19.02.24

26/02/2024

WEEK-09

Q:- 1) BFS

2) DFS

3) Delete a node in BST (Leet Code)

4) Bottom left tree value (LeetCode)

1) I/p:-

```
void bfs (int a[10][10], int n, int u)
{
    int f, r, q[10], v;
    int s[10] = {0};
    printf("Nodes visited from %d: ", u);
    f=0;
    r=-1;
    ~~q[++r]=u;~~
    ~~s[u]=1;~~
    printf("%d", u);
while( f<=r)
{   u=q[f++];
    for (v=0; v<n; v++)
    { if ( a[u][v]==1)
        {
                if (s[v]==0)
                {
                        printf("%d ", v);
                        s[v]=1;
                        q[++r]=v;
                }
        }
    }
}
printf("\n");
}
```

O/P:-

```
enter no. of nodes : 7
Enter the adjacency matrix: 0 111100
1 0 0 1 0 1 0
1 0 0 0 0 0 1
1 1 0 0 0 1 0
1 0 0 0 0 0 1
0 1 0 1 0 0 0
0 0 1 0 1 0 0
Nodes visited from 0: 0 1 2 3 4 5 6
                   1: 1 0 3 5 2 4 6
                   2: 2 0 6 1 3 4 5
                   3: 3 0 1 5 2 4 6
                   4: ~~5 1 3 0 2 4 6~~ 4 0 6 1 2 3 5
                   5: 5 1 3 0 2 4 6
6: 6 2 4 0 1 3 5
```

2) DFS =>

I/pt

```
void dfs (int v)
{
    int i;
    s[v] =1;
    for (i=1; i<=n; i++) {
    if (a[v][i]. && !s[i])
    {
        printf("\n %d -> %d", v, i);
        dfs (i);
    }
  }
}

if (count ==n)
    printf("Graph is connected");
else printf("Not connected");
```

O/pt

```
Enter the no. of vertices : 4
Enter the adjacency matrix : 0 1 0 1
  1 0 1 0
  0 1 0 1
  1 0 1 0

Graph is connected.
```

26.02.24