Assignment - 2

Name: Md. Minhazul Mowla

ID: 23201390

Section: 13

Course: CSE 221

Submission Date: 5/5/25

Ans. to the Ques No.-1

1. No, A and B can not be called adjacent/neighbour to each other.

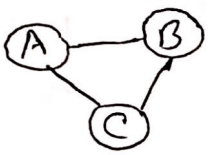
2. $\frac{n(n-1)}{2}$; n = no. of vencites

3. $n(n-1)$

4. $(n-1)$

5. $\frac{n(n-1)}{2}$

6. Connected graph and exactly 0 on 2 vertices having odd degree.

7. i. 

Possible → 1. A − C → C − B [∵ A-B ignored]
2. B − C → C − A [∵ B-A ignored]

ii. 

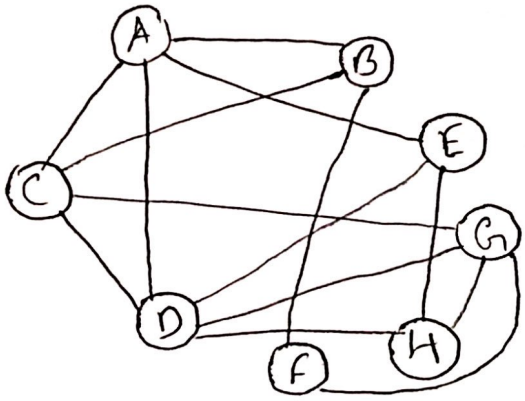
Impossible

8.

Graphs	Adjacency List	Adjacency matrix
Undirected	stores neighbor edges twice	stores edges symmetrically
Directed	stores outgoing edges only	Using 1's for edges
Weighted	Stores neighbor and weight-pairs	stores weights only
Unweighted	stores neighbors only	stores 0/1
Sparse	saves spaces	wastes space
Dense	Inefficient	Efficient

Ans. to the Ques. No. 2

A



B

A: B, C, D, E

B: A, C, F

C: A, B, D, G

D: A, C, E, G, H

E: A, D, H

F: B, G

G: C, D, F, H

H: D, E, G

Adjacency List

	A	B	C	D	E	F	G	H
A	0	1	1	1	1	0	0	0
B	1	0	1	0	0	1	0	0
C	1	1	0	1	0	0	1	0
D	1	0	1	0	1	0	1	1
E	1	0	0	1	0	0	0	1
F	0	1	0	0	0	0	1	0
G	0	0	1	1	0	1	0	1
H	0	0	0	1	1	0	1	0

Adjacency matrix

C

i. (A, B) → C (1)

ii. (A, C) → B, D (2)

iii. (A, D) → C, E (2)

iv. (A, E) → D (1)

~~v. (A, F)~~

~~v. (B, C) → A (1)~~

vi

v. (A, F) → B (1)

vi. (A, G) → C, D (2)

vii. (A, H) → D, E (2)

viii. (B, C) → A (1)

ix. (B, D) → C (1)

x. (B, E) → None (0)

xi. (B, F) → A (1)

xii. (B, G) → C, F (2)

xiii. (B, H) → None (0)

xiv. (C, D) → A, G (2)

xv. (C, E) → D (1)

xvi. (C, F) → G (1)

xvii. (C, G) → D, F (2)

xviii. (C, H) → D, G (2)

xix. (D, E) → A, H (2)

xx. (D, F) → G (1)

xxi. (D, G) → C, H (2)

xxii. (D, H) → E, G (2)

xxiii. (E, F) → None (0)

xxiv. (E, G) → D, H (2)

xxv. (E, H) → D, G (2)

xxvi. (F, G) → B, D (2)

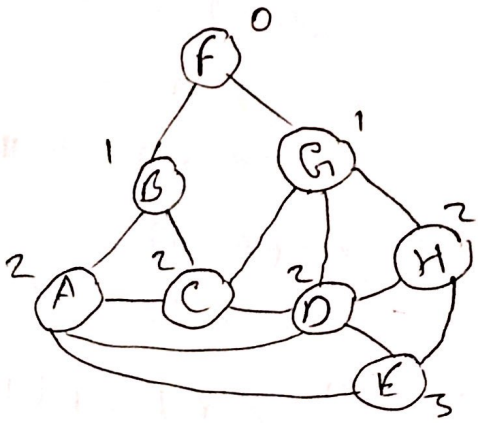
xxvii. (F, H) → G (1)

xxviii. (G, H) → D, E (2)

D

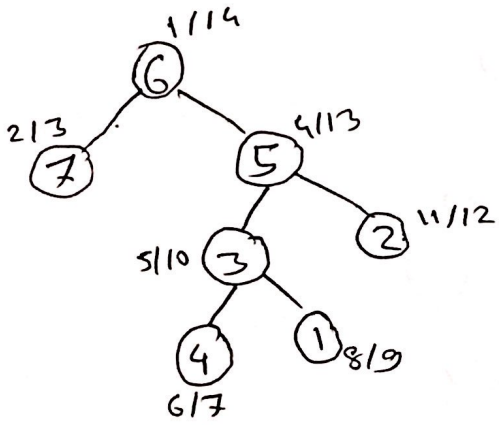
F, B, G, A, C, D, H, E : [Through BFS]

1 degree — 2 degree — 3 degree



Ans. to the Ques. No. - 3

A



B

Nodes	1	2	3	4	5	6	7
Parent	3	5	5	3	6	-	6
Starting Time	8	11	5	6	4	1	2
Finish Time	9	12	10	7	13	14	3
Distance from root	3	2	2	3	1	0	1

Ans. to the Ques. No.-4

A

A: B, C, S (3)

B: A, D, E, S (4)

C: ~~A, C, E~~ A, D, F, G (4)

D: B, C, E (3)

E: B, D, G (3)

F: C, G, H (3)

G: C, E, F, H (4)

H: F, G (2)

S: ~~Σ(A, B)~~ A, B (2)

$$\sum_{v \in V} deg(v) = 3+4+4+3+3+3+4+2+2$$

$$= 28$$

m = No. of Edges = SA, SB, AB, AC, BD, BE, CD, CE, CG, CF, DG, EG, FG, FH, GH (14)

~~= 14~~.

$\therefore 2m = 14 \times 2 = 28$

$\therefore \sum_{v \in V} deg(v) = 2m$

[Justified]

B

No. of nodes = 9

No. of edges = 14

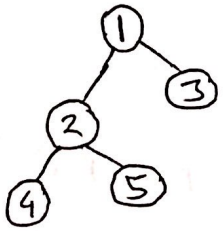
Maximum no. of edges = $\frac{9 \times (9-1)}{2} = 36$

∴ No. edges can be added = 36 - 14

= 22

Ans. to the Ques. No. - 5

i



Adjacency list

1: [2, 3]

2: [4, 5]

3: []

4: []

5: []

Python

```
N = int(input())
adj = [ ]
for i in range(N+1):
    adj += [[]]
for i in range(2, N+1):
    x = int(input())
    adj[x] += [i]
s = [0] * [N+1]
def dfs(j):
    s[j] = 1
    for i in adj[j]:
        dfs(i)
        s[j] += s[i]
dfs(1)
Q = int(input())
for i in range(Q):
    p = int(input())
    print(s[p])
```

Ans. to the Ques. No. - 6

Python

```
from collections import deque

def fun(v, e):
    rev = []
    for i in range(v+1):
        rev += [[]]
    for x,y in e:
        rev(v) += [x]

    vis = [False] * (v+1)
    q = deque([1])
    vis[1] = True
    c = 1
    while q:
        z = q.popleft()
        for i in rev[z]:
            if vis[i] == False:
                vis[i] = True
                c += 1
                q += [i]
    return c == v

v = 4
E = [(2,1), (3,1), (4,3)]
print(fun(v, E))
```