```cpp
#include"ALGraph.h"
#include"StackAndQueue.h"

//入度数组，存放每个顶点的入度
int indegree[MAX_VERTEX_NUM];

//对有向图进行拓扑排序，图G采用邻接表法的存储方式
//图中无回路，返回OK，并输出一条拓扑排序序列，若有回路则返回ERROR
Status TopologicalSort(ALGraph G){
    for(int i=0; i<G.vexnum; i++)                //求每个顶点的入度
        indegree[i] = FindInDegree(G, i);

    SqStack S;
    InitStack_Sq(S);
    for(int i=0; i<G.vexnum; i++)
        if(indegree[i]==0)
            Push_Sq(S, i);                  //入度为0者进栈

    int count = 0;  //对输出顶点计数
    int i,k;
    ArcNode *p;
    while(!StackEmpty_Sq(S)){    //栈不为空
        Pop_Sq(S, i);
        //输出第i个顶点并计数
        cout << i << ": " << G.vertices[i].data << endl;
        count ++;
        for(p=G.vertices[i].firstarc; p; p=p->nextarc){
            k = p->adjvex;
            if(0==(--indegree[k]))  //k的入度不为0，k入栈
                Push_Sq(S, k);
        }
    }
    if(count<G.vexnum)
        return ERROR;
    else
        return OK;
}


int main(){
    ALGraph G;
    CreateGraph(G);
    ALGraphShow(G);
    TopologicalSort(G);
    return 0;
}

```