

PROGNOZĂ DIDACTICĂ

LICEUL DE ARTĂ "IONEL PERLEA" SLOBOZIA

Disciplina: *Informatică*

Clasa: *a XI-a C*

Profesor: *Șcheaua Liliana*

Unitatea de învățare: *Grafuri orientate*

Tema: *Drum minim în grafuri orientate. Algoritmul lui Dijkstra*

Tipul lecției: *mixta*

COMPETENȚE GENERALE

- identificarea datelor care intervin într-o problemă și aplicarea algoritmilor fundamentali de prelucrare a acestora;
- realizarea unor subprograme pentru operații specifice grafurilor orientate;
- implementarea algoritmilor într-un limbaj de programare.

COMPETENȚE SPECIFICE

- analizarea unei probleme în scopul identificării datelor necesare și alegerea modalităților adecvate de structurare a datelor care intervin într-o problemă;
- descrierea unor algoritmi simpli de verificare a unor proprietăți specifice grafurilor.

OBIECTIVE PERFORMATIVE (OPERATIONALE):

- să construiască matricea costurilor unui graf orientat dat
- să reprezinte textual și grafic grafurile orientate;
- să identifice algoritmi necesari pentru rezolvarea problemelor
- să justifice necesitatea utilizării algoritmului Dijkstra
- să compare eficiența celor doi algoritmi studiați: Roy Floyd și Dijkstra.
- să înțeleagă exemplele date și să elaboreze programe corecte pentru aplicațiile propuse;

NIVELUL INITIAL AL CLASEI:

Elevii și-au însușit toate noțiunile teoretice predate referitoare la grafuri orientate și drumuri minime în graf orientat: algoritmul Roy Floyd

STRATEGII DIDACTICE

PRINCIPII DIDACTICE:

- principiul participării și învățării active
- principiul asigurării procesului gradat al performanței;
- principiul conexiunii inverse.

METODE DE ÎNVĂȚĂMANT:

- metode de comunicare orală: conversația, explicația;
- metode activ participative: problematizarea, exercițiul.

FORME DE ORGANIZARE: frontală și individuală;

RESURSE MATERIALE : manual, teste de evaluare, culegere

METODE DE EVALUARE:

- sumativă: test de evaluare

DESFĂȘURAREA ACTIVITĂȚII:

► *moment organizatoric*

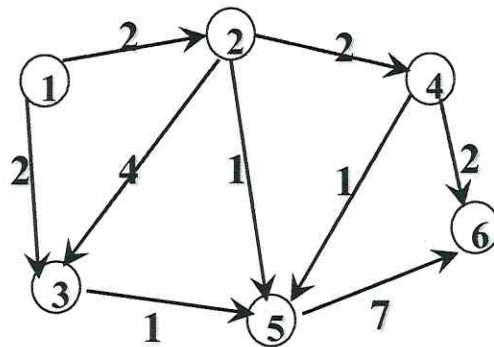
- întocmirea proiectului didactic;

- realizarea testului de evaluare
- ▶ *organizarea si pregătirea clasei*
 - verificarea frecvenței
 - verificarea cantitativa a temei, frontal, si verificarea calitativa, prin sondaj;
 - verificarea existentei resurselor materiale;

A. REACTUALIZAREA CUNOSTINTELOR

Fie graful din figura alăturată.

- a) Să se construiască matricea ponderilor in forma I
- b) Să se identifice valorile costurilor drumului minim de la nodul 2 catre toate celelalte noduri.
- c) Care este valoarea maxima dintre costurile drumurilor calculate? Sa se afiseze lista nodurilor care alcatuiesc drumul de la nodul 2 catre unul dintre nodurile care au costul maxim
- d) Cate drumuri minime putem calcula utilizand algoritmul Roy Floyd?



B. COMUNICAREA NOILOR CUNOSTINTE

Algoritmul Roy Floyd calculează drumul minim de la oricare nod x la oricare nod y . Algoritmul pe care îl studiem astăzi își propune să determine lungimea minimă a drumului de la un nod x la oricare nod y din mulțimea nodurilor unui graf orientat.

Prin comparatie cu algoritmul studiat care utiliza matricea pentru retinerea valorilor drumului acum vom utiliza 3 tablouri unidimensionale cu următorul rol:

- Vectorul COST- unde $cost[i]$ reprezintă costul drumului găsit la un moment dat între nodul i și nodurile i care apartin $\{1, \dots, n\}$
- Vectorul VIZ- indică mulțimea nodurilor selectate, vizitate, unde $viz[i]=0$ dacă nodul nu a fost selectat și $viz[i]=1$ dacă nodul a fost selectat.
- Vectorul T- reține pentru fiecare i nodul precedent pe unde trece drumul de la rădăcina la i (părintele sau tatăl nodului i). Pentru rădăcină se memorează 0.

PREZENTAREA ALGORITMULUI

ETAPA I

- Se citește nodul radacina, si se vizitează
ex. $rad=2$, $viz[rad]=1$;
- Se construiește vectorul cost cu valorile costurilor drumurilor de la rad la celelalte noduri (daca au arc se iau valorile din matricea ponderilor de pe linia i)
- Se construiește vectorul T: Pentru toate nodurile (diferite de radacina) care au costul mai mic de 32000 se completeaza vectorul $T[i]$ cu radacina

i	1	2	3	4	5	6
Cost[i]	32000	0	4	2	1	32000
T[i]	0	0	2	2	2	0

ETAPA a II-a

Se va executa de n-1 secventa următoare:

Pasul 1

Dintre nodurile nevizitate se caută cel aflat la distanță minimă față de rad, si se va păstra într-o variabila poz. Se vizitează viz[poz]:=1;

Pentru nodurile neselectate j, se actualizează în cost[j] costul drumurilor de la rad la j utilizand ca nod intermediar nodul selectat poz.

poz=5, j=6

Initial, cost[6]= 32000

daca $cost[j] > cost[poz] + a[poz,j]$ $cost[6] > cost[5] + a[5,6]$

atunci $cost[6] = cost[5] + a[5,6] = 1 + 7 = 8$

nodul i	1	2	3	4	5	6
Cost[i]	32000	0	4	2	1	8
T[i]	0	0	2	2	2	5
VIZ[i]	0	1	0	0	1	0

Pasul 2

Se repeta operatiile de la pasul 1. Se va selecta urmatorul nod nevizitat cu valoare minimă.

Nodul selectat va fi 4, se vizitează si va fi considerat nod intermediar poz=4, viz[poz]=1;

Se actualizează în cost[j] costul drumurilor de la rad la j utilizand ca nod intermediar nodul selectat poz=4

nodul i	1	2	3	4	5	6
Cost[i]	32000	0	4	2	1	4
T[i]	0	0	2	2	2	4
VIZ[i]	0	1	0	1	1	0

Pasul

3

Se repeta operatiile de la pasul 1. Se va selecta urmatorul nod nevizitat cu valoare minimă.

Nodul selectat va fi 3, se vizitează si va fi considerat nod intermediar poz=3, viz[poz]=1;

Se actualizează în cost[j] costul drumurilor de la rad la j utilizand ca nod intermediar nodul selectat poz=4.

nodul i	1	2	3	4	5	6
Cost[i]	32000	0	4	2	1	4
T[i]	0	0	2	2	2	4
VIZ[i]	0	1	1	1	1	0

Pasul 4

Se repeta operatiile de la pasul 1. Se va selecta urmatorul nod nevizitat cu valoare minimă.

Nodul selectat va fi 6, se vizitează si va fi considerat nod intermediar poz=6, viz[poz]=1;

Se actualizează în cost[j] costul drumurilor de la rad la j utilizand ca nod intermediar nodul selectat poz=6.

nodul i	1	2	3	4	5	6
Cost[i]	32000	0	4	2	1	4
T[i]	0	0	2	2	2	4
VIZ[i]	0	1	1	1	1	1

Pasul 5

Se va selecta urmatorul nod nevizitat 1. Nu se va realiza nicio modificare deoarece nu exista arc spre nodul 1.

nodul i	1	2	3	4	5	6
Cost[i]	32000	0	4	2	1	4
T[i]	0	0	2	2	2	4
VIZ[i]	0	1	1	1	1	1

C. INTENSIFICAREA RETINERII SI ASIGURAREA TRANSFERULUI DE INFORMATII

Se realizeaza pe parcursul orei prin intrebari, construirea celor trei vectori cu ajutorul elevilor in urma explicatiilor algoritmului precum si scrierea algoritmului Dijkstra.

```

cin>>rad; viz[rad]=1;
for(i=1;i<=n;i++)
{
cost[i]=a[rad][i];

    if ( i!=rad)
        if (cost[i]<32000 ) t[i]=rad;
}
for(i=1;i<n;i++)
{
    min=32000;
    for(j=1;j<=n;j++)
    if (viz[j]==0)
        if (cost[j]<min)
        {
            min=cost[j];
            poz=j;
        }
    viz[poz]=1;
    for(j=1;j<=n;j++)
    if (viz[j]==0)
        if (cost[j]>cost[poz]+a[poz][j] )
        {
            cost[j]=cost[poz]+a[poz][j];
            t[j]=poz;
        }
}

```

D.ASIGURAREA FEEDBACK-ULUI SI EVALUAREA PERFORMANTEI

Identificați drumurile de cost minim de la nodul 1 către celelalte noduri si construiti cei trei vectori cost, viz, t.

E.TEMA PENTRU ACASĂ

Analiza eficientei celor doi algoritmi cunoscuti pentru determinarea drumului minim in graf orientat.