1. 程序函数/变量说明

Graph是图的ADT，具体实现在graphm中

Graph中函数的解释：

void Init(int n)=0;//初始化有n个顶点的图

int n()=0;

int e()=0;//返回图的顶点数和边数

int first(int v)=0;//返回 顶点v的第一个数据

int next(int v,int w)=0;//回 顶点v的下一个数据

void setType(bool flag)=0;//设置图的类型

void setVex(int v,VT value)=0;//设置某个顶点的值

void setEdge(int v1,int v2,int wight)=0;//设置边权

void setValue(int v,int val)=0;//

bool getType()=0;//获取图的类型

VT getVex(int v)=0;//返回某个顶点的值

int getValue(int v)=0;//返回该顶点的标记值

int getMa(int i,int j)=0;//返回矩阵中i,j上的权值

void delEdge(int v1,int v2)=0;//删除边v1,v2

bool isEdge(int i,int j)=0;//判断边是否在图中

int weight(int v1,int v2)=0;//返回边的权值

graphm中成员的说明：

class Edge //边类，用做链表元素，包含vert, wt

struct graphlist//结构体数组，包含存储顶点信息的date和存储点之间关系的链表。

int numVert,numEdge; //顶点数和边数

bool undirected; // true表示无向图 false表示有向图

int\*mark; //指向标记数组

graphlist\* gl; //存储信息的结构体数组

link类中：

element 表示结点的数据域

next表示结点指针

link(const L& elemval,link\* nextval=NULL)是创建对象结点时的构造函数

list类中:

clear()是清空线性表元素

insert(L)是插入元素

append(L)是在表尾插入一个元素

remove()是删除当前元素

moveToXXX系列函数是移动当前指针位置

prev()函数是移动到前一个

next()是移动到后一个

length（）是求线性表长度

currPos()返回当前位置

FindValue()函数是查找元素是否存在于线性表中

getValue(int)函数是返回指定位置的元素值

Print()打印线性表中全部元素

llist类中：

head是头结点

tail是尾结点

curr是当前结点

size是表示线性表的大小

init()函数是对线性表进行初始化

deleteall()函数清空整个线性表

1. 说明

1.主程序的功能是对输入的图的顶点进行存储，然后通过点与点之间的关系建立图，输出该图对应的邻接矩阵，并且求出所有顶点中的最大出度和。

2.CG上测试结果

