Week 4 :)

23/05/2024

PAGE NO :
DATE :

Topological Sorting

```
#include<stdio.h>

void main(){
    int n, a[30][30], i, j, sum, in[30], s[30],
                t[30], k=0;
    printf("Enter no of vertices: ");
    scanf("%d", &n);
    printf("Enter adjacency matrix: \n");
    for(i=0; i<n; i++){
        for(j=0; j<n; j++)
        {
            scanf("%d", &a[i][j]);
        }
    }
    for(j=0; j<n; j++){
        sum=0;
        for(i=0; i<n; i++){
            sum+=a[i][j];
        }
        in[j]=sum;
    }
    int top=-1;
    for(i=0; i<n; i++){
        if(in[i]==0){
            top++;
            s[top]=i;
        }
    }
    while(top!=-1){
        int u=s[top];
        top--;
        t[k++]=u;
```

```
for (int i=0; i<n; i++) {
    if(a[u][i] == 1) {
        in[i]--;
        if(in[i]==0) {
            top++;
            s[top] = i;
        }
    }
}
printf("Sequence: ");
for(i=0; i<n; i++)
    printf("%d", s[i]);
}
```

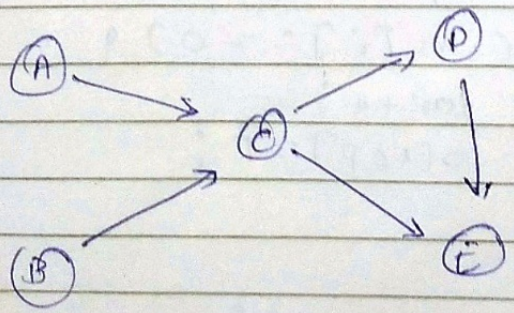
30/5/24

Output :-

Enter no of vertices : 5

Enter Adjacency matrix :

0 0 1 0 0
0 0 1 0 0
0 0 0 1 1
0 0 0 0 1
0 0 0 0 0



Sequence :- 1 0 2 3 4

PAGE NO :
DATE :

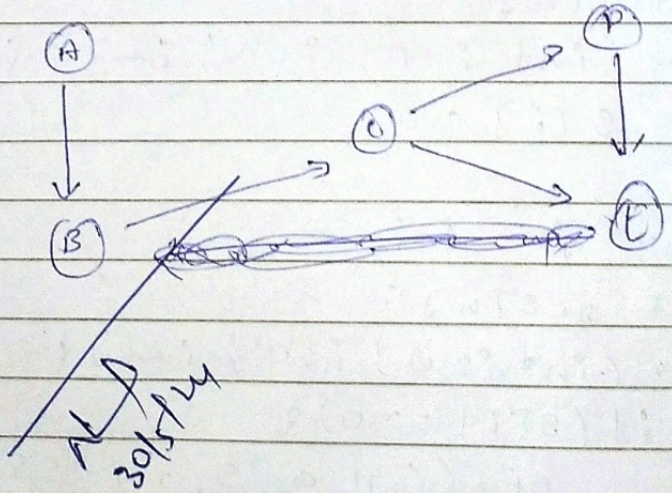
Topological sort using DFS

```
#include <stdio.h>
#include <stdio.h>

void DFS(int u, int n, int a[n][n], int s[],
         int *j, int res[]) {
    s[u] = 1;
    for(int v=0; v<n; v++) {
        if(a[u][v]==1 && s[v]==0) {
            DFS(v, n, a, s, j, res);
        }
    }
    (*j)++;
    res(*j) = u;
}

void topological_order(int n, int a[n][n]){
    int s[n];
    for(int i=0; i<n; i++) {
        s[i] = 0;
    }
    int j = -1;
    int res[n];
    for(int i=0; i<n; i++) {
        if(s[i]==0){
            DFS(i, n, a, s, j, res);
        }
    }
    for(int i=n-1; i>=0; i--) {
        printf("%d", res[i]);
    }
    printf("\n");
}
```

```
int main() {
    int adjacency_matrix[5][5] =
    {
        {0, 1, 0, 0, 0},
        {0, 0, 1, 0, 0},
        {0, 0, 0, 1, 1},
        {0, 0, 0, 0, 1},
        {0, 0, 0, 0, 0}
    };
    int num_vertices = 5;
    topological_order(num_vertices,
                      adjacency_matrix);
    return 0;
}
```

Output :-

Sequence : 0 1 2 3 4



ALP 30/5/24