$\frac{\Sigma(\deg v)}{2} = \text{Edge}(m)$ n vertices m edges

$m = n-1$ no cycle
$m \geq n$ there is a cycle

1. a) What is the relationship between the number of vertices and the number of edges of a graph?

$E \leq \frac{V(V-1)}{2}$ where $E \to$ no. of Edges
$V \to$ no. of vertices

b) Prove the result you stated in part (a).

In a complete graph every pair of vertices is connected by a unique edge. So, number of edges is no. of total possible pair of vertices.

$\therefore$ Max no. of possible edges is ${}^{V}C_2$

$= \frac{v(v-1)}{2}$

$$\boxed{\therefore \; E \leq \frac{V(V-1)}{2}}$$

c) What is the relationship between odd cycles and bipartite graphs?

The graph is bipartite if and only if it does not contain any odd simple cycles.

d) Prove the result you stated in part C.

Assume $G = (V, E)$ be bipartite

Then, $G(V) = X \cup Y$, where $X \cap Y = \phi$

For every $e \in E(G)$, $e = (u, v)$ where $u \in X$, $v \in Y$

Let G has one oddcycle C of length n

$C = (v_1, v_2, \ldots, v_n, v_1)$, where n is odd

So, $v_k \in \begin{cases} X : k \text{ is odd} \\ Y : k \text{ is even} \end{cases}$

But as n is odd $v_n \in X$

$(v_1, v_n) \in E(G)$ is a contradiction. [since $v_1 \in X$ & $v_n \in X$]

Hence, G is bipartite if it has no odd cycles. proved //

e) Prove that if G is disconnected, G^c is connected.

Assume G is disconnected.

Then we have to show that G^c is connected. We need to show there is a path from u to v in G^c. (where u and v are vertices)

Case 1: (u,v) is not an edge in G. Then it is an edge in G^c. So, we have a path u-v from u to v in G^c.

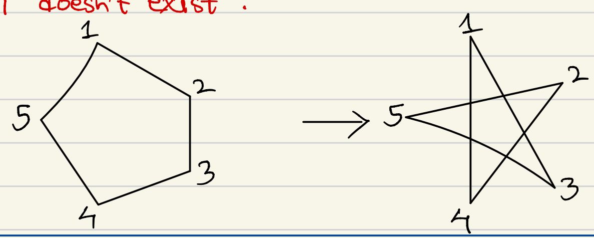
Case 2: (u,v) is an edge in G. This means that u and v are in the same component of G. Since G is disconnected, we can find a vertex w in different component such that

$$(u,w) \text{ and } (v,w) \notin E(G)$$

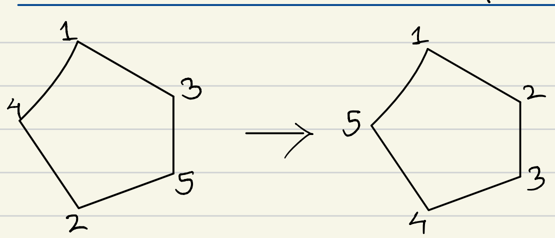
So, (u,w) and $(v,w) \in E(G^c)$

Thus, uwv is a path from u to v in G^c.

f) Draw a graph G with 5 or more vertices such that both G and G^c are connected. If such graph doesn't exist please write "G doesn't exist".



OR



2018 what I used

2. a) What are the properties of a Red-Black tree?

1. Every node is either red or black
2. Root node is always black.
3. Leaf nodes (NULL pointer) is always black.
4. If a node is red, both of it's children are black.
5. Every path from a node to it's descendant leaf has same no. of black nodes.

b) True or False: Number of Red Nodes $\leq$ Number of Black Nodes

True, [because every red node has 2 black children]
or 2 black nullpointer (leaf nodes)

c) True or False: The time complexity to build a n-node Red Black is $O(n)$

False: It is $O(\log n)$

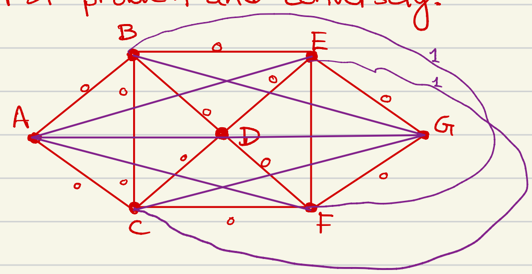
d) Write a non deterministic algorithm to search an "item" in an integer array. What is its time complexity?

Input: Array of Integers A, Item x
Output: Return true if x is present else return false

```
IsPresent (A, x)
    i ← guess(A, x)
    if (A[i] == x)
        return true
    else
        return false
```

The time complexity of above algorithm is $O(1)$.

e) Illustrate the proof that the Hamiltonian Cycle problem is polynomial reducable to TSP. By considering the following Hamiltonian graph - an instance of Hamitonian Cycle - and transforming it to a TSP instance in polynomial time so that a solution to the HC problem is a solution to the TSP problem and conversely.



Does it have HC?	Yes	No
Can TSP visit all city with cost 0?	Yes	No

Graph G has a hamiltonian cycle

Transforming G to a complete graph.

→ Assigned existing edges a cost of 0

→ Assigned added edges a cost of 1

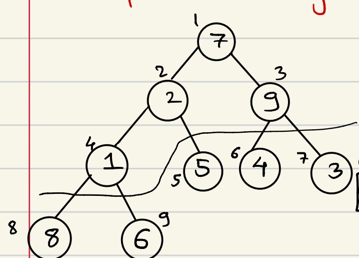
$$\text{Hamiltonian Cycle} \xrightarrow{\text{poly}} \text{TSP}$$

$k=0$

Analysing the TSP complete graph G, we can visit all the vertices at the cost 0 equivalent to k.

$$\therefore \text{Hamiltonian Cycle} \xrightarrow{\text{poly}} \text{TSP}$$

3. a) Heap sort [7,2,9,1,5,4,3,8,6] in ascending order using in-place bottom up iterative method. In this case, heap is maintained in an array as explained in your class notes.

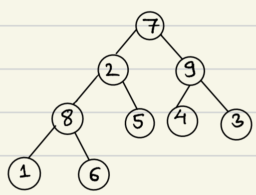


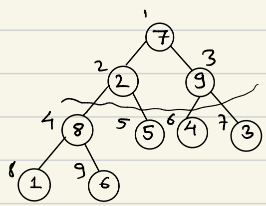
$\lceil n/2 \rceil = 5$ leaves

$\lfloor n/2 \rfloor = 4$ internal nodes

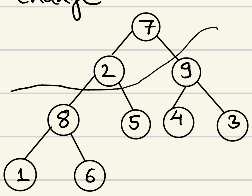
0	1	2	3	4	5	6	7	8	9
	7	2	9	1	5	4	3	8	6

i=4

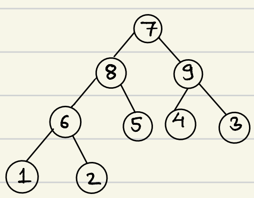


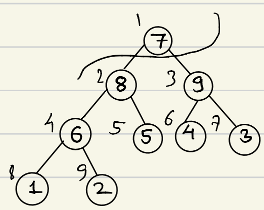


i=3 no change

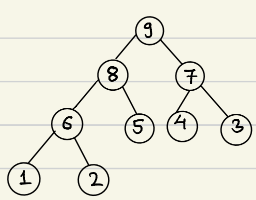


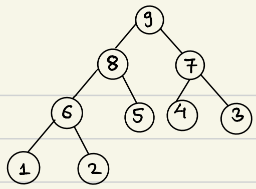
i=2



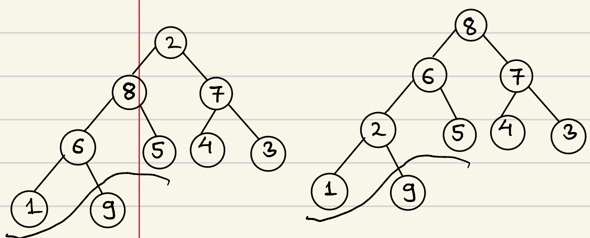


i=1

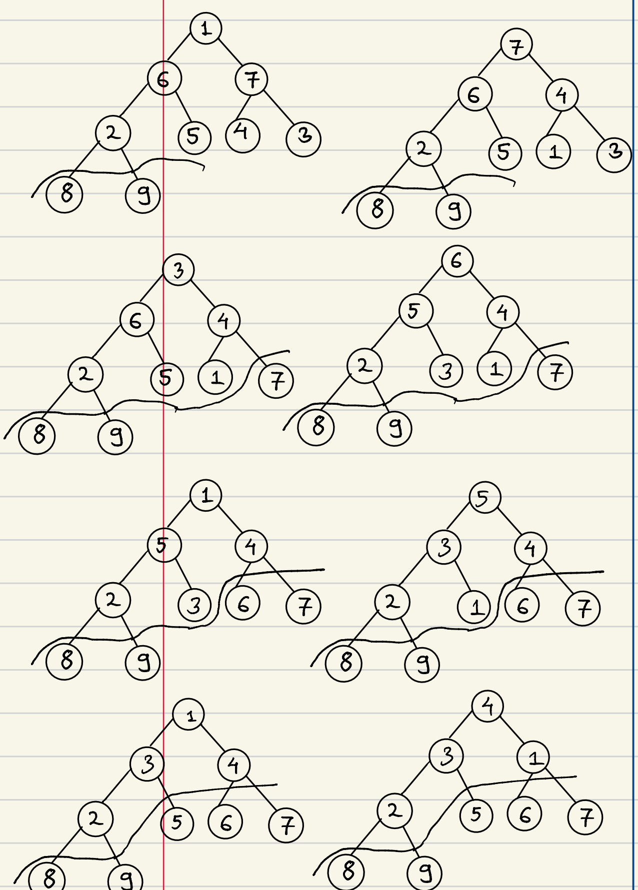


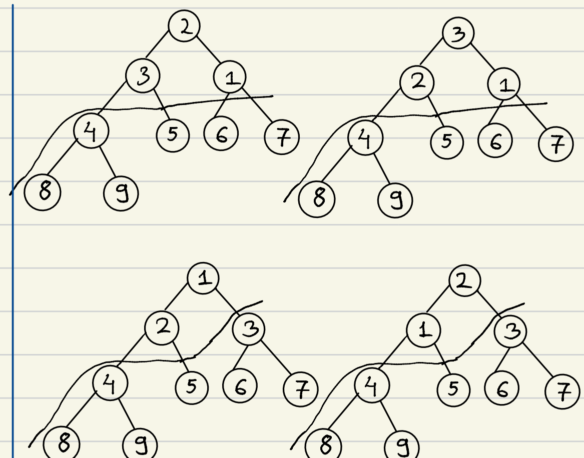


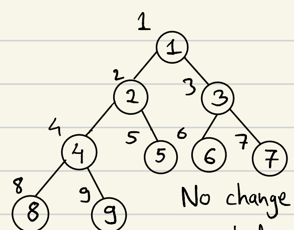
0	1	2	3	4	5	6	7	8	9
	9	8	7	6	5	4	3	2	1



0	1	2	3	4	5	6	7	8	9
	8	6	7	2	5	4	3	1	9







No change
sorted

0	1	2	3	4	5	6	7	8	9
	1	2	3	4	5	6	7	8	9

b) What is the time complexity to build a heap using in-place bottom-up iterative method?

$O(n) \rightarrow$ bottom up

$O(n\log n) \rightarrow$ top-down

c) Prove the result in b.

Assume $n = 2^{(h+1)} - 1$ where $n \rightarrow$ Number of nodes
$h \rightarrow$ Height of tree

Thus,

maximum no. of operations in worst case is

$$\sum_{j=0}^{h} j\,2^{h-j} = 2^h \sum_{j=0}^{h} j\,2^{-j}$$

Since, $\sum_{j=0}^{h} j2^{-j} < \sum_{j=0}^{\infty} j2^{-j} = 2$

$$S = \tfrac{1}{2} + \tfrac{2}{4} + \tfrac{3}{8} + \tfrac{4}{16} + \dots$$
$$\tfrac{1}{2}S = \tfrac{1}{4} + \tfrac{2}{8} + \tfrac{3}{16} + \dots$$
$$S - \tfrac{1}{2}S = \tfrac{1}{2} + \tfrac{1}{4} + \tfrac{1}{8} + \dots$$
$$\tfrac{1}{2}S = \frac{1/2}{1 - 1/2} = 1$$
$$\therefore S = 2$$

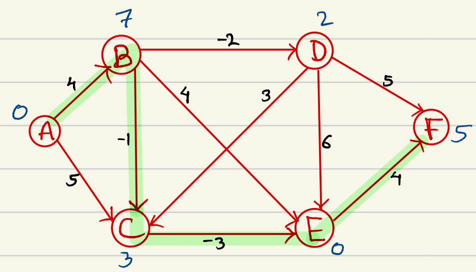
The maximum no. of operations in worst case is

$$\sum_{j=0}^{h} j\,2^{h-j} < 2^{h+1}$$

Since, $n = 2^{(h+1)} - 1$, $\sum_{j=0}^{h} j\,2^{h-j} < n+1 = O(n)$

4. a) Compute shortest path from A to F based on the adjacency matrix given below

	A	B	C	D	E	F
A	0	4	5	0	0	0
B	0	0	-1	-2	4	0
C	0	0	0	0	-3	0
D	0	0	3	0	6	5
E	0	0	0	0	0	4
F	0	0	0	0	0	0



Topological Order: ABDCEF

$$D[A] = 0$$

$$D[B] = D[A] + wt(A,B) = 0+4 = 7$$

$$D[D] = D[B] + wt(B,D) = 4-2 = 2$$

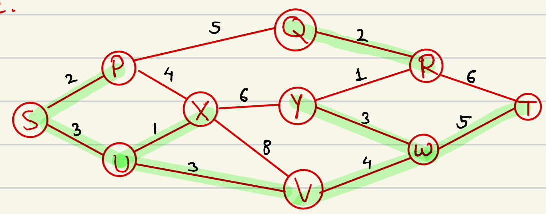
$$D[C] = \min \left\{ \begin{array}{l} D[A] + wt(A,C) = 0+5 = 5 \\ D[B] + wt(B,C) = 4-1 = 3 \\ D[D] + wt(D,C) = 2+3 = 5 \end{array} \right\} = 3$$

$$D[E] = \min \left\{ \begin{array}{l} D[C] + wt(C,E) = 3-3 = 0 \\ D[B] + wt(B,E) = 4+4 = 8 \\ D[D] + wt(D,E) = 2+6 = 8 \end{array} \right\} = 0$$

$$D[F] = \min \left\{ \begin{array}{l} D[D] + wt(D,F) = 0+5 = 5 \\ D[E] + wt(E,F) = 0+7 = 7 \end{array} \right\} = 5$$

$\therefore$ Shortest path A to F = {ABCEF}

b) Apply Kruskal's algorithm to the following graph to compute minimum spanning tree. Please show the minimum spanning tree.



(U,X)	(U,V)	(P,Q)	(X,V)
(Y,R)	(Y,W)	(W,T)	
(S,P)	(P,X)	(X,Y)	
(Q,R)	(V,W)	(R,T)	
(S,U)			

	S P U X Q Y V R W T
(U,X)	S P UX Q Y V R W T
(Y,R)	S P UX Q YR V W T
(S,P)	SP UX Q YR V W T
(Q,R)	SP UX QYR V W T
(S,U)	SPUX QYR V W T
(U,V)	SPUXV QYR W T
(Y,W)	SPUXV QYRW T
(P,X) → Cycle	
(V,W)	PQRSUVWXY T
(P,Q) → Cycle	
(W,T)	PQRSTUVWXY

Total Length = 1+1+2+2+3+3+3+4+5

= 24

5 a) What is the minimum number of edges required to guarantee that a graph on n vertices is connected (irrespective of which edges are present)

$$\varepsilon > {}^{(v-1)}C_2$$

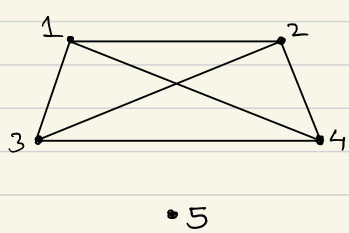
$$\Rightarrow \varepsilon > \frac{(v-1)(v-2)}{2}$$ where v is the no. of vertices

b) Prove the result (a).

$$\varepsilon > \frac{(v-1)(v-2)}{2}$$

Proof:-

Lets take the following graph of 5 vertices



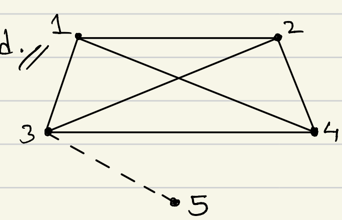
Here $v-1=4$

$${}^4C_2 = \frac{4\times3}{2} = 6$$

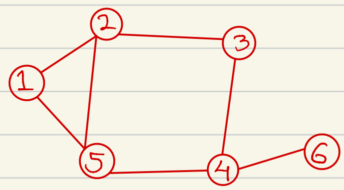
We have ${}^4C_2$ edges in this 5 vertices graph, but it's not connected.

So, the number of edges must be greater than ${}^4C_2=6$. If we add one more edge from any vertex, the graph will be connected.

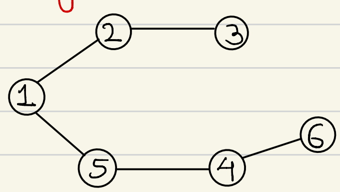
$\therefore\ \varepsilon > {}^{v-1}C_2$ is proved.//



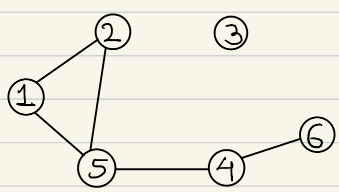
6. Consider the graph $G=(V,E)$

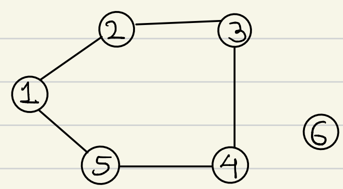


a) Give a spanning tree.



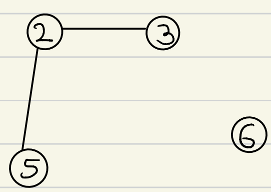
b) Give a spanning subgraph that is not a tree.





c) Let $W=\{2,3,5,6\}$. What is $G[W]$ ($G[W]$ is induced subgraph by G)

$G[W] =$



$G[W]$ is induced subgraph, which is a subset of the vertices of graph G together with any edges where endpoints are both in this subset.