TOPOLOGICAL SORTING

```
#include<stdio.h>
#include<math.h>
int front=-1, rear=-1;
void push(int)
int pop();
int st[10];
int adj[10][10]
int indegree[10];
int t[10];
int k
int n

Void push(int x)
{
   if(front==-1 && rear==-1)
      front=rear=0

   else if(rear==n-1)
      return

   else
   {
      rear++;
   }
   st[rear]=x;

}
```

```
int pop()
{
    int val;
    if (front == -1 || front > rear)
    {
        return -1;
    }
    val = st[front];
    if (front == rear || front > rear)
    {
        front = -1;
        rear = -1;
    }
    else
    {
        front++;
    }
    return val;
}

int main()
{
    int sum=0;
    printf("Enter number of vertices");
    scanf("%d", &n);
    printf("Enter the adjacency matrix\n");

    for(int i=0; i<n; i++)
    {
        for(int j=0; j<n; j++)
```

```
    {
        scanf("%d", &adj[i][j]);
    }
}
for(int i=0; i<n; i++)
{
    sum=0;
    for(int j=0; j<n; j++)
    {
        sum = sum + adj[i][i]
    }
    indegree[i] = sum;
}
for(int i=0; i<n; i++)
{
    if(indegree[i] == 0)
        push(i);
}
while(front != -1)
{
    int u = pop()
    if(u == -1)
        break;
    t[k] = u
    k++;
```

```
for(int j=0; j<n; j++)
    if (adj[v][i] == 1)
        indegree[j] --
        if (indegree[j] == 0)
            push(j);
    }
}
printf("Final Solution:");
for(int i=0; i<k; i++)
    printf("%d\t", t[i]);
}
}
```