Homework #3

- Program:
  - Input file: input.txt
    - Write a parser to read input.txt and construct a connected graph.
    - The input.txt contains all of the vertices' information, such as the number of vertices, their names, and their coordinates.
  - Output file: output.txt
    - Once the connected graph is constructed, the challenge lies in finding the minimum spanning tree.
    - The edge cost is determined by the Manhattan distance between two vertices. Transform each edge of the minimum spanning tree into an L-shape and add Steiner vertices.
    - Output.txt should include the names and coordinates of the vertices (Steiner points should be named S1, S2, ...) and the connections between points.

Homework #3

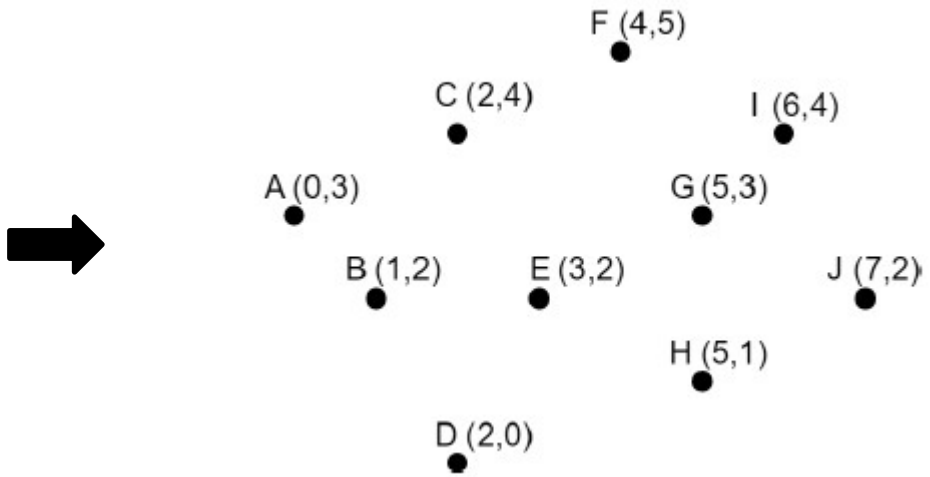
- The format for the **input.txt** should be as follows:

```
1   NumVertices: 10
2   A  0  3
3   B  1  2
4   C  2  4
5   D  2  0
6   E  3  2
7   F  4  5
8   G  5  3
9   H  5  1
10  I  6  4
11  J  7  2
```

The number of vertices

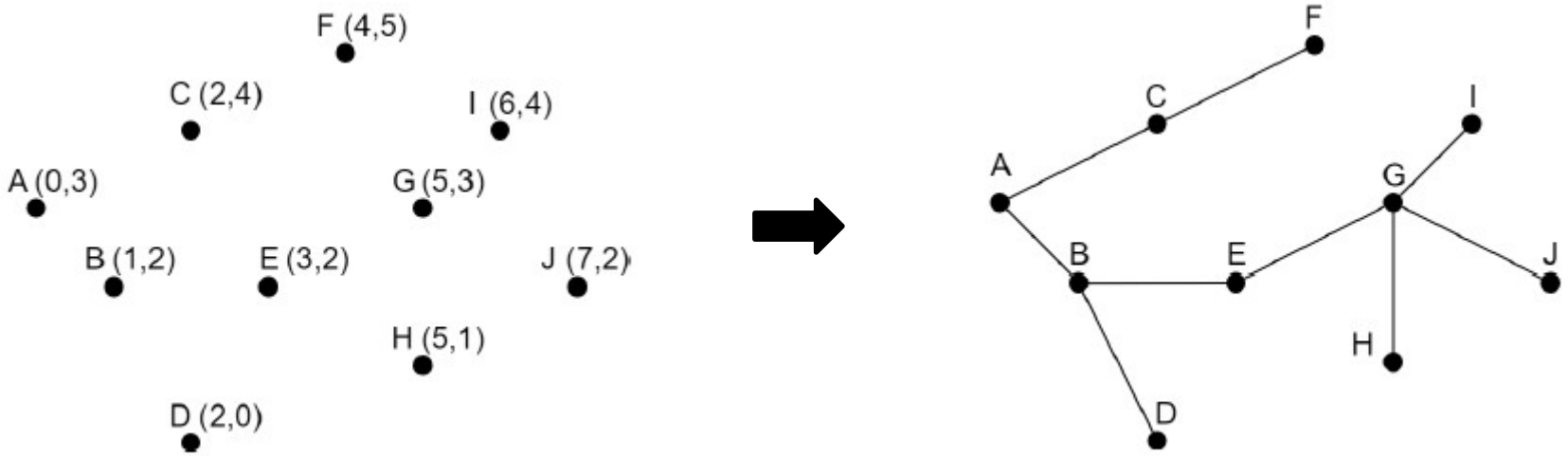
The coordinates of vertices (x,y)

The name of vertices



Homework #3

- Please create a connected graph based on the coordinates of the vertices and find the minimum spanning tree. Set the edge cost to be the Manhattan distance between two vertices.



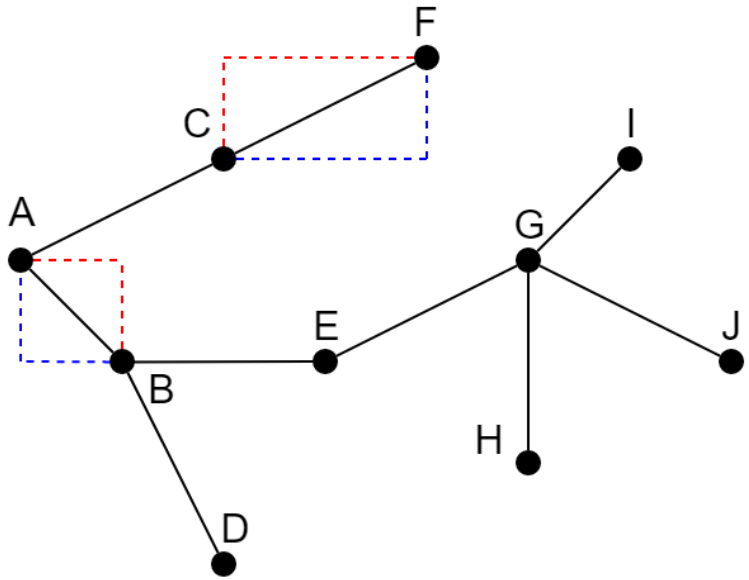
Homework #3

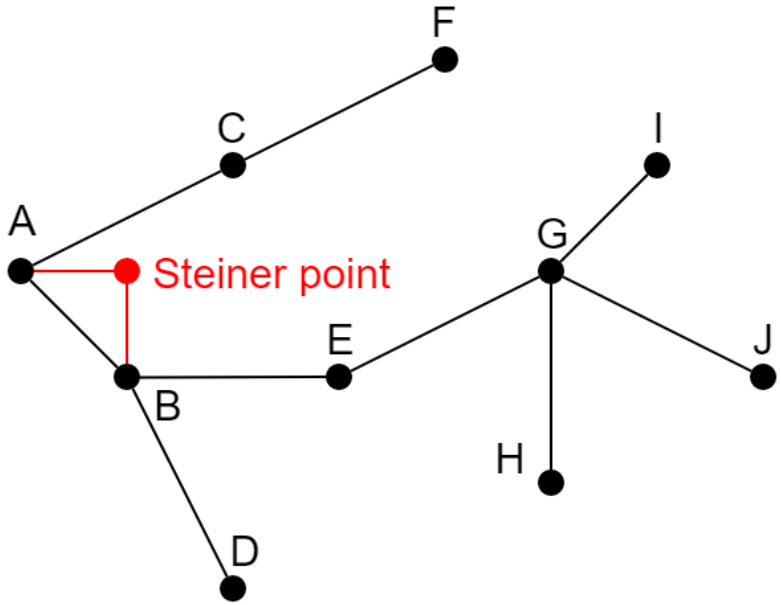
- Please use Prim's Algorithm to find the minimum spanning tree.

```
MST-PRIM(G, w, r)
 1  for each u ∈ G.V
 2      u.key = ∞
 3      u.π = NIL
 4  r.key = 0
 5  Q = G.V
 6  while Q ≠ ∅
 7      u = EXTRACT-MIN(Q)
 8      for each v ∈ G.Adj[u]
 9          if v ∈ Q and w(u, v) < v.key
10              v.π = u
11              v.key = w(u, v)
```

Homework #3

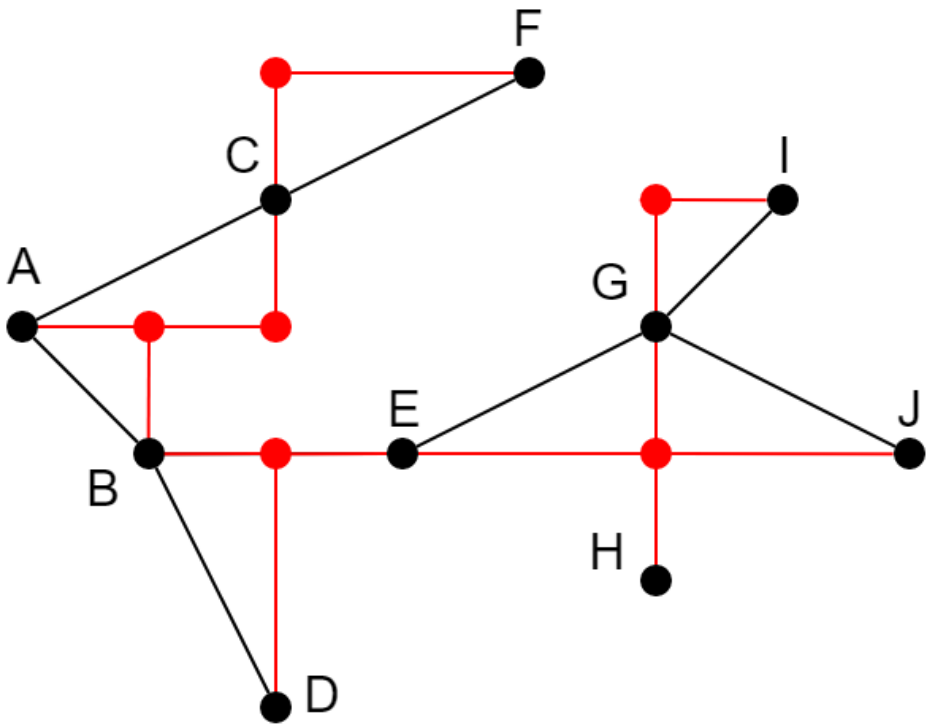
- Please convert the edges of the minimum spanning tree into L-shapes, either horizontal or vertical, as shown by the red or blue dashed lines in the diagram.
- You need to add Steiner points during the conversion process, as shown in the diagram.





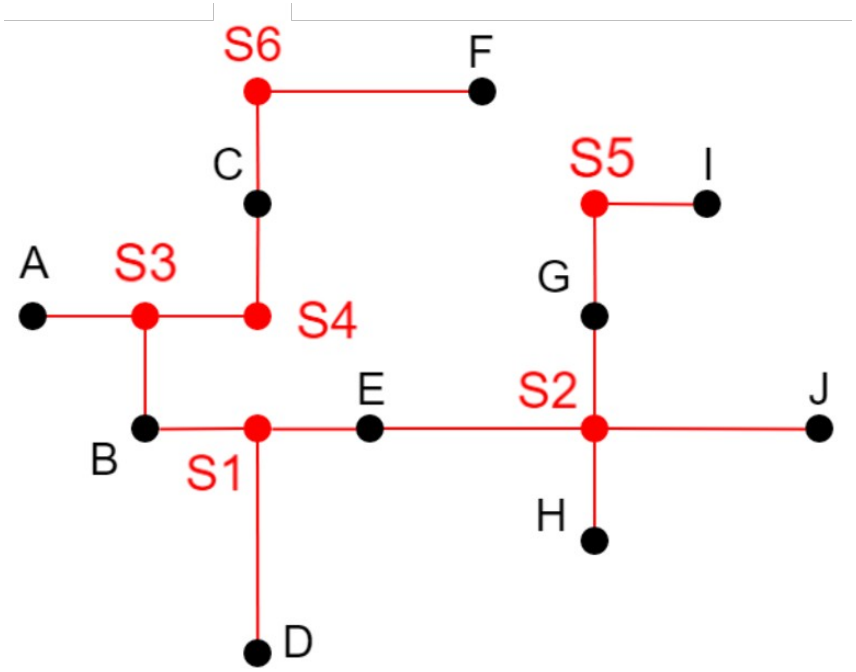
Homework #3

- After converting all edges of the minimum spanning tree, you will obtain the result shown in the diagram. The output file should only record the horizontal and vertical connections.



Homework #3

- The format for the **output.txt** should be as follows:
  - In output.txt, please keep the input vertices in the same order as in input.txt and place them above the Steiner points.
  - The order of the Steiner points doesn't matter. The only thing you have to do is keep the edge relations correct.




```
NumVertices: 10
A    0    3
B    1    2
C    2    4
D    2    0
E    3    2
F    4    5
G    5    3
H    5    1
I    6    4
J    7    2
S1    2    2
S2    5    2
S3    1    3
S4    2    3
S5    5    4
S6    2    5

NumEdges: 15
E1   B   S1
E2   G   S2
E3   S1   E
E4   S1   D
E5   S2   H
E6   S2   E
E7   S2   J
E8   A   S3
E9   S3   B
E10   S3   S4
E11   S4   C
E12   S5   G
E13   S5   I
E14   S6   C
E15   S6   F

WireLength: 19
```

Homework #3

- **Deadline: 2024/06/17 11:55 pm**

- Please use C++ language to implement your program.

- **TA email: m16121093@gs.ncku.edu.tw**
- If you have any questions, please get in touch with the TA email.
- The mail‘s title should be:
  - DS_HW3_your name_your student ID_your question in brief.

Homework #3

- Please upload your source code to Moodle.
  - e.g., HW3_E12345678.cpp
- Notes:
  - The specified algorithm must be used to construct the minimum spanning tree.
  - Steiner points should be represented as S1, S2, etc.
  - Edges of the Steiner tree should be represented as E1, E2, etc.
- The **WireLength** represents the total wirelength of the steiner tree.
- Scoring:
  - There are 3 test cases, 1 of them is public case and 2 of them are hidden cases.
  - **Higher scores are given for shorter total wirelength.**