

Week4_Q2

```
def searchInsert(nums, target):
    left = 0
    right = len(nums) - 1

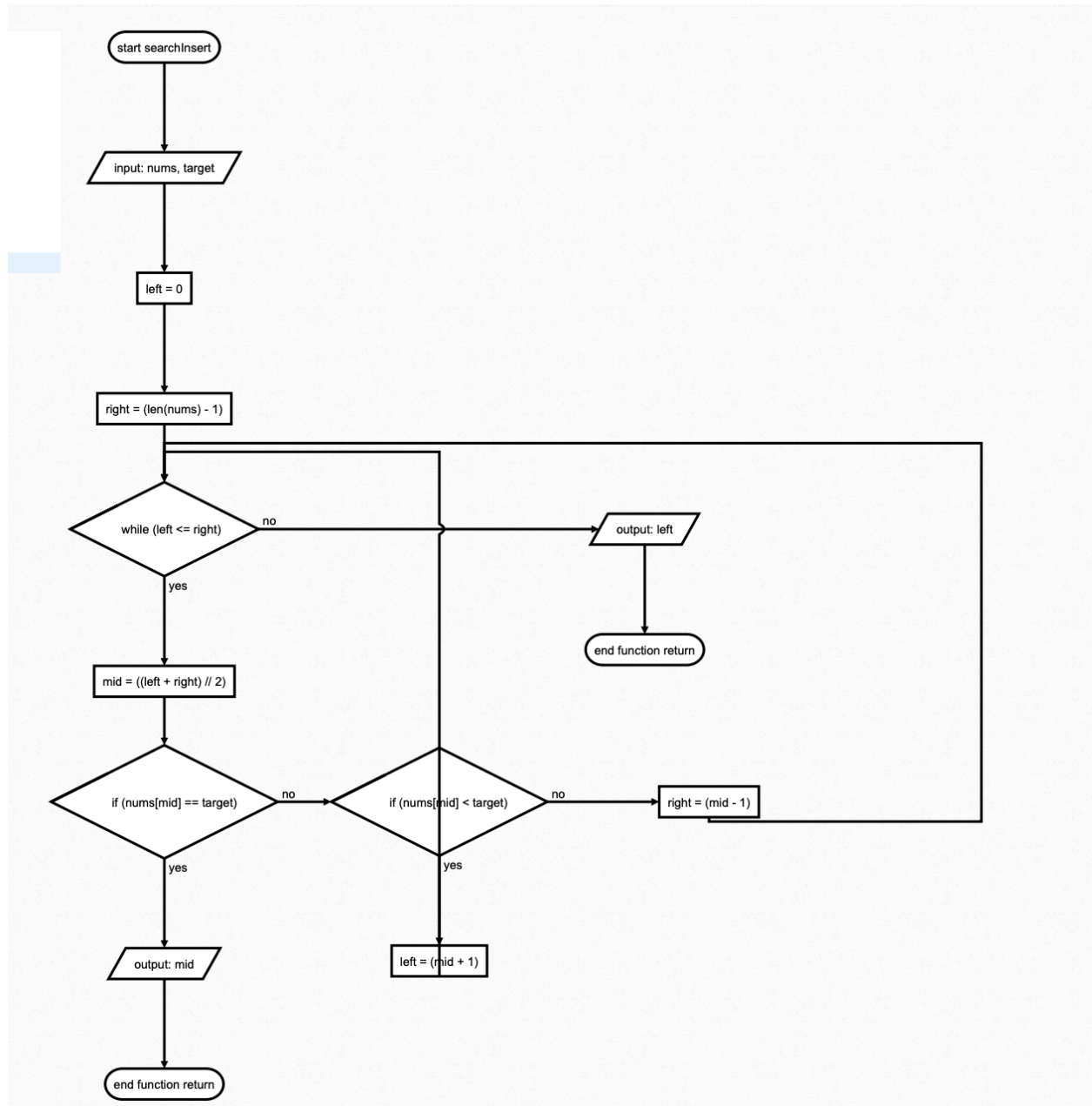
    while left <= right:
        mid = (left + right) // 2

        if nums[mid] == target:
            return mid
        elif nums[mid] < target:
            left = mid + 1
        else:
            right = mid - 1

    return left

# Test case
nums = [1, 3, 5, 6]
target = 5

result = searchInsert(nums, target)
print(result)
```



Trace table:

line	nums	target	left	right	left <= right	mid	nums[mid] == target	nums[mid] < target	return
1	[1,3,5,6]	5							
2			0						
3				4-1=3					
5					0 < 3 true				
6						(3+1)/2 = 2			
8							False		
10								3 < 5 = true	
11			0+1=1						
5					1 < 3 = true				
6						(1+3)/2 = 2			
8							True		
9									2

Test cases:

```
def searchInsert(nums, target):
    left = 0
    right = len(nums) - 1

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

        if nums[mid] == target:
            return mid
        elif nums[mid] < target:
            left = mid + 1
        else:
            right = mid - 1

    return left
```

```
# Test case
nums = [1, 3, 5, 6]
```

```
target = 5
```

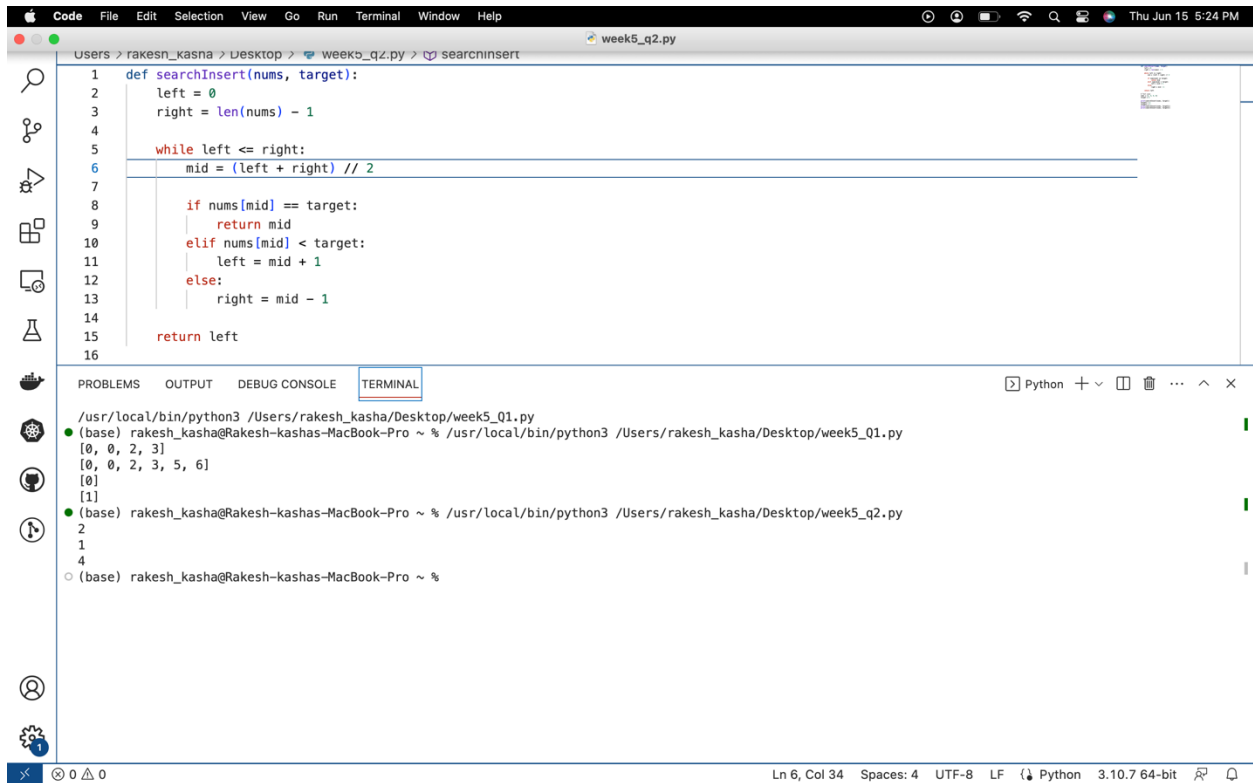
```
print(searchInsert(nums, target))
```

```
target1 = 2
```

```
target2 = 7
```

```
print(searchInsert(nums, target1))
```

```
print(searchInsert(nums, target2))
```



The screenshot shows a VS Code editor window with a file named `week5_q2.py` open. The file contains a Python function `searchInsert` that implements a binary search algorithm. The function takes a list `nums` and a `target` as input and returns the index of the target if it exists, or the index where it should be inserted. The code is as follows:

```
1 def searchInsert(nums, target):
2     left = 0
3     right = len(nums) - 1
4
5     while left <= right:
6         mid = (left + right) // 2
7
8         if nums[mid] == target:
9             return mid
10        elif nums[mid] < target:
11            left = mid + 1
12        else:
13            right = mid - 1
14
15    return left
16
```

The terminal output shows the execution of two Python scripts:

```
(base) rakesh_kasha@Rakesh-kashas-MacBook-Pro ~ % /usr/local/bin/python3 /Users/rakesh_kasha/Desktop/week5_q1.py
[0, 0, 2, 3]
[0]
[1]
(base) rakesh_kasha@Rakesh-kashas-MacBook-Pro ~ % /usr/local/bin/python3 /Users/rakesh_kasha/Desktop/week5_q2.py
2
1
4
(base) rakesh_kasha@Rakesh-kashas-MacBook-Pro ~ %
```

The status bar at the bottom indicates the current line and column (Ln 6, Col 34), the number of spaces (4), the encoding (UTF-8), the line feed (LF), the Python version (3.10.7 64-bit), and the file type (Python).