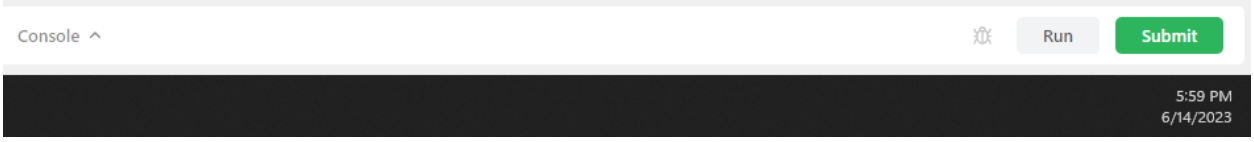
Step 1: Study [69. Sqrt(x)]

```python
class Solution:
    def mySqrt(self, x: int) -> int:
        left, right = 1, x
        while left <= right:
            mid = (left + right)//2
            if mid*mid == x:
                return mid
            elif mid*mid > x:
                right = mid - 1
            else:
                left = mid + 1
        return right
```



Step 2: Ask ChatGPT: "How to use Binary Search to implement LeetCode question: '69. Sqrt(x)', with the test data: Input: x = 8, Output: 2"

To solve the LeetCode question '69. Sqrt(x)' using binary search, you can follow these steps:

1. Define the search space: In this problem, the search space is the range of possible square root values for the given number 'x'. Since the square root of a positive number is always between 0 and x, we can define the search space as [0, x].
2. Initialize two pointers: Set the left pointer to 0 and the right pointer to x.
3. Perform binary search: While the left pointer is less than or equal to the right pointer, do the following steps:
   a. Calculate the middle value: Find the middle value by taking the average of the left and right pointers. Use integer division or floor function to obtain an integer value.
   b. Calculate the square of the middle value: Let's call this value 'midSquare'. Calculate 'midSquare = mid * mid'.
   c. Compare 'midSquare' with 'x': If 'midSquare' is equal to 'x', return the middle value 'mid' as the square root of 'x'.
   d. Adjust the pointers: If 'midSquare' is less than 'x', update the left pointer to 'mid + 1' since the square root must be larger. Otherwise, update the right pointer to 'mid - 1' since the square root must be smaller.
4. Return the result: If the loop ends without finding an exact square root, return the value of the right pointer, which would be the largest integer whose square is less than or equal to 'x'.

Here's the implementation of the algorithm in Python:

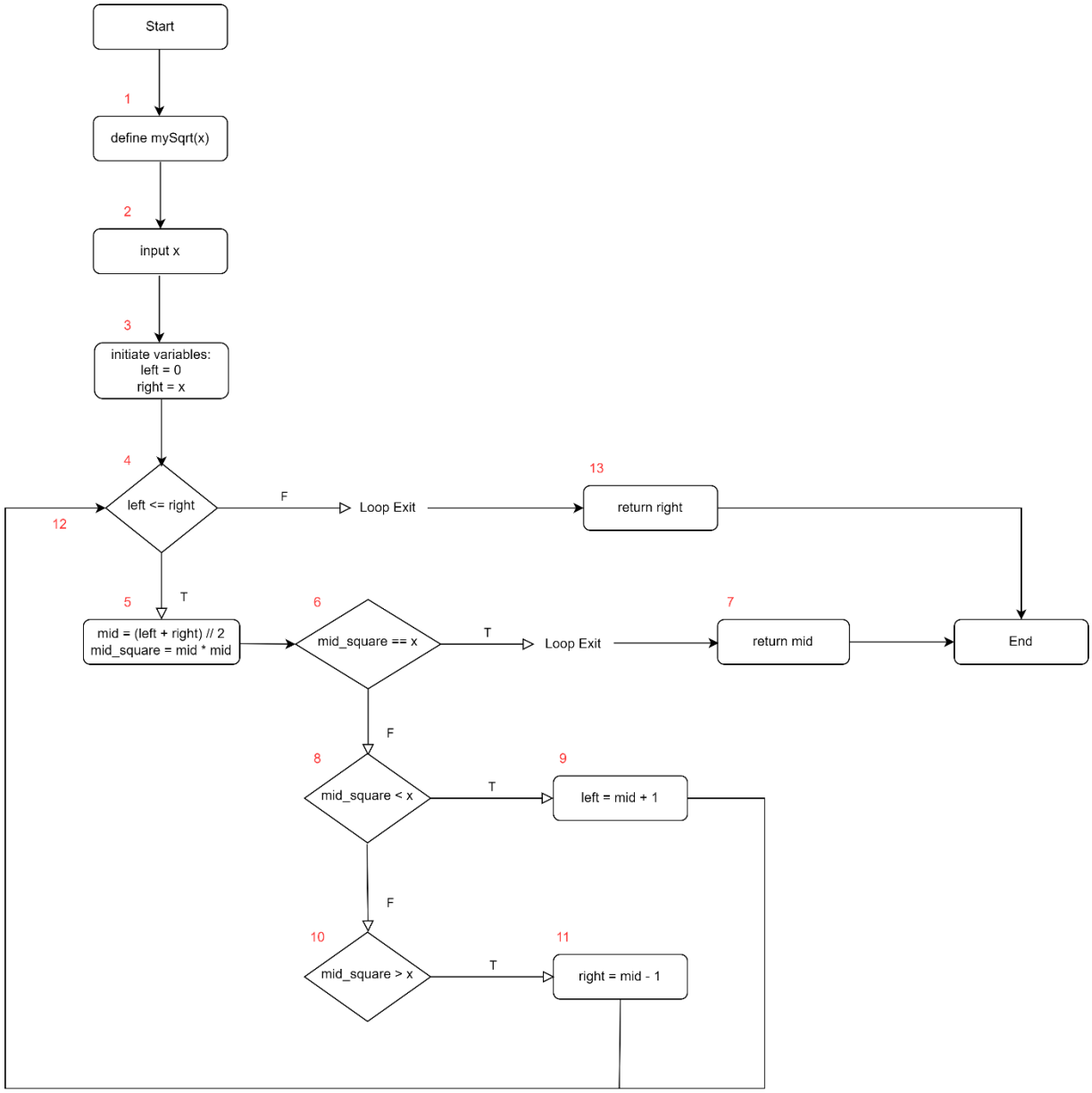
python Copy code

```python
def mySqrt(x):
    left = 0
    right = x

    while left <= right:
        mid = (left + right) // 2
        mid_square = mid * mid

        if mid_square == x:
            return mid
        elif mid_square < x:
            left = mid + 1
        else:
            right = mid - 1

    return right

# Test case
x = 8
result = mySqrt(x)
print(result)  # Output: 2
```

By using binary search, the algorithm can find the square root of 'x' in an efficient manner, with a time complexity of O(log x).

Step 3: Convert ChatGPT's Python code into a flowcharts.

- The syntax of flowchart
- Flowchart and Trace Table
- Single/Double Loops and Flow Chart
- Flowchart and Trace Table for Recursive Function

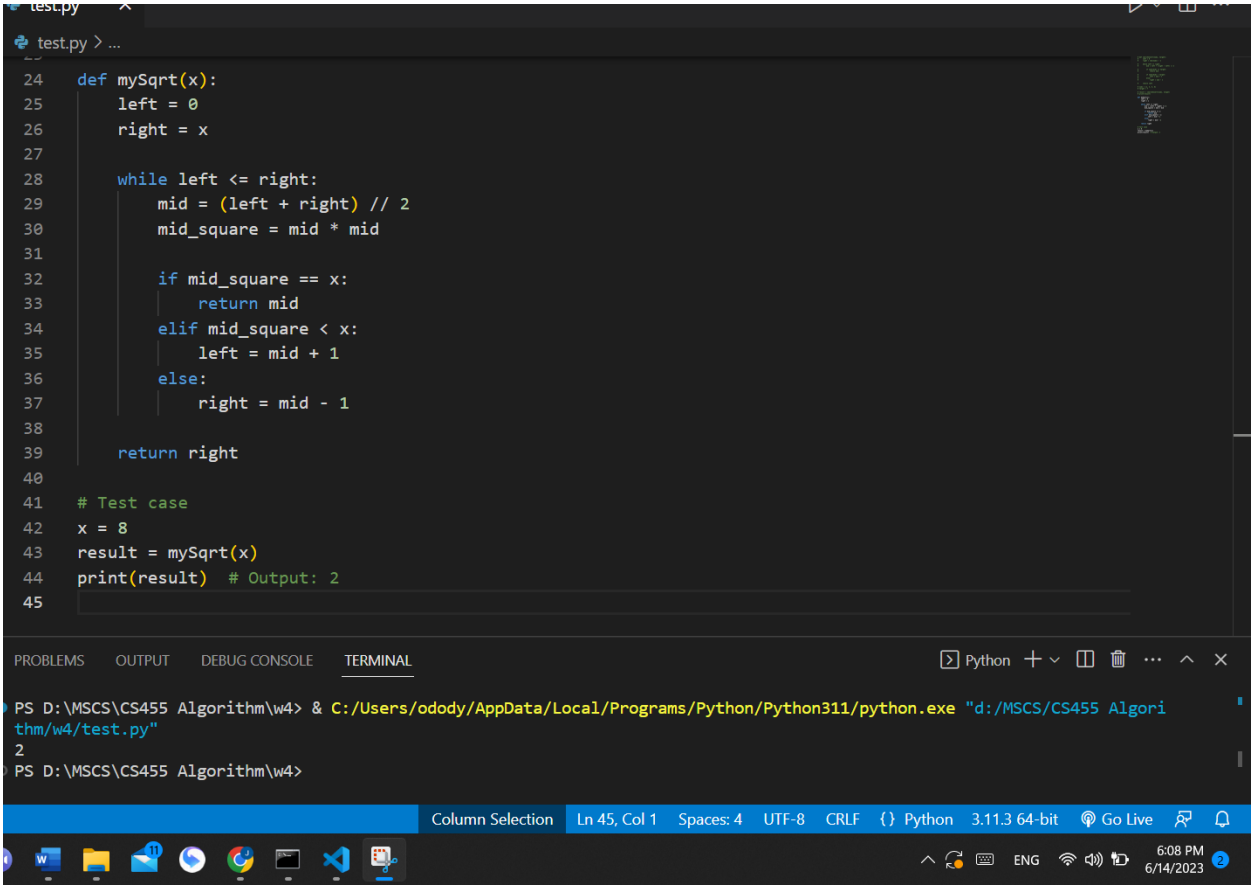


Step 4: Verify the flowchart with a trace tables. You only need to trace the test data: "Input: x = 8, Output: 2"

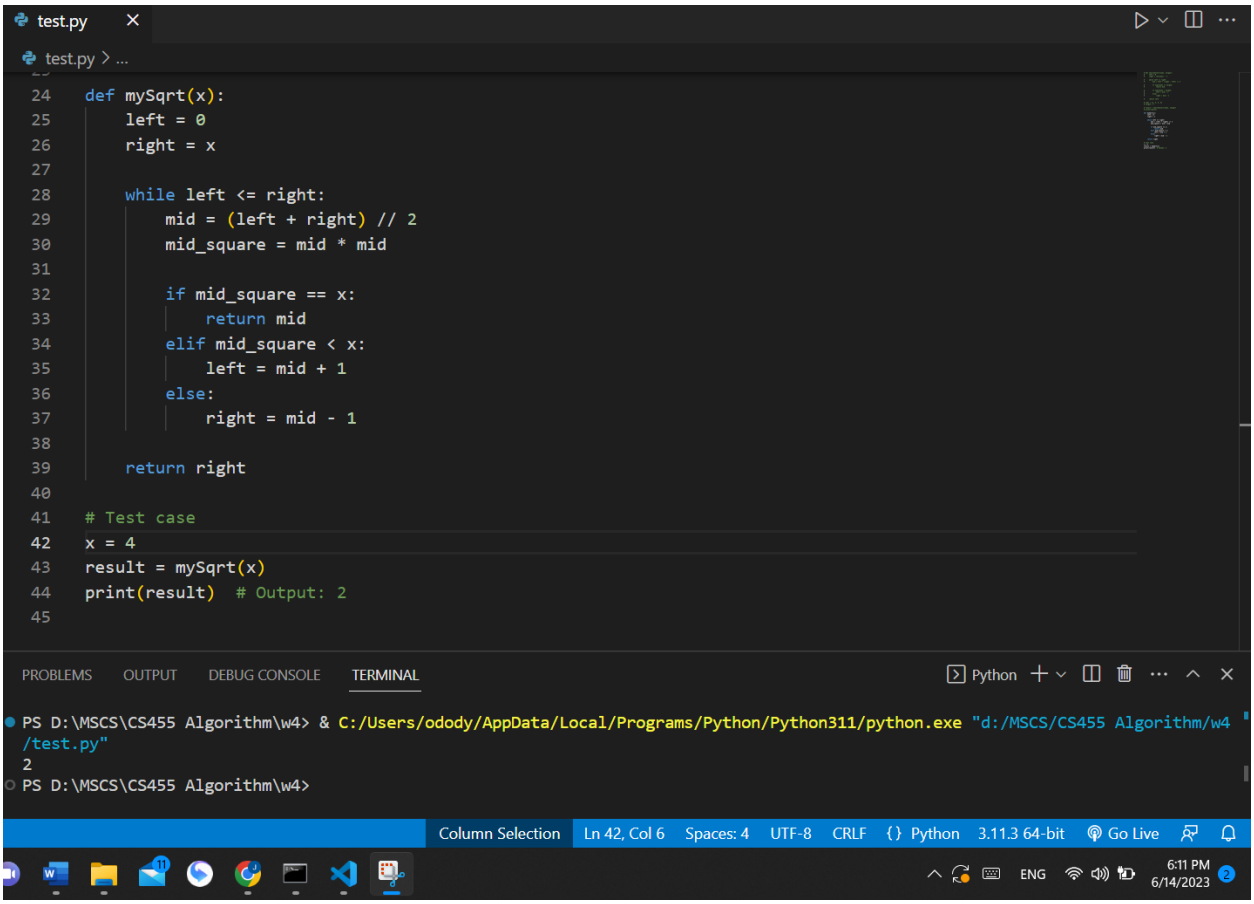
- Flowchart and Trace Table
- Single/Double Loops and Flow Chart
- Flowchart and Trace Table for Recursive Function

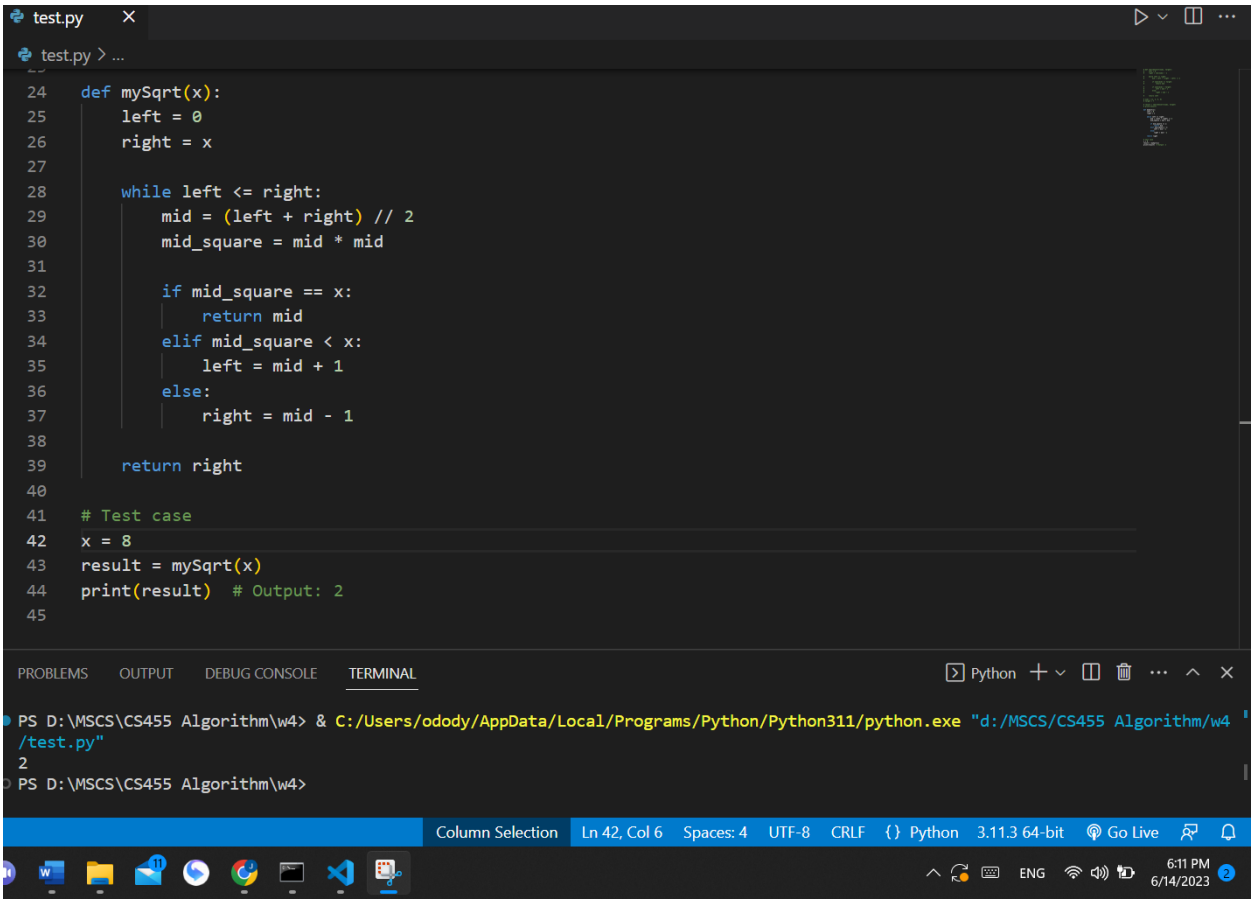
	RAM										screen
Step	define function	x	left	right	left <= right	mid	mid_square	mid_square == x	mid_square < x	mid_square > x	
1	mySqrt(x)										
2		x = 8									
3			0	8							
4					T						
5						4	16				
6								F			
8									F		
10										T	
11				3							
12					T						
5						1	1				
6								F			
8									T		
9			2								
12					T						
5						2	4				
6								F			
8									T		
9			3								
12					T						
5						3	9				
6								F			
8									F		
10										T	
11				2							
12					F						
13											2

Step 5: Implement the Python code created by ChatGPT



Step 6: Test the Python code with all the test cases provided by [69. Sqrt(x)]() ([local copy]())


test.py
test.py > ...
24 def mySqrt(x):
25 left = 0
26 right = x
27
28 while left <= right:
29 mid = (left + right) // 2
30 mid_square = mid * mid
31
32 if mid_square == x:
33 return mid
34 elif mid_square < x:
35 left = mid + 1
36 else:
37 right = mid - 1
38
39 return right
40
41 # Test case
42 x = 4
43 result = mySqrt(x)
44 print(result) # Output: 2
45
PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
Python
PS D:\MSCS\CS455 Algorithm\w4> & C:/Users/odody/AppData/Local/Programs/Python/Python311/python.exe "d:/MSCS/CS455 Algorithm/w4/test.py"
2
PS D:\MSCS\CS455 Algorithm\w4>
Column Selection
Ln 42, Col 6
Spaces: 4
UTF-8
CRLF
{} Python
3.11.3 64-bit
Go Live
ENG
6:11 PM
6/14/2023


test.py
test.py > ...
24 def mySqrt(x):
25 left = 0
26 right = x
27
28 while left <= right:
29 mid = (left + right) // 2
30 mid_square = mid * mid
31
32 if mid_square == x:
33 return mid
34 elif mid_square < x:
35 left = mid + 1
36 else:
37 right = mid - 1
38
39 return right
40
41 # Test case
42 x = 8
43 result = mySqrt(x)
44 print(result) # Output: 2
45
PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
Python
PS D:\MSCS\CS455 Algorithm\w4> & C:/Users/odody/AppData/Local/Programs/Python/Python311/python.exe "d:/MSCS/CS455 Algorithm/w4/test.py"
2
PS D:\MSCS\CS455 Algorithm\w4>
Column Selection
Ln 42, Col 6
Spaces: 4
UTF-8
CRLF
{} Python
3.11.3 64-bit
Go Live
ENG
6:11 PM
6/14/2023