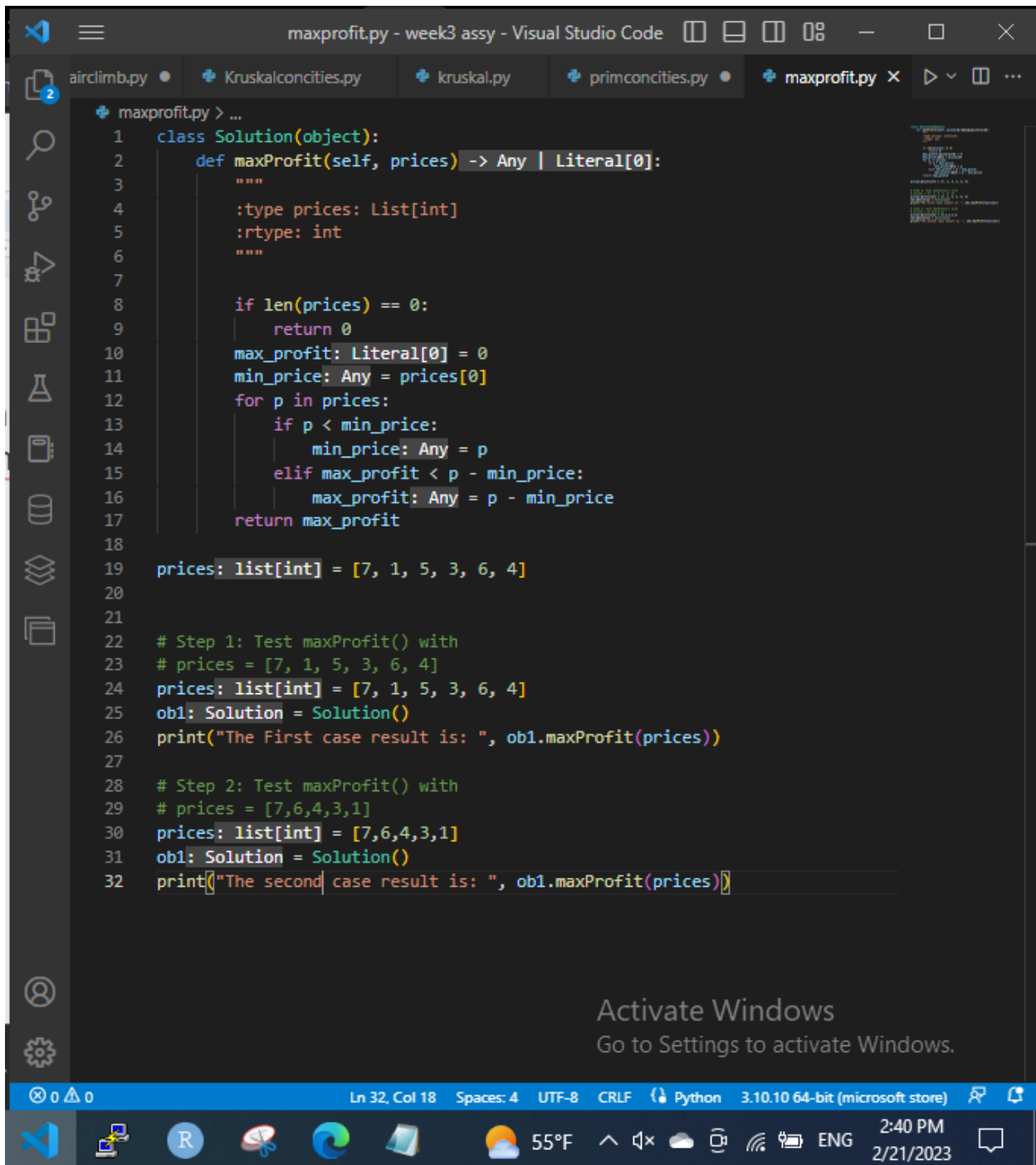


Week 5 Homework 3: Design Strategy: Greedy Algorithm / Dynamic Programming : Easy Problems - LC

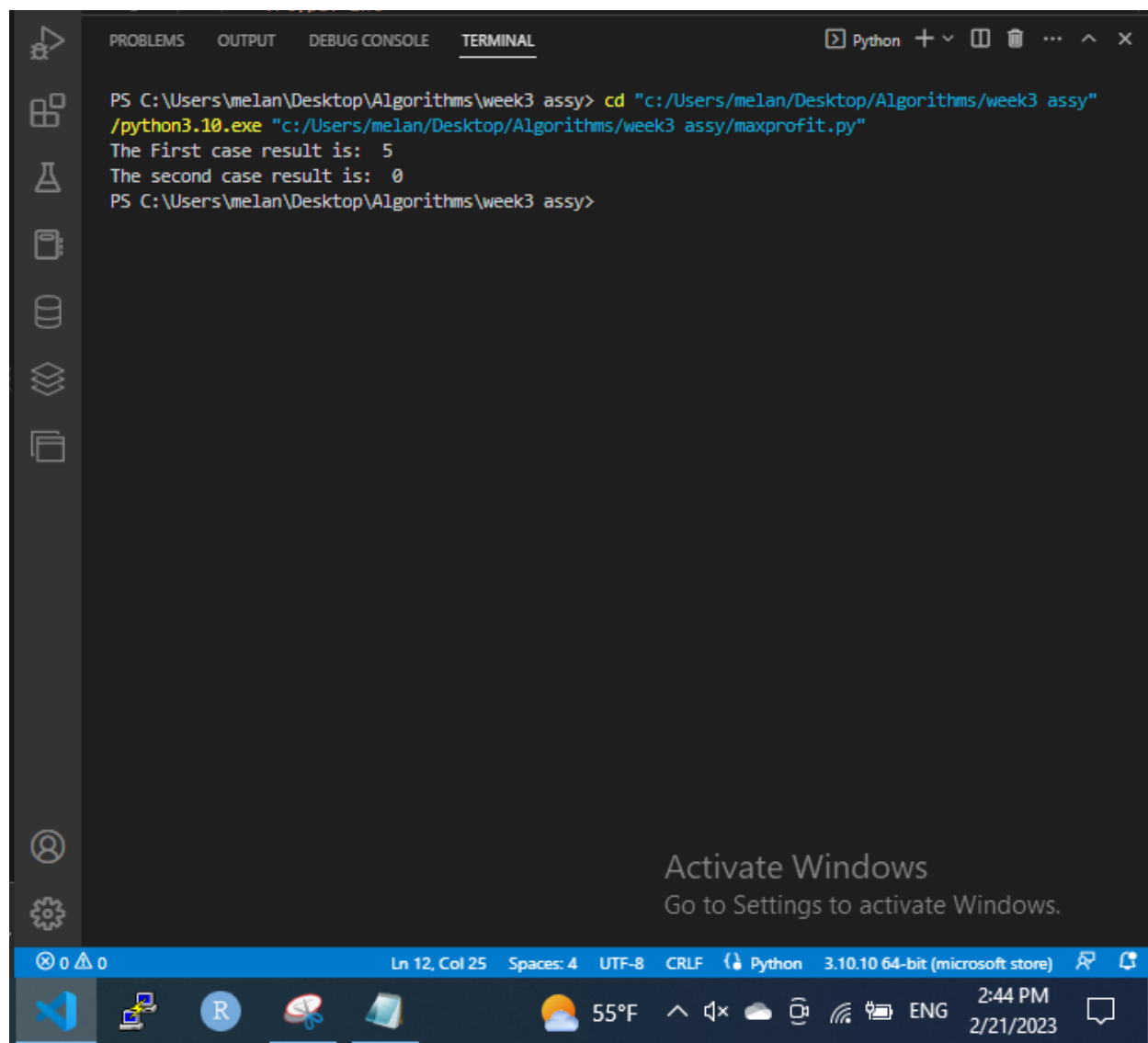


```
maxprofit.py - week3 assy - Visual Studio Code
airclimb.py • Kruskalconcities.py • kruskal.py • primconcities.py • maxprofit.py x
maxprofit.py > ...
1 class Solution(object):
2     def maxProfit(self, prices) -> Any | Literal[0]:
3         """
4         :type prices: List[int]
5         :rtype: int
6         """
7
8         if len(prices) == 0:
9             return 0
10        max_profit: Literal[0] = 0
11        min_price: Any = prices[0]
12        for p in prices:
13            if p < min_price:
14                min_price: Any = p
15            elif max_profit < p - min_price:
16                max_profit: Any = p - min_price
17        return max_profit
18
19 prices: list[int] = [7, 1, 5, 3, 6, 4]
20
21
22 # Step 1: Test maxProfit() with
23 # prices = [7, 1, 5, 3, 6, 4]
24 prices: list[int] = [7, 1, 5, 3, 6, 4]
25 ob1: Solution = Solution()
26 print("The First case result is: ", ob1.maxProfit(prices))
27
28 # Step 2: Test maxProfit() with
29 # prices = [7,6,4,3,1]
30 prices: list[int] = [7,6,4,3,1]
31 ob1: Solution = Solution()
32 print("The second case result is: ", ob1.maxProfit(prices))
```

Activate Windows
Go to Settings to activate Windows.

Ln 32, Col 18 Spaces: 4 UTF-8 CRLF Python 3.10.10 64-bit (microsoft store)

2:40 PM
2/21/2023



The image shows a Visual Studio Code interface with a terminal window open. The terminal displays the execution of a Python script. The command prompt is 'PS C:\Users\melan\Desktop\Algorithms\week3 assy>'. The user enters 'cd "c:/Users/melan/Desktop/Algorithms/week3 assy"' followed by '/python3.10.exe "c:/Users/melan/Desktop/Algorithms/week3 assy/maxprofit.py"'. The output shows 'The First case result is: 5' and 'The second case result is: 0'. The terminal window has a title bar with 'Python' and various icons. The status bar at the bottom shows 'Ln 12, Col 25', 'Spaces: 4', 'UTF-8', 'CRLF', 'Python', '3.10.10 64-bit (microsoft store)', and a system tray with weather, time, and language settings.

```
PS C:\Users\melan\Desktop\Algorithms\week3 assy> cd "c:/Users/melan/Desktop/Algorithms/week3 assy"
/python3.10.exe "c:/Users/melan/Desktop/Algorithms/week3 assy/maxprofit.py"
The First case result is: 5
The second case result is: 0
PS C:\Users\melan\Desktop\Algorithms\week3 assy>
```

Activate Windows
Go to Settings to activate Windows.

Ln 12, Col 25 Spaces: 4 UTF-8 CRLF Python 3.10.10 64-bit (microsoft store)

55°F 2:44 PM 2/21/2023 ENG

CODE

```
class Solution(object):
    def maxProfit(self, prices):
        """
        :type prices: List[int]
        :rtype: int
        """

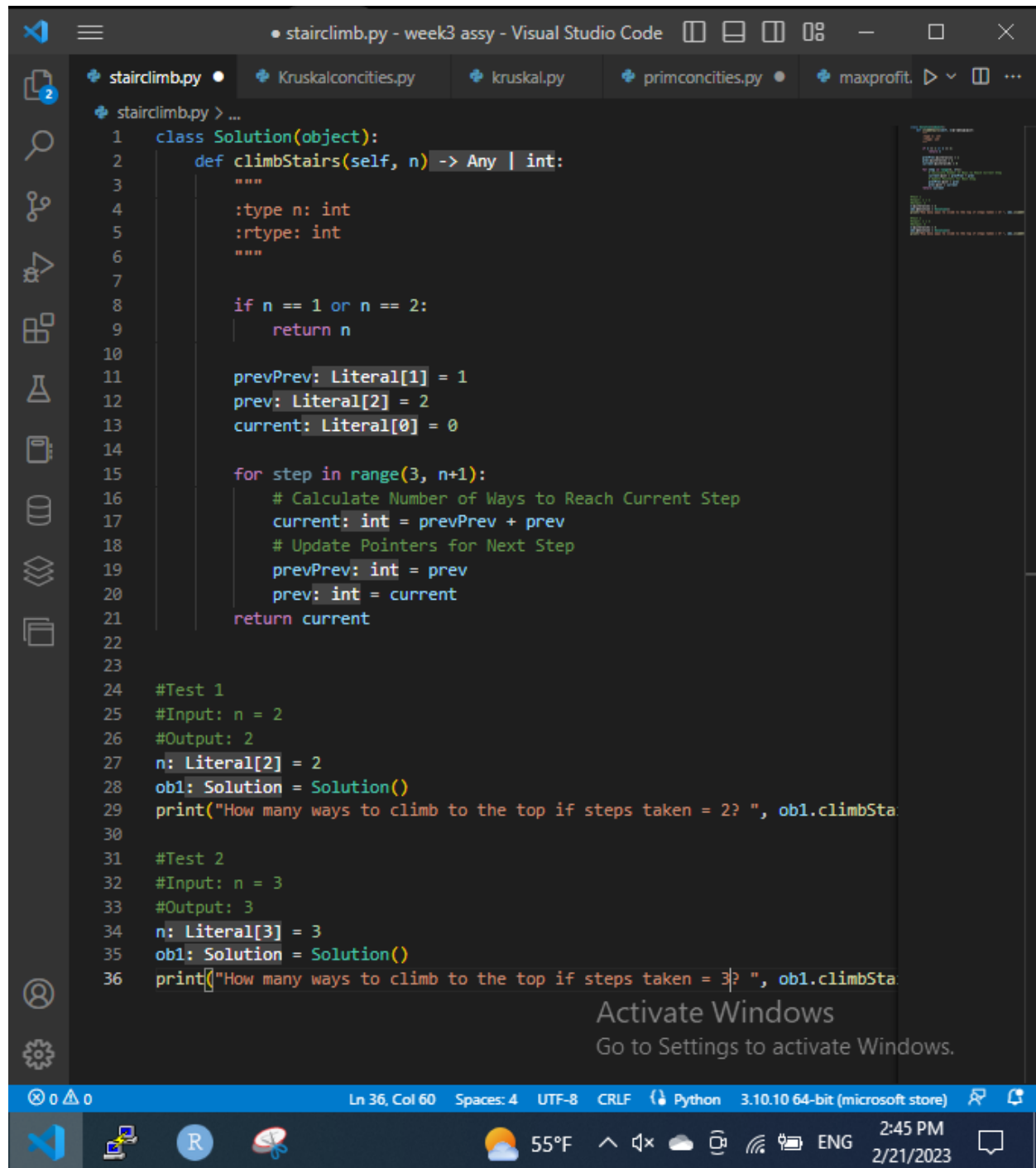
        if len(prices) == 0:
            return 0
        max_profit = 0
        min_price = prices[0]
        for p in prices:
            if p < min_price:
                min_price = p
            elif max_profit < p - min_price:
                max_profit = p - min_price
        return max_profit
```

```
# Step 1: Test maxProfit() with
# prices = [7, 1, 5, 3, 6, 4]
```

```
prices = [7, 1, 5, 3, 6, 4]
ob1 = Solution()
print("The First case result is: ",
ob1.maxProfit(prices))
```

```
# Step 2: Test maxProfit() with
# prices = [7,6,4,3,1]
prices = [7,6,4,3,1]
ob1 = Solution()
print("The second case result is: ",
ob1.maxProfit(prices))
```

OPTIONAL

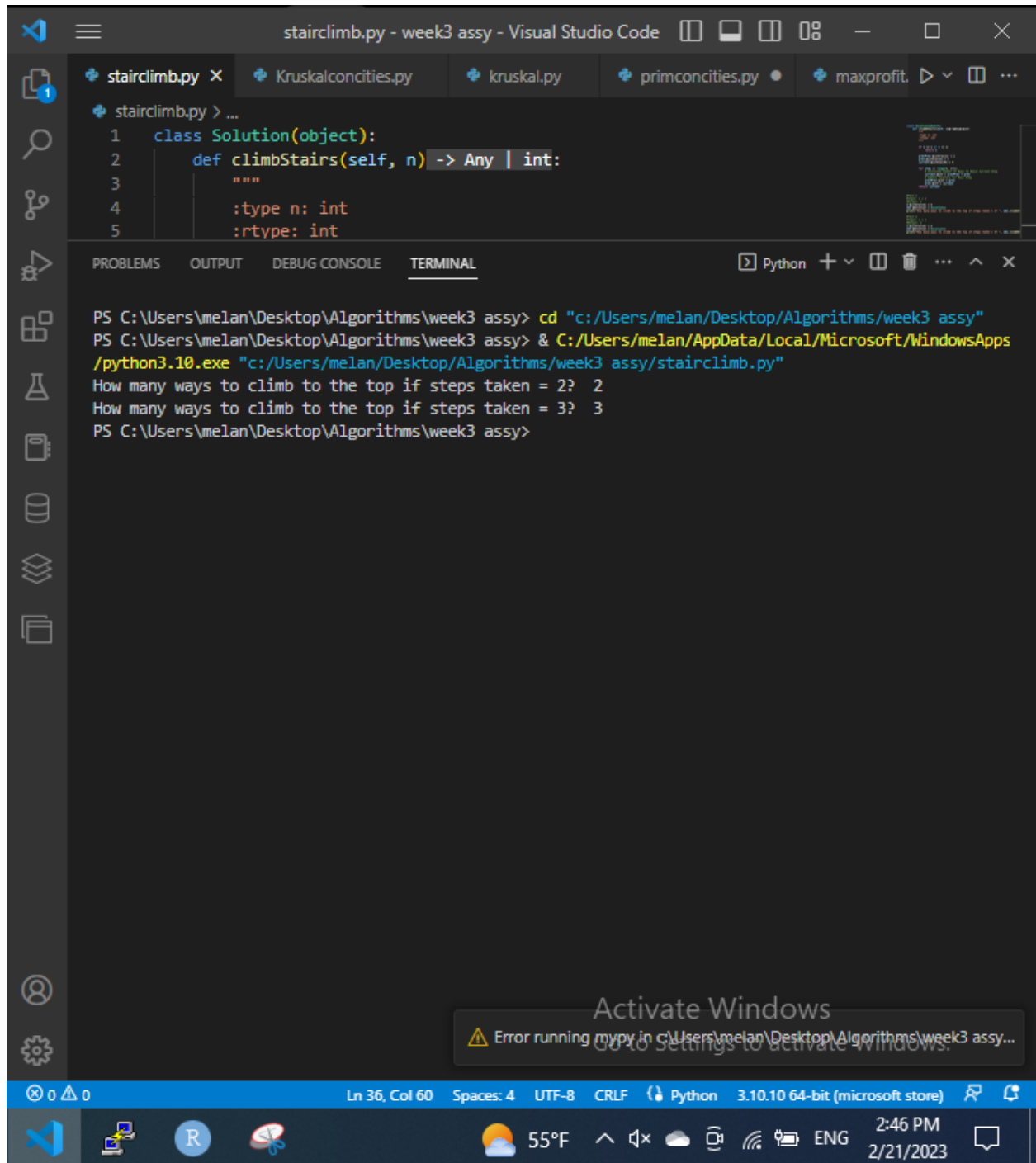


```
1 class Solution(object):
2     def climbStairs(self, n) -> Any | int:
3         """
4         :type n: int
5         :rtype: int
6         """
7
8         if n == 1 or n == 2:
9             return n
10
11         prevPrev: Literal[1] = 1
12         prev: Literal[2] = 2
13         current: Literal[0] = 0
14
15         for step in range(3, n+1):
16             # Calculate Number of Ways to Reach Current Step
17             current: int = prevPrev + prev
18             # Update Pointers for Next Step
19             prevPrev: int = prev
20             prev: int = current
21         return current
22
23
24 #Test 1
25 #Input: n = 2
26 #Output: 2
27 n: Literal[2] = 2
28 ob1: Solution = Solution()
29 print("How many ways to climb to the top if steps taken = 2? ", ob1.climbSta
30
31 #Test 2
32 #Input: n = 3
33 #Output: 3
34 n: Literal[3] = 3
35 ob1: Solution = Solution()
36 print("How many ways to climb to the top if steps taken = 3? ", ob1.climbSta
```

Activate Windows
Go to Settings to activate Windows.

Ln 36, Col 60 Spaces: 4 UTF-8 CRLF Python 3.10.10 64-bit (microsoft store)

2:45 PM
2/21/2023



CODE

```
class Solution(object):
    def climbStairs(self, n):
        """
        :type n: int
        :rtype: int
        """

        if n == 1 or n == 2:
            return n

        prevPrev = 1
        prev = 2
        current = 0

        for step in range(3, n+1):
            # Calculate Number of Ways to Reach
            Current Step
            current = prevPrev + prev
            # Update Pointers for Next Step
            prevPrev = prev
            prev = current
        return current
```

#Test 1

#Input: n = 2

#Output: 2

n = 2

ob1 = Solution()

print("How many ways to climb to the top if steps
taken = 2? ", ob1.climbStairs(n))

#Test 2

#Input: n = 3

#Output: 3

n = 3

ob1 = Solution()

print("How many ways to climb to the top if steps
taken = 3? ", ob1.climbStairs(n))