**SD – Seminar 8 – GRAFURI. DIGRAFURI**

**08.12.2020**

**Graf.**

**Digraf.**

Exemplu de digraf:

|  |  |
| --- | --- |
|  | n = 6, V = {0, 1, 2, 3, 4, 5}  m = 9  A = {(0,1), (0,4), (1,2), (1,3), (1,4), (2,5), (3,5), (4,2), (5,2)} |

**Parcurgere BFS** (din varful 0**):**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| coada: | **0** | **1** | **4** | **2** | **3** | **5** |
| vizitare: | **0** | **1** | **4** | **2** | **3** | **5** |

**Parcurgere DFS:** (din varful 0)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| stiva: | **0** | **1** | **2** | **5** | **3** | **4** |
| vizitare: | **0** | **1** | **2** | **5** | **3** | **4** |

***I Reprezentarea cu matrici de adiacenta***

n = 6

a[0..n-1, 0..n-1], a[i,j] = 1 daca (i,j) ∊A si 0 in caz contrar.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | 0 | 1 | 2 | 3 | 4 | 5 |
| 0 |  | 1 |  |  | 1 |  |
| 1 |  |  | 1 | 1 | 1 |  |
| 2 |  |  |  |  |  | 1 |
| 3 |  |  |  |  |  | 1 |
| 4 |  |  | 1 |  |  |  |
| 5 |  |  | 1 |  |  |  |

struct Digraf{

int n

// int m

int a[0..nMax-1][0..nMax-1]

}

***II Reprezentarea cu liste de adiacenta (exterioara)***

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 |  |  | 1 |  |  | 4 |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
| 1 |  |  | 2 |  |  | 3 |  |  | 4 |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
| 2 |  |  | 5 |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
| 3 |  |  | 5 |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
| 4 |  |  | 2 |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
| 5 |  |  | 2 |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |

struct nod {

Elt inf

nod \* succ

}

struct Digraf {

int n

// int m

nod \* a[0..nMax-1]

}

**Pr. 1** Citirea unui digraf in cele doua reprezentari

n

m

perechi de forma x, y pentru cele m arce

**procedure** citesteDigraf(Digraf D)

**begin**

D <- digrafVid()

read D.n

read D.m

for i <- 0 to D.m-1 do {

read x, y

insereazaArc(D, x, y)

}

**end**

**I matrice de adiacenta**

**procedure** insereazaArc(D, x, y)

**begin**

**\\ TEMA**

**end**

**II liste de adiacenta**

**procedure** insereazaArc(D, x, y)

**begin**

**\\ TEMA**

**end**

**Pr. 2** Determinarea componentelor conexe cu numar maxim de varfuri a unui graf implementat cu matrici de adiacenta / liste de adiacenta.

|  |  |
| --- | --- |
|  | G = (V, E), G.n = 8, G.m = 5 |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| index | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| c | 2 | 3 | 3 | 2 | 3 | 3 | 1 | 2 |
| d | 3 | 4 | 4 | 3 | 4 | 4 | 1 | 3 |

struct Graf{

int n

int m

int a[0..nMax-1][0..nMax-1]

}

**Input:** Graf G

**Output:** Componentele conexe cu numar maxim de varfuri

x R y daca exista drum de la x la y.

V1 = {6}, V2 = {0,3,7}, V3 = {1,2,4,5}

? Cum reprezentam o solutie ?

c[0..n-1], c[i] = eticheta componentei conexe din care face parte varful i.

d[0..n-1], d[i] = dimensiunea componentei conexe din care face parte varful i.

procedure compConexeMax(Graf G, int c[], int d[])

begin

for i <- 0 to G.n-1 do { c[i] <- i; d[i] <- 1 }

for i <- 0 to G.n -2 do

for j <- i+1 to G.n-1 do

if G.a[i,j] == 1 then {

ci <- c[i]; cj <- c[j]; nrVarfuri <- d[i] + d[j]

for k <- 0 to G.n – 1 do

if c[k] == ci or c[k] == cj then {

c[k] <- ci; d[k] <- nrVarfuri

}

}

max <- d[0]

for k <- 1 to G.n – 1 do if d[k] > max then max <- d[k]

for k <-0 to G.n -1 do

if d[k] == max then print k, c[k]

end