```c
//Topological using source removal
#include<stdio.h>
#include<stdlib.h>
#include<time.h>
int a[10][10],indegree[10],front=-1,rear=-1,stack[10],t[10],k,n;
void get(int n){
    int i,j;
    printf("Enter the adjacency matrix :\n");
    for(i=0;i<n;i++){
        printf("Enter row %d\n",(i+1));
        for(j=0;j<n;j++)
            scanf("%d",&a[i][j]);
    }
}
void cal_indegree(int n){
    int i,j;
    for(i=0;i<n;i++){
        for(j=0;j<n;j++){
            indegree[i]=indegree[i]+a[j][i];
        }
    }
}
void push(int x){
    if(front==-1&&rear==-1)
        front=rear=0;
    else if(rear==n-1)
        return;
    else{
        rear++;
    }
    stack[rear]=x;
}
int pop()
```

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

}
int main(){
    int i,v;
    printf("Enter the no of vertices :\n");
    scanf("%d",&n);
    get(n);
    for(i=0;i<n;i++){
        indegree[i]=0;
    }
    double t_time=0.0;
    clock_t begin=clock();
    cal_indegree(n);
    printf("\nThe topological order is : ");
    for(int i=0;i<n;i++){
        if(indegree[i]==0){
            push(i);
        }
    }
```

```
668     while(front!=-1){
669         int u=pop();
670         if(u==-1)
671             break;
672         t[k]=u;
673         k++;
674         for(v=0;v<n;v++){
675             if(a[u][v]==1){
676                 ((indegree[v])--);
677                 if(indegree[v]==0){
678                     push(v);
679                 }
680             }
681         }
682     }
683     clock_t end=clock();
684     t_time+=(double)(end-begin)/CLOCKS_PER_SEC;
685     printf("Topological Order :\n");
686     for(int i=0;i<k;i++){
687         printf("%d\t",t[i]);
688     }
689     printf("\nFor matrix n=%d\tTime:%f\n",n,t_time);
690     return 0;
691 }
692
```

```
Enter the no of vertices :
6
Enter the adjacency matrix :
Enter row 1
0 0 0 0 0 0
Enter row 2
0 0 0 0 0 0
Enter row 3
0 0 0 1 0 0
Enter row 4
0 1 0 0 0 0
Enter row 5
1 1 0 0 0 0
Enter row 6
1 0 1 0 0 0

The topological order is : Topological Order :
4       5       0       2       3       1
For matrix n=6  Time:0.000018


...Program finished with exit code 0
Press ENTER to exit console.
```