Topological Sort (DFS-based)

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

#define MAX 100

int adj[MAX][MAX], visited[MAX], stack[MAX], top = -1, n;

void dfs(int v) {
    visited[v] = 1;
    for (int i = 0; i < n; i++)
        if (adj[v][i] && !visited[i])
            dfs(i);
    stack[++top] = v;
}

int main() {
    int edges, u, v;
    printf("Enter number of vertices: ");
    scanf("%d", &n);
    printf("Enter number of edges: ");
    scanf("%d", &edges);
    for (int i = 0; i < edges; i++) {
        printf("Enter edge (u v): ");
        scanf("%d %d", &u, &v);
        adj[u][v] = 1;
    }

    for (int i = 0; i < n; i++)
        if (!visited[i])
            dfs(i);

    printf("Topological order: ");
    while (top >= 0)
        printf("%d ", stack[top--]);
    printf("\n");
    return 0;
}
```

OUTPUT:

```
Enter number of vertices: 5
Enter number of edges: 5
Enter edge (u v): 11
22
Enter edge (u v): 33
44
Enter edge (u v): 55
66
Enter edge (u v): 77
88
Enter edge (u v): 99
10
Topological order: 4 3 2 1 0

Process returned 0 (0x0)   execution time : 56.455 s
Press any key to continue.
```