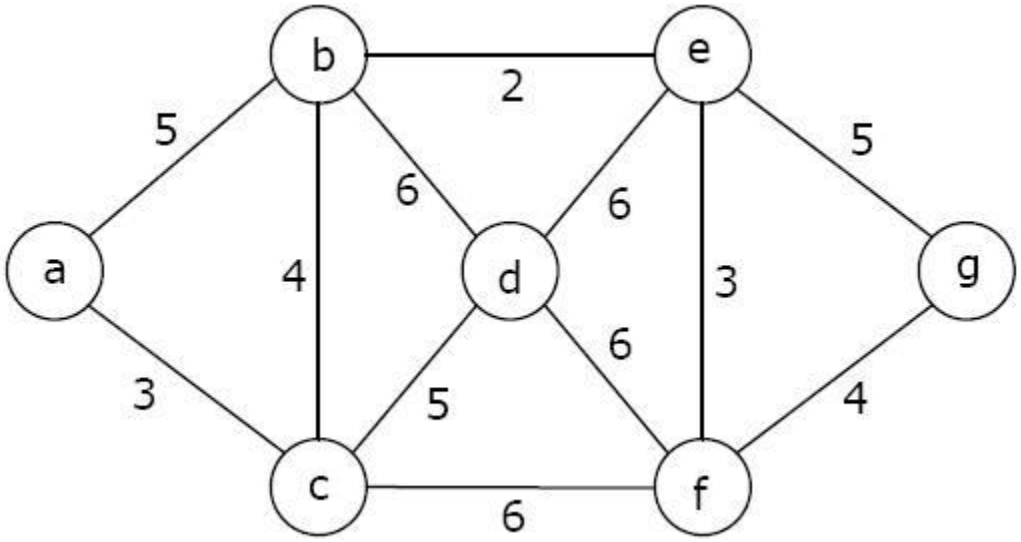
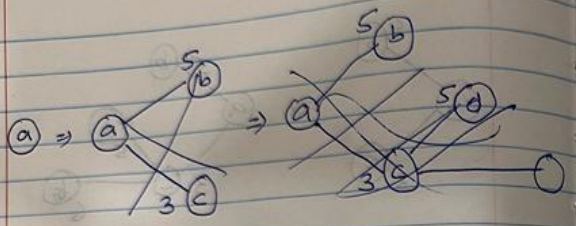
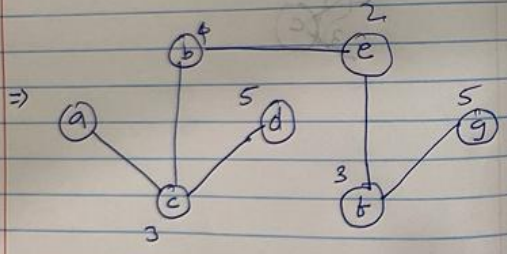
Hw5q1

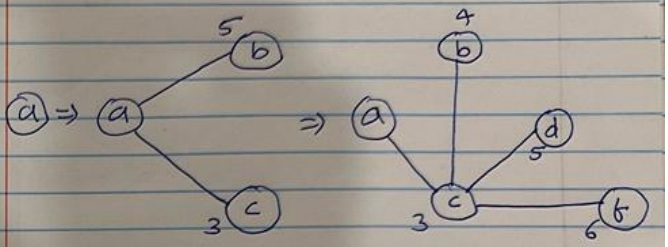


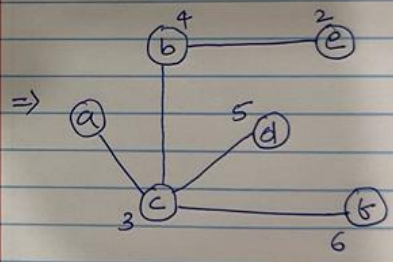


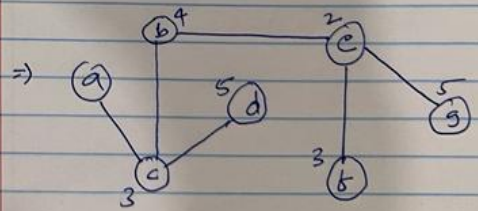


Total weight = 3+4+5+2+3+4

= 21

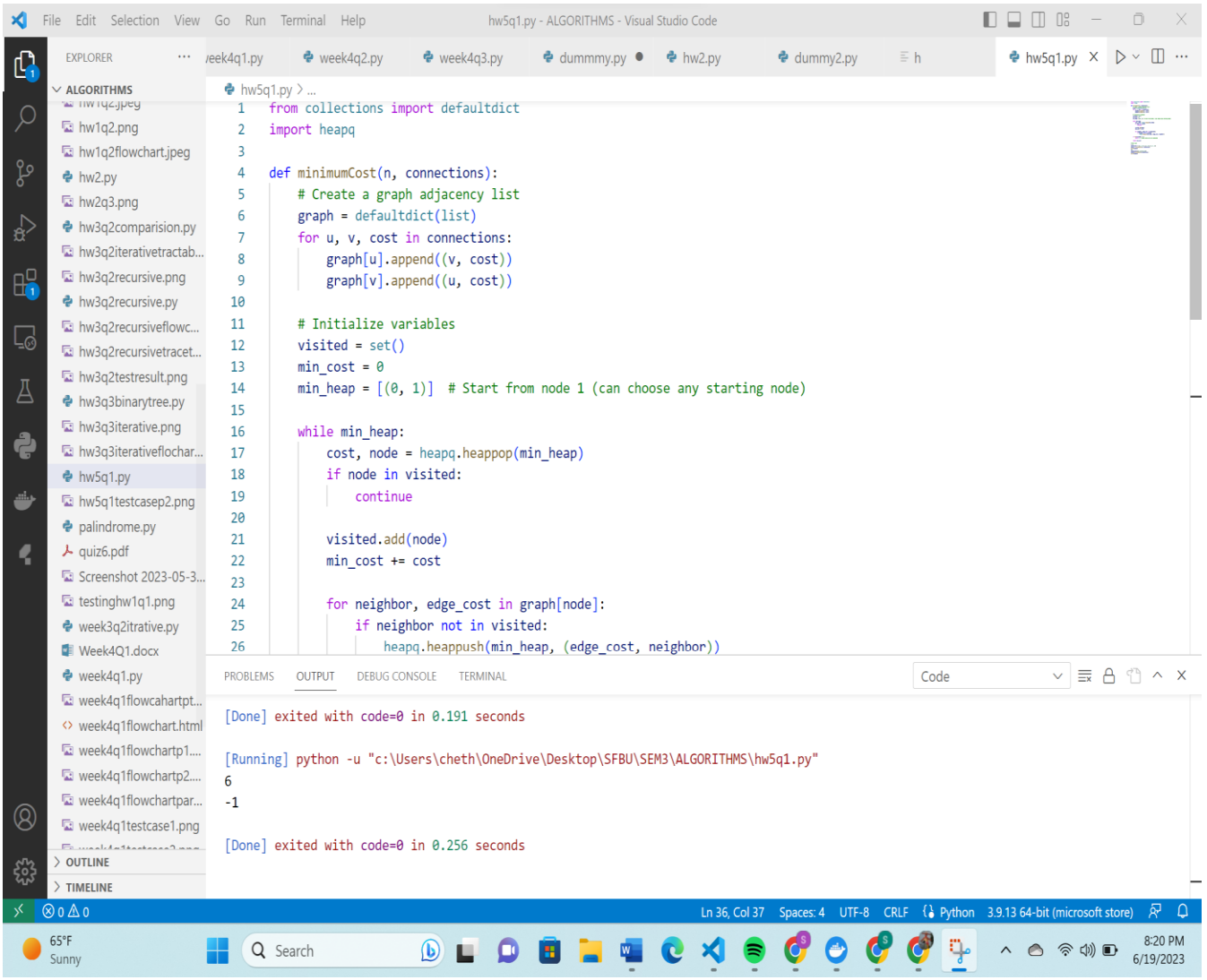


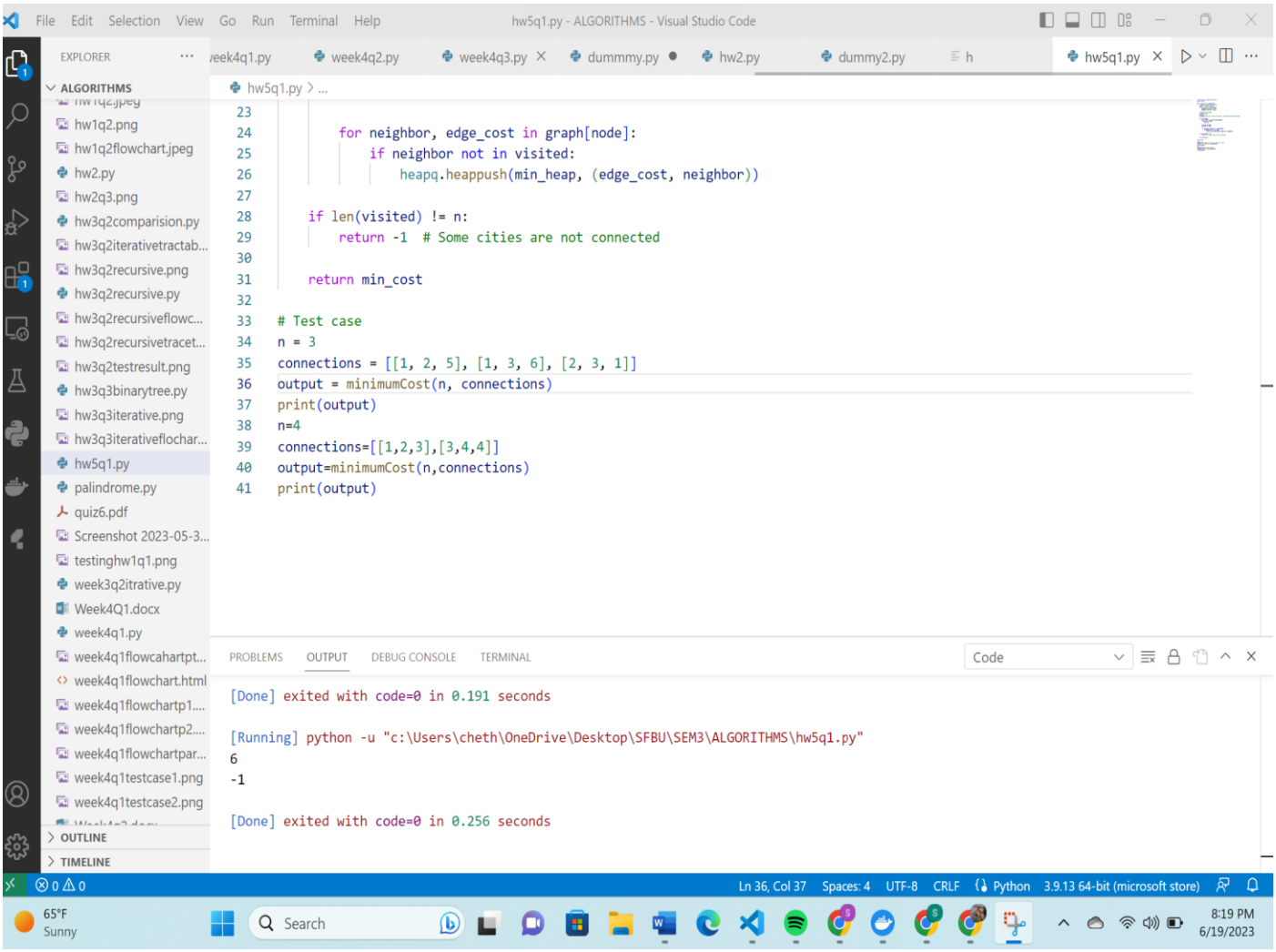




Python code

```python
from collections import defaultdict
import heapq

def minimumCost(n, connections):
    # Create a graph adjacency list
    graph = defaultdict(list)
    for u, v, cost in connections:
        graph[u].append((v, cost))
        graph[v].append((u, cost))

    # Initialize variables
    visited = set()
    min_cost = 0
    min_heap = [(0, 1)]  # Start from node 1 (can choose any starting node)

    while min_heap:
        cost, node = heapq.heappop(min_heap)
        if node in visited:
            continue

        visited.add(node)
        min_cost += cost

        for neighbor, edge_cost in graph[node]:
            if neighbor not in visited:
                heapq.heappush(min_heap, (edge_cost, neighbor))

    if len(visited) != n:
        return -1  # Some cities are not connected

    return min_cost

# Test case
n = 3
connections = [[1, 2, 5], [1, 3, 6], [2, 3, 1]]
output = minimumCost(n, connections)
print(output)
n=4
connections=[[1,2,3],[3,4,4]]
output=minimumCost(n,connections)
print(output)
```


File Edit Selection View Go Run Terminal Help
hw5q1.py - ALGORITHMS - Visual Studio Code
EXPLORER
ALGORITHMS
hw1q2.png
hw1q2flowchart.jpeg
hw2.py
hw2q3.png
hw3q2comparision.py
hw3q2iterativetractab...
hw3q2recursive.png
hw3q2recursive.py
hw3q2recursiveflowc...
hw3q2recursivetracet...
hw3q2testresult.png
hw3q3binarytree.py
hw3q3iterative.png
hw3q3iterativeflochar...
hw5q1.py
hw5q1testcasep2.png
palindrome.py
quiz6.pdf
Screenshot 2023-05-3...
testinghw1q1.png
week3q2itrative.py
Week4Q1.docx
week4q1.py
week4q1flowcahartpt...
week4q1flowchart.html
week4q1flowchartp1....
week4q1flowchartp2....
week4q1flowchartpar...
week4q1testcase1.png
OUTLINE
TIMELINE
week4q1.py
week4q2.py
week4q3.py
dummmy.py
hw2.py
dummy2.py
h
hw5q1.py
hw5q1.py > ...
1 from collections import defaultdict
2 import heapq
3
4 def minimumCost(n, connections):
5 # Create a graph adjacency list
6 graph = defaultdict(list)
7 for u, v, cost in connections:
8 graph[u].append((v, cost))
9 graph[v].append((u, cost))
10
11 # Initialize variables
12 visited = set()
13 min_cost = 0
14 min_heap = [(0, 1)] # Start from node 1 (can choose any starting node)
15
16 while min_heap:
17 cost, node = heapq.heappop(min_heap)
18 if node in visited:
19 continue
20
21 visited.add(node)
22 min_cost += cost
23
24 for neighbor, edge_cost in graph[node]:
25 if neighbor not in visited:
26 heapq.heappush(min_heap, (edge_cost, neighbor))
PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
Code
[Done] exited with code=0 in 0.191 seconds
[Running] python -u "c:\Users\cheth\OneDrive\Desktop\SFBU\SEM3\ALGORITHMS\hw5q1.py"
6
-1
[Done] exited with code=0 in 0.256 seconds
Ln 36, Col 37
Spaces: 4
UTF-8
CRLF
Python
3.9.13 64-bit (microsoft store)
65°F
Sunny
Search
8:20 PM
6/19/2023


File Edit Selection View Go Run Terminal Help
hw5q1.py - ALGORITHMS - Visual Studio Code
EXPLORER
week4q1.py
week4q2.py
week4q3.py
dummmy.py
hw2.py
dummy2.py
h
hw5q1.py
ALGORITHMS
hw1q2.png
hw1q2flowchart.jpeg
hw2.py
hw2q3.png
hw3q2comparision.py
hw3q2iterativetractab...
hw3q2recursive.png
hw3q2recursive.py
hw3q2recursiveflowc...
hw3q2recursivetracet...
hw3q2testresult.png
hw3q3binarytree.py
hw3q3iterative.png
hw3q3iterativeflochar...
hw5q1.py
palindrome.py
quiz6.pdf
Screenshot 2023-05-3...
testinghw1q1.png
week3q2itrative.py
Week4Q1.docx
week4q1.py
week4q1flowcahartpt...
week4q1flowchart.html
week4q1flowchartp1....
week4q1flowchartp2....
week4q1flowchartpar...
week4q1testcase1.png
week4q1testcase2.png
OUTLINE
TIMELINE
hw5q1.py > ...
for neighbor, edge_cost in graph[node]:
if neighbor not in visited:
heapq.heappush(min_heap, (edge_cost, neighbor))
if len(visited) != n:
return -1 # Some cities are not connected
return min_cost
# Test case
n = 3
connections = [[1, 2, 5], [1, 3, 6], [2, 3, 1]]
output = minimumCost(n, connections)
print(output)
n=4
connections=[[1,2,3],[3,4,4]]
output=minimumCost(n,connections)
print(output)
PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
Code
[Done] exited with code=0 in 0.191 seconds
[Running] python -u "c:\Users\cheth\OneDrive\Desktop\SFBU\SEM3\ALGORITHMS\hw5q1.py"
6
-1
[Done] exited with code=0 in 0.256 seconds
Ln 36, Col 37
Spaces: 4
UTF-8
CRLF
Python
3.9.13 64-bit (microsoft store)
65°F
Sunny
Search
8:19 PM
6/19/2023