**Ministerul Educației al Republicii Moldova**

**Universitatea Tehnică a Moldovei**

**Facultatea Calculatoare, Informatică și Microelectronică**

## RAPORT

**la disciplina: “Matematica discretă”**

**Lucrare de laborator Nr. 1**

**Tema: “Păstrarea grafurilor în memoria calculatorului”**

**Varianta 3**

**A verificat : Dohotaru Leonid**

**A efectuat: st. Brăduleac Vadim , gr. AI-151**

**Chișinău 2016**

Se numeşte graf, ansamblul format dintr-o mulţime finită *X* şi o aplicaţie *F* a lui *X* în *X*. Se notează *G = (X,F)*. Numărul elementelor mulţimii *X* determină ordinul grafului finit. Dacă *card X = n*, graful *G = (X,F)* se numeşte *graf finit de ordinul n*. Elementele mulţimii *X* se numesc *vârfurile* grafului. Geometric, vârfurile unui graf le reprezentăm prin puncte sau cerculeţe. Perechea de vârfuri *(x,y)* se numeşte *arc*; vârful *x* se numeşte originea sau extremitatea iniţială a arcului *(x,y)*, iar vârful *y* se numeşte extremitatea finală sau terminală. Un arc *(x,y)* îl reprezentăm geometric printr-o săgeată orientată de la vârful *x* la vârful *y*.

Există trei metode de bază de definire a unui graf:

1. *Matricea de incidenţă;*
2. *Matricea de adiacenţă;*
3. *Lista de adiacenţă (incidenţă).*

Vom face cunoştinţă cu fiecare dintre aceste metode.

## 4.2.1. Matricea de incidenţă

Este o matrice de tipul *m*x*n*, în care *m* este numărul de muchii sau arce (pentru un graf orientat), iar *n* este numărul vârfurilor. La intersecţia liniei *i* cu coloana *j* se vor considera valori de *0* sau *1* în conformitate cu următoarea regulă:

* *1* - dacă muchia *i* este incidentă cu vârful *j* (dacă arcul *i* "intră" în vârful *j* în cazul unui graf orientat);
* *0* - dacă muchia (arcul) *i* şi vârful *j* nu sunt incidente;
* *-1* - numai pentru grafuri orientate, dacă arcul *i* "iese" din vârful *j*.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | *x1* | *x2* | *x3* | *x4* | *x5* | *x6* | *x7* |
| *u1* | -1 | 0 | 1 | 0 | 0 | 0 | 0 |
| *u2* | -1 | 0 | 0 | 1 | 0 | 0 | 0 |
| *u3* | 0 | 0 | 0 | -1 | 0 | 0 | 1 |
| *u4* | 0 | 0 | -1 | 0 | 0 | 0 | 1 |
| *u5* | 0 | -1 | 1 | 0 | 0 | 0 | 0 |
| *u6* | 0 | -1 | 0 | 0 | 1 | 0 | 0 |
| *u7* | 0 | -1 | 0 | 0 | 0 | 1 | 0 |
| *u8* | 0 | 0 | -1 | 0 | 0 | 1 | 0 |

Fig. 1 Exemplu de matrice de incidenţă

Este uşor de observat că această metodă este de o eficacitate mică în sensul utilizării memoriei calculatorului: fiecare linie conţine doar două elemente diferite de zero (o muchie poate fi incidentă cu nu mai mult de două vârfuri).

În limbajul Pascal matricea de incidenţă poate fi redată printr-un tablou bidimensional *m*x*n*:

Matr\_Incd: array [1..LinksCount, 1..TopsCount] of integer;

## 4.2.2. Matricea de adiacenţă

Este o matrice pătrată *n*x*n*, aici *n* este numărul de vârfuri. Fiecare element poate fi *0*, dacă vârfurile respective nu sunt adiacente, sau 1, în caz contrar. Pentru un graf fără bucle putem observa următoarele:

* diagonala principală este formată numai din zerouri;
* pentru grafuri neorientate matricea este simetrică faţă de diagonala principală.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | *x1* | *x2* | *x3* | *x4* | *x5* | *x6* | *x7* |
| *x1* | 0 | 0 | 1 | 1 | 0 | 0 | 0 |
| *x2* | 0 | 0 | 1 | 0 | 1 | 1 | 0 |
| *x3* | 0 | 0 | 0 | 0 | 0 | 1 | 1 |
| *x4* | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| *x5* | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| *x6* | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| *x7* | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Fig. 2. Exemplu de matrice de adiacenţă

În limbajul Pascal matricea de adiacenţă poate fi reprezentată în modul următor:

Matr\_Ad :array [1..TopsCount,1..TopsCount] of byte,

sau

Matr\_Ad :array [1..TopsCount,1..TopsCount] of boolean;

După cum este lesne de observat şi în acest caz memoria calculatorului este utilizată nu prea eficace din care cauză matricea de adiacenţă ca şi matricea de incidenţă se vor utiliza de obicei doar în cazul în care se va rezolva o problemă concretă pentru care reprezentarea grafului în această formă aduce unele facilităţi algoritmului respectiv.

Pentru păstrarea grafurilor în memoria calculatorului (în deosebi, memoria externă) se va utiliza una din posibilităţile de mai jos.

## Lista de adiacenţă şi lista de incidenţă

**Lista de adiacenţă** este o listă cu *n* linii (după numărul de vârfuri *n*), în linia cu numărul *i* vor fi scrise numerele vârfurilor adiacente cu vârful *i*.

**Lista de incidenţă** se defineşte analogic cu deosebirea că în linia *i* vor fi scrise numerele muchiilor (arcelor) incidente cu vârful *i*.

Reprezentarea grafurilor prin intermediul acestor liste permite utilizarea mai eficace a memoriei calculatorului, însă aceste forme sunt mai complicate atât în realizare, cât şi în timpul procesării. Pentru a lua în consideraţie lungimea variabilă a liniilor vor fi utilizate variabile dinamice şi pointeri.

Vom exemplifica pentru un graf cu *n* vârfuri. Deoarece fiecare element al listei conţine numere de vârfuri este evident să considerăm că vom avea un şir de variabile dinamice de tip INTEGER care se vor afla în relaţia respectivă de precedare (succedare). Această relaţie se va realiza prin pointeri, uniţi împreună cu variabila de tip întreg în înregistrarea (Pascal: record). Pentru a păstra indicatorii de intrare în aceste şiruri se va folosi un tablou unidimensional de indicatori de lungime *n*. În calitate de simbol de terminare a şirului se va utiliza un simbol care nu a fost folosit la numeraţia vârfurilor (de exemplu *0*), care va fi introdus în calitate de variabilă de tip întreg al ultimului bloc.

**Variante 3**

F(x1)={x2,x3};

F(x2)={x3,x5};

F(x3)={x4,x6};

F(x4)={x6,x7};

F(x5)={x6,x8};

F(x6)={x8};

F(x7)={x6,x8};

F(x8)={Ø};

**Scopul lucrarii:**

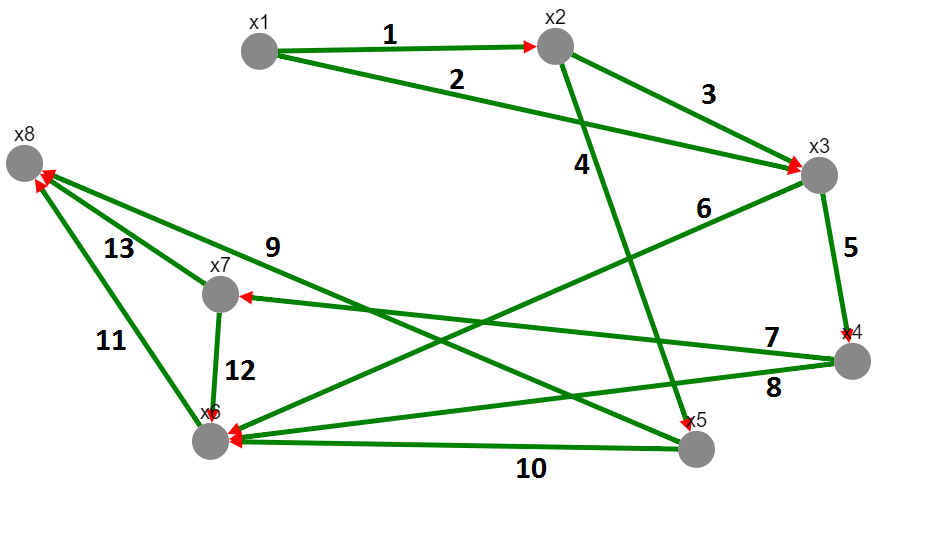
Se dă un graf G=(X,F) unde X este mulțimeade vîrfuri X={x1,x2,x3,x4,x5,x6,x7,x8}, F:X-->P(X)

**Se cere:**

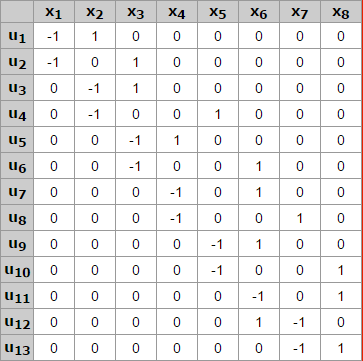
**1 A doua formă analitică G=(X,U),unde U este mulțimea de arce.**

U={(x1,x2)(x1,x3)(x2,x3)(x2,x5)(x3,x4)(x3,x6)(x4,x6)(x4,x7)(x5,x6)(x5,x8)(x6,x8)(x7,x6)(x7,x8)}

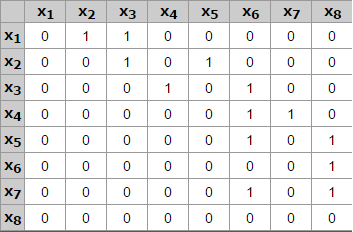
**2 Reprezentarea grafică (geometrică).**



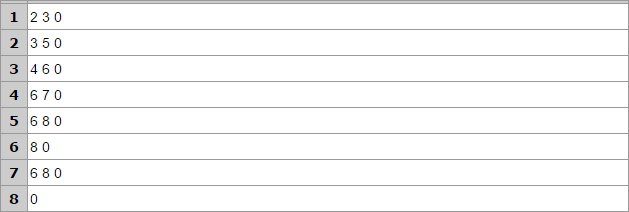
**3 Matricea de incidență.**



**4 Matricea de adiacență.**



**5 Lista de adiacență.**



**6 Gradele exterioare ,interioare și gradele vîrfurilor.**

g+(x1)=2 g-(x1)=0 g(x1)=2

g+(x2)=2 g-(x2)=1 g(x1)=3

g+(x3)=2 g-(x3)=2 g(x1)=4

g+(x4)=2 g-(x4)=1 g(x1)=3

g+(x5)=2 g-(x5)=1 g(x1)=3

g+(x6)=1 g-(x6)=4 g(x1)=5

g+(x7)=2 g-(x7)=1 g(x1)=3

g+(x8)=3 g-(x8)=0 g(x1)=3

**7 Clasificarea vîrfurilor: inițiale,finale,intermediare,izolate.**

**Izolat g(x)=0**

**Inițial g+(x)>0 g-(x)=0**

**Final g+(x)=0 g-(x)>0**

**Intermediar g+(x)>0 g-(x)>0**

**Nod g(x)>2**

Izolat- nu există

Inițial-x1,x8

Final- nu există

Intermediar-x1,x2,x3,x4,x5,x6,x7

Noduri-x2,x3,x4,x5,x6,x7,x8.

**8 Buclele.**

Nu există.

**9 Vîrfurile adiacente X3.**

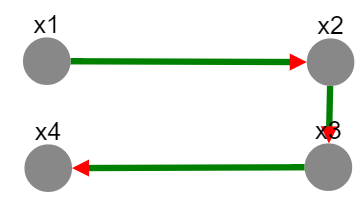
🡪X4,x6

**10 Arcele de incidență cu X4.**

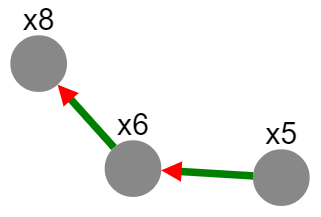
**🡪** x3

**11 Subgraficele G1=(X1,U1) și G2=(X2,U2) unde X1={x1,x2,x3,x4},X2={x5,x6,x7,x8}.**

**X1={x1,x2,x3,x4}:**

****

**X2={x5,x6,x7,x8}:**

****

**Concluzie :** *Efectuând aceasta lucrare am făcut cunoştinţă: cu noţiuni generale în teoria grafurilor, metodele de reprezentare ale grafurilor: lista de adiacenţă, matricea de adiacenţă şi matricea de incidenţă şi procedurile de introducere, păstrare şi transformare a diferitor forme de reprezentare internă a grafurilor. Cel mai eficient mod de reprezentare a grafurilor în memoria calculatorului este lista de adiacenţă, deoarece menajeaza memoria calculatorului prin folosirea structurilor dinamice de date şi în acelaşi timp păstrează toată informaţia despre graf într-o formă simplă şi explicită.*