**CODE THI THỰC HÀNH LÍ THUYẾT ĐỒ THỊ**

***1. Thuật toán Prim tìm cây khung nhỏ nhất:***

#include <stdio.h>

#include <conio.h>

#include <string.h>

#define MAX\_ARR 10

#define fi "CAYKHUNG.TXT"

#define oo 100000

struct GRAPH

{

int n;

int a[MAX\_ARR][MAX\_ARR];

};

struct EDGE

{

int u;

int v;

int value;

};

int Free[MAX\_ARR];

EDGE T[MAX\_ARR];

int ReadData(char finame[20],GRAPH &g)

{

FILE \*f;

f=fopen(finame,"rt");

if (f==NULL)

{

printf("Khong doc duoc file!\n");

return 0;

}

fscanf(f,"%d",&g.n);

for (int i=0;i<g.n;i++)

for (int j=0;j<g.n;j++)

fscanf(f,"%d",&g.a[i][j]);

fclose(f);

return 1;

}

void WriteData(GRAPH g)

{

printf("So dinh cua do thi: %d\n",g.n);

printf("Ma tran ke cua do thi:\n");

for (int i=0;i<g.n;i++)

{

for (int j=0;j<g.n;j++)

printf("%d\t",g.a[i][j]);

printf("\n");

}

}

void FindConnect(GRAPH g,int label[MAX\_ARR],int i)

{

for (int j=0;j<g.n;j++)

if (g.a[i][j]!=0 && label[j]!=label[i])

{

label[j]=label[i];

FindConnect(g,label,j);

}

}

int Connect(GRAPH g)

{

int label[MAX\_ARR];

memset(label,0,sizeof(label));

int count=0;

for (int i=0;i<g.n;i++)

if (label[i]==0)

{

count++;

label[i]=count;

FindConnect(g,label,i);

}

return count;

}

void Prim(GRAPH g)

{

if (Connect(g)!=1)

{

printf("Do thi khong lien thong, do do khong thuc hien duoc thuat toan Prim...\n");

return;

}

int nT=0;

memset(Free,0,sizeof(Free));

Free[0]=1;

while (nT < g.n-1)

{

EDGE MinEdge;

int MinVal=oo;

for (int i=0;i<g.n;i++)

if (Free[i]==1)

{

for (int j=0;j<g.n;j++)

if (Free[j]==0 && g.a[i][j]!=0 && MinVal > g.a[i][j])

{

MinEdge.u=i;

MinEdge.v=j;

MinEdge.value=g.a[i][j];

MinVal=g.a[i][j];

}

}

T[nT]=MinEdge;

nT++;

Free[MinEdge.v]=1;

}

int SumVal=0;

printf("Cay khung nho nhat cua do thi la:\n");

for (int i=0;i<nT-1;i++)

{

printf("(%d,%d), ",T[i].u,T[i].v);

SumVal+=T[i].value;

}

printf("(%d,%d).\n",T[nT-1].u,T[nT-1].v);

SumVal+=T[nT-1].value;

printf("Tong trong so cua cay khung: %d\n",SumVal);

}

int main()

{

GRAPH g;

if (ReadData(fi,g)==1)

{

printf("Doc thong tin do thi thanh cong...\n");

WriteData(g);

printf("Bam phim bat ki de bat dau tim cay khung nho nhat...\n");

getch();

Prim(g);

}

return 0;

}

***2. Thuật toán Dijkstra tìm đường đi ngắn nhất:***

#include <stdio.h>

#include <conio.h>

#include <memory.h>

#define MAX\_ARR 10

#define fi "DUONGDI.TXT"

#define oo 10000

struct GRAPH

{

int n;

int a[MAX\_ARR][MAX\_ARR];

};

int Trace[MAX\_ARR];

int Free[MAX\_ARR];

int Path[MAX\_ARR];

int ReadData(char finame[20],GRAPH &g)

{

FILE \*f;

f=fopen(finame,"rt");

if (f==NULL)

{

printf("Khong doc duoc file!\n");

return 0;

}

fscanf(f,"%d",&g.n);

for (int i=0;i<g.n;i++)

for (int j=0;j<g.n;j++)

fscanf(f,"%d",&g.a[i][j]);

fclose(f);

return 1;

}

void WriteData(GRAPH g)

{

printf("So dinh cua do thi: %d\n",g.n);

printf("Ma tran ke cua do thi:\n");

for (int i=0;i<g.n;i++)

{

for (int j=0;j<g.n;j++)

printf("%d\t",g.a[i][j]);

printf("\n");

}

}

int FindMinPath(GRAPH g)

{

int li=-1; float p=oo;

for (int i=0;i<g.n;i++)

if (Free[i]==0 && Path[i]<p)

{

p=Path[i];

li=i;

}

return li;

}

void UpdatePath(int u,GRAPH g)

{

Free[u]=1;

for (int v=0;v<g.n;v++)

{

if (Free[v]==0 && g.a[u][v]>0 && g.a[u][v]<oo)

if (Path[v]>Path[u]+g.a[u][v])

{

Path[v]=Path[u]+g.a[u][v];

Trace[v]=u;

}

}

}

void Dijkstra(int s,int f,GRAPH g)

{

memset(Free,0,sizeof(Free));

memset(Path,oo,sizeof(Path));

memset(Trace,-1,sizeof(Trace));

Path[s]=0;

while (Free[f]==0)

{

int u=FindMinPath(g);

if (u==-1) break;

UpdatePath(u,g);

}

if (Free[f]==1)

{

printf("Duong di ngan nhat tu dinh %d den dinh %d la:\n",s,f);

int i=f;

printf("%d ",f);

while (Trace[i]!=s)

{

printf("<- %d ",Trace[i]);

i=Trace[i];

}

printf("<- %d\n",Trace[i]);

printf("\tVoi do dai la: %d\n",Path[f]);

}

else

printf("Khong co duong di tu dinh %d den dinh %d.",s,f);

}

int main()

{

GRAPH g;

if (ReadData(fi,g)==1)

{

printf("Doc thong tin do thi thanh cong...\n");

WriteData(g);

printf("Bam phim bat ki de bat dau tim duong di ngan nhat...\n");

getch();

int s,f;

printf("Nhap dinh bat dau: ");

scanf("%d",&s);

printf("Nhap dinh ket thuc: ");

scanf("%d",&f);

Dijkstra(s,f,g);

}

return 0;

}

***3. Bộ dữ liệu test:***

|  |  |
| --- | --- |
| CAYKHUNG.TXT | DUONGDI.TXT |
| 8  0 42 4 0 10 0 0 0  42 0 0 0 3 3 0 0  4 0 0 3 0 0 15 0  0 0 3 0 1 0 5 0  10 3 0 1 0 11 0 0  0 3 0 0 11 0 0 0  0 0 15 5 0 0 0 7  0 0 0 0 0 0 7 0 | 5  0 1 3 10 6  1 0 8 9 2  3 8 0 4 7  10 9 4 0 5  6 2 7 5 0 |

Good luck to me ☺

Nguyễn Trọng Minh Hồng Phước

NTMHP – 15DTH04 – 1511060249

hongphuoc123456@gmail.com