```c
#include <stdio.h>
void main(){
    int i,j,k,n,a[10][10],indeg[10],flag[10],count=0;
    printf("Enter the no of vertices:\n");
    scanf("%d",&n);
    printf("Enter the 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;
        flag[i]=0;
    }
    for(i=0;i<n;i++)
        for(j=0;j<n;j++)
            indeg[i]=indeg[i]+a[j][i];
    printf("\nThe topological order is:");
    while(count<n){
        for(k=0;k<n;k++){
            if((indeg[k]==0) && (flag[k]==0)){
                printf("%d ",(k+1));
                flag [k]=1;
            }
            for(i=0;i<n;i++){
                if(a[i][k]==1)
                    indeg[k]--;
            }
        }
        count++;
    }
}
```

× Output

Enter the no of vertices:
>>>5
Enter the 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
...

The topological order is:1 2 3 4 5

```c
1   #include<stdio.h>
2   int a[20][20],q[20],visited[20],n,i,j,f=0,r=-1;
3   void bfs(int v) {
4       for (i=1;i<=n;i++)
5           if(a[v][i] && !visited[i])
6               q[++r]=i;
7       if(f<=r) {
8           visited[q[f]]=1;
9           bfs(q[f++]);
10      }
11  }
12  void main() {
13      int v;
14      printf("Enter the Number of Vertices:");
15      scanf("%d",&n);
16      for (i=1;i<=n;i++) {
17          q[i]=0;
18          visited[i]=0;
19      }
20      printf("\nEnter Adjacency Matrix:\n");
21      for (i=1;i<=n;i++)
22          for (j=1;j<=n;j++)
23              scanf("%d",&a[i][j]);
24      printf("\nEnter the Source Node:");
25      scanf("%d",&v);
26      bfs(v);
27      printf("\nThe Node which are reachable are:\n");
28      for (i=1;i<=n;i++)
29          if(visited[i])
30              printf("%d\t",i);
31  }
32
```

X Output

Enter the Number of Vertices:>>>5

Enter Adjacency Matrix:
>>>0 1 1 1 0
... 0 0 1 0 0
... 0 1 0 0 0
... 0 0 0 1 0
... 0 1 0 0 0

...

Enter the Source Node:>>>1

The Node which are reachable are:
2   3   4