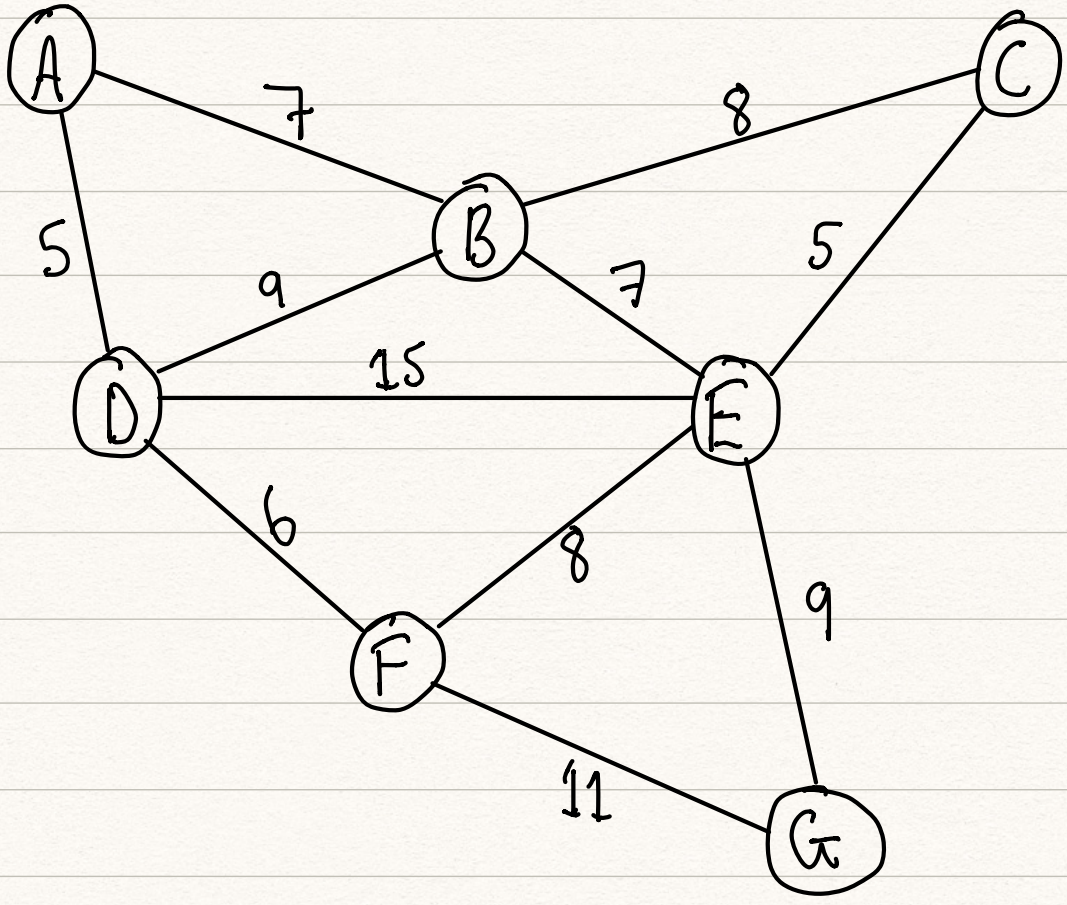
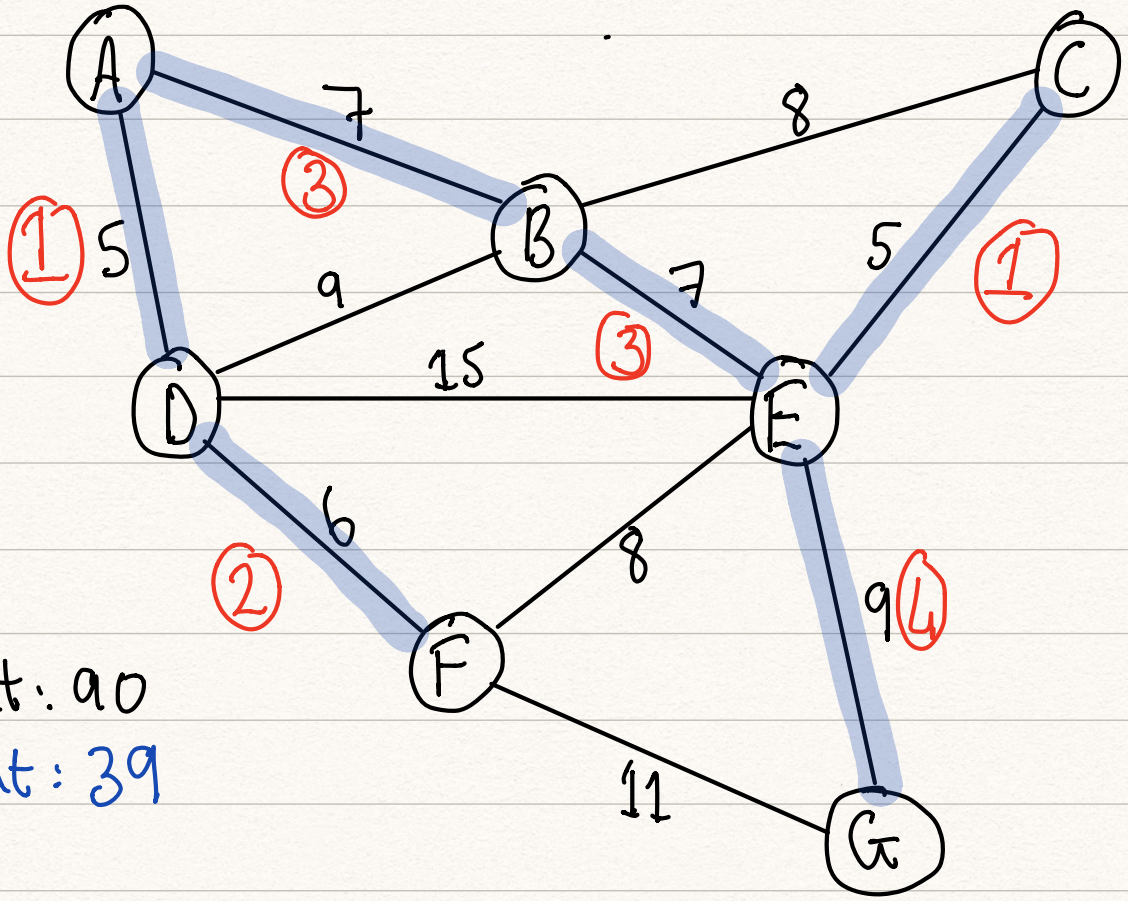
Given this following tree structure



Kruskal's Algorithm

Finds a safe edge to add to a growing forest by finding all the edges that connect any two trees in the forest an edge (u,v) of least weight. It uses a disjoint-set data structure where each vertex is initally in its own set. This is known as UnionFind in my implementation.

Minimum Weight + Union Find



Weight: 90

Weight: 39

Path Compression

1. [A D], [B], [C E], [F] [G]
2. [A D F], [B], [C E], [G]
3. [A D F C E B], [G]
4. [A D F C E B G]

Kruskal Pseudo Code

```
MST-KRUSKAL (G, w)
 MST = ∅
 for each vertex v in G.VL)
      MAKE-SET(v)
sort edges of G.EC) ascendingly by weight
for each edge (u,v) in G.EC) order by weight
     if (! connected (u,v))
        UNION (u,v)
        MST.insert(Edge)
return MST
```

Expected Output

A	->	D	5
C	->	E	5
D	->	F	6
A	->	B	7
B	->	E	7
E	->	G	9

39 total weight

39 < 90 therefore is minimum tree.

How to compile my code

```
Eamonn Keogh@DESKTOP-0A91N31 MINGW64 ~/eclipse-workspace/Algorithms-DS/src (master)
$ javac kruskal/*.java
Note: kruskal\Graph.java uses unchecked or unsafe operations.
Note: Recompile with -Xlint:unchecked for details.
```

There is a warning because java doesn't like casting generic arrays

How to run my Code (cmd line input)

```
Eamonn Keogh@DESKTOP-0A91N31 MINGW64 ~/eclipse-workspace/Algorithms-DS/src (master)
$ java kruskal.Main myGraph.txt
```

My code takes command line arguements myGraph.txt is the arguement

Program Output

```
Eamonn Keogh@DESKTOP-0A91N31 MINGW64 ~/eclipse-workspace/Algorithms-DS/src (master)
$ java kruskal.Main myGraph.txt
Vertices: 7 Edges: 11

Adjacency List: 90 weighting
A -> [B, 7] [D, 5]
B -> [C, 8] [D, 9] [E, 7]
C -> [E, 5]
D -> [E, 15] [F, 6]
E -> [F, 8] [G, 9]
F -> [G, 11]

Minimum Spanning Tree: 39 weighting
[D, 5]
[E, 5]
[F, 6]
[B, 7]
[E, 7]
[G, 9]


Eamonn Keogh@DESKTOP-0A91N31 MINGW64 ~/eclipse-workspace/Algorithms-DS/src (master)
$
```