

Competitive Algorithm Design and Practice

Minimum Spanning Tree

2018/05/02

國立成功大學ACM-ICPC程式競賽培訓隊

Department of Computer Science and Information Engineering
National Cheng Kung University
Tainan, Taiwan



Spanning Tree

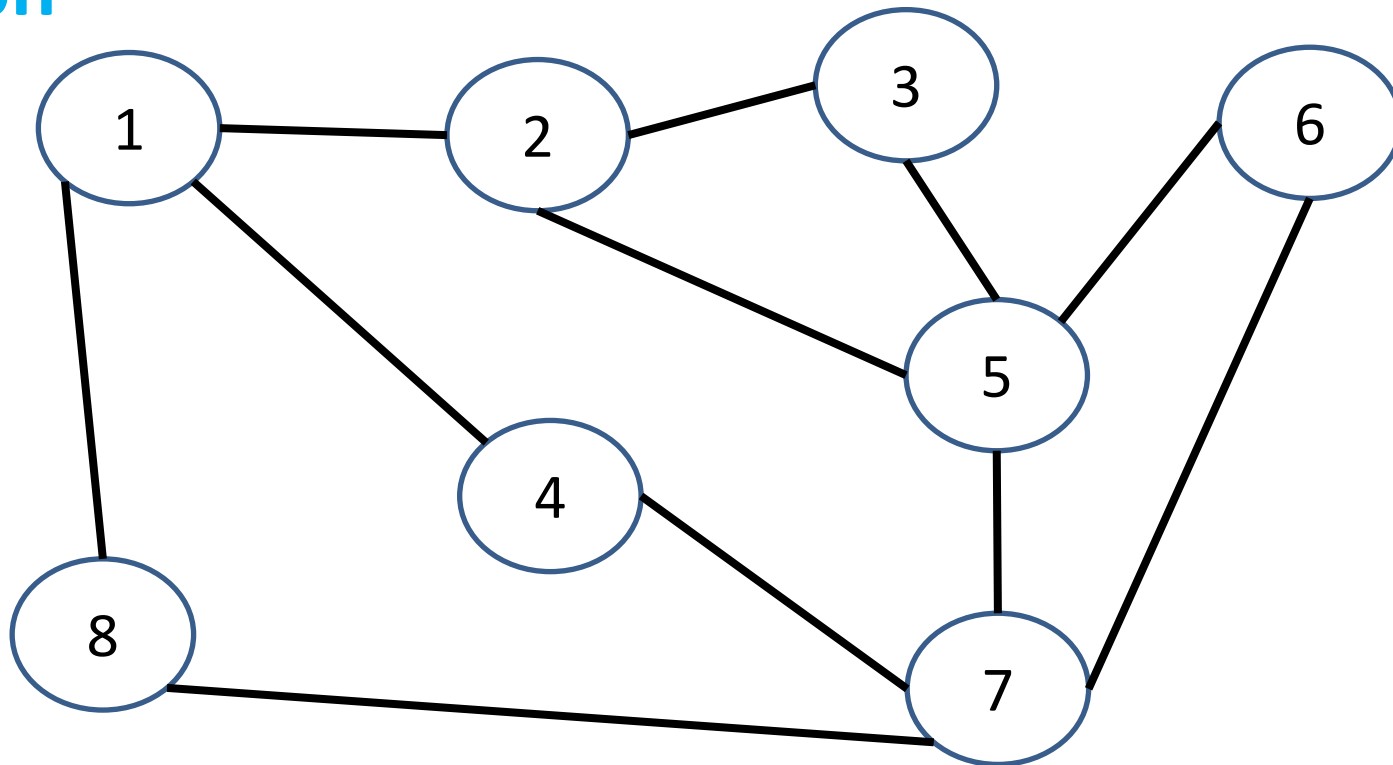
Definition:

A spanning tree of the graph is a tree that contains **all vertices** in the graph.



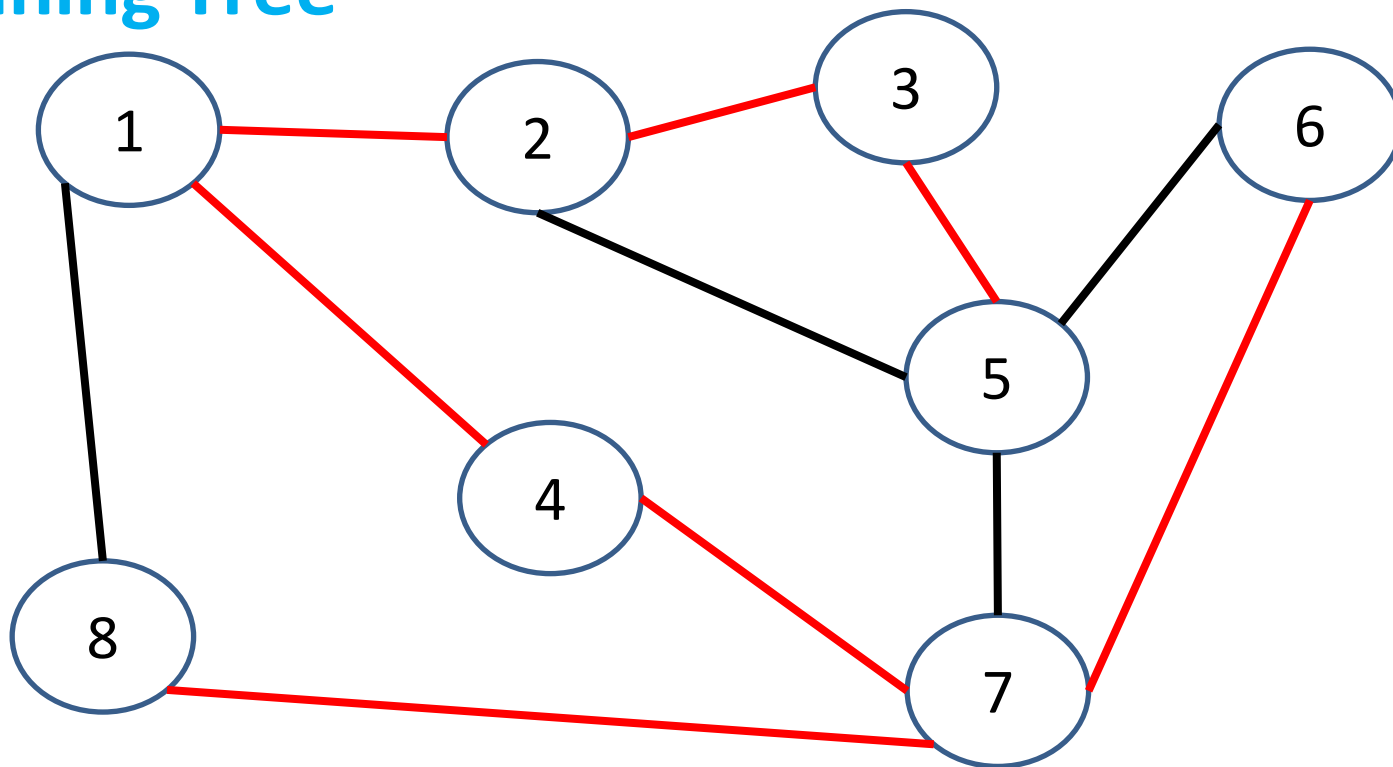
Example

Graph



Example

Spanning Tree

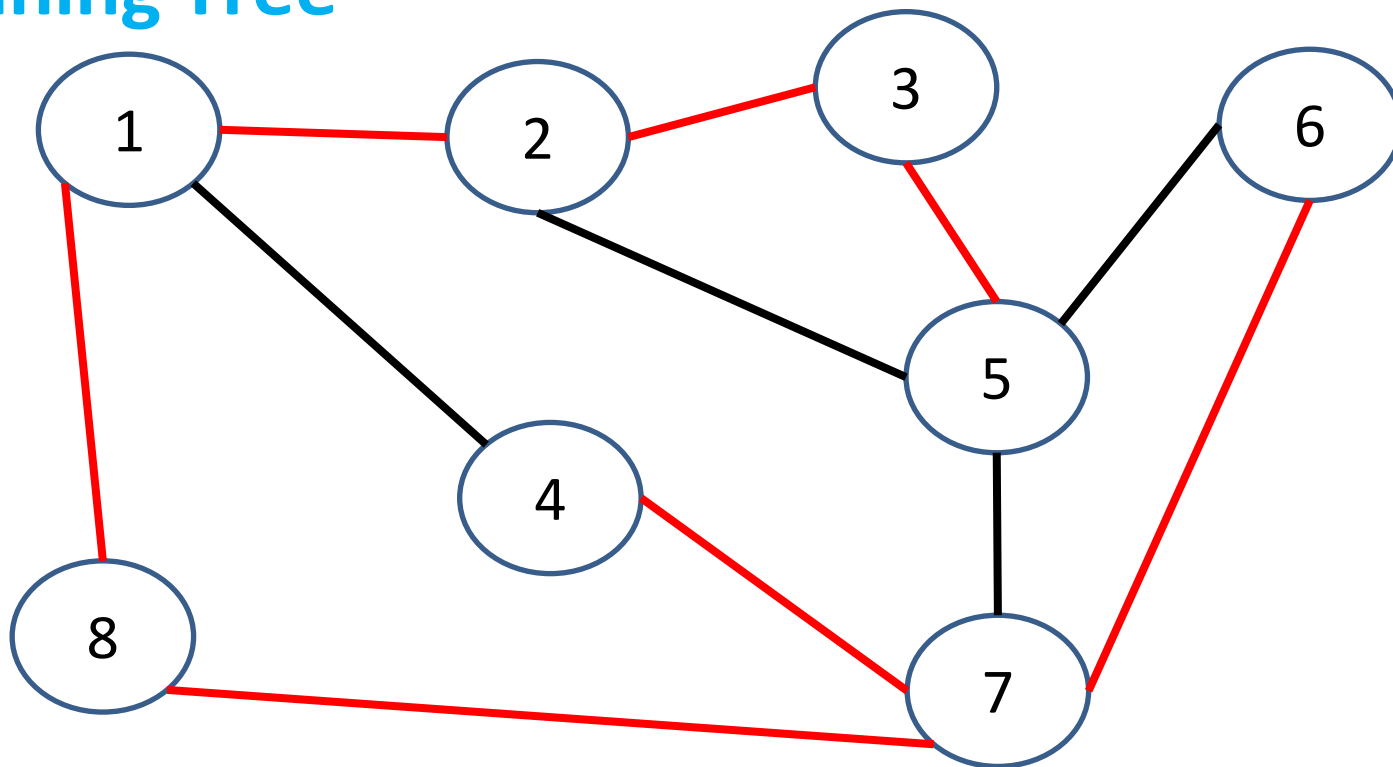


8 nodes , 7 edges



Example

Spanning Tree

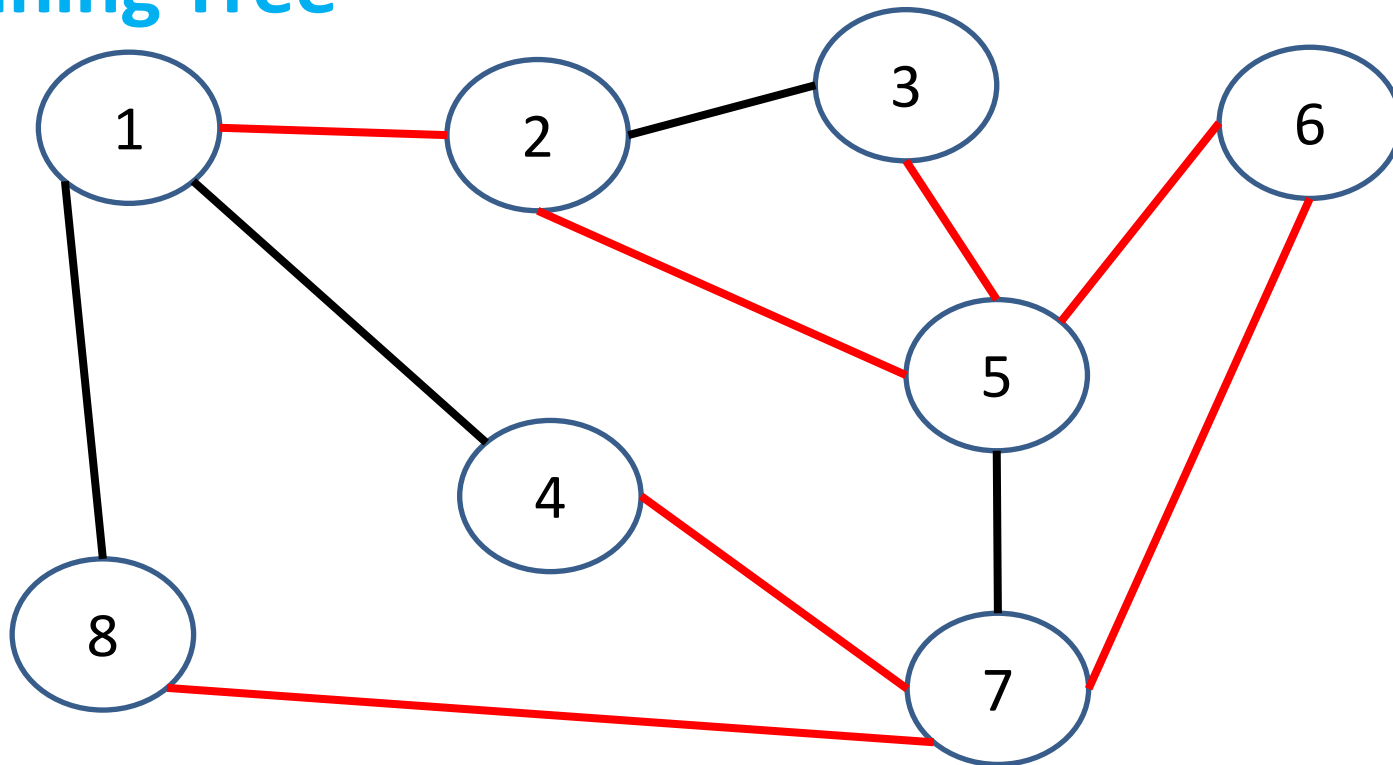


8 nodes , 7 edges



Example

Spanning Tree



8 nodes , 7 edges



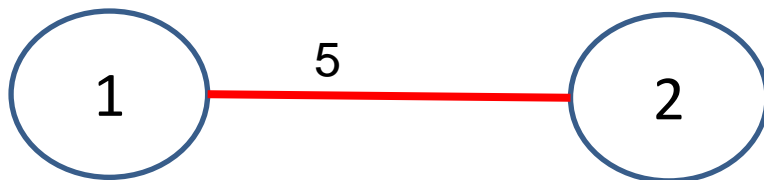
Spanning Tree

A graph may have many spanning trees.



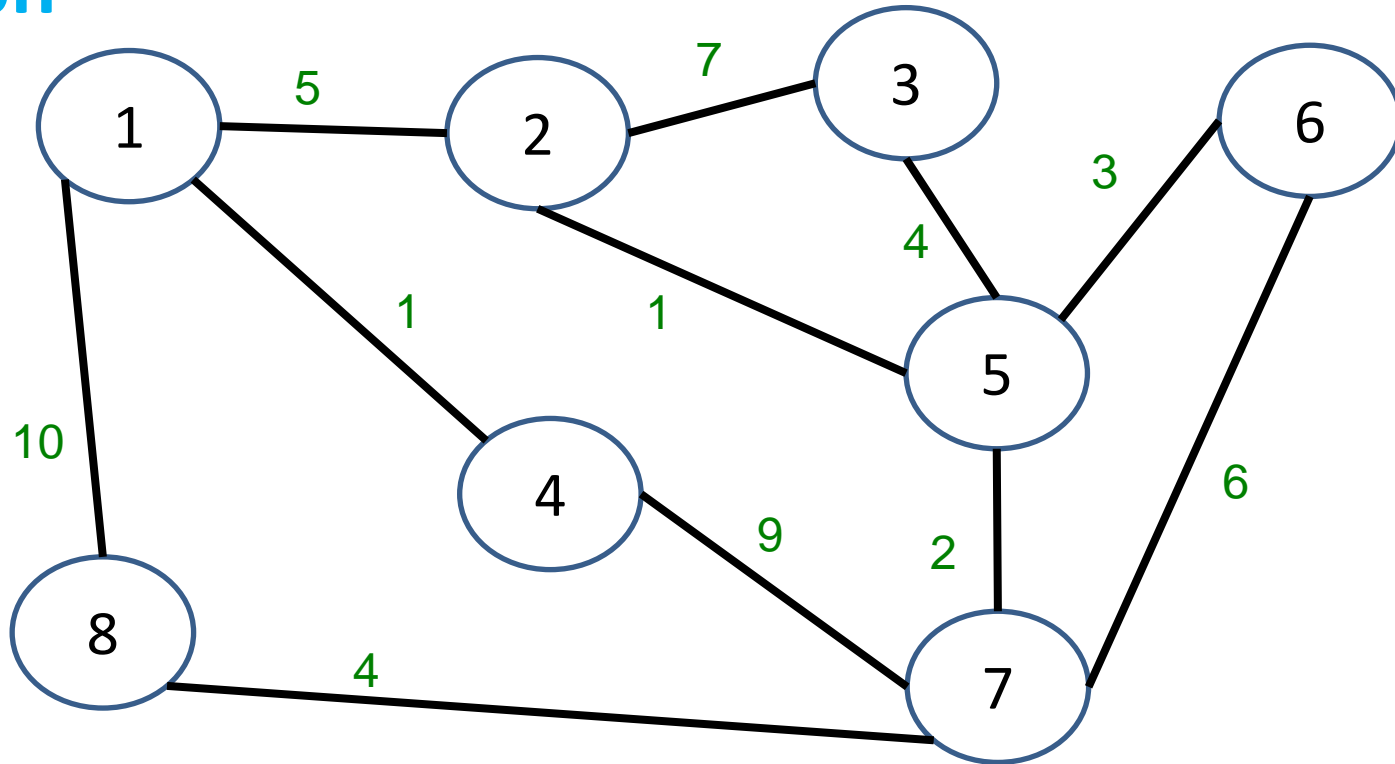
Weighted Graph

- Each edge in the graph has a weight(cost).



Weighted Graph

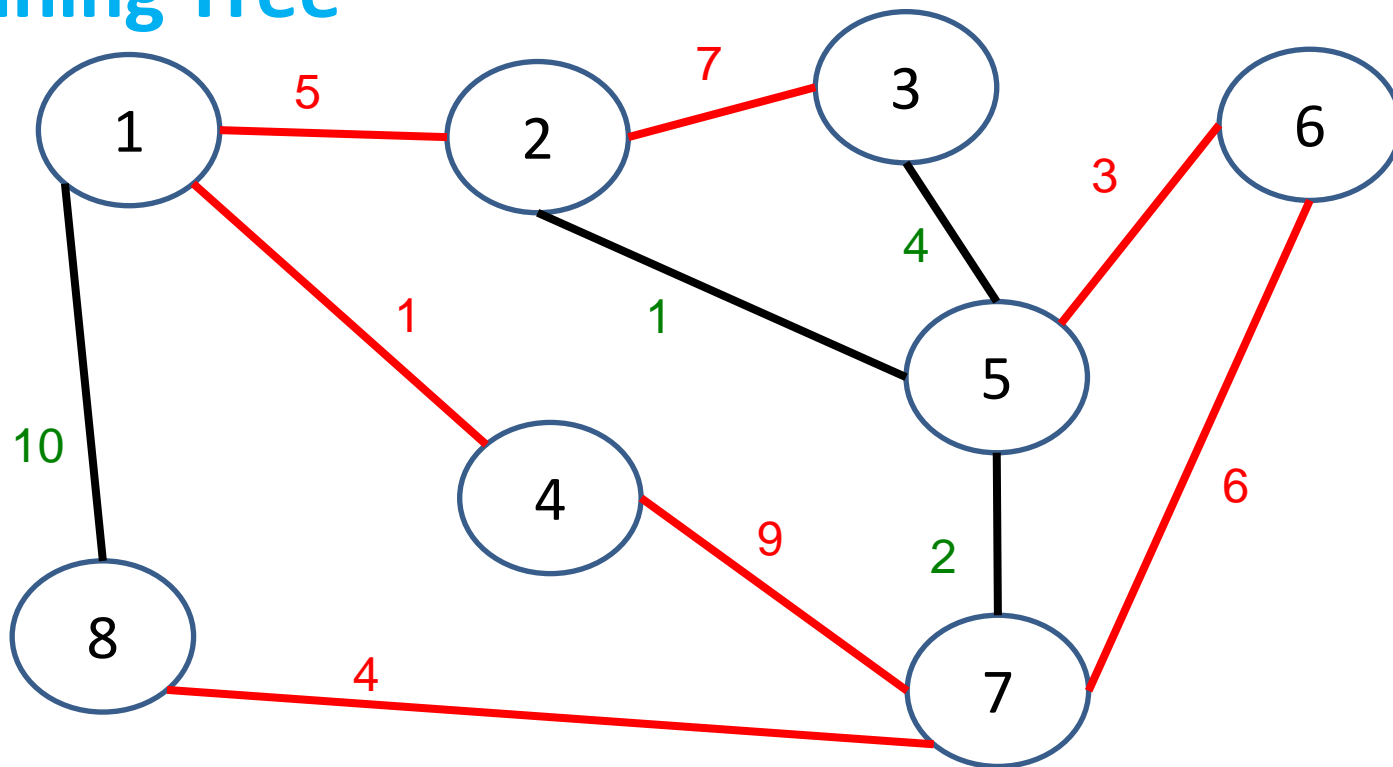
Graph



Spanning Tree

Spanning Tree

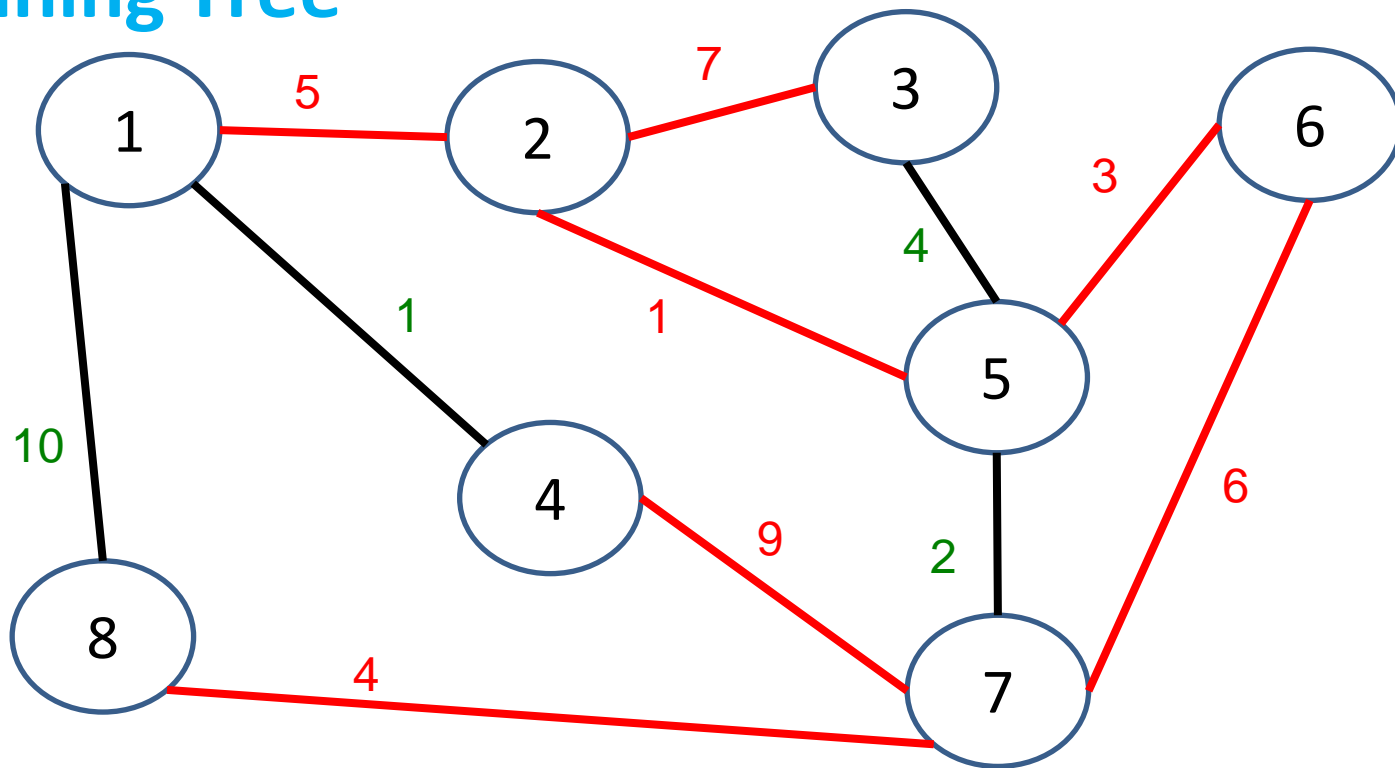
35



Spanning Tree

Spanning Tree

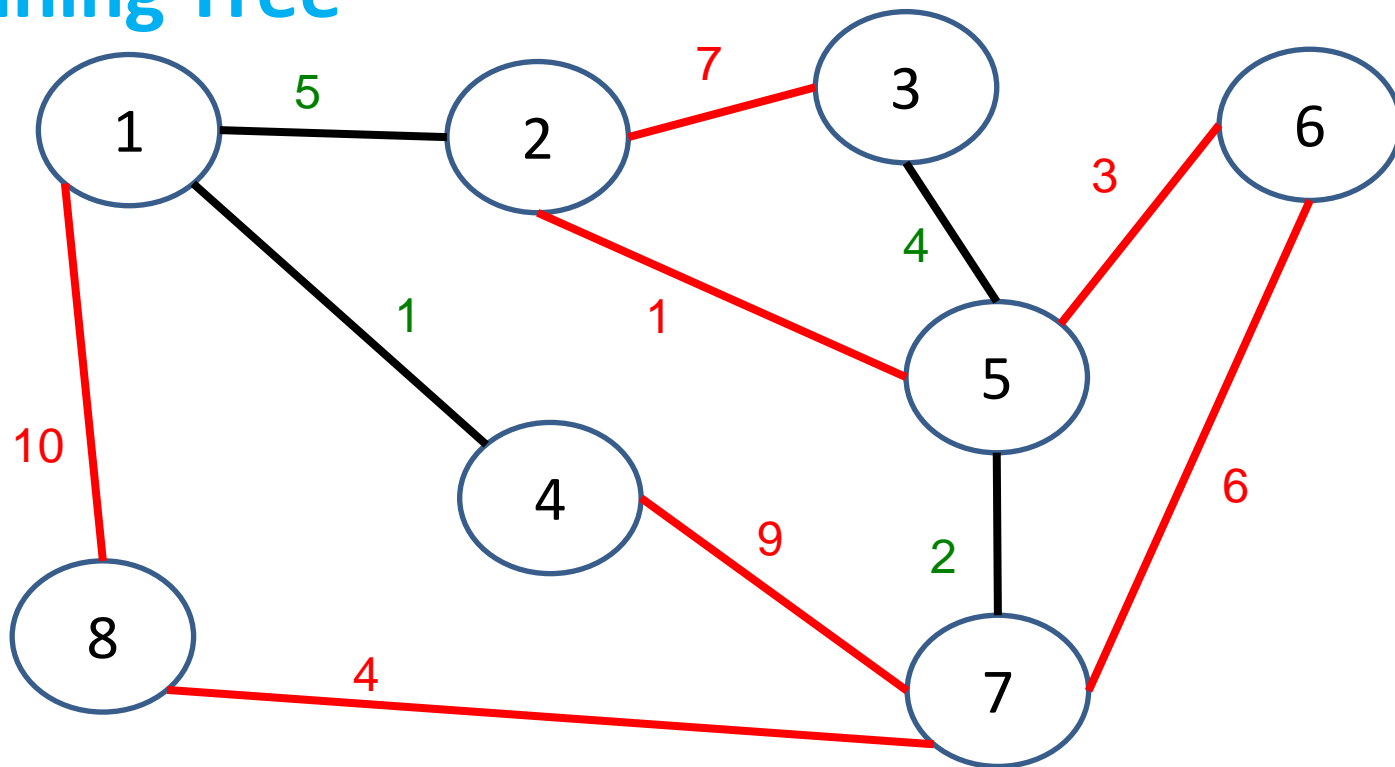
35



Spanning Tree

Spanning Tree

40



Minimum Spanning Tree

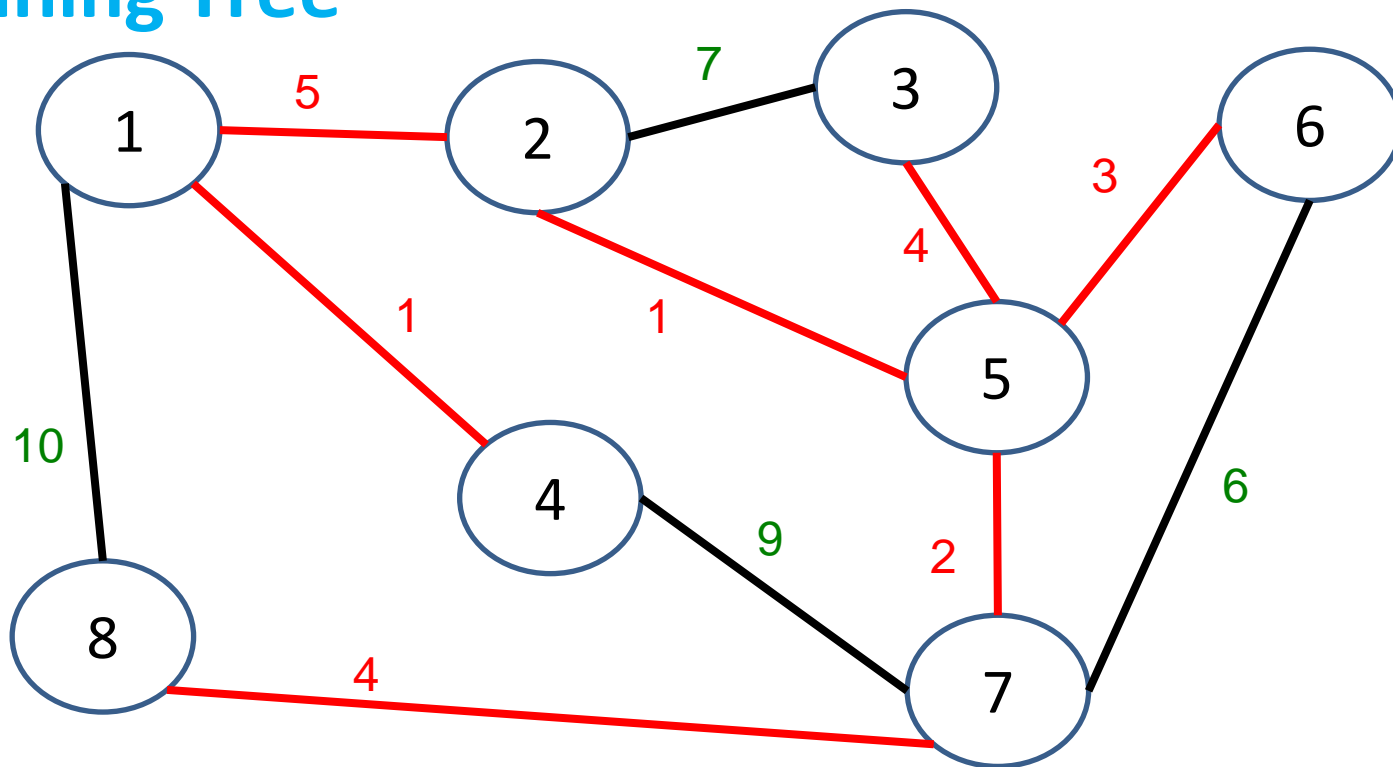
Spanning Tree with **minimum cost**



Minimum Spanning Tree

Spanning Tree

20



Minimum Cost!!



How to find MST?

Kruskal's Algorithm

Prim's Algorithm



Kruskal's Algorithm

Pseudo code

totalcost \leftarrow 0

for each $v \in V$

do MAKE-SET (v)

sort the edges into non-decreasing order by weight

for each edge $(u, v) \in E$, taken in non-decreasing order

do if FIND-SET (u) \neq FIND-SET (v)

then UNION (u, v)

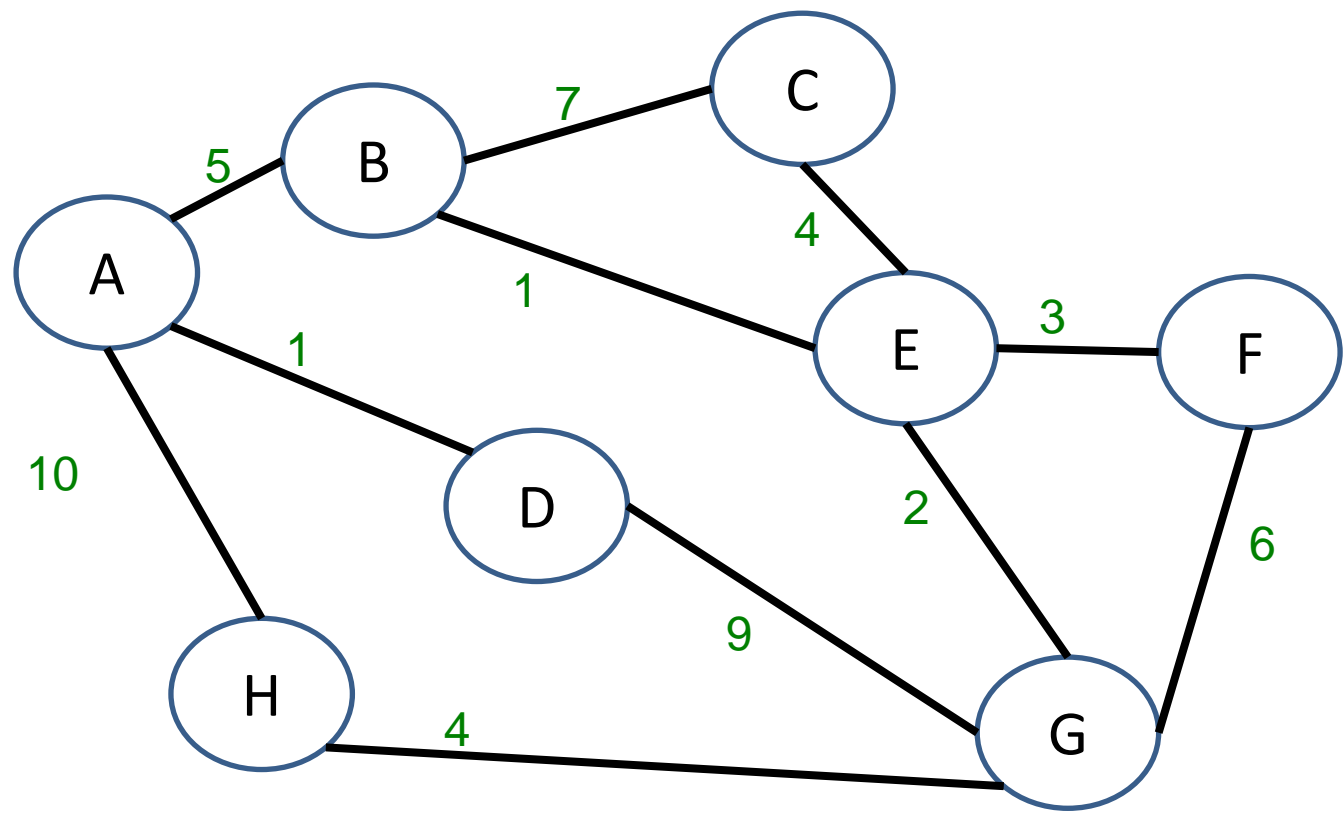
totalcost \leftarrow totalcost + $w(u, v)$

return totalcost



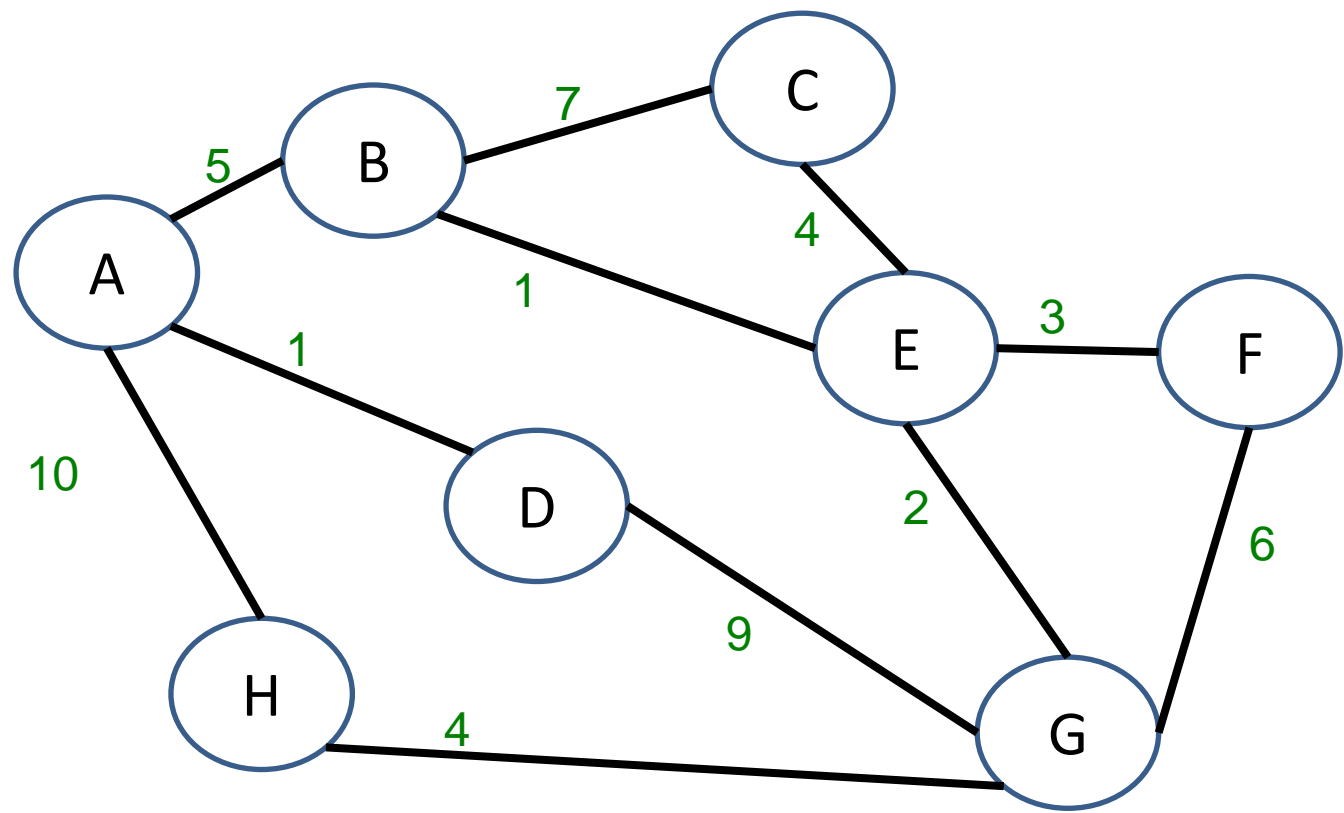
Kruskal's Algorithm

- AB - 5
- AD - 1
- AH - 10
- BC - 7
- BE - 1
- CE - 4
- DG - 9
- EF - 3
- EG - 2
- FG - 6
- GH - 4



Kruskal's Algorithm

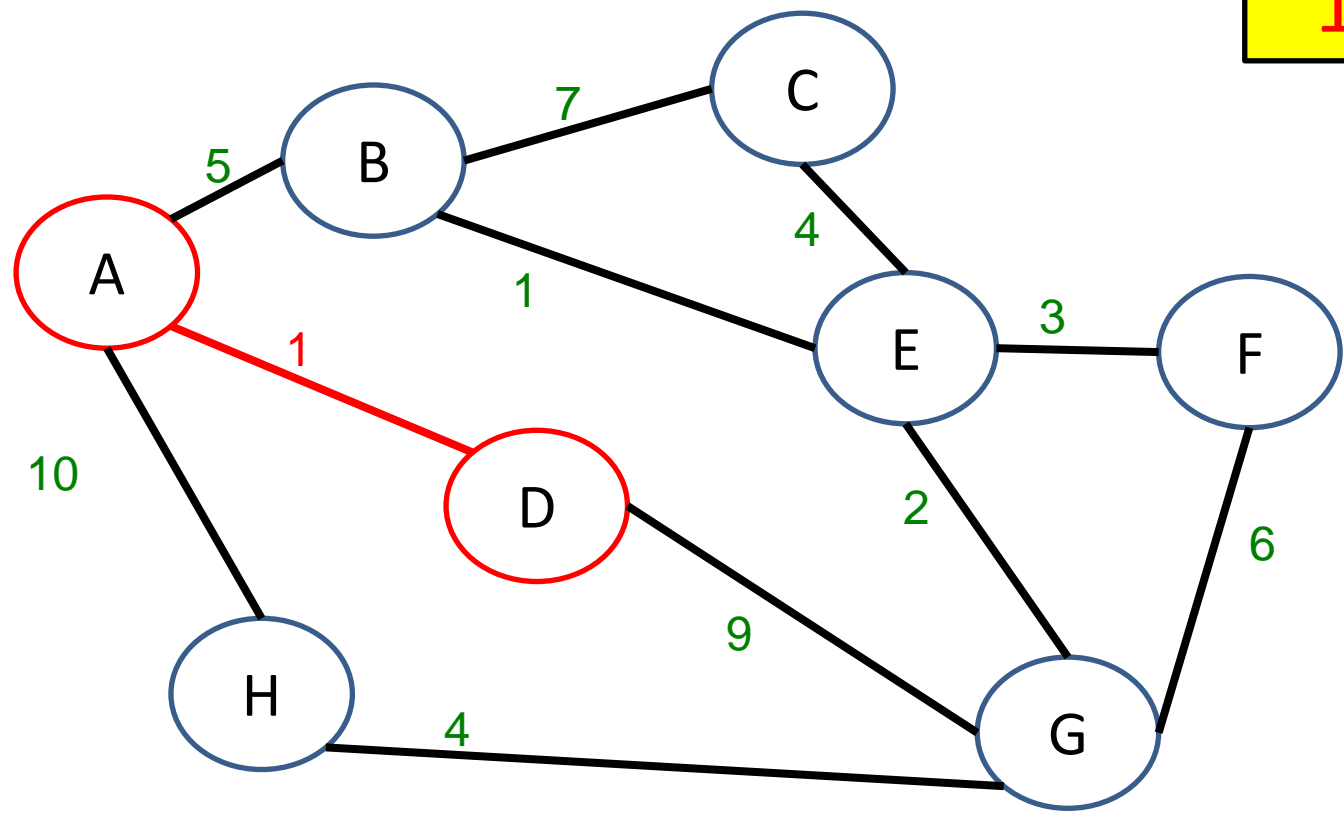
- AD - 1
- BE - 1
- EG - 2
- EF - 3
- CE - 4
- GH - 4
- AB - 5
- FG - 6
- BC - 7
- DG - 9
- AH - 10



Kruskal's Algorithm

- AD - 1
- BE - 1
- EG - 2
- EF - 3
- CE - 4
- GH - 4
- AB - 5
- FG - 6
- BC - 7
- DG - 9
- AH - 10

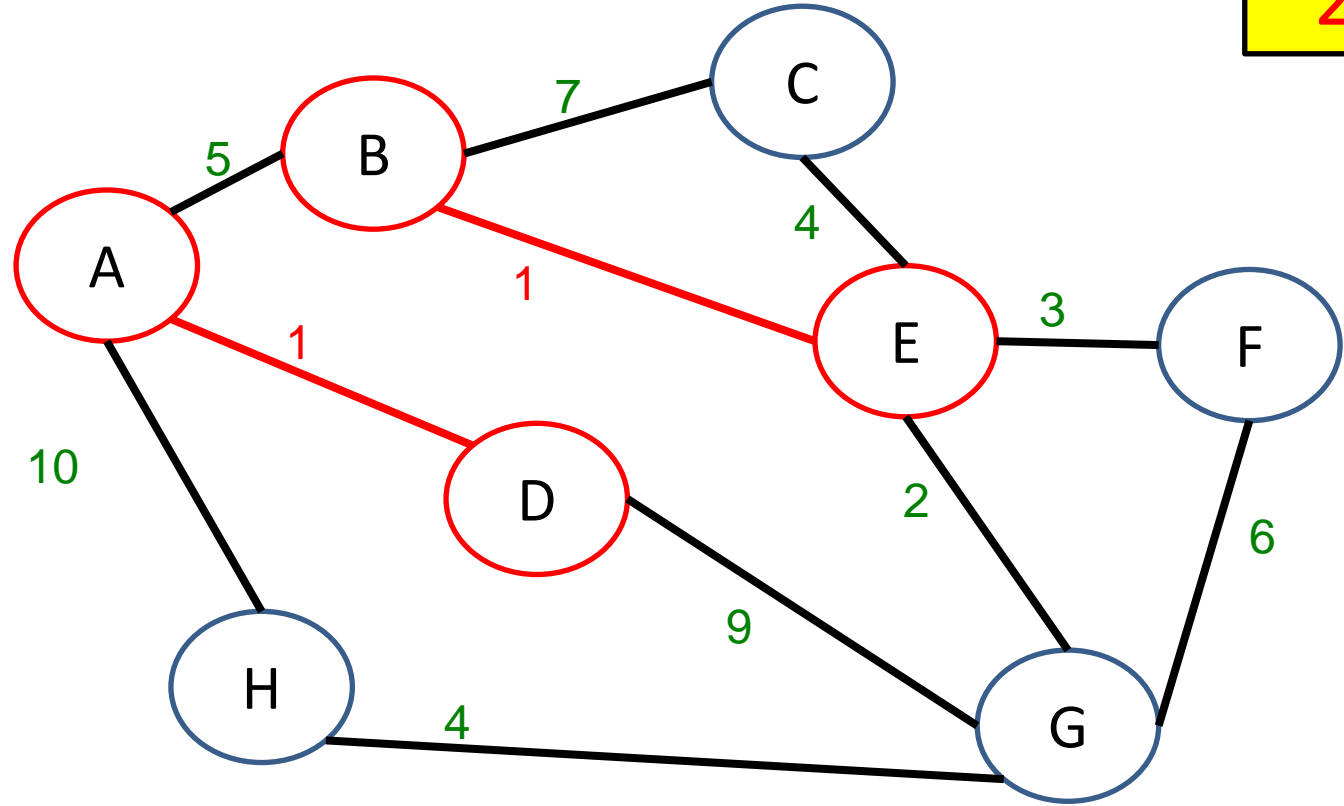
1



Kruskal's Algorithm

2

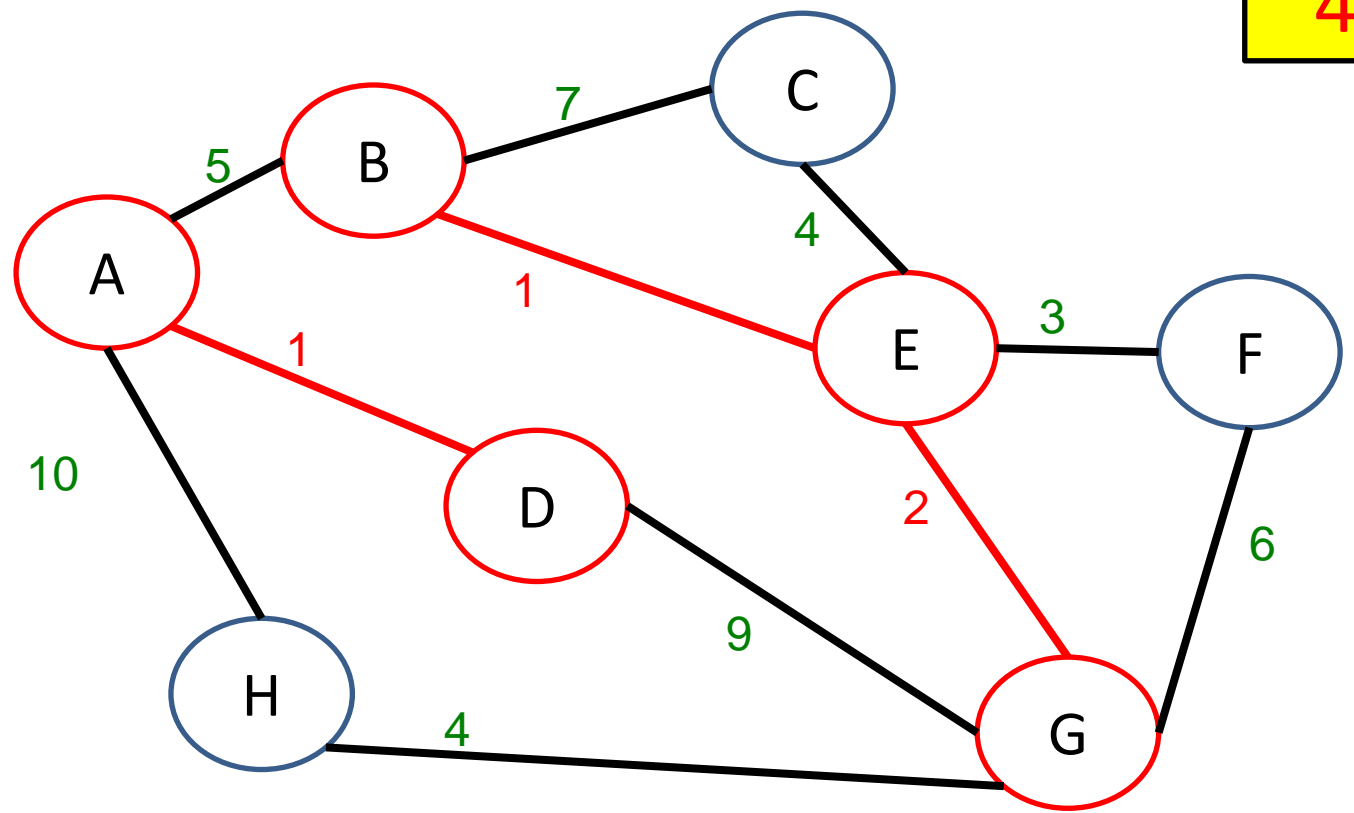
- AD - 1
- BE - 1
- EG - 2
- EF - 3
- CE - 4
- GH - 4
- AB - 5
- FG - 6
- BC - 7
- DG - 9
- AH - 10



Kruskal's Algorithm

- AD - 1
- BE - 1
- EG - 2
- EF - 3
- CE - 4
- GH - 4
- AB - 5
- FG - 6
- BC - 7
- DG - 9
- AH - 10

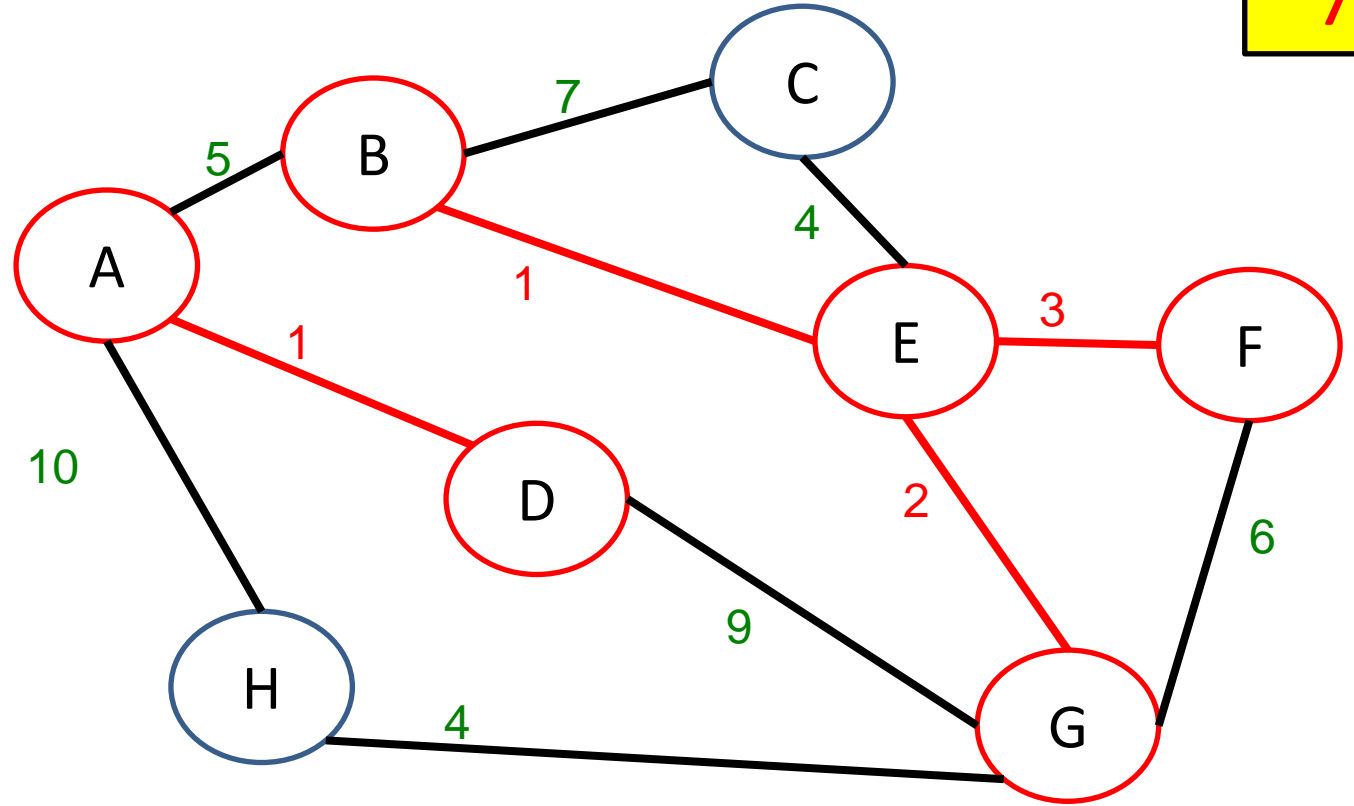
4



Kruskal's Algorithm

- AD - 1
- BE - 1
- EG - 2
- EF - 3
- CE - 4
- GH - 4
- AB - 5
- FG - 6
- BC - 7
- DG - 9
- AH - 10

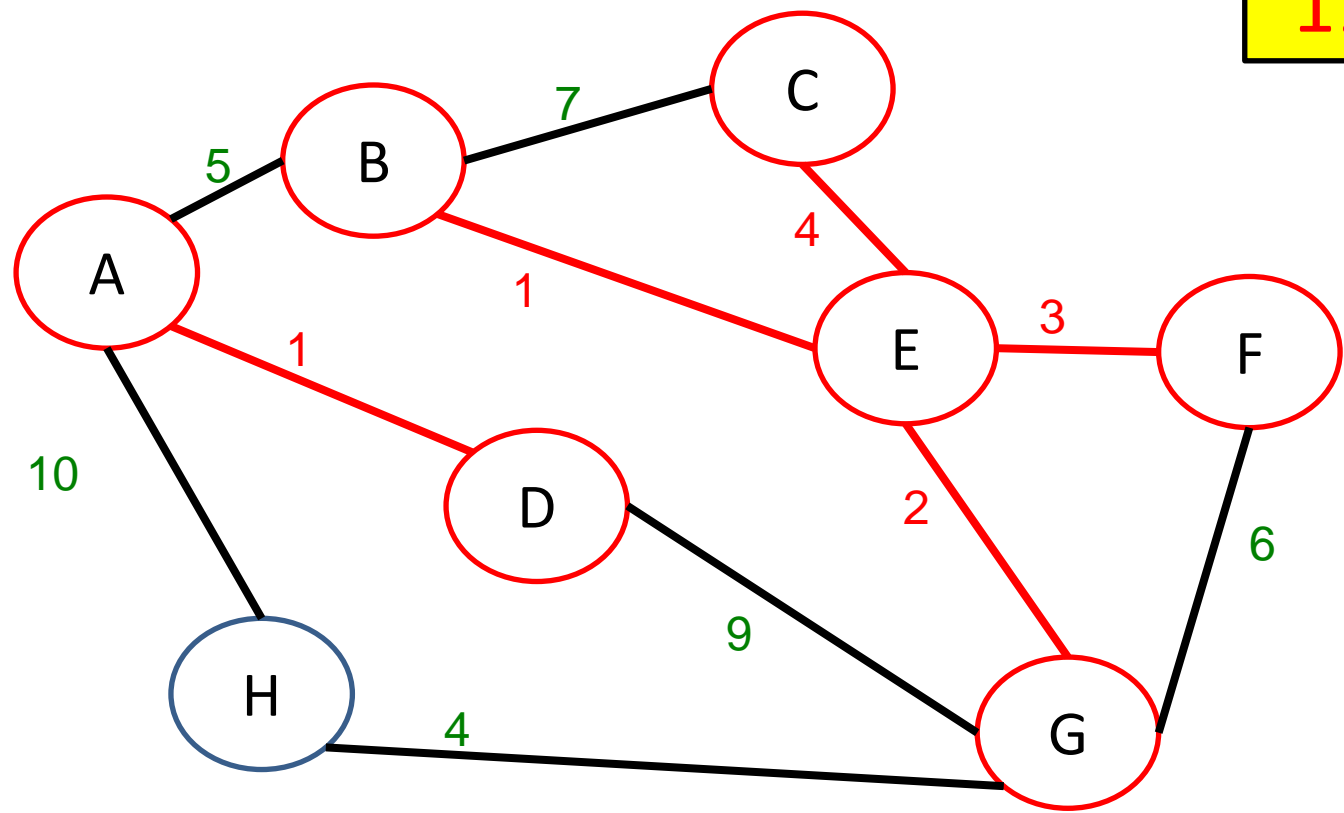
7



Kruskal's Algorithm

11

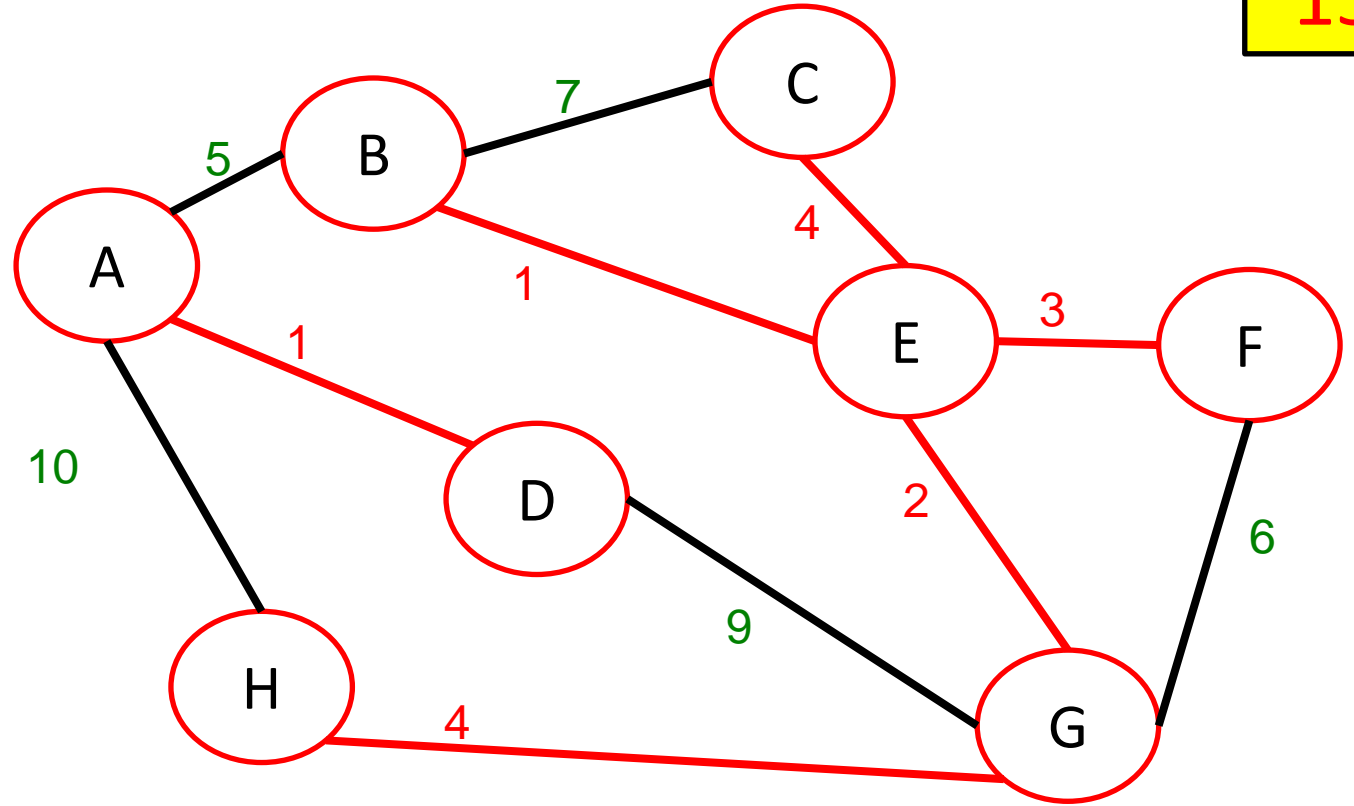
- AD - 1
- BE - 1
- EG - 2
- EF - 3
- CE - 4
- GH - 4
- AB - 5
- FG - 6
- BC - 7
- DG - 9
- AH - 10



Kruskal's Algorithm

15

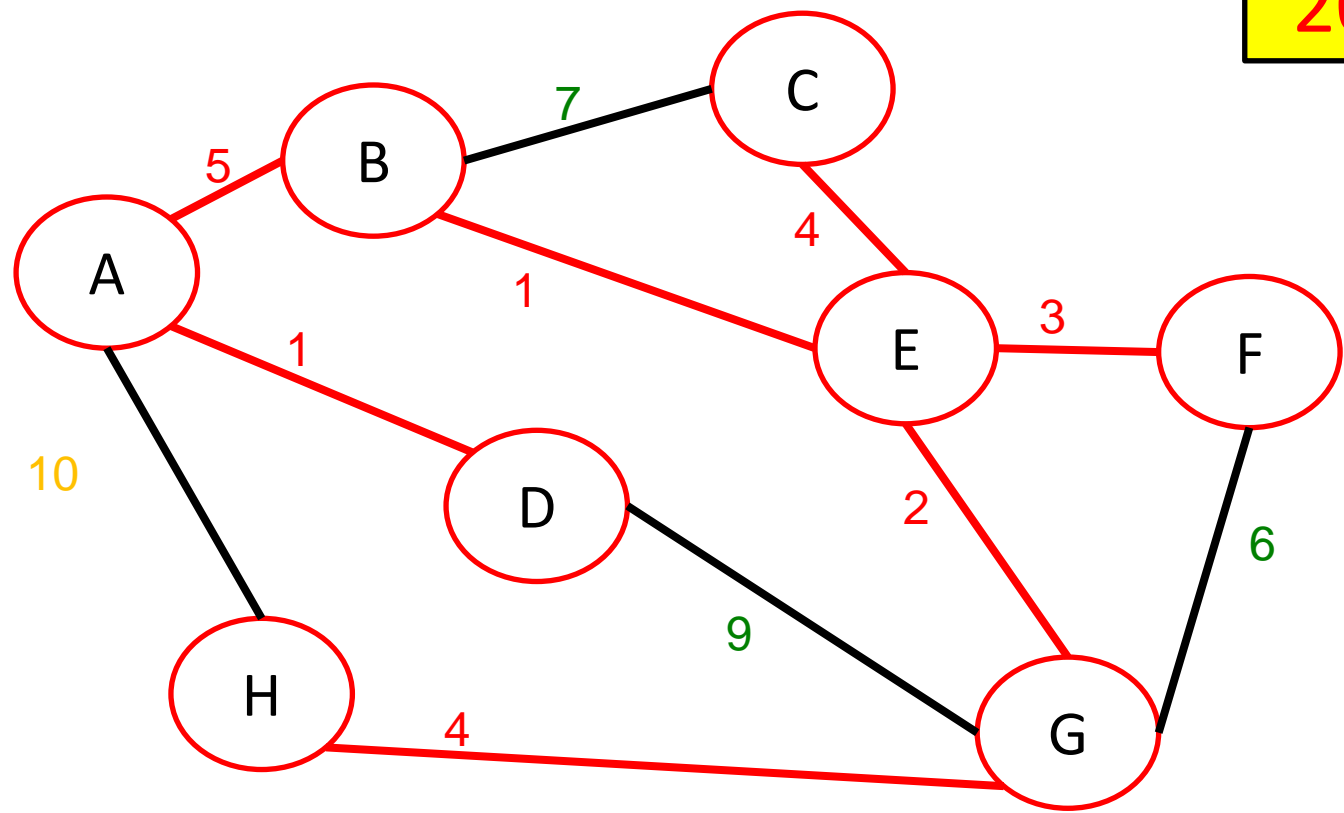
- AD - 1
- BE - 1
- EG - 2
- EF - 3
- CE - 4
- GH - 4
- AB - 5
- FG - 6
- BC - 7
- DG - 9
- AH - 10



Kruskal's Algorithm

20

- AD - 1
- BE - 1
- EG - 2
- EF - 3
- CE - 4
- GH - 4
- AB - 5
- FG - 6
- BC - 7
- DG - 9
- AH - 10



Kruskal's Algorithm

MST 並不唯一!!!

時間複雜度: $O(E \lg E)$



Prim's Algorithm

Pseudo code

add source vertex u to Q

for $i=1$ to $V-1$

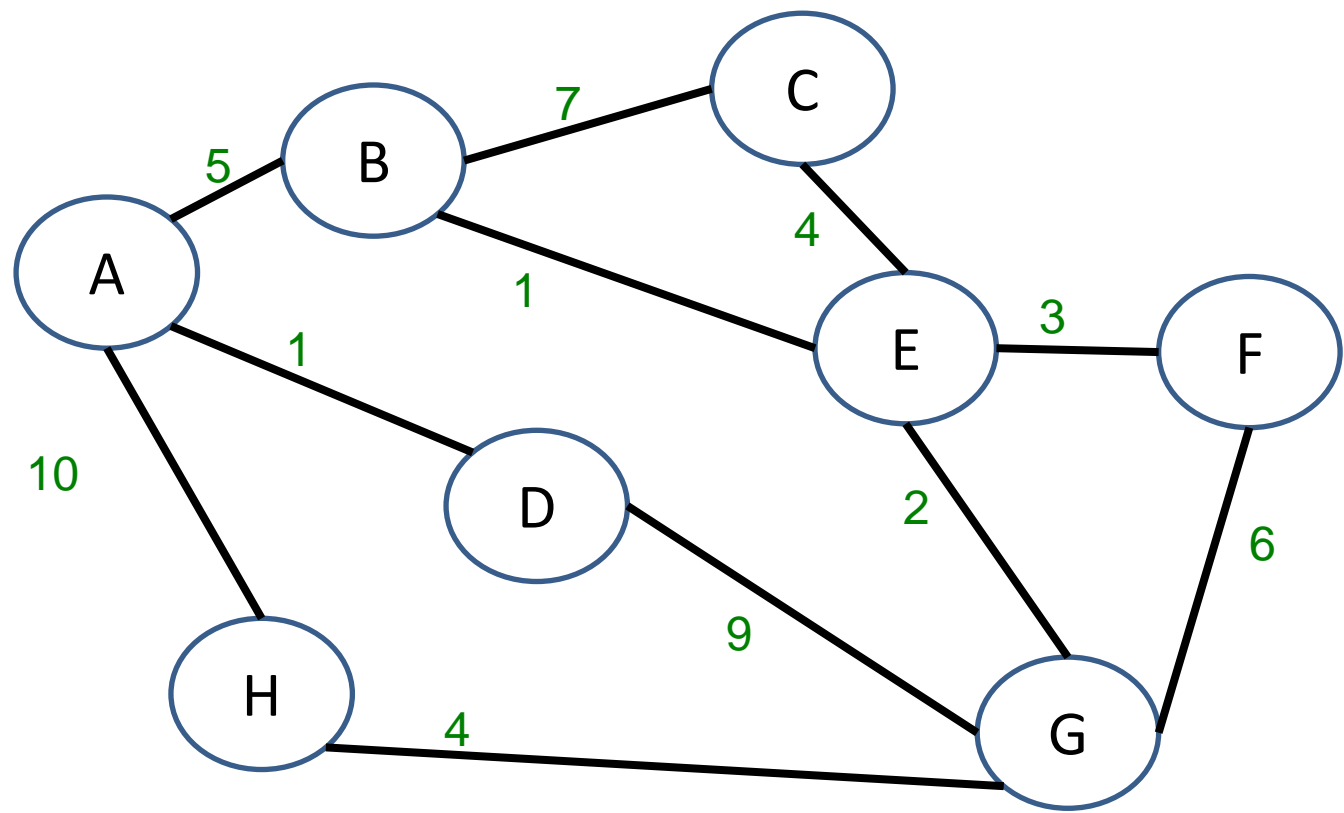
find an edge $u-w$ $\{u \in Q, w \notin Q\}$ s.t. $u-w$ is the smallest weighted
edge connected with Q

add w to Q



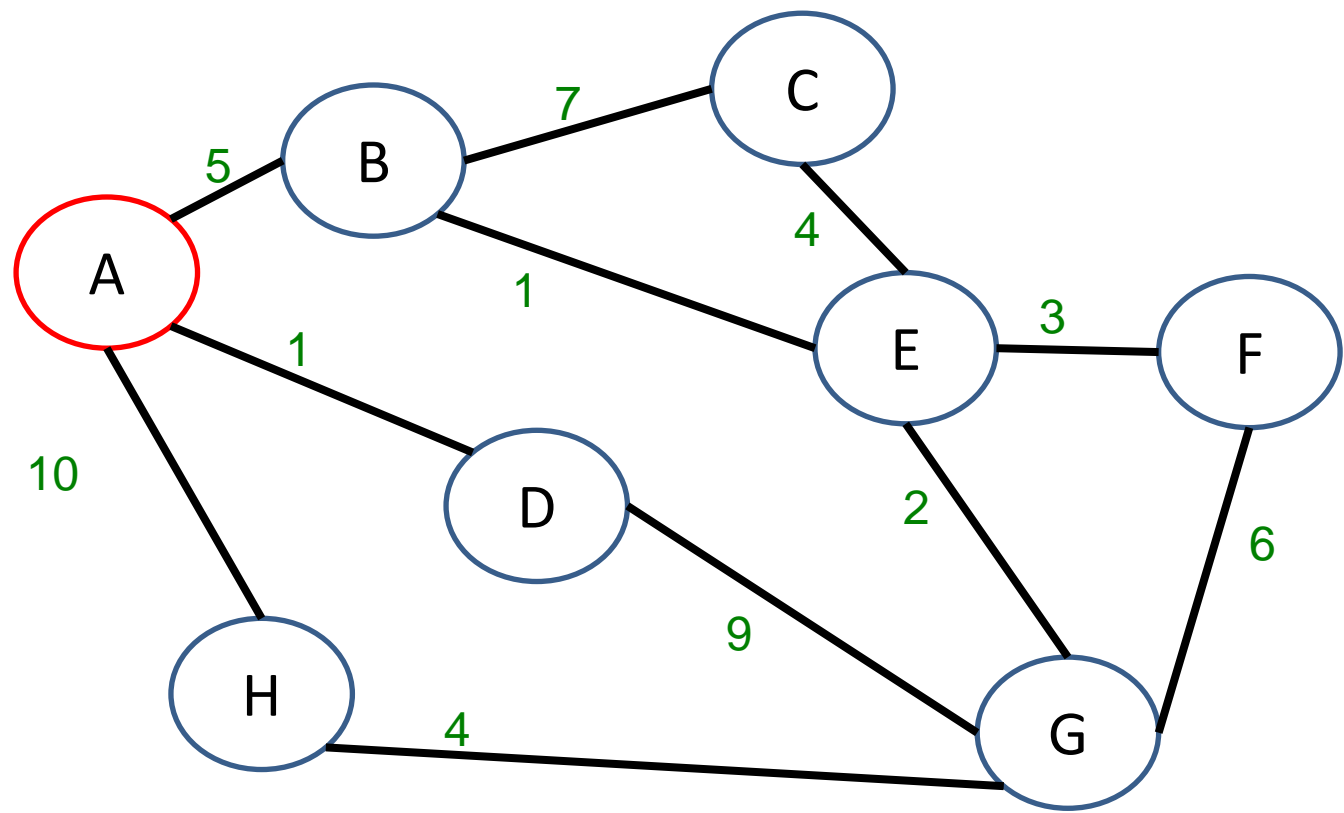
Prim's Algorithm

- AB - 5
- AD - 1
- AH - 10
- BC - 7
- BE - 1
- CE - 4
- DG - 9
- EF - 3
- EG - 2
- FG - 6
- GH - 4



Prim's Algorithm

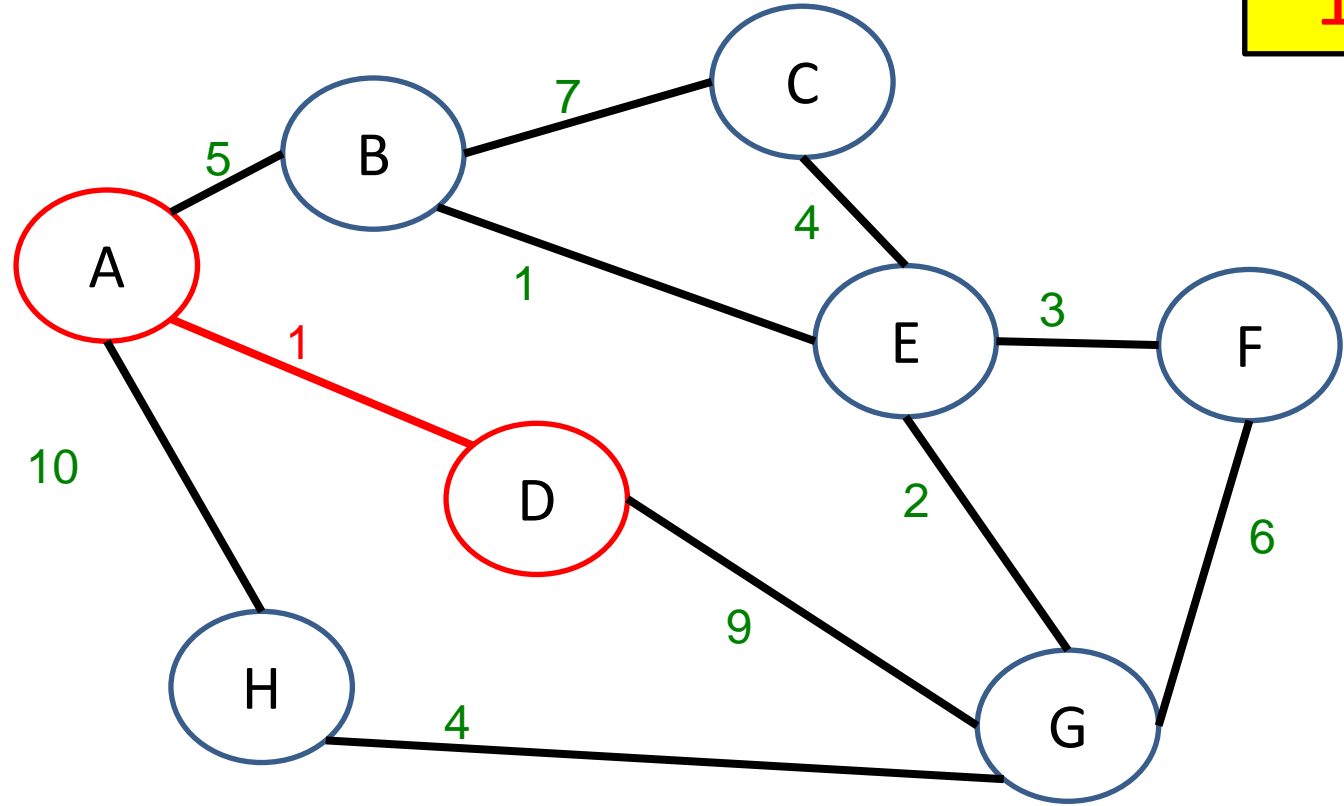
- AB - 5
- AD - 1
- AH - 10
- BC - 7
- BE - 1
- CE - 4
- DG - 9
- EF - 3
- EG - 2
- FG - 6
- GH - 4



Prim's Algorithm

1

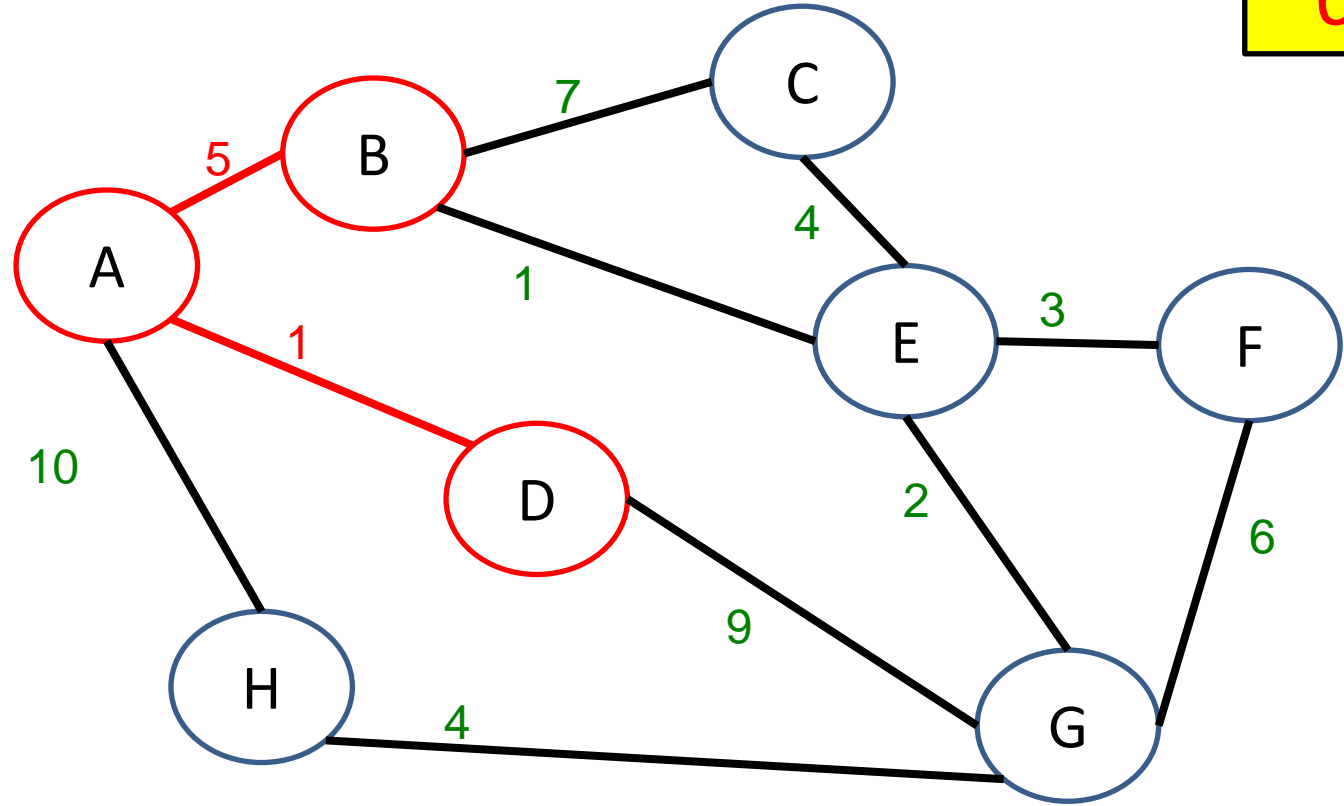
- AB - 5
- AD - 1
- AH - 10
- BC - 7
- BE - 1
- CE - 4
- DG - 9
- EF - 3
- EG - 2
- FG - 6
- GH - 4



Prim's Algorithm

6

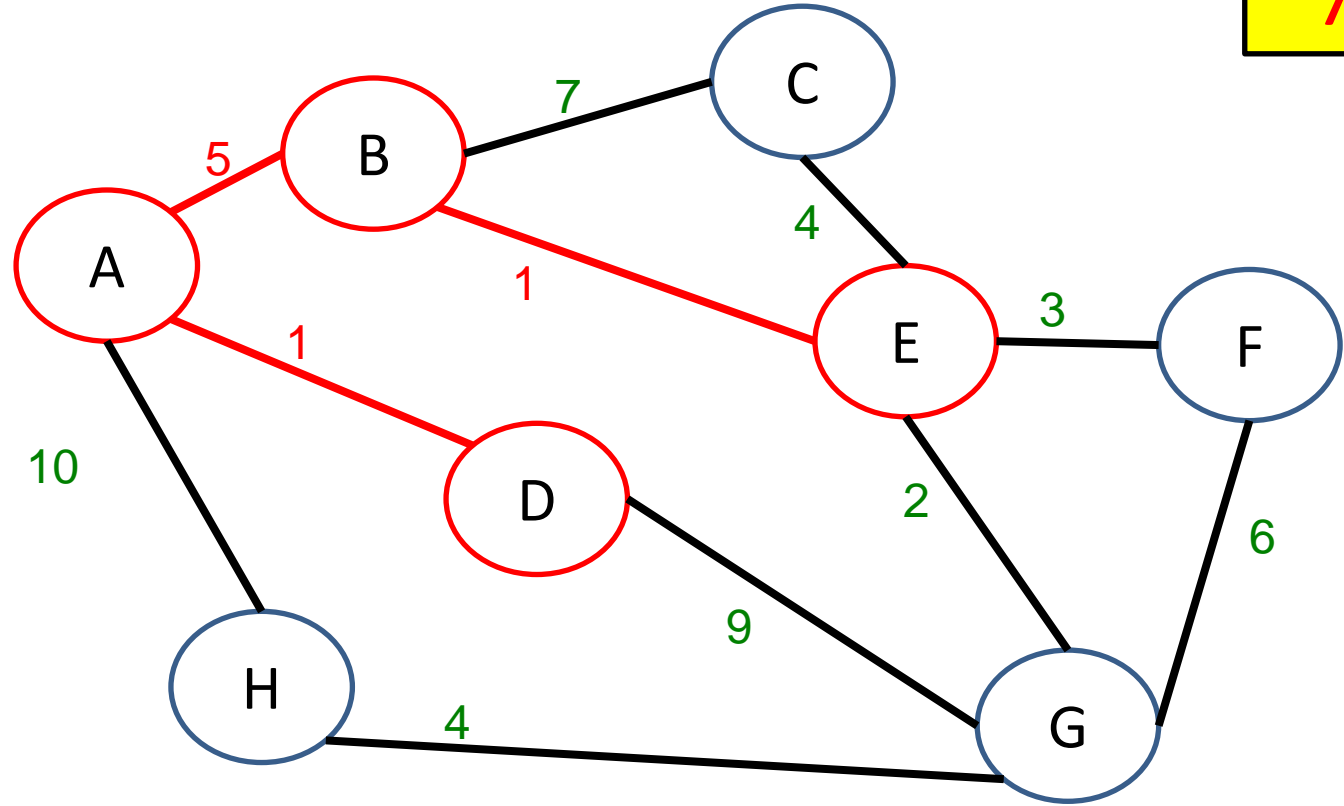
- AB - 5
- AD - 1
- AH - 10
- BC - 7
- BE - 1
- CE - 4
- DG - 9
- EF - 3
- EG - 2
- FG - 6
- GH - 4



Prim's Algorithm

7

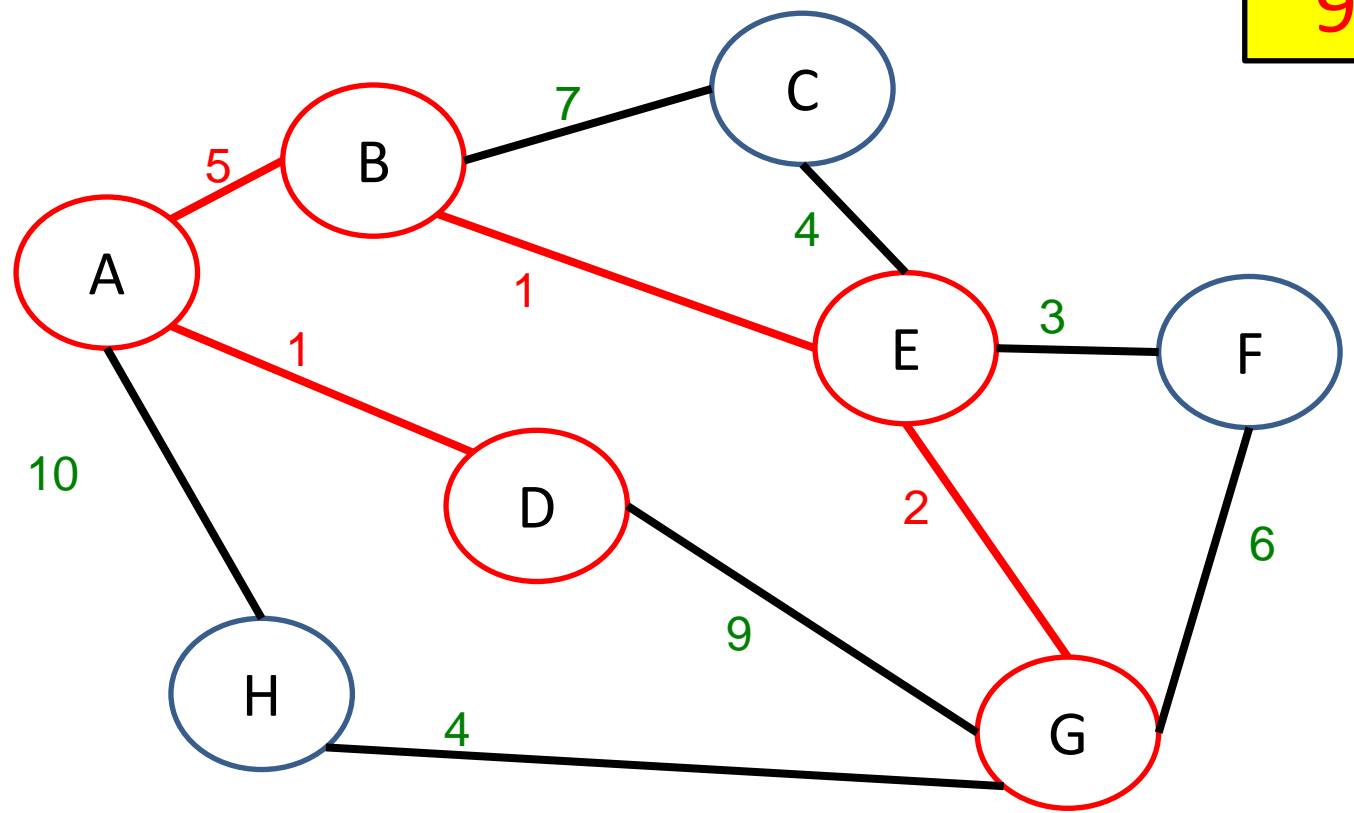
- AB - 5
- AD - 1
- AH - 10
- BC - 7
- BE - 1
- CE - 4
- DG - 9
- EF - 3
- EG - 2
- FG - 6
- GH - 4



Prim's Algorithm

9

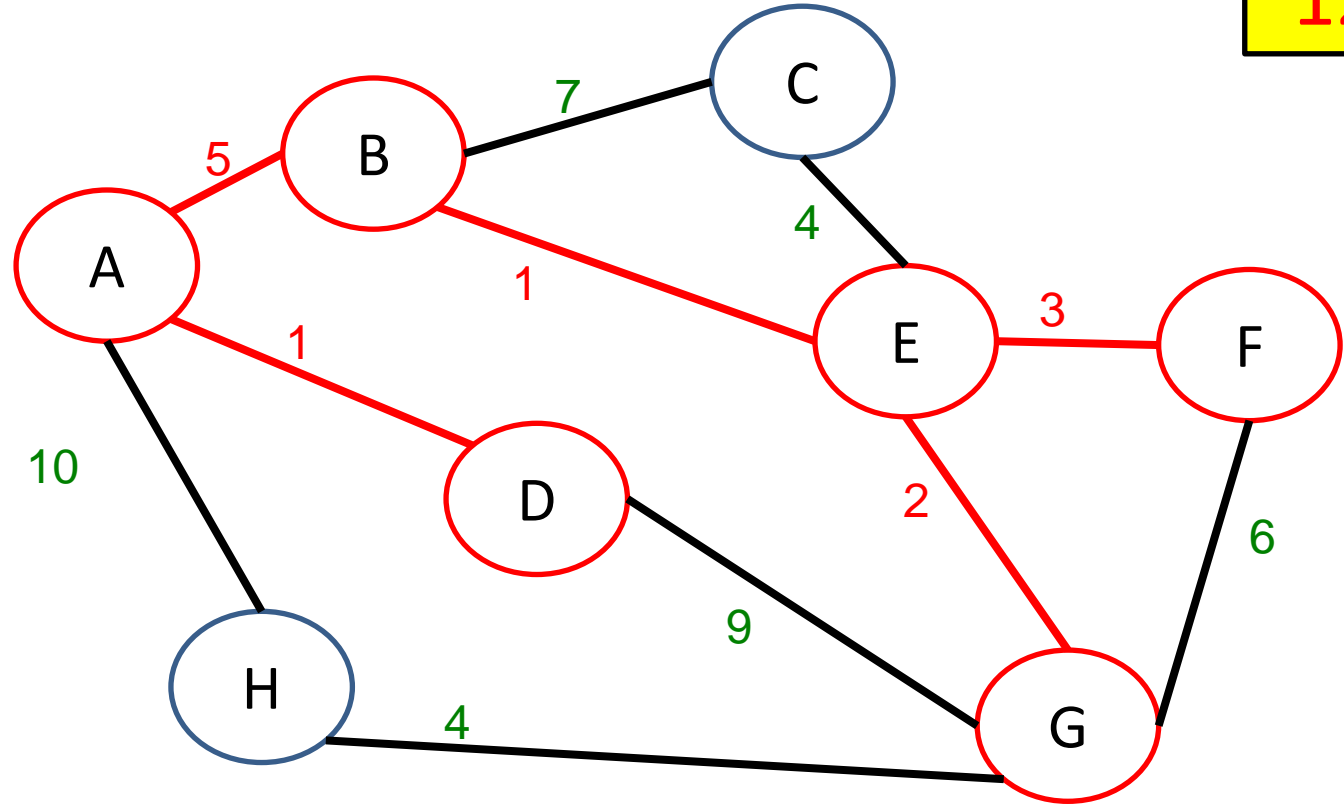
- AB - 5
- AD - 1
- AH - 10
- BC - 7
- BE - 1
- CE - 4
- DG - 9
- EF - 3
- EG - 2
- FG - 6
- GH - 4



Prim's Algorithm

12

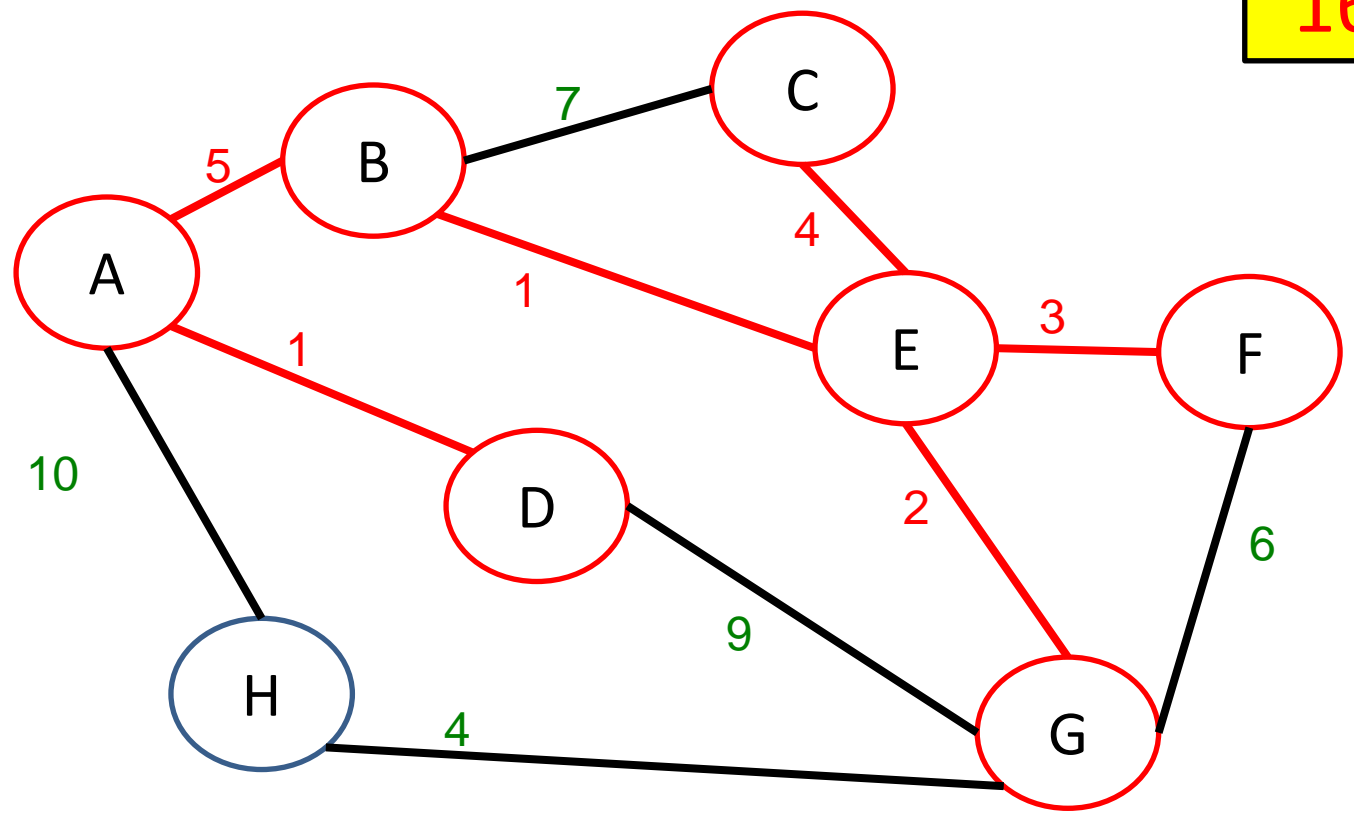
- AB - 5
- AD - 1
- AH - 10
- BC - 7
- BE - 1
- CE - 4
- DG - 9
- EF - 3
- EG - 2
- FG - 6
- GH - 4



Prim's Algorithm

16

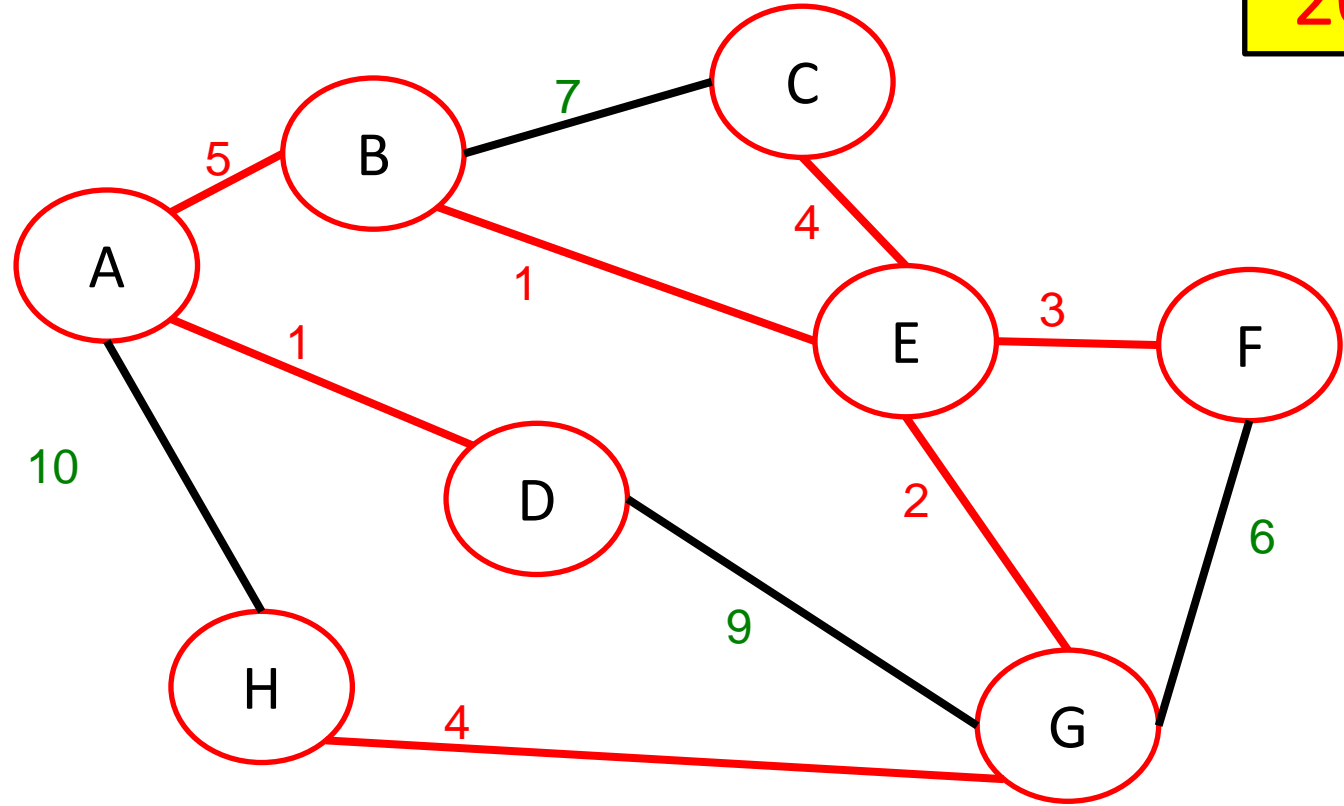
- AB - 5
- AD - 1
- AH - 10
- BC - 7
- BE - 1
- CE - 4
- DG - 9
- EF - 3
- EG - 2
- FG - 6
- GH - 4



Prim's Algorithm

20

- AB - 5
- AD - 1
- AH - 10
- BC - 7
- BE - 1
- CE - 4
- DG - 9
- EF - 3
- EG - 2
- FG - 6
- GH - 4



Prim's Algorithm

Time Complexity: $O(V^2)$



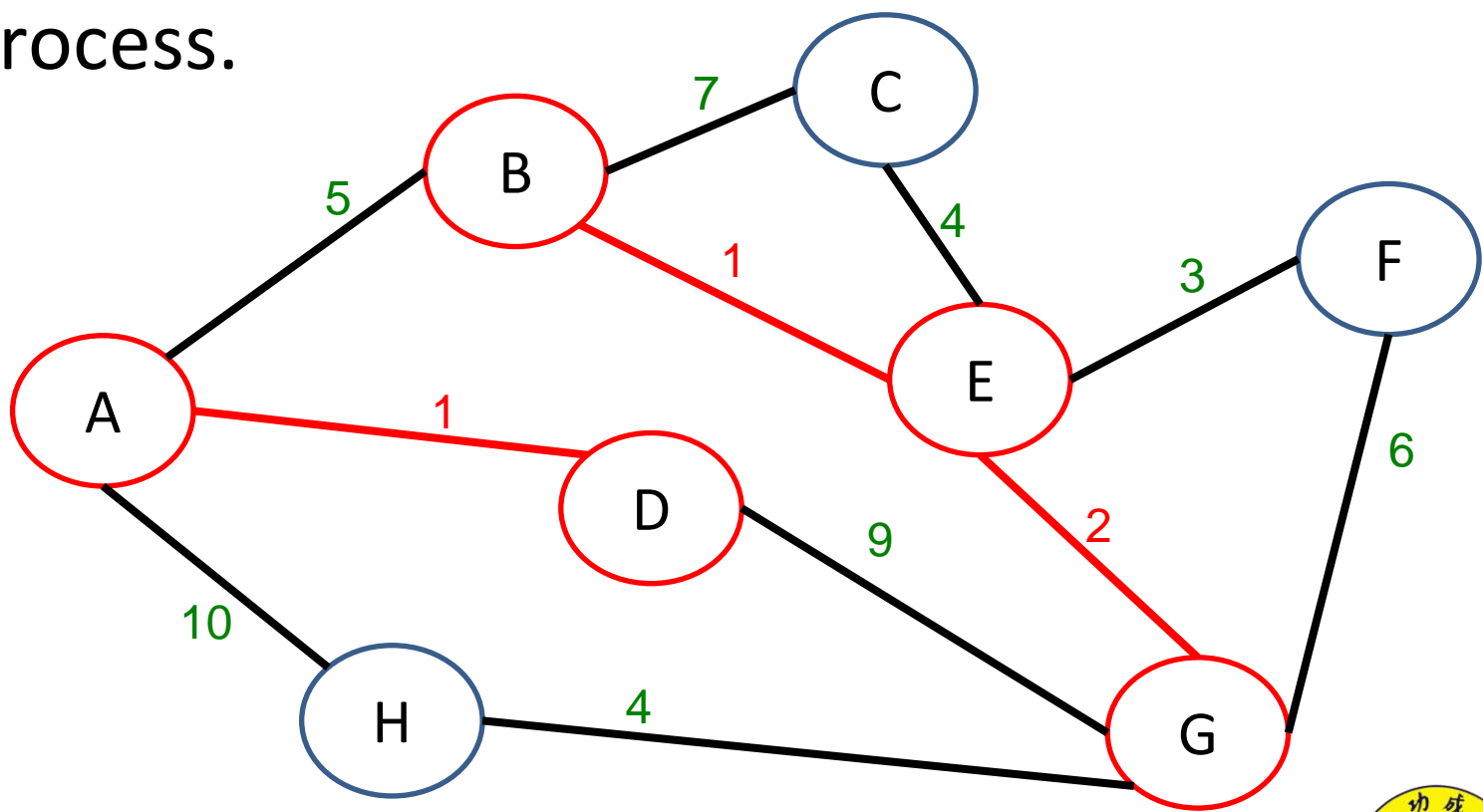
Prim's Algorithm

With Binary-Heap : $O((V+E) \lg V)$



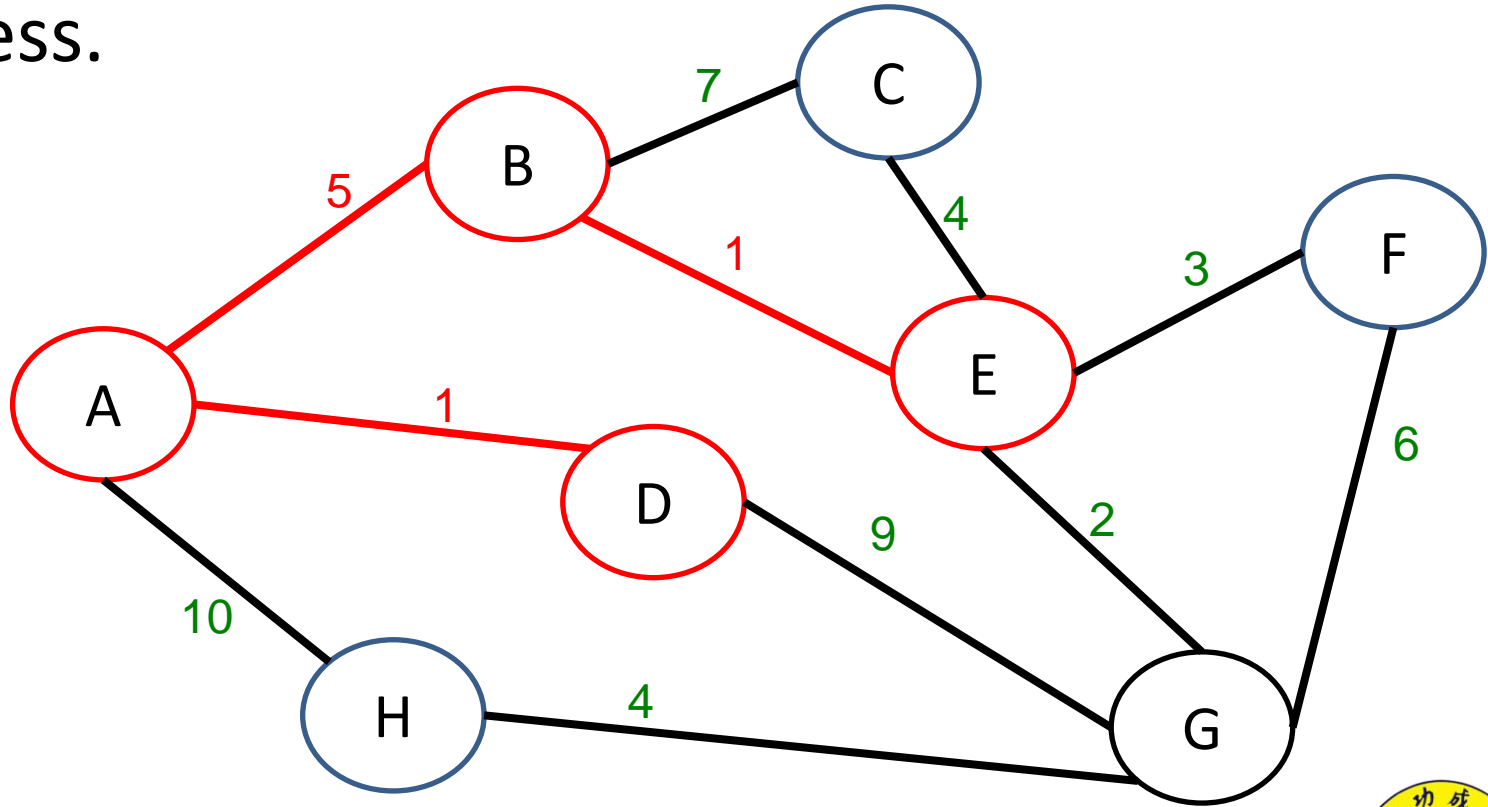
Compare

Kruskal's algorithm may form a **forest** during the process.



Compare

Prim's algorithm only maintain a **tree** during the process.



Practice

UVa -10034: Freckles



Thank you for your attention!

