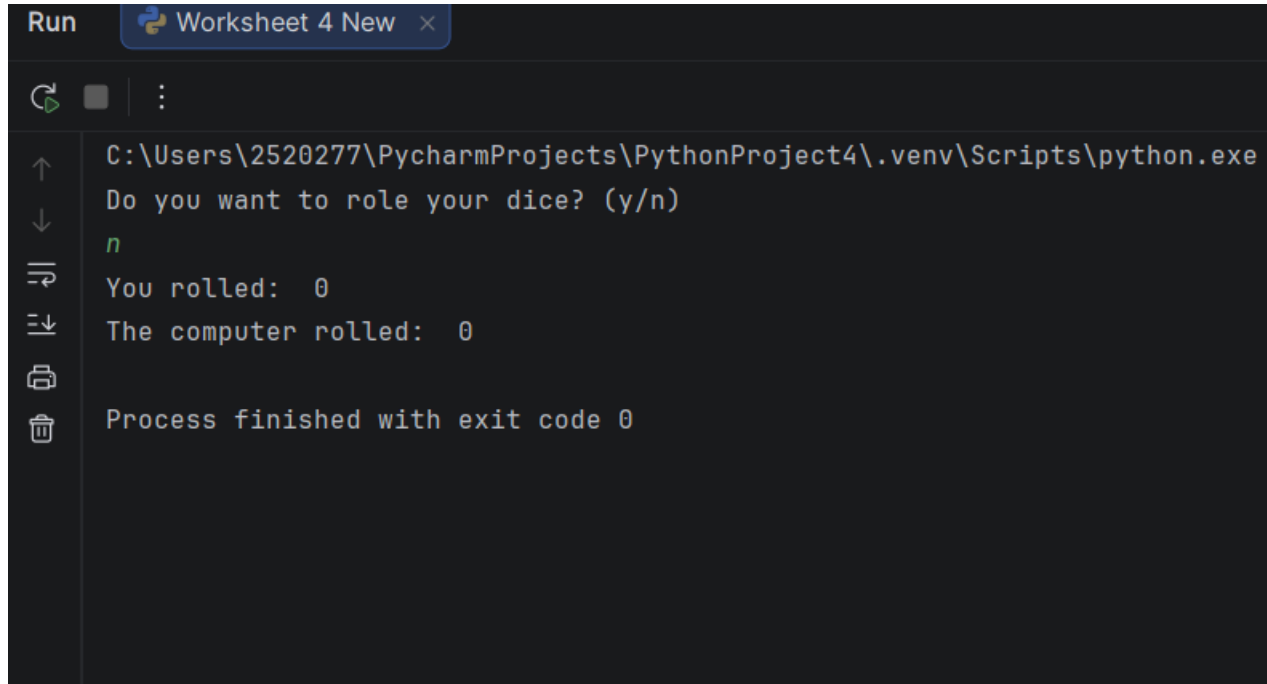
Result: Did the programme behave as expected ie syntactically and logically correct?

NO

The program accepted y and completed one round of dice rolls. It correctly rolled three dice for the player and three dice for the computer, then displayed both totals. However, the program did not ask the user if they wanted to go again after the round finished. This means the "go again" feature is missing from the original code and should be added in the revised version.

Test: v. "Go again" – enter n

Screenshot(s) of output:



```
Run Worksheet 4 New x
C:\Users\2520277\PycharmProjects\PythonProject4\.venv\Scripts\python.exe
Do you want to role your dice? (y/n)
n
You rolled: 0
The computer rolled: 0
Process finished with exit code 0
```

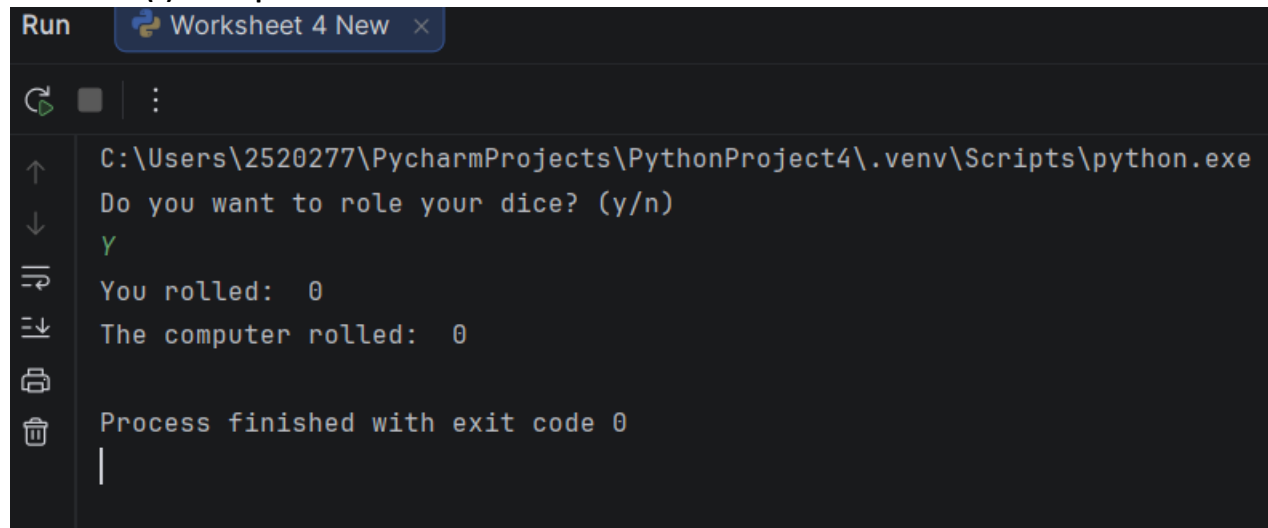
Result: Did the programme behave as expected ie syntactically and logically correct?

NO

The program accepted n, but it still displayed totals for the player and computer as 0. This is not logically correct because no dice were rolled. If the user enters n, the program should stop or display a message such as "Okay" instead of showing both scores as 0. This happened because the total calculations are outside the if choice == 'y' block, so they still run even when the user chooses not to roll.

Test: vi. "Go again" – enter Y

Screenshot(s) of output:



```
Run Worksheet 4 New x
C:\Users\2520277\PycharmProjects\PythonProject4\.venv\Scripts\python.exe
Do you want to role your dice? (y/n)
Y
You rolled:  0
The computer rolled:  0
Process finished with exit code 0
```

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

NO

The program did not treat uppercase Y as a valid yes input. It did not roll any dice, but still displayed both totals as 0. This is not logically correct because Y should ideally be accepted in the same way as lowercase y. This could be improved by converting the user's input to lowercase before checking it.

Test: vii. "Go again" – enter b

Screenshot(s) of output:



```
Run Worksheet 4 New x
C:\Users\2520277\PycharmProjects\PythonProject4\.venv\Scripts\python.exe
Do you want to role your dice? (y/n)
b
You rolled: 0
The computer rolled: 0
Process finished with exit code 0
|
```

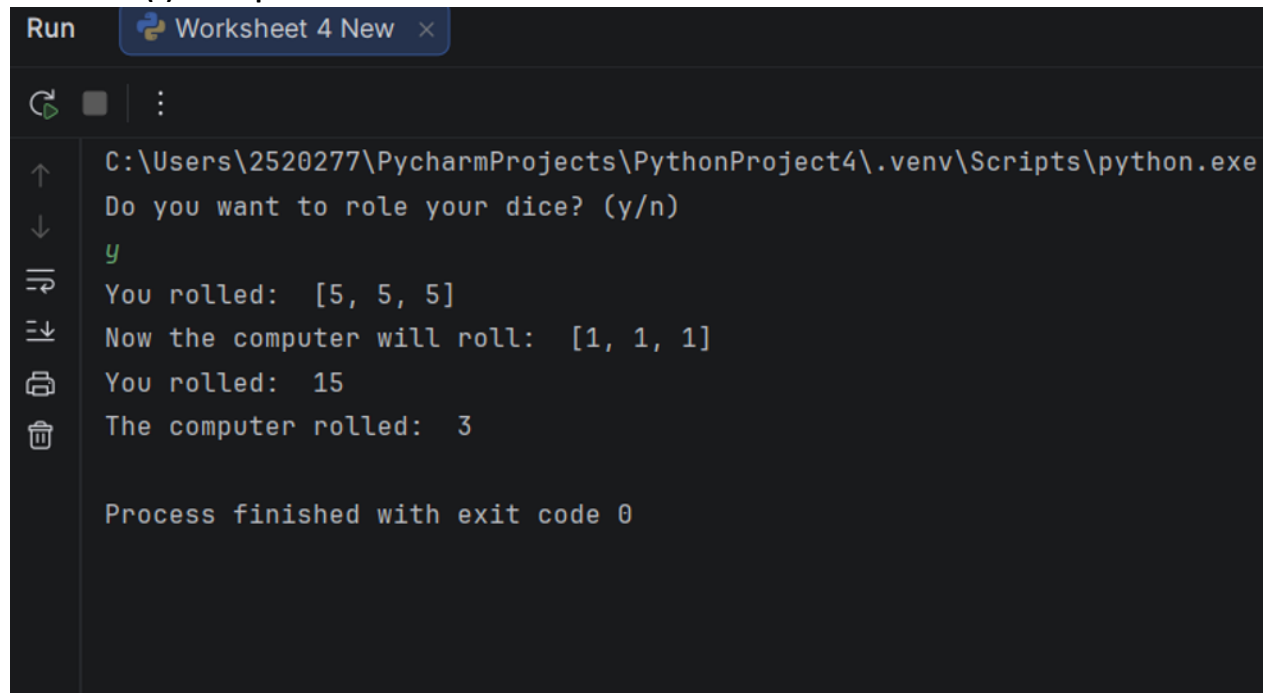
Result: Did the programme behave as expected ie syntactically and logically correct?

NO

The program did not handle the invalid input b. It did not roll any dice, but still displayed both totals as 0. This is not logically correct because b is not a valid option. The program should display an error message and ask the user to enter either y or n. This could be improved by adding input validation.

Test: viii. Show score / score tracking

Screenshot(s) of output:



```
Run Worksheet 4 New x
C:\Users\2520277\PycharmProjects\PythonProject4\.venv\Scripts\python.exe
Do you want to role your dice? (y/n)
y
You rolled:  [5, 5, 5]
Now the computer will roll:  [1, 1, 1]
You rolled:  15
The computer rolled:  3

Process finished with exit code 0
```

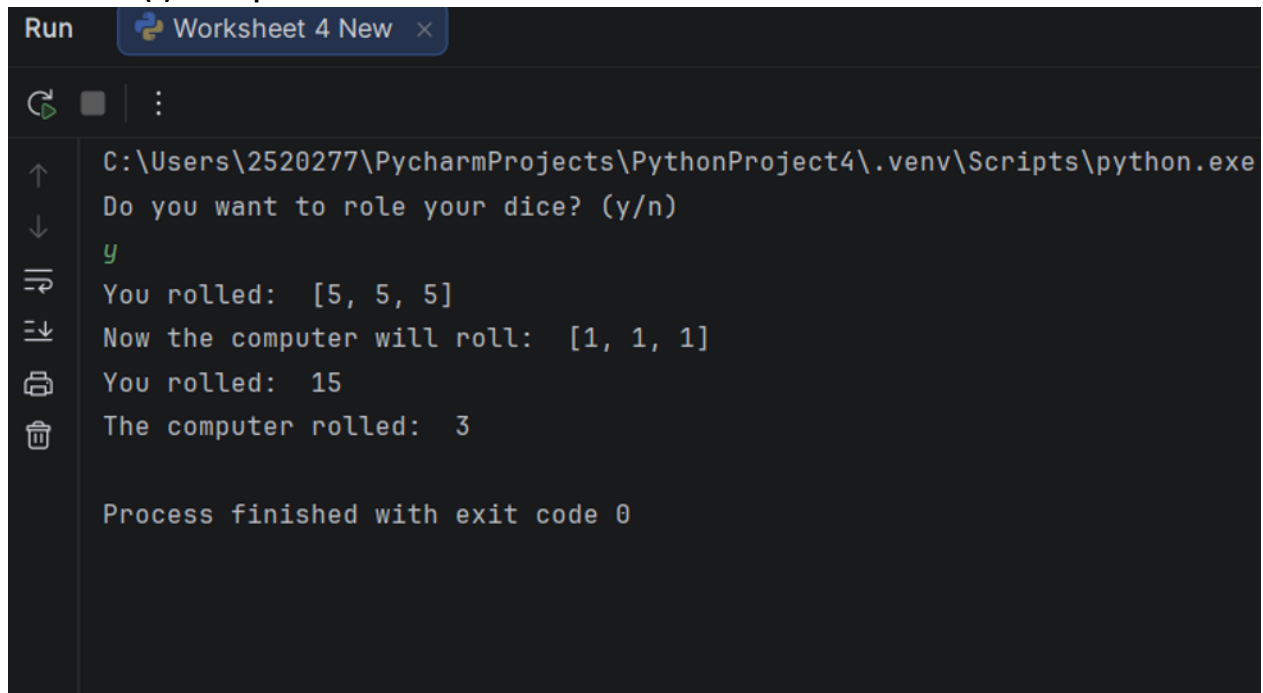
Result: Did the programme behave as expected ie syntactically and logically correct?

NO

The program displays the player total and computer total for one round, but it does not keep track of scores across multiple rounds. There is no player score, computer score, or round history shown. This means score tracking is missing from the original code and should be added in the revised version.

Test: ix. Show an overall win

Screenshot(s) of output:



```
Run Worksheet 4 New x
C:\Users\2520277\PycharmProjects\PythonProject4\.venv\Scripts\python.exe
Do you want to role your dice? (y/n)
y
You rolled:  [5, 5, 5]
Now the computer will roll:  [1, 1, 1]
You rolled:  15
The computer rolled:  3

Process finished with exit code 0
```

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

NO

The program does not show an overall win. It only plays one round and displays the player total and computer total. It does not repeat rounds, track the number of wins, or decide when the player or computer has won the overall duel. This feature is missing from the original code and should be added in the revised version.

Test: x. Efficiency analysis paragraph

The original code is not very efficient because it only plays one round and does not use a loop to repeat the game. It also repeats similar code for the player roll and computer roll, which could be improved by using a function or a loop. The code does not compare the totals to decide who won, does not track scores, and does not validate user input. The total calculations are also outside the if choice == 'y' block, so the program still prints scores of 0 when the user enters n, Y, or an invalid letter such as b. The efficiency could be improved by using loops, input validation, score variables, and possibly a function to roll three dice.

3. Consider the results of the above tests and your analysis of program efficiency and revisit your original code in order to address any tests that failed and/or improve code efficiency.

(Note: you may not be able to solve all of the problems that were encountered, but solve as many as you can.)

Paste your full revised code here :

[Copy and paste your full revised code here – note, do not use a screenshot]

```
import random

playerRoll = []
comRoll = []

playerScore = 0
comScore = 0

print("Do you want to role your dice? (y/n)")
choice = input()
choice = choice.lower()

if choice == 'y':
    for counter in range(0, 3):
        playerRoll.append(random.randint(1, 6))
    print("You rolled: ", playerRoll)

    for counter in range(0, 3):
        comRoll.append(random.randint(1, 6))
    print("Now the computer will roll: ", comRoll)

    playerTotal = sum(playerRoll)
    print("You rolled: ", playerTotal)

    comTotal = sum(comRoll)
    print("The computer rolled: ", comTotal)

    if playerTotal > comTotal:
        print("You won the round")
        playerScore = playerScore + 1

    elif comTotal > playerTotal:
        print("The computer won the round")
        comScore = comScore + 1

    else:
        print("Draw, no points")

    print("Current scores are: You:", playerScore, "Computer:", comScore)

elif choice == 'n':
    print("Okay")
```

```
else:
```

```
    print("Invalid choice. Please enter y or n.")
```