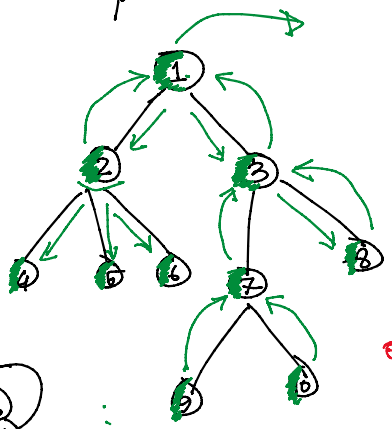
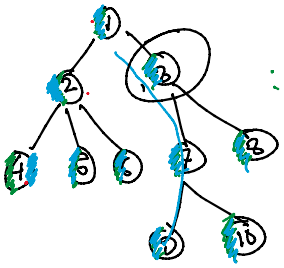
DFS

Friday, September 11, 2020 9:59 PM

DFS → Depth First Search



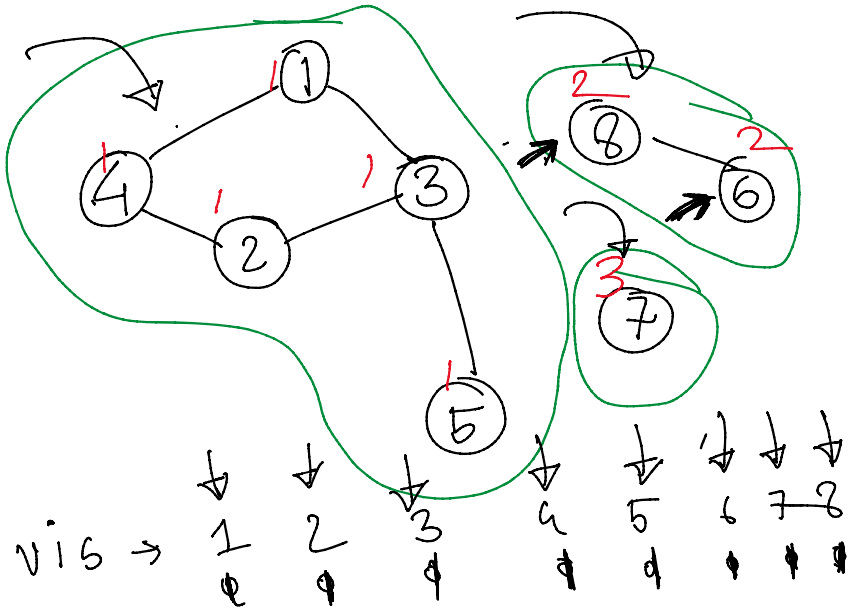


Pre → 1, 2, 4, 5, 6, 3, 7, 9, 10, 8

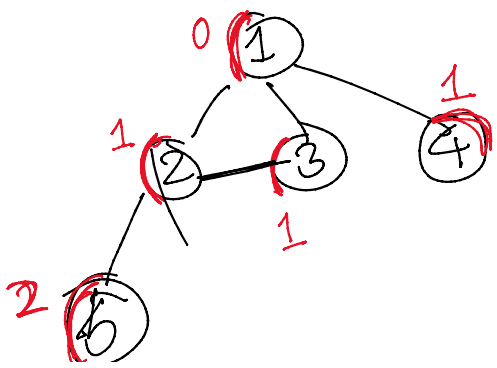
In → 4, 2, 5, 6, 1, 9, 7, 10, 3, 8

Application

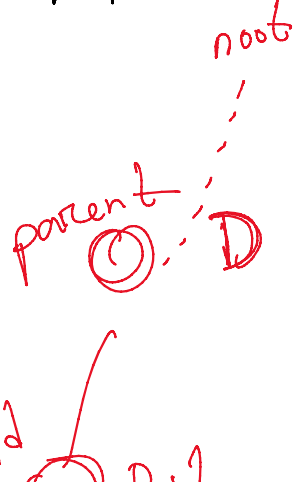
1) Traverse
2) Finding Path
3) Count component
4) Cycle graph



vis → 1 2 3 4 5 6 7 8



1 ... 3

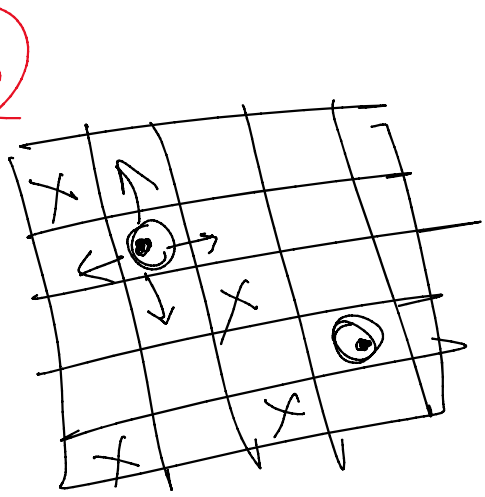


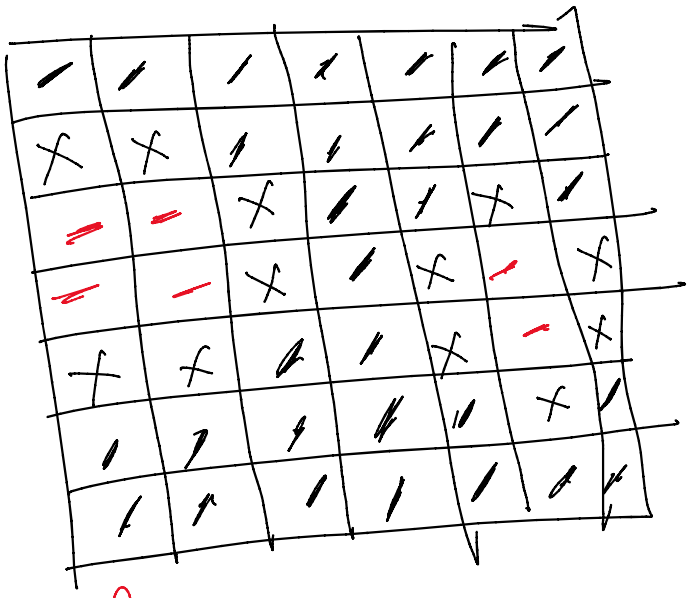
2 ⑥

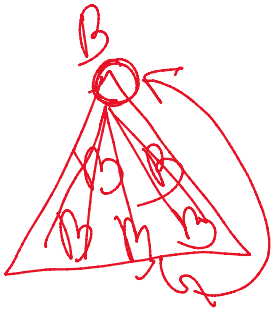
child ○ D+1

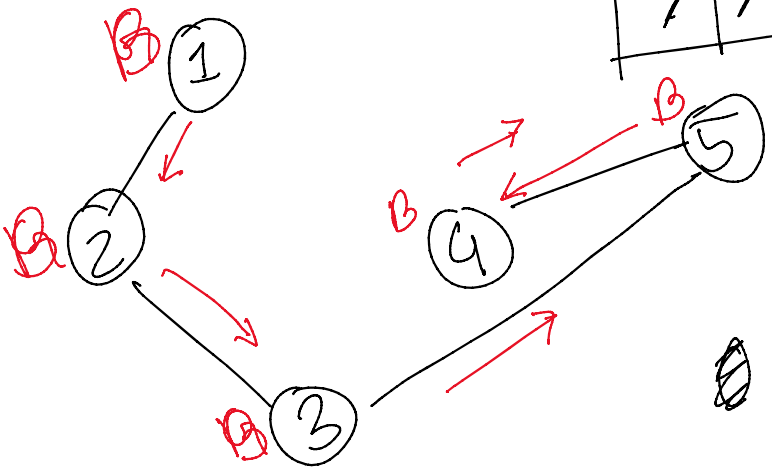
1 → 3 ①

1st



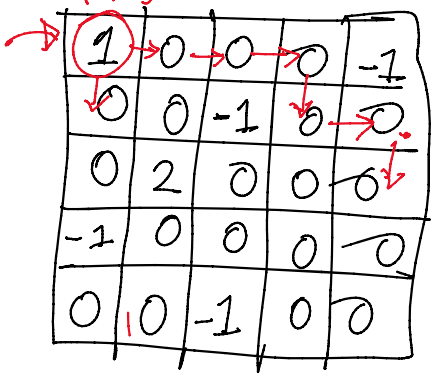






W G B

5 5



dfs (0,0)

↓

dfs (0,1)

↓

dfs (0,2)

↓

dfs (0,3) → dfs (1,3)

↑

dfs (1,4)

```
void dfs(x, y)
{
    vis[x][y] = 1;
    for(i=0; i<4; i++)
    {
        X = x+dx[i]
        Y = y+dy[i]
    }
}
```

x+0, y+1

x+1, y+0

x+0, y-1

x-1, y+0

dx[] = {+0, +1, +0, -1}

dy[] = {+1, +0, -1, +0}