/\*\*

\* ToplogicalSort.c 拓扑排序算法

\* @author Darbuly 970073804@qq.con

\* @copylight Darbuly 2018-2019

\*

\*/

#include <stdio.h>

#include <stdlib.h>

#define VRType int

#define InfoType int

#define VertexType char

#define OK 1

#define ERROR 0

#define MAX\_VERTEX\_NUM 20

typedef int Status;

typedef enum{DG,DN,AG,AN}GraphKind;

typedef struct ArcNode{

int adjvex;

struct ArcNode \*nextarc;

InfoType \*info;

}ArcNode;

typedef struct VNode

{

VertexType data;

int in;

ArcNode \*firstarc;

}VNode,AdjList[MAX\_VERTEX\_NUM];

typedef struct

{

AdjList vertices;

int vexnum,arcnum;

GraphKind kind;

}ALGraph;

void InsertNode(ALGraph \*G,int i, int j)

{

printf("\ni=%d,j=%d\n",i,j);

ArcNode \*ptrArcNode = (ArcNode\*)malloc(sizeof(ArcNode));

ptrArcNode->adjvex = j;

ptrArcNode->nextarc = G->vertices[i].firstarc;

G->vertices[i].firstarc = ptrArcNode;

G->vertices[j].in++;

printf("%c-%c\n",G->vertices[i].data,G->vertices[j].data);

}

void CreateALGraph(ALGraph \*G)

{

int i,j,k;

if(G==NULL) G = (ALGraph\*)malloc(sizeof(ALGraph));

printf("Please input the vexnum and arcnum:\n");

scanf("%d%d",&G->vexnum,&G->arcnum);

getchar();

printf("Please input the Vertices:\n");

for(i=0;i<G->vexnum;i++)

{

printf("Vertex %d:",i+1);

scanf("%c",&G->vertices[i].data);

G->vertices[i].in = 0;

G->vertices[i].firstarc=NULL;

getchar();

}

for(k=0;k<G->arcnum;k++)

{

printf("Please input (vi,vj) vertex:\n");

scanf("%d%d",&i,&j);

InsertNode(G,i,j);

}

}

void ShowALGraph(ALGraph \*G)

{

ArcNode \*ptrArcNode;

int i;

for(i=0;i<G->vexnum;i++)

{

printf("The Vertex %d is %c ,in %d:",i+1,G->vertices[i].data,G->vertices[i].in);

ptrArcNode = G->vertices[i].firstarc;

while(1)

{

if(ptrArcNode ==NULL)break;

printf("->%c",G->vertices[ptrArcNode->adjvex]);

if(ptrArcNode->nextarc==NULL)break;

ptrArcNode = ptrArcNode->nextarc;

}

printf("\n");

}

}

Status ToplogicalSort(ALGraph G)

{

ArcNode \*e;

int i,k,gettop;

int top = 0;

int count = 0;

int \*stack;

printf("\nToplogicalSort:\n");

stack = (int \*)malloc(G.vexnum\*sizeof(int));

for(i=0;i<G.vexnum;i++)

{

if(0==G.vertices[i].in)

{

// printf("The Head is %d\n",i);

stack[++top] = i;

}

}

while(0!=top)

{

gettop = stack[top--];

printf("%c -> ",G.vertices[gettop].data);

count++;

for(e=G.vertices[gettop].firstarc;e;e=e->nextarc)

{

k = e->adjvex;

if(!(--G.vertices[k].in))

{

stack[++top]=k;

}

}

}

if(count<G.vexnum)

{

return ERROR;

}

}

int main()

{

printf("ToplogicalSort.c Test\n");

ALGraph G;

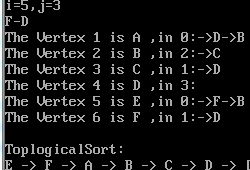
CreateALGraph(&G);

ShowALGraph(&G);

ToplogicalSort(G);

return 0;

}



代码分析

1. 时间复杂度：

有n个元素，有e条边

O(n+e);