1. Formularea enuntului problemei

Se da un graf orientat si valorile arcelor. Sa se conceapa un algoritm care sa determine drumul de valoare minima de la varful p la varful q (p si q date , oarecare ) , folosind algoritmul lui Yen.

Datele de intrare se dau de la tastatura .

Datele de iesire care prezinta drumul de valoare minima de la varful p la varful q se vor afisa pe ecran .

2. Dezvoltarea algoritmului

Algoritmul presupune doi pasi :

a. Citirea matricei costurilor (valorilor arcelor)

b.Modificarea acesteia la calcularea valorilor drumurilor minime dintre varfuri.

3. Descrierea algoritmului

Algoritmul lui Yen realizeaza generarea matricii drumurilor optime

dintre oricare 2 varfuri date .

*Pentru j=1,n executa*

*Pentru i ϵ Г-j ( sau i=1,n ) executa*

*Daca ʎi + li j < ʎj atunci ʎj := ʎi + li j*

*k := 1*

*SfDaca*

*SfPentru*

*SfPentru*

*Pentru j=n,1,-1 executa*

*Pentru i ϵ Г-j ( sau i=1,n ) executa*

*Daca ʎi + li j < ʎj atunci ʎj := ʎi + li j*

*k := 1*

*SfDaca*

*SfPentru*

*SfPentru*

4. Demonstrarea corectitudinii algoritmului

Tot ceea ce am folosit se gaseste in manual si ni s-a predat la curs

5. Cod sursa

#include <iostream>

#include <fstream>

using namespace std;

#define inf 10000000

int v[100][100],n , la[100],l[100][100];

void citire()

{int i,j;

ifstream f("graf.in");

f>>n;

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

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

f>>v[i][j];

}

void tipar(int la[])

{

int i;

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

cout<<la[i]<<" ";

cout<<endl;

}

void yen(int vf)

{int i,j,modif;

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

if(i==vf)

la[i]=0;

else la[i]=v[vf][i];

tipar(la);

do{modif=0;

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

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

if(v[i][j]!=inf && la[i]!=inf)

if(la[j]>la[i]+v[i][j])

{la[j]=la[i]+v[i][j];

modif=1;

}

for(j=n;j>=1;j--)

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

if(v[i][j]!=inf && la[i]!=inf)

if(la[j]>la[i]+v[i][j])

{la[j]=la[i]+v[i][j];

modif=1;

}

}while(modif);

tipar(la);

}

void drum(int p,int q)

{int d[100],k,i,j;

//se porneste invers, se adauga varful final

k=1;

d[k]=q;

while(d[k]!=p)

{

//drumul de la p la d[k](initial q) trece prin varful intermediar j

//daca costul minim al drumului p->d[k] este egal cu costul minim al drumului

//p->j la care se adauga arcul (j,d[k])

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

if(v[j][d[k]]!=inf && v[j][d[k]]!=0) //sa existe arcul(j,d[k])

if(la[j]+v[j][d[k]]==la[d[k]])

{

k++;

d[k]=j;

}

}

for(i=k;i>=1;i--)cout<<d[i]<<" ";

}

int main()

{

citire();

int p,q;

cin>>p>>q;

cout<<endl<<"yen:"<<endl;

yen(p);

cout<<"drum"<<endl;

drum(p,q);

return 0;

}

6. Date de test

|  |  |
| --- | --- |
| Date de intrare | Date de iesire |
| 6  0 7 8 10000000 10000000 10000000  10000000 0 10000000 -4 1 10000000  10000000 2 0 10000000 10000000 2  10000000 10000000 10000000 0 10000000 10000000  10000000 10000000 -2 2 0 10000000  10000000 10000000 10000000 10000000 3 0  Varful p =1 ;  Varful q = 6 ; | Yen:  0 7 8 10000000 10000000 10000000  0 7 6 3 8 8  Drum:  1 2 5 3 6 |
| 5  0 2 10000000 10 10000000  2 0 3 10000000 10000000  10000000 3 0 1 8  10 10000000 1 0 10000000  10000000 10000000 8 10000000 0  Varful p =2 ;  Varful q = 5 ; | Yen:  2 0 3 10000000 10000000  2 0 3 4 11  Drum:  2 3 5 |