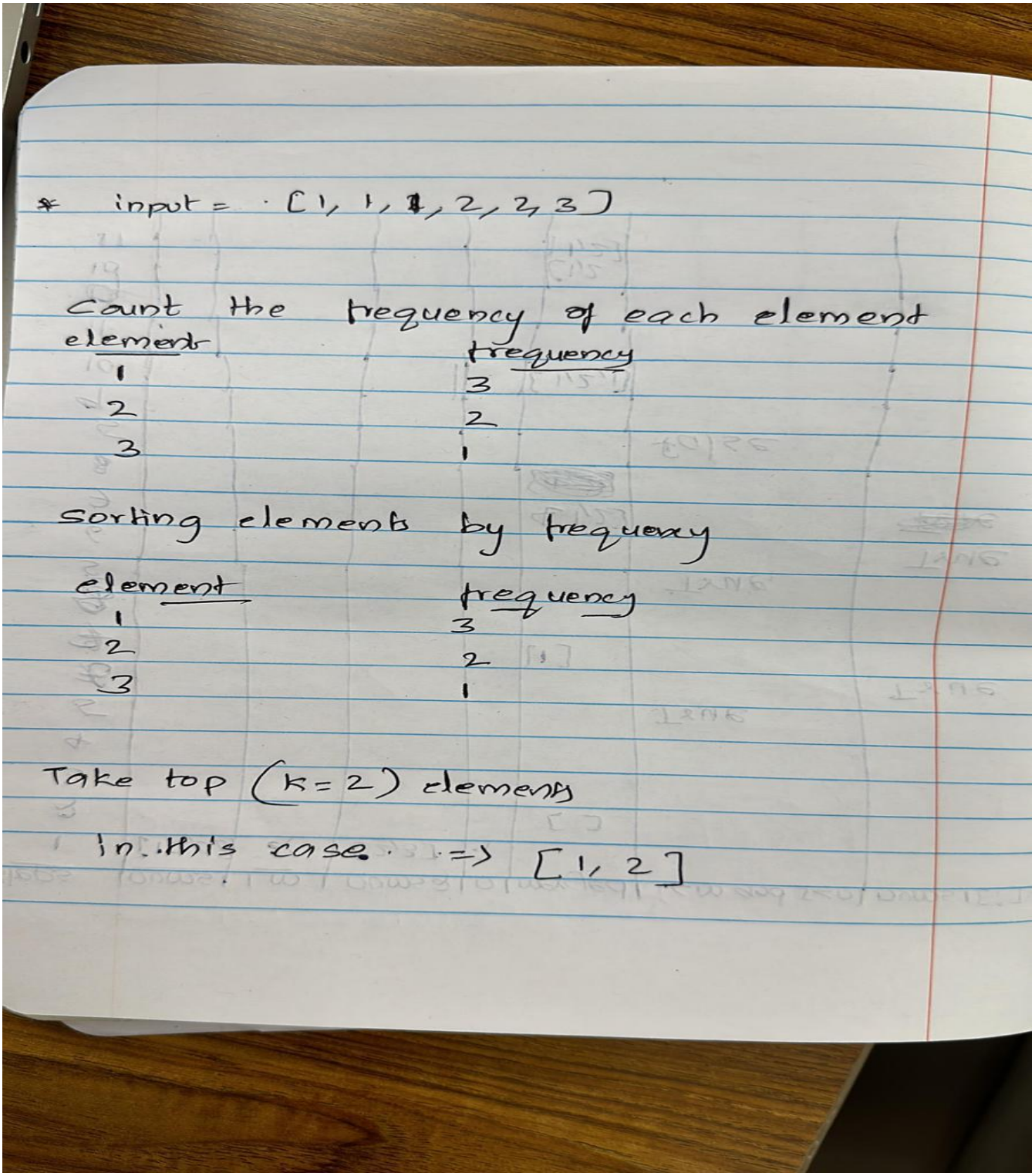
Week 9 hw1



* input = [1, 1, 1, 2, 2, 3]

Count the frequency of each element

element	frequency
1	3
2	2
3	1

Sorting elements by frequency

element	frequency
1	3
2	2
3	1

Take top (k = 2) elements

In this case => [1, 2]

Chatgpt python code

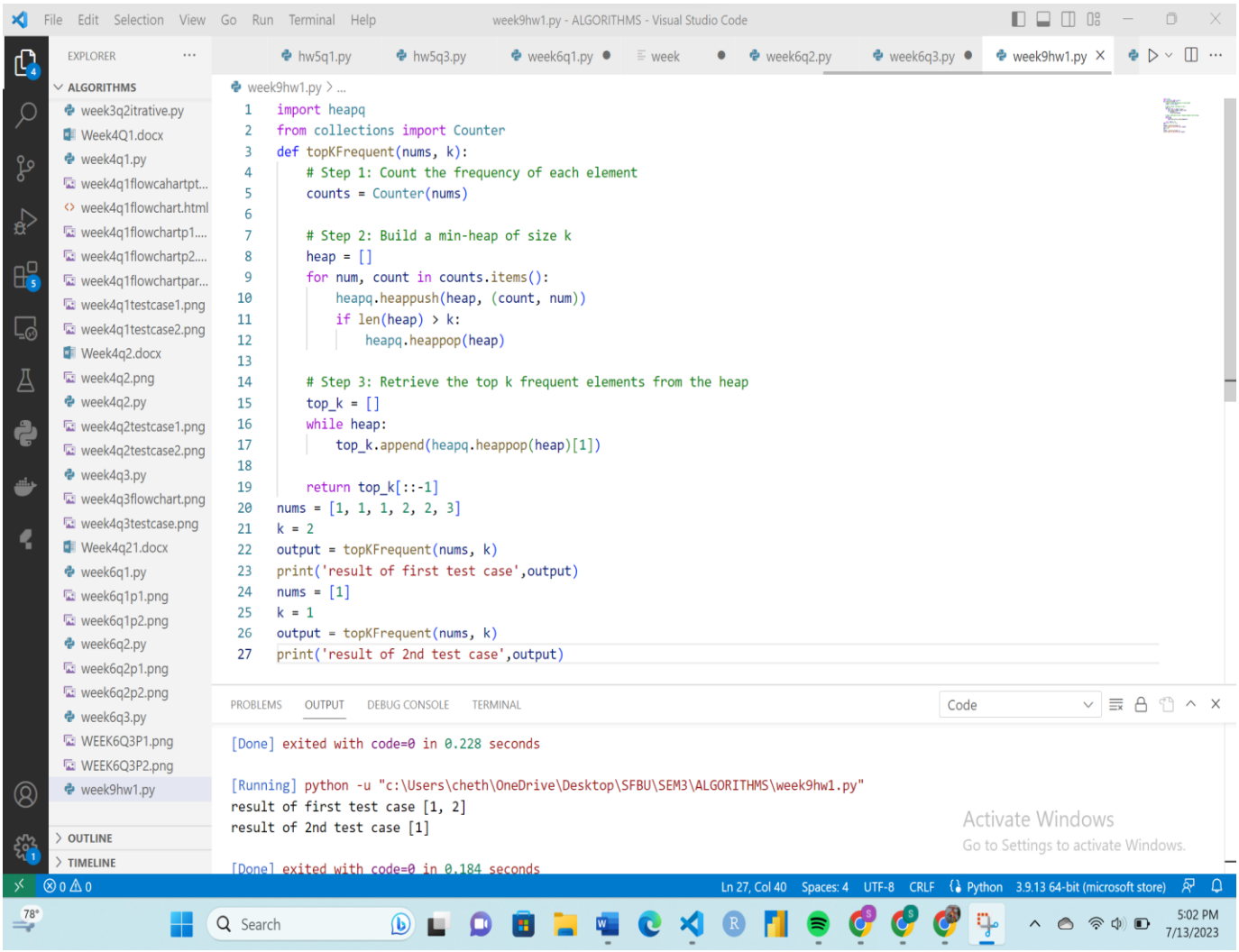
```python
import heapq
from collections import Counter
def topKFrequent(nums, k):
    # Step 1: Count the frequency of each element
    counts = Counter(nums)

    # Step 2: Build a min-heap of size k
    heap = []
    for num, count in counts.items():
        heapq.heappush(heap, (count, num))
        if len(heap) > k:
            heapq.heappop(heap)

    # Step 3: Retrieve the top k frequent elements from the heap
    top_k = []
    while heap:
        top_k.append(heapq.heappop(heap)[1])

    return top_k[::-1]
nums = [1, 1, 1, 2, 2, 3]
k = 2
output = topKFrequent(nums, k)
print(output)
```

Tests case


week9hw1.py - ALGORITHMS - Visual Studio Code
EXPLORER
ALGORITHMS
week3q2itrative.py
Week4Q1.docx
week4q1.py
week4q1flowcahartpt...
week4q1flowchart.html
week4q1flowchartp1....
week4q1flowchartp2....
week4q1flowchartpar...
week4q1testcase1.png
week4q1testcase2.png
Week4q2.docx
week4q2.png
week4q2.py
week4q2testcase1.png
week4q2testcase2.png
week4q3.py
week4q3flowchart.png
week4q3testcase.png
Week4q21.docx
week6q1.py
week6q1p1.png
week6q1p2.png
week6q2.py
week6q2p1.png
week6q2p2.png
week6q3.py
WEEK6Q3P1.png
WEEK6Q3P2.png
week9hw1.py
OUTLINE
TIMELINE
hw5q1.py
hw5q3.py
week6q1.py
week
week6q2.py
week6q3.py
week9hw1.py
week9hw1.py > ...
1 import heapq
2 from collections import Counter
3 def topKFrequent(nums, k):
4 # Step 1: Count the frequency of each element
5 counts = Counter(nums)
6
7 # Step 2: Build a min-heap of size k
8 heap = []
9 for num, count in counts.items():
10 heapq.heappush(heap, (count, num))
11 if len(heap) > k:
12 heapq.heappop(heap)
13
14 # Step 3: Retrieve the top k frequent elements from the heap
15 top_k = []
16 while heap:
17 top_k.append(heapq.heappop(heap)[1])
18
19 return top_k[::-1]
20 nums = [1, 1, 1, 2, 2, 3]
21 k = 2
22 output = topKFrequent(nums, k)
23 print('result of first test case',output)
24 nums = [1]
25 k = 1
26 output = topKFrequent(nums, k)
27 print('result of 2nd test case',output)
PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
Code
[Done] exited with code=0 in 0.228 seconds
[Running] python -u "c:\Users\cheth\OneDrive\Desktop\SFBU\SEM3\ALGORITHMS\week9hw1.py"
result of first test case [1, 2]
result of 2nd test case [1]
[Done] exited with code=0 in 0.184 seconds
Activate Windows
Go to Settings to activate Windows.
Ln 27, Col 40
Spaces: 4
UTF-8
CRLF
Python
3.9.13 64-bit (microsoft store)
Search
5:02 PM
7/13/2023