**云南大学数学系《算法图论实验》课程上机实验报告**

|  |  |  |
| --- | --- | --- |
| **课程名称**：算法图论实验 | **学期：**2016-2017学年第一学期 | **成绩**： |
| **指导教师**：李建平 | **学生姓名**：金洋 | **学生学号**：20131910023 |
| **实验名称**： **弧连通度** | | |
| **实验编号**：No.4 | **实验日期**：**2016.10.20** | **实验学时**： **1** |
| **学院：** 数学与统计学院 | **专业： 信息与计算科学** | **年级**： **2013** |

**一、实验目的**

使用c语言实现求有向图D的弧连通度；

**二、实验内容**

给定任意有向图D=(V,A)，设计算法求其弧连通度；

**三、使用环境**

平台：Microsoft Visual C++ 6.0

语言：C语言

**四、算法介绍**

Algorithm 弧连通度

Input 有向图D=(V, A);

Output D的弧连通度;

Begin

Step 1:令；

Step 2: For i=1 through n

For j=1 through n

if (i≠j) then

1. 构造网络，对N中任意的弧e，令c(e)=1，初始流为零流；
2. 利用Edmonds-Karp算法求得N上从到的最大整数流值；
3. ；

Step 3: 输出D的弧连通度；

End.

**五、调试过程**

1．程序代码

**ArcConnectivity.c**

#include <stdio.h>

#include <string.h>

#define MAXN 999999999

#define MAXV 100

int n;//顶点数

int f;//记录弧连通度

int C[MAXV][MAXV],F[MAXV][MAXV];//C,F分别记录N的容量、流量

int Cf[MAXV][MAXV];//剩余网络权值

int LAM[MAXV],L[MAXV],X[MAXV];

void input() {

int i,j;

//freopen("ArcConnectivity.in", "r", stdin);

printf("请输入有向图D的顶点个数n=");

scanf("%d",&n);

printf("请输入有向图D的邻接矩阵:\n");

for (i=1;i<=n;i++)

for (j=1;j<=n;j++)

scanf("%d",&C[i][j]);

}

/\*构造剩余网络\*/

void buildNf() {

int i,j;

memset(Cf, 0, sizeof(Cf));

for (i=1;i<=n;i++)

for (j=1;j<=n;j++)

if (C[i][j]) {

Cf[i][j]=C[i][j]-F[i][j];

Cf[j][i]=F[i][j];

}

for (i=1;i<=n;i++)

for (j=1;j<=n;j++)

if (Cf[i][j]==0)

Cf[i][j]=MAXN;//去掉Nf中容量值为0的所有弧

}

/\*最短路径初始化\*/

void shortestPathInitial(int VS) {

int i;

for (i=1;i<=n;i++) {

X[i]=0;

LAM[i]=MAXN;

}

LAM[VS]=0;

X[VS]=1;

X[0]=1;//X[0]记录X集合中顶点个数,初始时Vs进入X集合

for (i=1;i<=n;i++) L[i]=0;

}

/\*φ(X)是否为空? 是:返回0;否:返回1\*/

int PHi(int M[MAXV][MAXV]) {

int i,j;

for(i=1;i<=n;i++)

for (j=1;j<=n;j++)

if ((i!=j)&&(X[i])&&(X[j]==0)&&(M[i][j]<MAXN)) return(1);

return(0);

}

/\*最短路\*/

void prim(int M[MAXV][MAXV]) {

int i,j,min;

int x,y;

min=MAXN;

for(i=1;i<=n;i++)

if (X[i])

for(j=1;j<=n;j++)

if (!X[j] && i!=j && LAM[i]+M[i][j]<min ) {

min=LAM[i]+M[i][j];

x=i;

y=j;

}

if (min!=MAXN) {

X[y]=1;

X[0]++;

L[y]=x;

LAM[y]=min;

}

}

/\*改变N中的流量\*/

void changeFlow(int VT) {

int theta=MAXN;

int y=VT;

while (L[y]!=0) {

if (theta>Cf[L[y]][y] ) theta=Cf[L[y]][y];

if (theta>Cf[y][L[y]] ) theta=Cf[y][L[y]];

y=L[y];

}

y=VT;

while (L[y]!=0) {

if (F[L[y]][y]<C[L[y]][y]) F[L[y]][y]+=theta;

if (F[y][L[y]]>0) F[y][L[y]]-=theta;

y=L[y];

}

}

void main() {

int i,j,k,tempVf;

input();

f=MAXN;//记录弧连通度

for (i=1;i<=n;i++)

for (j=1;j<=n;j++)

if (i!=j) {

memset(F, 0, sizeof(F));//流量初始为全0

while (1) {

buildNf();

shortestPathInitial(i);//最短路径初始化

while (!X[j] && PHi(Cf)) prim(Cf);

if (X[j]) changeFlow(j);

else break;

}

tempVf=0;

for(k=1;k<=n;k++) tempVf+=F[i][k];

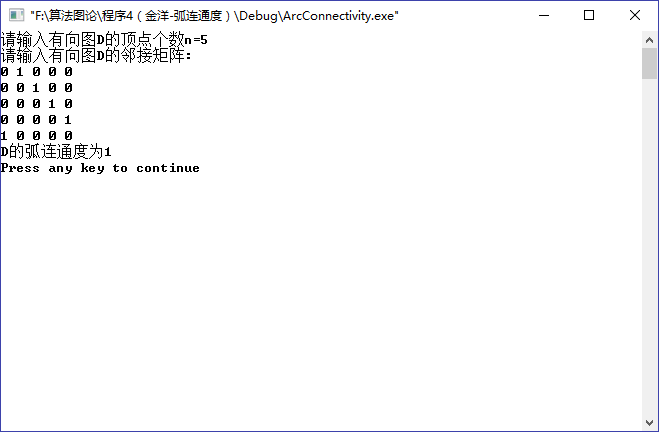
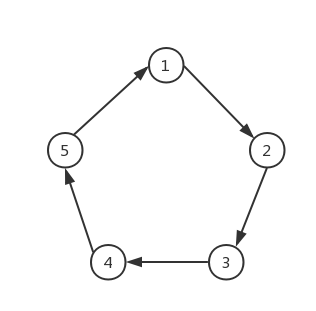
if (tempVf<f) f=tempVf;

}

printf("D的弧连通度为%d\n",f);

}

1. 运行窗口



**六、总结**

1.学会使用c语言实现求有向图D的弧连通度；

**七、参考文献**

[1] 谭浩强，c程序设计（第三版），清华大学出版社，2005年7月；

[2] 《运筹学》教材编写组，运筹学（第四版），清华大学出版社，2012年9月；

**八、教师评语**