Corso di Laurea in INFORMATICA

Algoritmi e Strutture Dati

MODULO 11

STRUTTURE NON LINEARI

Il tipo astratto grafo: specifiche sintattiche e semantiche. Realizzazioni. Visita di un grafo.

Questi lucidi sono stati preparati da Oronzo Altamura, Maria Francesca Costabile e Floriana Esposito, Università degli Studi di Bari, per uso didattico. Essi contengono materiale originale di proprietà dell'Università degli Studi di Bari e/o figure di proprietà di altri autori, società e organizzazioni di cui e' riportato il riferimento. Tutto o parte del materiale può essere fotocopiato per uso personale o didattico ma non può essere distribuito per uso commerciale. Qualunque altro uso richiede una specifica autorizzazione da parte dell'Università degli Studi di Bari e degli altri autori coinvolti.

IL TIPO ASTRATTO GRAFO

COME SI E' DETTO IL GRAFO E' UNA STRUTTURA COMPOSTA DA NODI E ARCHI CHE RAPPRESENTA UNA RELAZIONE BINARIA SULL'INSIEME COSTITUITO DAI NODI.

IN GENERALE, I NODI SONO USATI PER RAPPRESENTARE OGGETTI E GLI ARCHI PER RAPPRESENTARE RELAZIONI TRA COPPIE DI OGGETTI.

CONSIDERIAMO DI SEGUITO SOLO GRAFI ORIENTATI O DIRETTI, NEI QUALI GLI ARCHI HANNO UNA DIREZIONE DA UN CERTO NODO (DI PARTENZA) A UN ALTRO NODO (DI ARRIVO). IN TAL CASO IL GRAFO G = (N,A) , DOVE N E' L'INSIEME FINITO DEI NODI, PREVEDE CHE A SIA UN INSIEME FINITO DI COPPIE ORDINATE DI NODI, RAPPRESENTANTI GLI ARCHI ORIENTATI. POICHE'

OGNI GRAFO NON ORIENTATO G' PUO' ESSERE VISTO COME UN GRAFO ORIENTATO G, OTTENUTO DA G', SOSTITUENDO OGNI ARCO (i,j) E (j,i), DIAMO L'ALGEBRA RELATIVA AI SOLI GRAFI ORIENTATI.

2

SPECIFICA SINTATTICA

TIPI: GRAFO, BOOLEAN, NODO, LISTA

OPERATORI:

CREAGRAFO:	()	→	GRAFO
GRAFOVUOTO:	(GRAFO)	→	BOOLEAN
INSNODO:	(NODO,GRAFO)	→	GRAFO
INSARCO:	(NODO,NODO,GRAFO)	→	GRAFO
CANCNODO:	(NODO,GRAFO)	→	GRAFO
CANCARCO:	(NODO,NODO,GRAFO)	→	GRAFO
PRECEDENTI:	(NODO,GRAFO)	→	LISTA
ESISTENODO:	(NODO,GRAFO)	→	BOOLEAN
ESISTEARCO:	(NODO,NODO,GRAFO)	→	BOOLEAN

3

SPECIFICA SEMANTICA

TIPI: GRAFO: INSIEME G = (N,A) CON N SOTTOINSIEME FINITO DI ELEMENTI DI TIPO "NODO" E A Í NxN

NODO: INSIEME FINITO QUALSIASI

LISTA: LISTA DI ELEMENTI DI TIPO NODO

BOOLEAN: INSIEME DEI VALORI DI VERITA'

OPERATORI:

CREAGRAFO = G

PRE: NESSUNA

POST: G = (N,A) CON N = Æ E A = Æ

4

GRAFOVUOTO (G) = b

PRE: NESSUNA

POST: b=VERO SE N=0 E A=0

b=FALSO ALTRIMENTI

INSNODO (u,G) = G'

PRE: G = (N,A) u Ï N (E' DI TIPO "NODO")

POST: G' = (N,A') A' = A È {(u,v)}

CANCNODO (u,G) = G'

PRE: G = (N,A), u Î N

NON ESISTE v Î N '¢ (u,v) Î A O (v,u) Î A

POST: G' = (N',A), N' = N – {u}

5

CANCARCO (u,v,G) = G'

PRE: G = (N,A), u Î N, v Î N, (u,v) Î A

POST: G' = (N,A') , A' = A – {(u,v)}

ADIACENTI: (u,G) = L

PRE: G = (N,A), u Î N

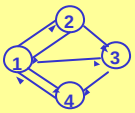
POST: L E' UNA LISTA CHE CONTIENE UNA E UNA SOLA VOLTA GLI ELEMENTI DI A(u) = {v ¡ (u,v) Î A}

QUANDO AI NODI E/O AGLI ARCHI SONO ASSOCIATE INFORMAZIONI (ETICHETTE, PESI), SI PARLA DI GRAFI ETICHETTATI: IN TAL CASO VANNO INTRODOTTI NUOVI OPERATORI PER RITROVARE O MODIFICARE LE INFORMAZIONI ASSOCIATE AI NODI E/O AGLI ARCHI.

6

RAPPRESENTAZIONE CON MATRICE DI ADIACENZA

LA PIU' SEMPLICE RAPPRESENTAZIONE UTILIZZA UNA MATRICE E = [e i j] TALE CHE e i j = 1 NEL CASO (i,j) Î A, MENTRE e i j = 0 SE (i,j) Ï A.



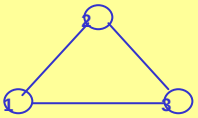
	1	2	3	4
1	0	1	1	1
2	1	0	1	0
3	0	0	0	1
4	1	0	0	0

SE IL GRAFO E' PESATO, NELLA MATRICE SI UTILIZZANO I PESI DEGLI ARCHI AL POSTO DEGLI ELEMENTI BINARI. SE Pij E' IL PESO DELL'ARCO (i,j) ALLORA L'ELEMENTO DELLA MATRICE E DIVENTA

$$e_{ij} = \begin{cases} P_{ij} & \text{SE } (i,j) \in A \\ +\infty\ (-\infty) & \text{SE } (i,j) \notin A \end{cases}$$

7

RAPPRESENTAZIONE CON MATRICE DI ADIACENZA



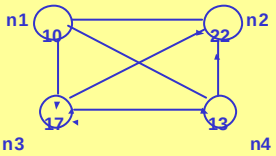
	1	2	3
1	0	1	1
2	1	0	1
3	1	1	0

NATURALMENTE E' POSSIBILE UTILIZZARE LA MEDESIMA RAPPRESENTAZIONE PER GRAFI NON ORIENTATI: NE RISULTERA' UNA MATRICE SIMMETRICA RISPETTO ALLA DIAGONALE PRINCIPALE CON e i j = e j i

8

LA MATRICE DI ADIACENZA PER GRAFI ETICHETTATI

NEL CASO IL GRAFO SIA ETICHETTATO POSSIAMO ASSOCIARE AL NODO ALTRE INFORMAZIONI.



POSSIAMO SEMPRE UTILIZZARE LA RAPPRESENTAZIONE CON MATRICE DI ADIACENZA.

9

LA MATRICE DI ADIACENZA

	LABEL	MARK	ARCHI	RIGA				
n=1	10	1	3	0	1	1	1	
n=2	22	1	3	0	0	1	0	
n=3	17	1	3	0	0	0	0	
n=4	13	1	3	0	1	1	0	
n=5	24	0	2					

MARK E' UN FLAG CHE HA VALORE FALSO O 0 SE IL NODO E' STATO RIMOSSO.

ARCHI CONTIENE IL NUMERO SOMMA DEGLI ENTRANTI h E USCENTI k DAL GENERICO NODO.

10

RAPPRESENTAZIONE CON MATRICI D'INCIDENZA

UN GRAFO G = (N,A) PUO' ANCHE ESSERE RAPPRESENTATO MEDIANTE UNA MATRICE (n x m), B = [b ik], NELLA QUALE CIASCUNA RIGA RAPPRESENTA UN NODO E CIASCUNA COLONNA RAPPRESENTA UN ARCO.

PER UN GRAFO NON ORIENTATO

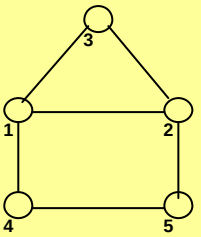
$$b_{ij} = \begin{cases} 1 & \text{SE L'ARCO j-ESIMO E' INCIDENTE} \\ & \text{NEL NODO i} \\ 0 & \text{ALTRIMENTI} \end{cases}$$

NEL CASO DI GRAFI ORIENTATI O DIRETTI IL GENERICO ELEMENTO DI B DIVIENE

$$b_{ij} = \begin{cases} +1 & \text{SE L'ARCO j-ESIMO ENTRA NEL NODO i} \\ -1 & \text{SE L'ARCO j-ESIMO ESCE DAL NODO i} \\ 0 & \text{ALTRIMENTI} \end{cases}$$

11

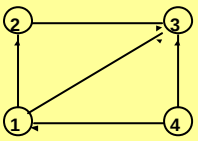
RAPPRESENTAZIONE CON MATRICI D'INCIDENZA GRAFO NON ORIENTATO



	(1,2)	(1,3)	(1,4)	(2,3)	(2,5)	(4,5)
1	1	1	1	0	0	0
2	1	0	0	1	1	0
3	0	1	0	1	0	0
4	0	0	1	0	0	1
5	0	0	0	0	1	1

12

RAPPRESENTAZIONE CON MATRICI D'INCIDENZA GRAFO ORIENTATO



	(1,2)	(1,3)	(2,3)	(4,1)	(4,3)
1	-1	-1	0	+1	0
2	+1	0	-1	0	0
3	0	+1	+1	0	+1
4	0	0	0	-1	-1

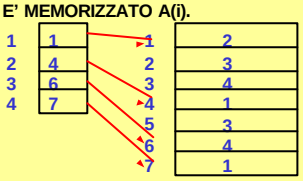
DATO UN NODO NON E' FACILE RICAVARE L'INSIEME DI ADIACENZA.

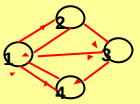
PER CALCOLARE A(u) E' NECESSARIO SCANDIRE LA RIGA u DI B ALLA RICERCA DELLE COLONNE k $\neq b_{uk} = -1$, E PER OGNI COLONNA k SCANDIRE L'INDICE DI RIGA i $\neq b_{ik} = +1$.

13

RAPPRESENTAZIONE CON VETTORI DI ADIACENZA (GRAFO ORIENTATO)

E' POSSIBILE RAPPRESENTARE IL GRAFO (N,A) CON DUE VETTORI, IL VETTORE NODI E IL VETTORE ARCHI. IL VETTORE NODI E' FORMATO DA N ELEMENTI E NODI(i) CONTIENE UN CURSORE ALLA POSIZIONE DI ARCHI A PARTIRE DALLA QUALE E' MEMORIZZATO A(i).





NEL CASO IL GRAFO SIA ETICHETTATO POSSIAMO ASSOCIARE AL NODO ALTRE INFORMAZIONI.

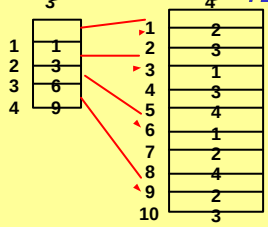
14

RAPPRESENTAZIONE CON VETTORI DI ADIACENZA (GRAFO NON ORIENTATO)

E' NECESSARIO RAPPRESENTARE OGNI ARCO [i,j] DUE VOLTE.



SE I GRAFI SONO ETICHETTATI SUI NODI E/O SUGLI ARCHI I PESI POSSONO ESSERE MEMORIZZATI IN VETTORI PESINODI(n) E PESIARCHI (m)

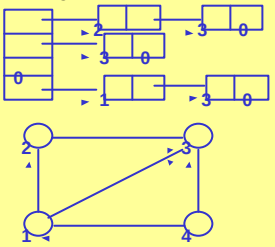


15

RAPPRESENTAZIONE CON LISTE DI ADIACENZA

E' POSSIBILE ANCHE UTILIZZARE UN VETTORE DI NODI A (1..n) ED n LISTE.

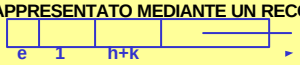
UNA GENERICA COMPONENTE A(i) DEL VETTORE E' IL PUNTATORE ALLA LISTA i-ESIMA IN CUI SONO MEMORIZZATI I NODI ADIACENTI A i.



16

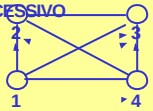
RAPPRESENTAZIONE CON STRUTTURA A PUNTATORI

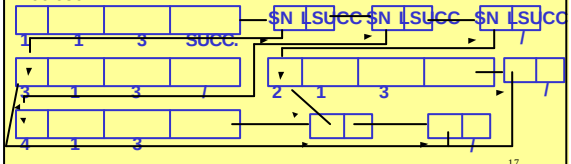
IL NODO v CON ETICHETTA e, h ARCHI ENTRANTI E k ARCHI USCENTI E' RAPPRESENTATO MEDIANTE UN RECORD



LABEL MARK ARCHI SUCCESSIVO

SUCCESSIVO CONTIENE UN RIFERIMENTO ALLA LISTA DEI SUCCESSORI DEL NODO. OGNI ELEMENTO DI QUESTA LISTA CORRISPONDE AD IN ARCO USCENTE.





17

VISITA DI UN GRAFO

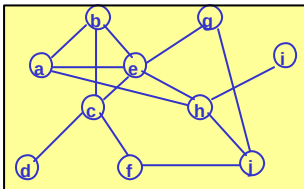
IL PRIMO SCHEMA DI VISITA E' QUELLO IN PROFONDITA'

(DFS ° DEPTH-FIRST-SEARCH)

L'ALGORITMO PREVEDE CHE CONTRASSEGNAMO UN VERTICE APPENA LO VISITIAMO E POI CERCHIAMO DI SPOSTARCI IN UN VERTICE ADIACENTE NON CONTRASSEGNATO. SE NON CI SONO VERTICI NON MARCATI INDIETREGGIAMO LUNGO I VERTICI GIA' VISITATI FINCHE' ARRIVIAMO AD UN VERTICE CHE RISULTA ADIACENTE AD UNO NON VISITATO E CONTINUIAMO COSI' IL PROCEDIMENTO. L'ALGORITMO IN PASCAL PUO' ESSERE:

```
PROCEDURE DFS ( VAR G:GRAFO; u:NODO);
VAR v:NODO;
BEGIN
      {ESAMINA IL NODO u E MARCALO "VISITATO"}
      FOR (TUTTI I v ? AD(u)) DO BEGIN
            {ESAMINA L'ARCO (u,v)}
            IF v NON E' "VISITATO" THEN
                  DFS(G,v);
            END;
END;
```

18



LE CHIAMATE RICORSIVE

- a — AD{b,e,h}
  - b — AD{(a),c,e}
    - c — AD{(b),d,e,f}
      - d — AD{(c)}
      - e — AD{(a),(b),(c),g,h}
        - g — AD{(e),j}

19

- j — AD{f,(g),h}
  - f — AD{(c),(j)}
    - h — AD{(a),(e),i,(j)}
      - i — AD{(h)}

20

LA VISITA IN AMPIEZZA

(BFS ° BREADTH-FIRST-SEARCH)

IN QUESTO SCHEMA OGNI VERTICE ADIACENTE AL VERTICE CORRENTE E' VISITATO PRIMA DI SPOSTARSI DAL VERTICE CORRENTE STESSO.

I NODI SONO VISITATI IN ORDINE DI DISTANZA CRESCENTE DAL NODO DI PARTENZA **u**, DOVE LA **DISTANZA** DA **u** AD UN GENERICO NODO **v** E' IL **MINIMO NUMERO DI ARCHI IN UN CAMMINO DA u A v.**

CONVIENE TENERE IN UNA CODA I VERTICI VISITATI MA NON COMPLETAMENTE "ESAMINATI" COSI' CHE, QUANDO SIAMO PRONTI A SPOSTARCI SU UN VERTICE ADIACENTE AL CORRENTE, POSSIAMO RITORNARE AL VECCHIO VERTICE CORRENTE DOPO IL NOSTRO MOVIMENTO.

21

UN ALGORITMO GENERALE

VISITA-DI-UN-GRAFO

ESAMINATO UN NODO INIZIALE i, SI VISITA UN NODO j COLLEGATO A i E, SUCCESSIVAMENTE, SI CONSIDERA UN NUOVO NODO e, DIVERSO DA i E DA j, E ADIACENTE A i OPPURE A j.

SI UTILIZZA UN INSIEME R DEI NODI VISITATI E UN INSIEME Q DEI NODI UTILI PER IL PROSEGUIMENTO DELLA RICERCA.

INFINE E' UTILIZZATO L'INSIEME B IN CUI VENGONO MEMORIZZATI GLI ARCHI UTILIZZATI PER LA SCOPERTA DEI NUOVI NODI

OUTPUT : R e B

22

ALGORITMO

```
Q:={i};          B:=0;
REPEAT
"SCEGLI UN NODO j Î Q";
IF (ESISTE UN NODO NON VISITATO ADIACENTE A j)
        THEN
                BEGIN
                "VISITA E PONI IN R E IN Q UNO O PIU' NODI
                k1,k2,... ADIACENTI A j E NON ANCORA VISITATI";
                "PONI IN B GLI ARCHI (j, k1),(j, k2),..."
                END
        ELSE "ELIMINA j DA Q"
UNTIL (Q E' VUOTO);
```

23

ALCUNE CONSIDERAZIONI

IL CRITERIO DI SCELTA DEL NODO STABILISCE LA MODALITA' DI VISITA.

NELLA RICERCA IN PROFONDITA' (**DEPTH-FIRST**), CHE E' DIRETTA ESTENSIONE DELLA VISITA IN ORDINE ANTICIPATO DI UN ALBERO RADICATO, TRA I VARI NODI UTILI PER IL PROSEGUIMENTO DELLA RICERCA VIENE SEMPRE SCELTO QUELLO VISITATO PIU' RECENTEMENTE.

NELLA RICERCA IN AMPIEZZA (**BREADTH-FIRST**) I NODI SONO VISITATI IN ORDINE DI DISTANZA CRESCENTE DAL NODO DI PARTENZA r, DOVE LA DISTANZA DA r AD UN GENERICO NODO u E' IL NUMERO DI ARCHI IN UN CAMMINO DA r AD u CHE SIA IL PIU' CORTO POSSIBILE. DUNQUE LA RICERCA IN AMPIEZZA FORNISCE SEMPRE UN CAMMINO "OTTIMO".

24

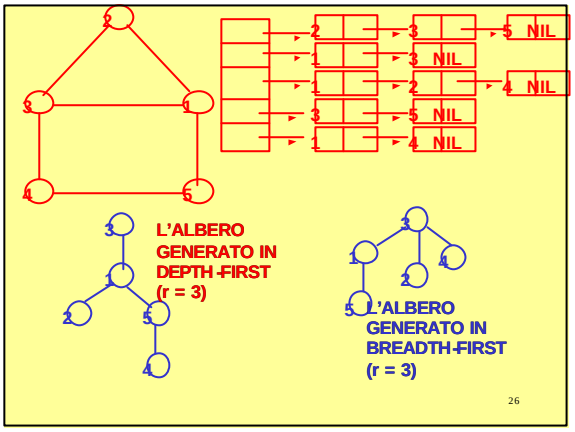
LE PROCEDURE **DFS** E **BFS** SONO METODI DI VISITA SISTEMATICA CHE VENGONO USATI PER RISOLVERE MOLTI PROBLEMI. AD ESEMPIO:

- ***VERIFICARE SE UN GRAFO NON ORIENTATO E' CONNESSO OPPURE NO***
- ***TROVARE TUTTI I SOTTOGRAFI CONNESSI DI UN GRAFO NON ORIENTATO (COMPONENTI CONNESSE)***

INOLTRE, SIA LA VISITA DFS CHE LA VISITA BFS DI UN GRAFO, CONNETTENDO I NODI MARCATI AD UNO AD UNO A PARTIRE DA UN GENERICO NODO i, **GENERANO UN ALBERO RADICATO NEL NODO INIZIALE i**.

QUESTO RISULTA INTERESSANTE QUANDO SI VOGLIA RISOLVERE UN PROBLEMA CHE RICHIEDE DI SELEZIONARE, TRA TUTTI I PERCORSI CHE CONNETTONO DUE NODI IN UN GRAFO, QUELLO CHE RISULTA OTTIMO, AD ESEMPIO IN BASE AL CRITERIO DI MINIMIZZAZIONE DELLA SOMMA DEI PESI ASSOCIATI AGLI ARCHI (**RICERCA DELL'ALBERO MINIMO DI COPERTURA O DI ATTRAVERSAMENTO**).

25



26