

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

int main ()
{
    int i, j, K, n, a[10][10], indeg[10], flag[10], count = 0;
    clock_t start, end;

    double time_u;

    printf ("enter the no. of vertices");
    scanf ("%d", &n);

    printf ("Enter the adjacency matrix\n");
    start = clock();

    for (i = 0; i < n; i++)
    {
        printf ("enter row %d\n", i+1);
        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[j] = indeg[j] + a[j][i];

    end = clock();
```

time_u = (double)(end - start) / CLOCKS_PER_SEC;

printf ("Time complexity = %fs\n", time_u);

printf ("In the topological order is : %d", count);

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++;

}

// time complexity //.

return 0;

time complexity