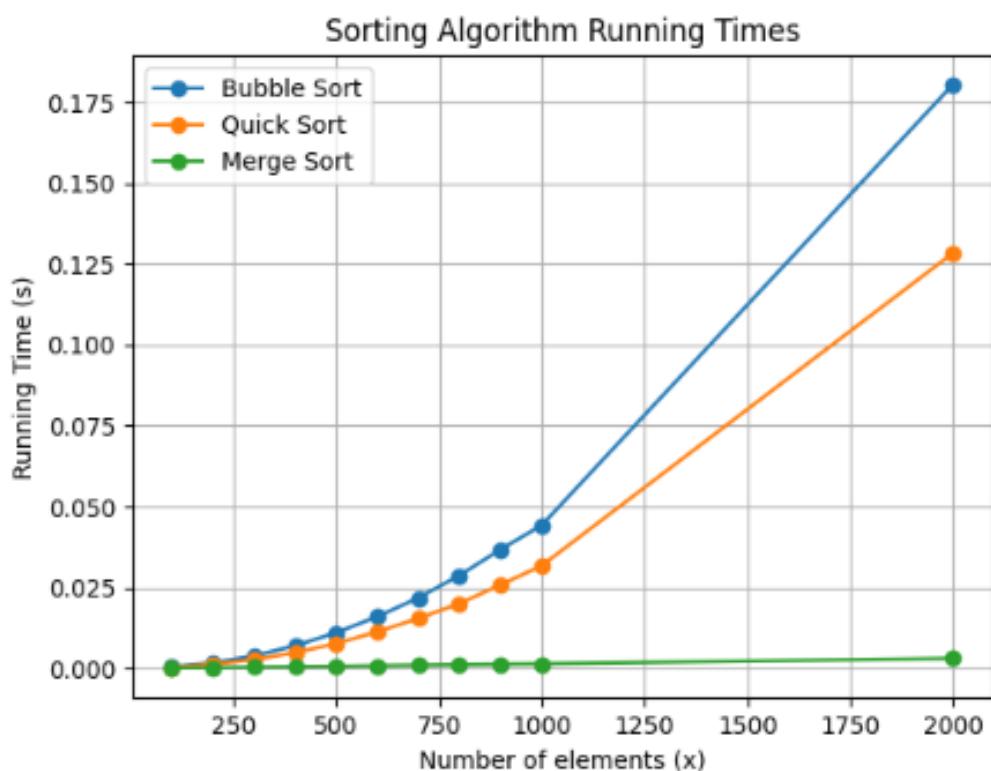


**PDF File DSA Assignment II**  
**Anil Vercruyse**  
**Student ID 22202474**

**Part I - Task III**



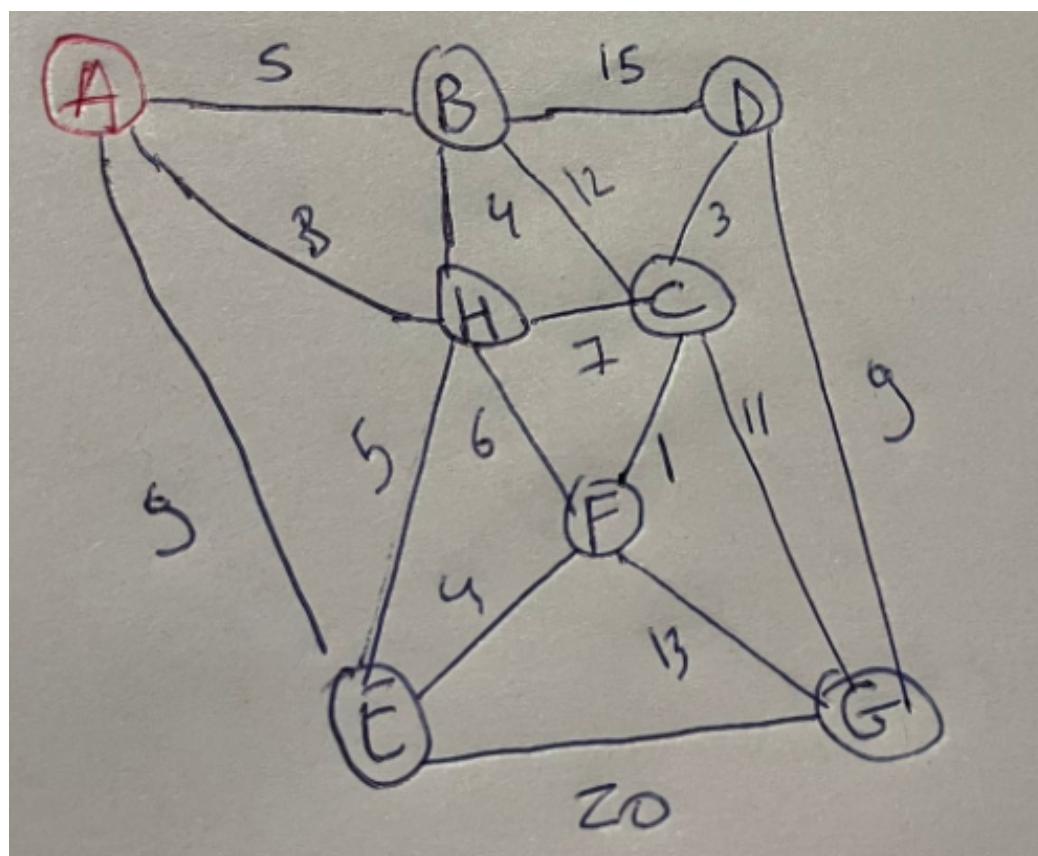
```
x: 2000
Average Bubble Sort time: 0.180353146666666788
Average Quick Sort time: 0.12830757333332865
Average Merge Sort time: 0.003090473333334861
```

## Part II - Tasks I to III (pen and paper)

This section shows the step-by-step evolution of the Part II weighted graph. Although the drawing style from lecture 16 was suggested, a slightly different approach was taken to ensure the readability of the required information. Kindly note that visited nodes are coloured in red.

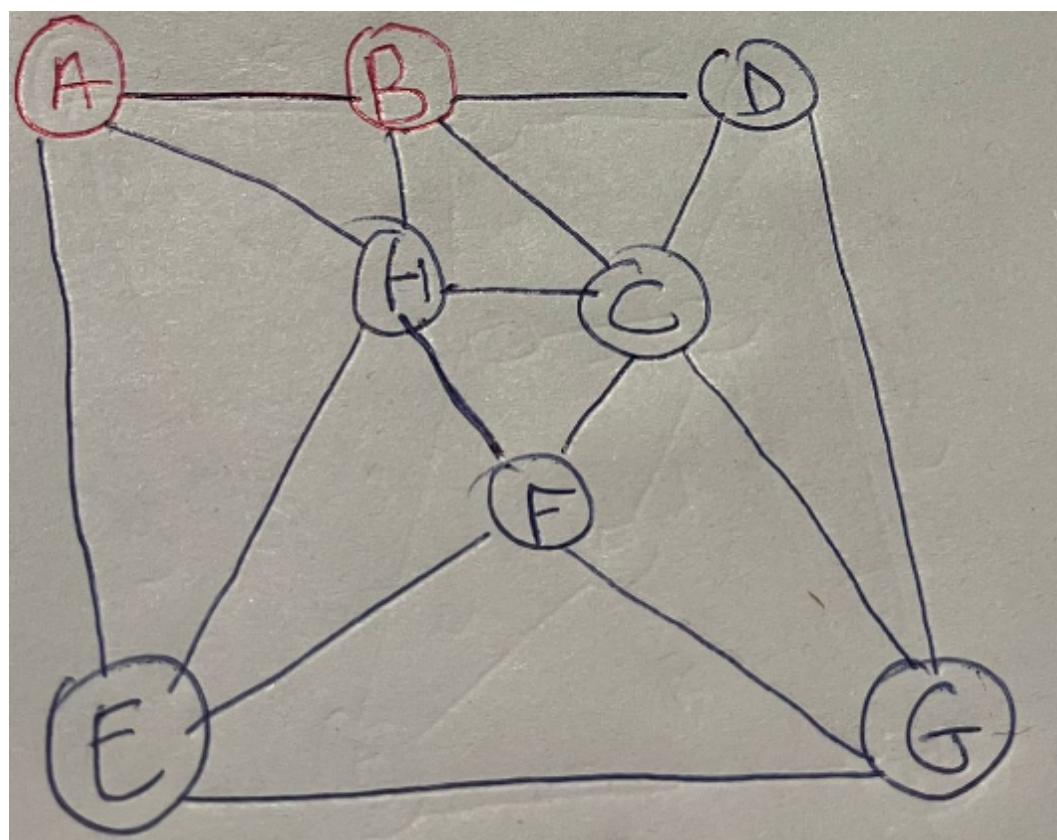
**Step 1:**

Node	Visited	Distance	Path
A	1	0	['A']
B	0	5	['A', 'B']
C	0	inf	[]
D	0	inf	[]
E	0	9	['A', 'E']
F	0	inf	[]
G	0	inf	[]
H	0	8	['A', 'H']



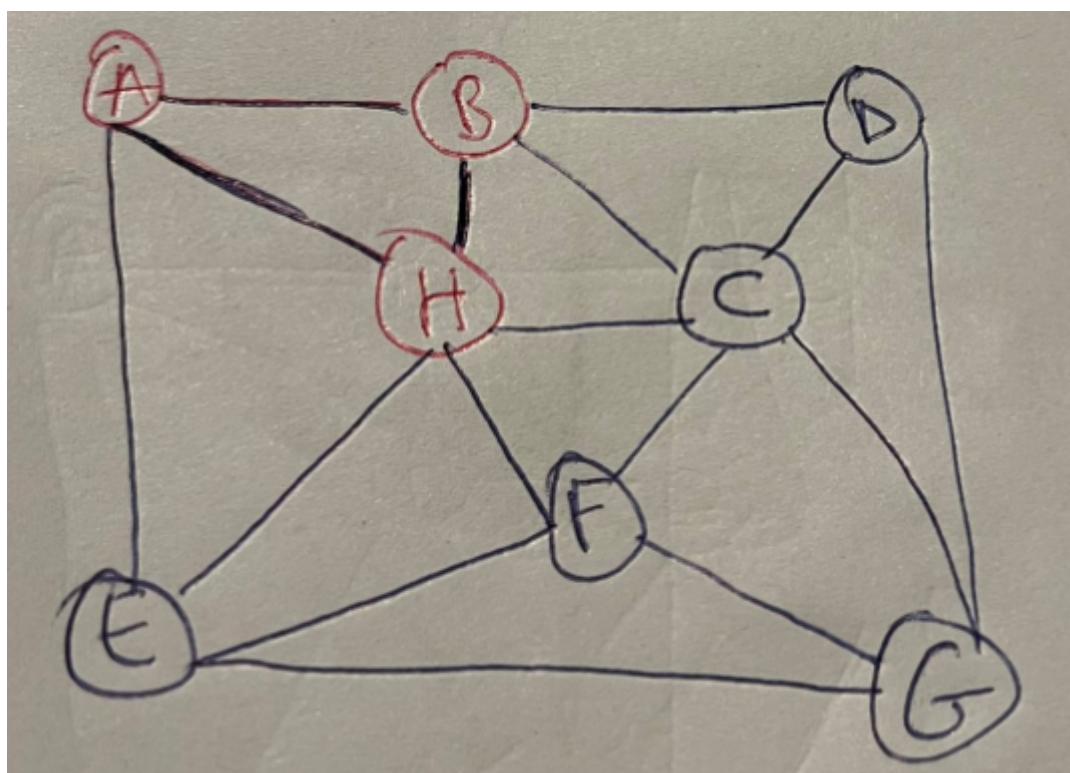
Step 2:

Node	Visited	Distance	Path
A	1	0	['A']
B	1	5	['A', 'B']
C	0	17	['A', 'B', 'C']
D	0	20	['A', 'B', 'D']
E	0	9	['A', 'E']
F	0	inf	[]
G	0	inf	[]
H	0	8	['A', 'H']



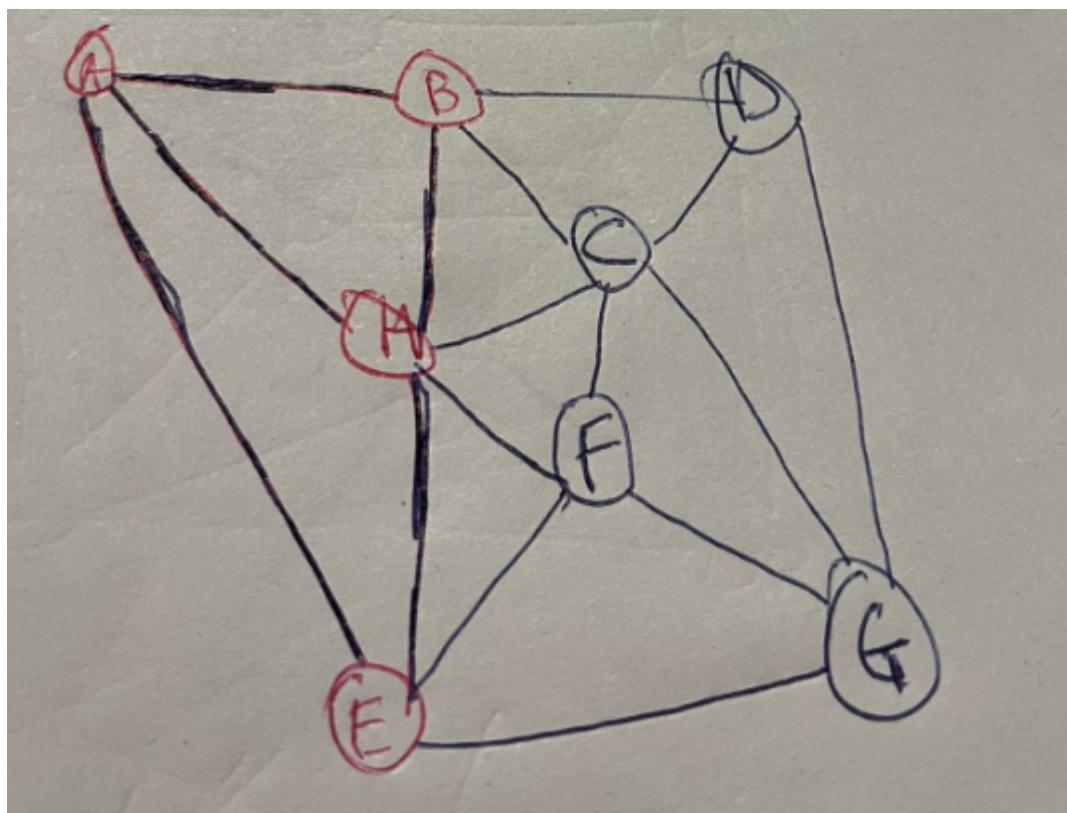
**Step 3:**

Node	Visited	Distance	Path
A	1	0	['A']
B	1	5	['A', 'B']
C	0	15	['A', 'H', 'C']
D	0	20	['A', 'B', 'D']
E	0	9	['A', 'E']
F	0	14	['A', 'H', 'F']
G	0	inf	[]
H	1	8	['A', 'H']



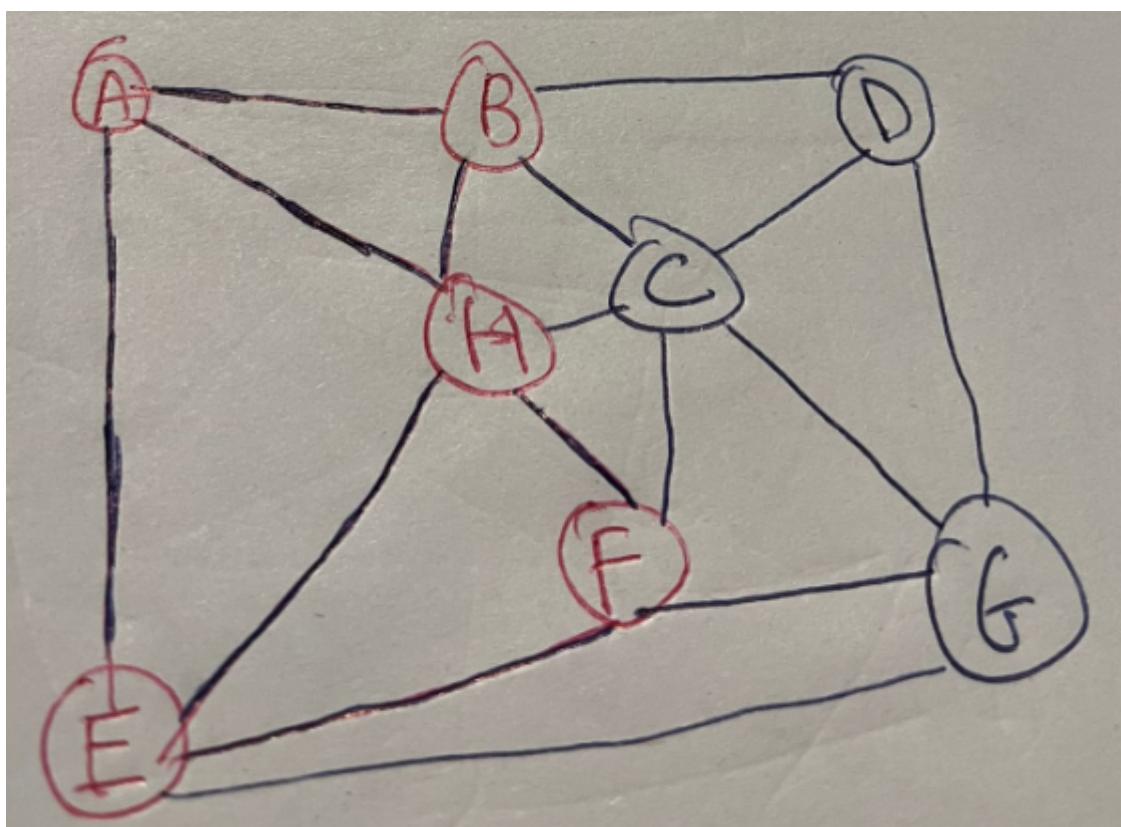
Step 4:

Node	Visited	Distance	Path
A	1	0	['A']
B	1	5	['A', 'B']
C	0	15	['A', 'H', 'C']
D	0	20	['A', 'B', 'D']
E	1	9	['A', 'E']
F	0	13	['A', 'E', 'F']
G	0	29	['A', 'E', 'G']
H	1	8	['A', 'H']



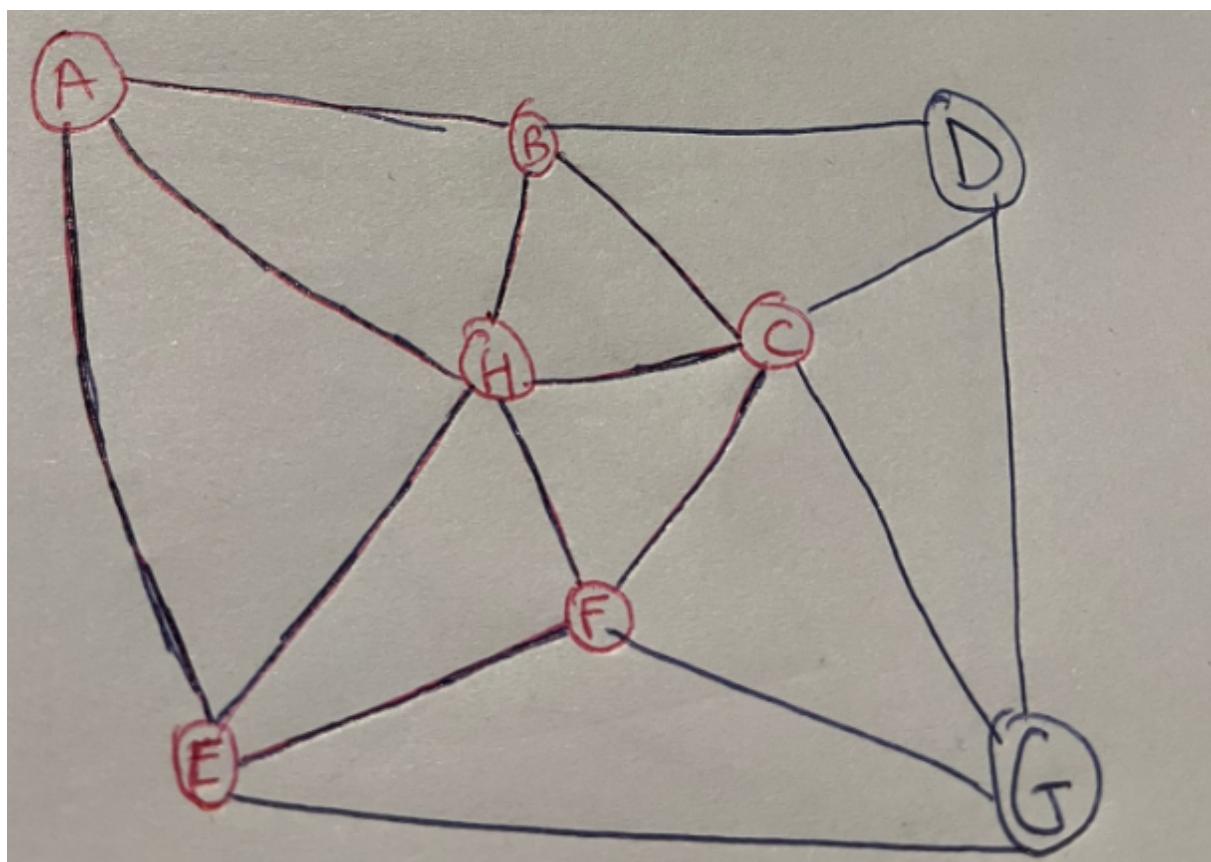
**Step 5:**

Node	Visited	Distance	Path
A	1	0	['A']
B	1	5	['A', 'B']
C	0	14	['A', 'E', 'F', 'C']
D	0	20	['A', 'B', 'D']
E	1	9	['A', 'E']
F	1	13	['A', 'E', 'F']
G	0	26	['A', 'E', 'F', 'G']
H	1	8	['A', 'H']



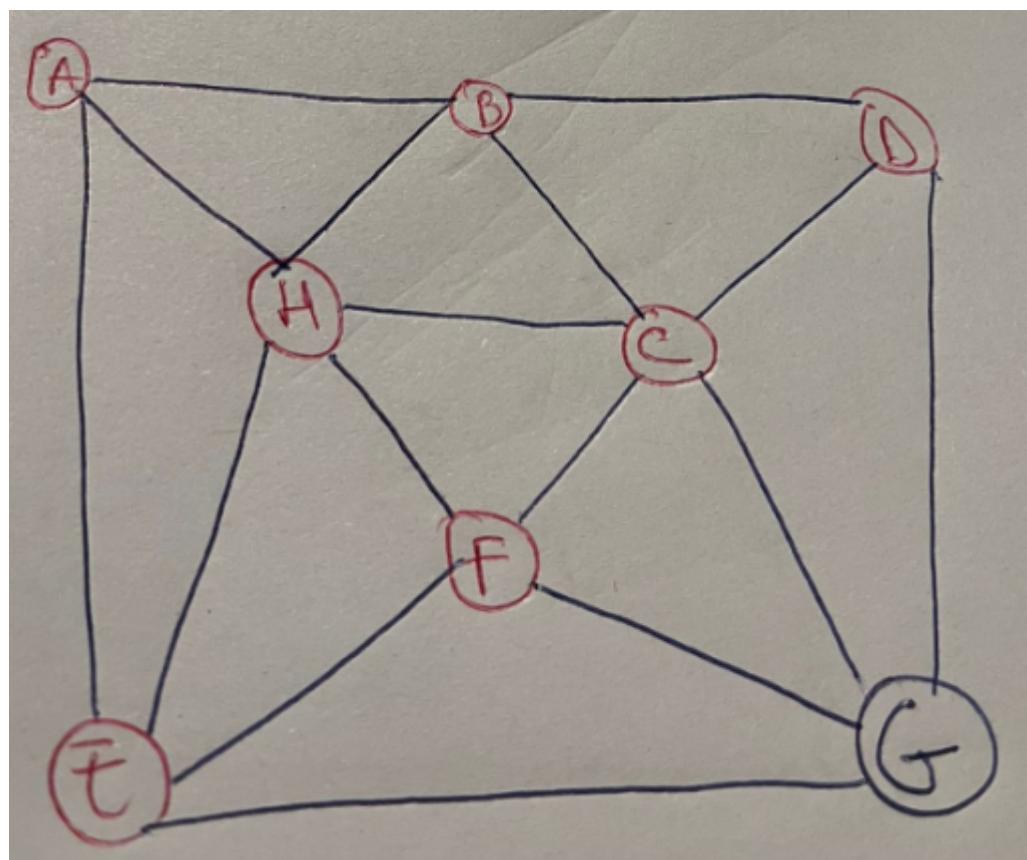
Step 6:

Node	Visited	Distance	Path
A	1	0	['A']
B	1	5	['A', 'B']
C	1	14	['A', 'E', 'F', 'C']
D	0	17	['A', 'E', 'F', 'C', 'D']
E	1	9	['A', 'E']
F	1	13	['A', 'E', 'F']
G	0	25	['A', 'E', 'F', 'C', 'G']
H	1	8	['A', 'H']



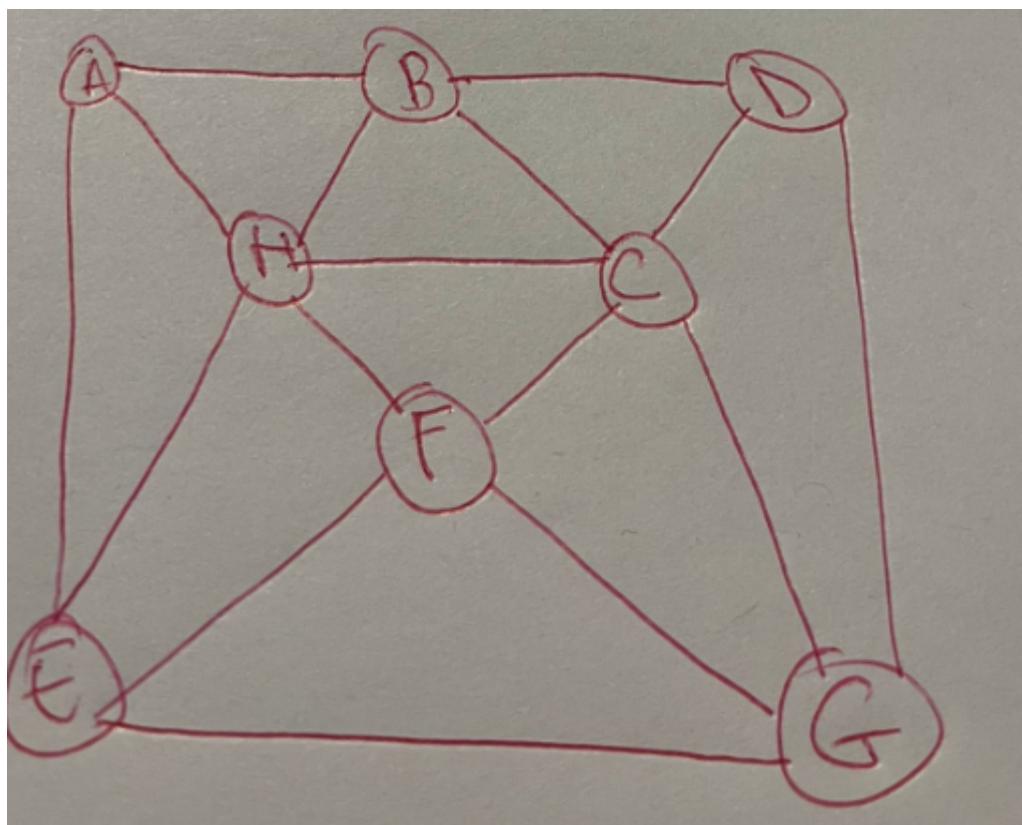
Step 7:

Node	Visited	Distance	Path
A	1	0	['A']
B	1	5	['A', 'B']
C	1	14	['A', 'E', 'F', 'C']
D	1	17	['A', 'E', 'F', 'C', 'D']
E	1	9	['A', 'E']
F	1	13	['A', 'E', 'F']
G	0	25	['A', 'E', 'F', 'C', 'G']
H	1	8	['A', 'H']



**Step 8:**

Node	Visited	Distance	Path
A	1	0	['A']
B	1	5	['A', 'B']
C	1	14	['A', 'E', 'F', 'C']
D	1	17	['A', 'E', 'F', 'C', 'D']
E	1	9	['A', 'E']
F	1	13	['A', 'E', 'F']
G	1	25	['A', 'E', 'F', 'C', 'G']
H	1	8	['A', 'H']



The shortest path from A to G is therefore: [A, E, F, C, G] = 25.

### Part III - Tasks I to III

This section shows the written work for part III, please refer to the table on the second page for the distances between each island.

**Step A:** add edge 1-8 with minimum cost 120. Edges in the tree after adding edge 1-8: [(1, 8, 120)].

**Step B:** add edge 8-2 with minimum cost 155. Edges in the tree after adding edge 8-2: [(1, 8, 120), (8, 2, 155)].

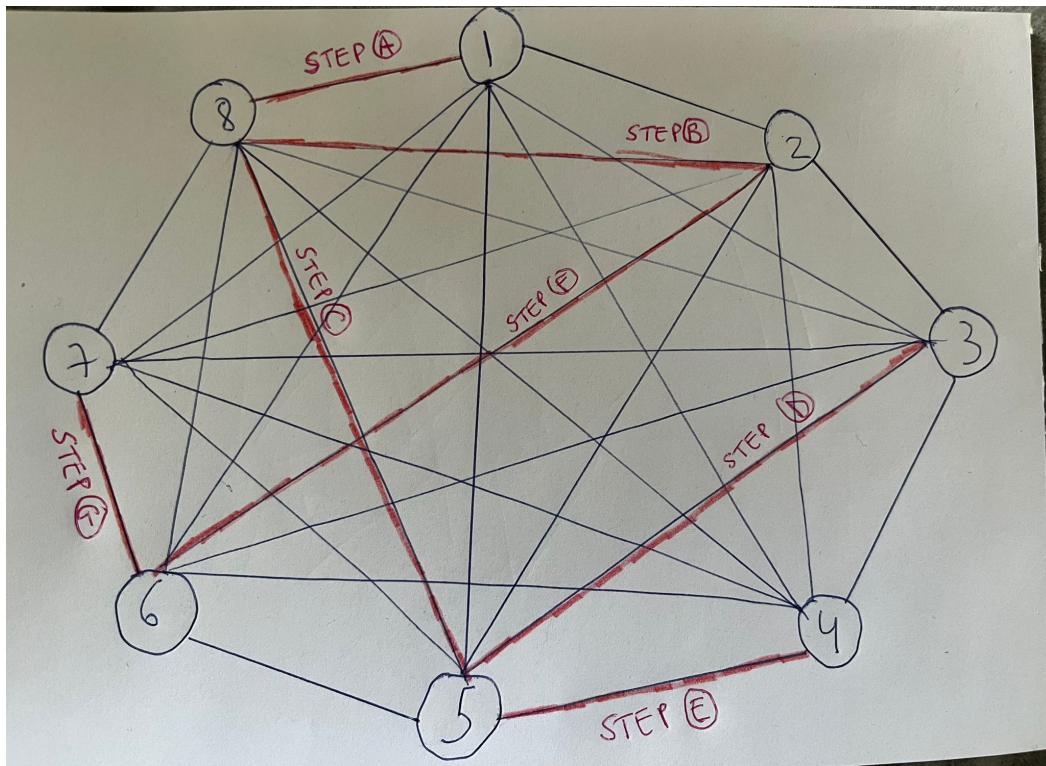
**Step C:** add edge 8-5 with minimum cost 170. Edges in the tree after adding edge 8-5: [(1, 8, 120), (8, 2, 155), (8, 5, 170)].

**Step D:** add edge 5-3 with minimum cost 115. Edges in the tree after adding edge 5-3: [(1, 8, 120), (8, 2, 155), (8, 5, 170), (5, 3, 115)].

**Step E:** add edge 5-4 with minimum cost 160. Edges in the tree after adding edge 5-4: [(1, 8, 120), (8, 2, 155), (8, 5, 170), (5, 3, 115), (5, 4, 160)].

**Step F:** add edge 2-6 with minimum cost 180. Edges in the tree after adding edge 2-6: [(1, 8, 120), (8, 2, 155), (8, 5, 170), (5, 3, 115), (5, 4, 160), (2, 6, 180)].

**Step G:** add edge 6-7 with minimum cost 175. Edges in the tree after adding edge 6-7: [(1, 8, 120), (8, 2, 155), (8, 5, 170), (5, 3, 115), (5, 4, 160), (2, 6, 180), (6, 7, 175)].



Based on the implementation of steps A to G, the Minimum Spanning Tree Edges are  $[(1, 8, 120), (8, 2, 155), (8, 5, 170), (5, 3, 115), (5, 4, 160), (2, 6, 180), (6, 7, 175)]$ , with a total construction cost of 1075.