LAB-2

Q1. Write a program to obtain topology ordering of vertices in a given diagraph.

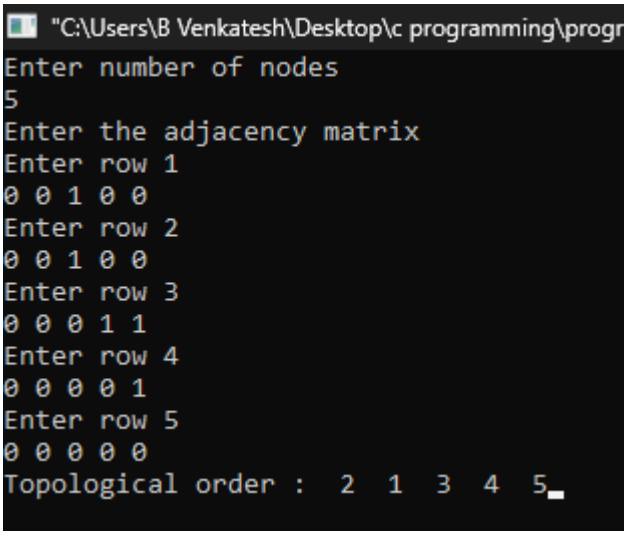
Aim: To write a c program for topological sort algorithm.

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

void source_removal(int n, int a[10][10]) {
   int i,j,k,u,v,top,s[10],t[10],indeg[10],sum;
   for(i=0;i<n;i++) {
      sum=0;
      for(j=0;j<n;j++)
         sum+=a[j][i];
      indeg[i]=sum;
   }
   top=-1;
   for(i=0;i<n;i++) {
      if(indeg[i]==0)
         s[++top]=i;
   }
   k=0;
   while(top!=-1) {
      u=s[top--];
      t[k++]=u;
      for(v=0;v<n;v++) {
         if(a[u][v]==1) {
            indeg[v]=indeg[v]-1;
            if(indeg[v]==0)
               s[++top]=v;
         }
      }
   }
   printf("Topological order :");
   for(i=0;i<n;i++)
      printf("  %d", t[i]);
}

void main() {
int i,j,a[10][10],n;
printf("Enter number of nodes\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]);
source_removal(n,a);
getch();
}
```

RESULT:


"C:\Users\B Venkatesh\Desktop\c programming\progr
Enter number of nodes
5
Enter the adjacency matrix
Enter row 1
0 0 1 0 0
Enter row 2
0 0 1 0 0
Enter row 3
0 0 0 1 1
Enter row 4
0 0 0 0 1
Enter row 5
0 0 0 0 0
Topological order : 2 1 3 4 5

22/6/23 LAB-2

Q] Write a program to obtain topological ordering of vertices in a given digraph.

Soln

```
#include<stdio.h>
#include<conio.h>
void todo-degree(int n, int a[10][10])
{
 int i, j, k, u, v, top, s[10], t[10], indeg[10], sum;
 for (i=0;i<n; i++)
 { sum=0;
 for(j=0; j<n; j++)
    sum+=a[j][i];
   indeg[i] =sum;
  }
 top=-1;
 for (i=0; i<n; i++)
  {
   if (indeg[i]==0)
      s[++top]=i;
  }
  k=0;
 while (top!=-1) {
   u=s[top--];
   t[k++]=u;
   for (v=0; v<n; v++){
   if(a[u][v]==1){
     indeg[v]=indeg[v]-1;
     if (indeg[v]==0)
        s[++top]=v;
   }
  }
 printf("Topological order:");
  for (i=0; i<n; i++)
    printf(" %d ", (t[i]+1));
 }
void main()
{
 int i, j, a[10][10], n;
 printf("Enter number of nodes\n");
 scanf("%d", &n);
 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]);
  }
 todo-degree(n,a);
 getch();
}
```

Result:-

Enter number of nodes

5

- Enter the adjacency matrix

Enter ROW 1

0 0 1 0 0

Enter ROW 2

0 0 1 0 0

Enter ROW 3

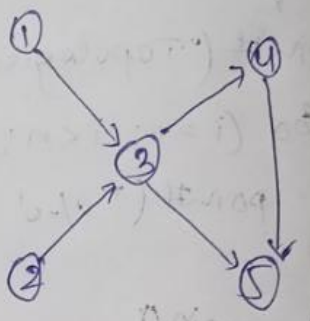
0 0 0 1 1

Enter ROW 4

0 0 0 0 1

Enter ROW 5

0 0 0 0 0



$$\begin{bmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Topological order of the given adjacency matrix is : 2 1 3 4 5

Ex-1
22/9/23