

2. Write program to obtain the Topological ordering of vertices in a given digraph.

**Code:**

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

void main(){
    int a[10][10],n,i,j;
    int indeg[10],flag[10],c=0;

    printf("Enter number of vertices \n");
    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(i=0;i<n;i++)
        indeg[i]=0;

    for(i=0;i<n;i++)
        flag[i]=0;

    for(i=0;i<n;i++)
        for(j=0;j<n;j++)
            if(a[i][j]==1)
                indeg[j]+=1;

    printf("Order is : ");
    while(c<=n)
    {
        for(i=0;i<n;i++)
        {
            if(indeg[i]==0 && flag[i]==0)
            {
                printf("%d  ",i+1);
                flag[i]=1;
            }
        }
        for(i=0;i<n;i++)
        {
            if(flag[i]==1)
            {
```

```

        for(j=0;j<n;j++)
        {
            if(a[i][j]==1)
            {
                indeg[j]-=1;
                a[i][j]=0;
            }
        }
    }
    c++;
}
}
}

```

Output:

```

PS D:\VS Code\OS> cd "d:\VS Code\OS\" ; if ($?) { gcc ada.c -o ada } ; if ($?) { .\ada }
Enter number 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
Order is : 1 2 3 4 5

```

Observation :

Write Program to find the Topological ordering  
vertices in a given digraph

AB-2

Hint: <Stack>  
Hint: <Queue>

Code

```
int n[10], indeg[10], pos[10], c=0;

printf("Enter number of vertices 'n':\n");
scanf("%d", &n);

printf("Enter adjacency matrix 'n':\n");
for(i=0; i<n; i++)
    for(j=0; j<n; j++)
        scanf("%d", &a[i][j]);

for(i=0; i<n; i++)
    indeg[i]=0;

for(i=0; i<n; i++)
    pos[i]=0;

for(i=0; i<n; i++)
    for(j=0; j<n; j++)
        if(a[i][j]==1)
            indeg[j]++;
```

```

print("Order is : ")
while (c <= n)
{
    for (i = 1; i <= n; i++)
    {
        if (indeg[i] == 0 || outdeg[i] == 0)
        {
            print(" ", i, " ");
            outdeg[i] = 1;
        }
    }
    for (i = 0; i <= n; i++)
    {
        if (indeg[i] == 1)
        {
            for (j = 0; j <= n; j++)
            {
                if (a[i][j] == 1)
                {
                    indeg[j] = 1;
                    a[i][j] = 0;
                }
            }
        }
    }
    c++;
}

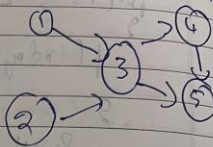
```

Output

Enter the no. of vertices: 5

Enter the adjacency matrix of graph

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



The topological ordering of vertices

2 1 3 4 5

~~Don't~~

