

Unità didattica: rappresentazione in memoria di un grafo

[2-T]

Titolo: Rappresentazione in memoria di una struttura dati
reticolare

Argomenti trattati:

- ✓ Grafo non orientato rappresentato mediante matrice di adiacenze
- ✓ Grafo orientato rappresentato mediante matrice di adiacenze
- ✓ Grafo rappresentato mediante lista di adiacenza

Prerequisiti richiesti: generalità sulla struttura dati grafo, matrici,
liste multiple

Tipo strutturato grafo

rappresentazione in memoria

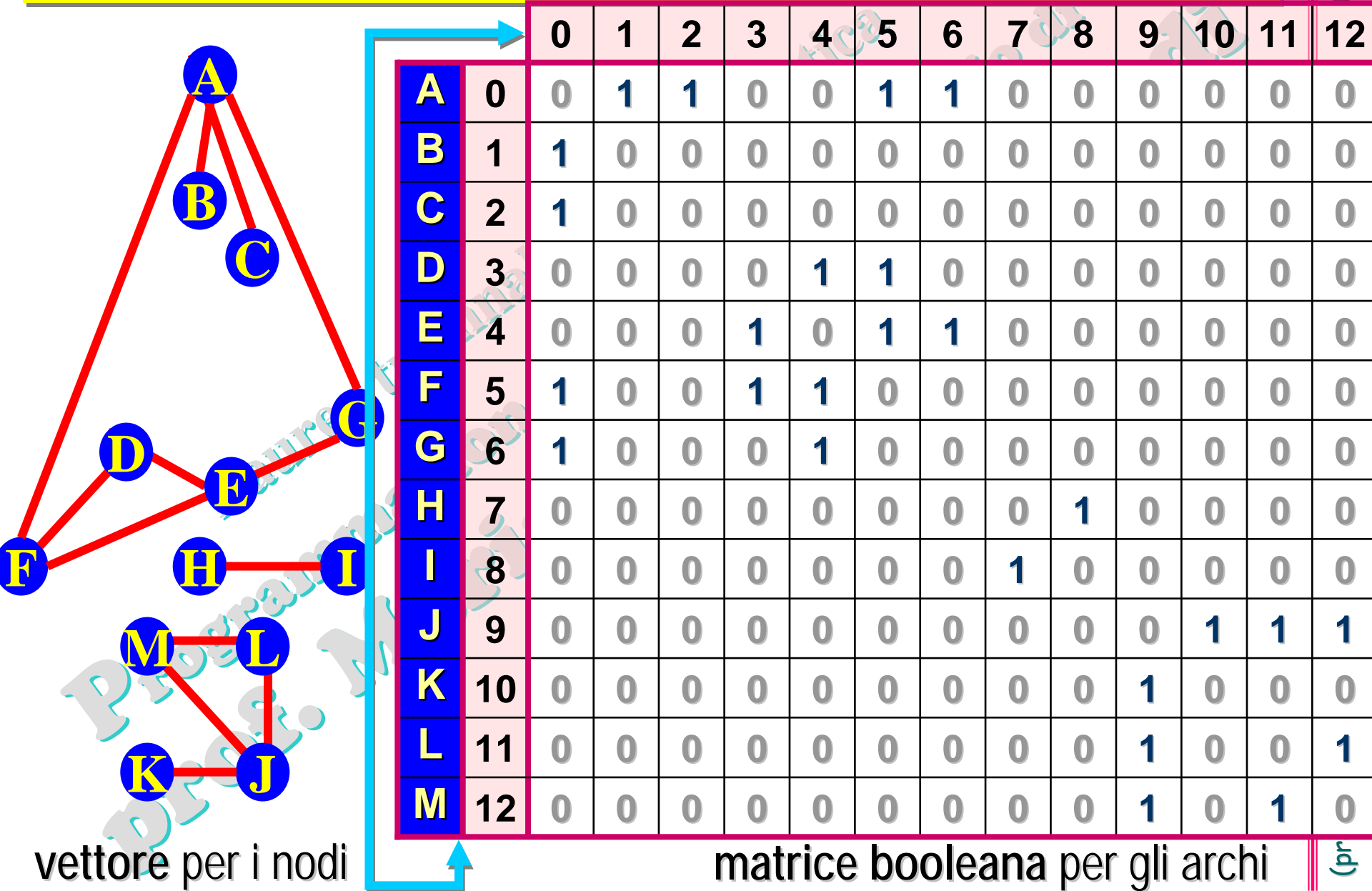
grafo



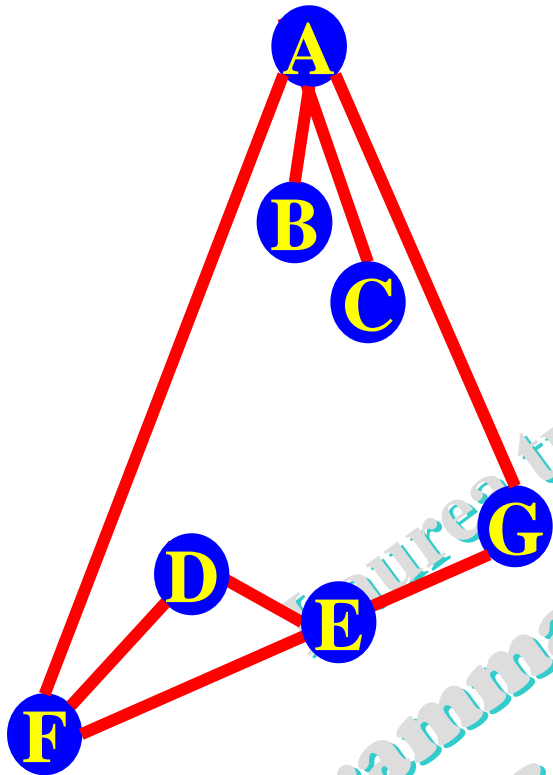
matrice di adiacenze

lista di adiacenze

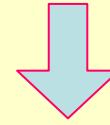
Esempio: grafo mediante matrice di adiancenze



Costruzione della matrice di adiacenze di un grafo non orientato



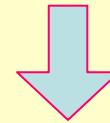
N=numero dei nodi



M=matrice di adiacenze

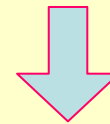
M($N \times N$)

grafo non orientato



matrice **M** simmetrica

INPUT



per ogni nodo sono elencati i nodi adiacenti

esempio:

(nodo) **A**

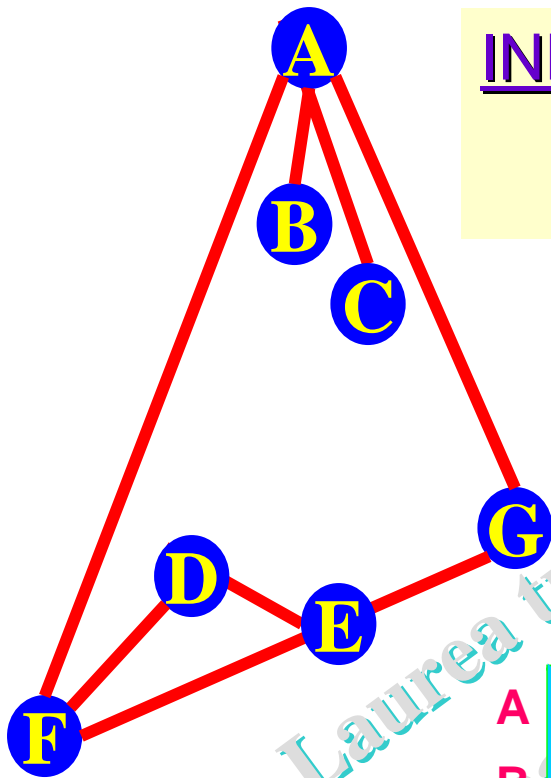
(adiac.) F,B,C,G

(nodo) **F**

(adiac.) A,D,E

...

Costruzione della matrice di adiacenze di un grafo non orientato



INPUT

per ogni nodo sono elencati i nodi adiacenti

Esempio:

	A	B	C	D	E	F	G
A	0	1	1	0	0	1	1
B	1	0	0	0	0	0	0
C	1	0	0	0	0	0	0
D	0	0	0	0	1	1	0
E	0	0	0	1	0	1	1
F	1	0	0	1	1	0	0
G	1	0	0	0	1	0	0

(nodo) **A** (adiac.) **F,B,C,G**

(nodo) **B** (adiac.) **A**

(nodo) **C** (adiac.) **A**

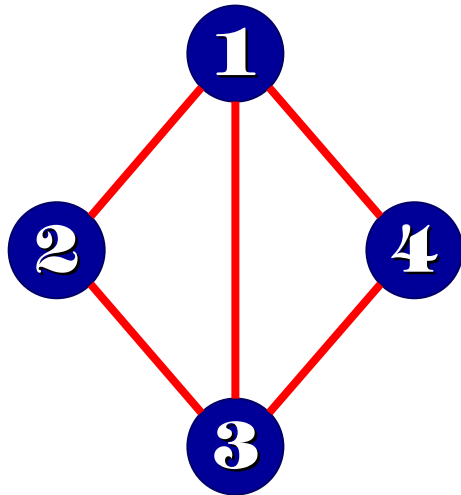
(nodo) **D** (adiac.) **E,F**

(nodo) **E** (adiac.) **F,D,G**

(nodo) **F** (adiac.) **A,D,E**

(nodo) **G** (adiac.) **A,E**

uso della matrice di adiacenze di grafi non orientati in MATLAB



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

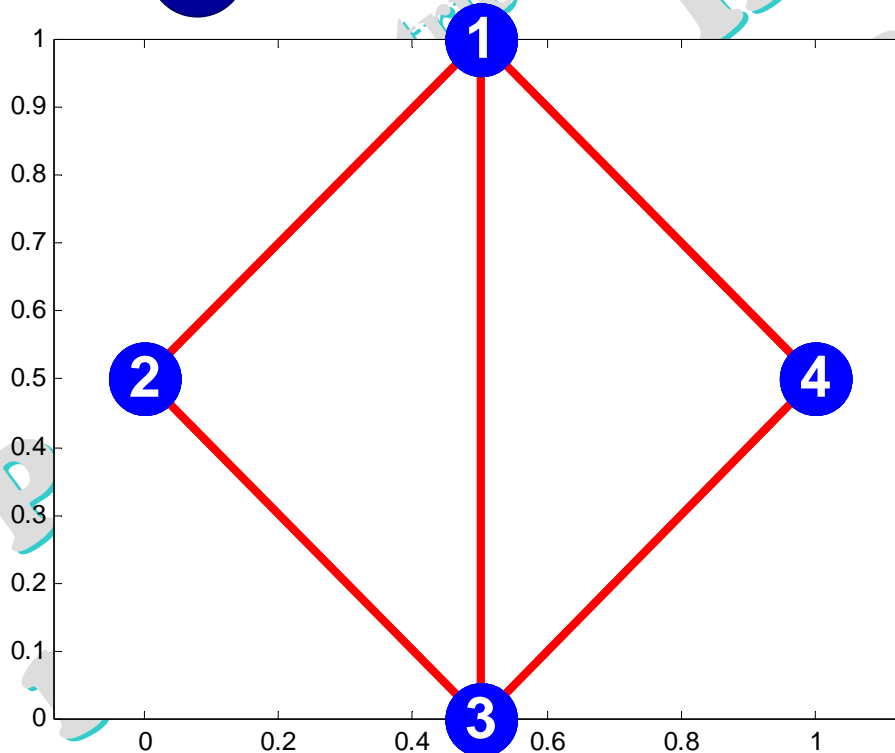
```
M = [0 1 1 1; 1 0 1 0;  
      1 1 0 1; 1 0 1 0];
```

```
X = [.5 0 .5 1]';  
Y = [1 .5 0 .5]';  
XY = [X Y];
```

```
gplot(M, XY, '-ro')
```

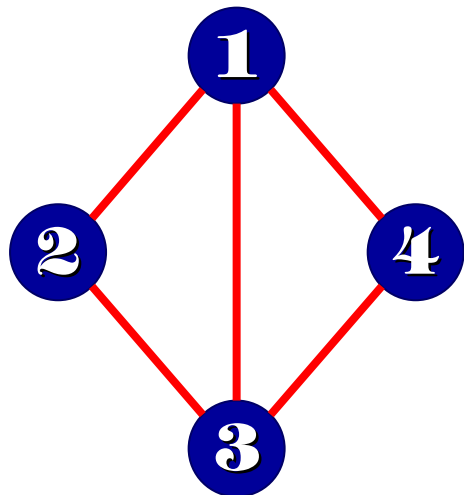
```
...
```

coordinate punti



Un'applicazione della matrice di adiacenze

... senza dettagli matematici!



grafo connesso

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

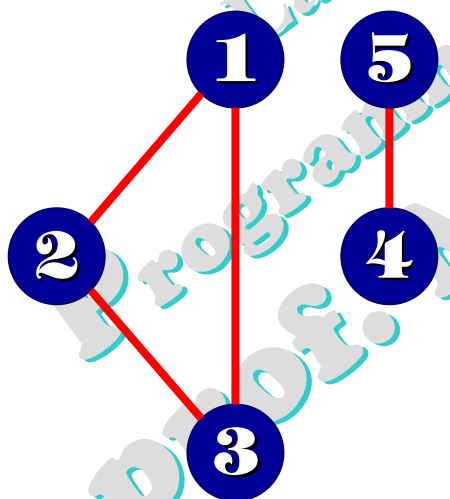
```
M=[0 1 1 1; 1 0 1 0; 1 1 0 1; 1 0 1 0];
```

```
eig(M) % autovalori
```

```
ans =
```

**Autovalori
tutti
diversi!**

-1.5616
-1.0000
0.0000
2.5616



grafo non connesso

	1	2	3	4	5
1	0	1	1	0	0
2	1	0	1	0	0
3	1	1	0	0	0
4	0	0	0	0	1
5	0	0	0	1	0

```
M=[0 1 1 0 0; 1 0 1 0 0; 1 1 0 0 0; 0 0 0 1; 0 0 0 1 0];
```

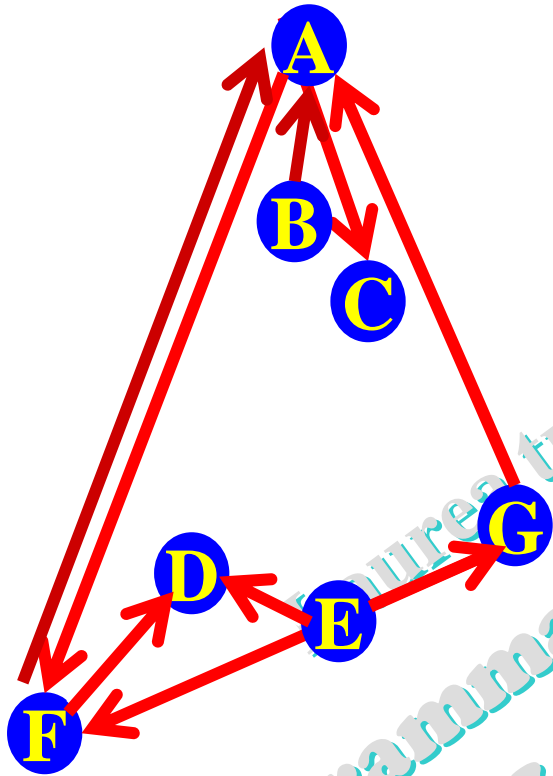
```
eig(M) % autovalori
```

```
ans =
```

**Autovalori
non tutti
diversi!**

-1.0000
-1.0000
-1.0000
1.0000
2.0000

Costruzione della matrice di adiacenze di un grafo orientato



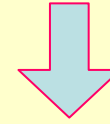
N=numero dei nodi



M=matrice di adiacenze

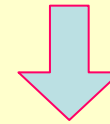
M($N \times N$)

grafo orientato



matrice **M** non simmetrica

INPUT



per ogni nodo sono elencati i nodi raggiungibili

esempio:

(nodo) **A**

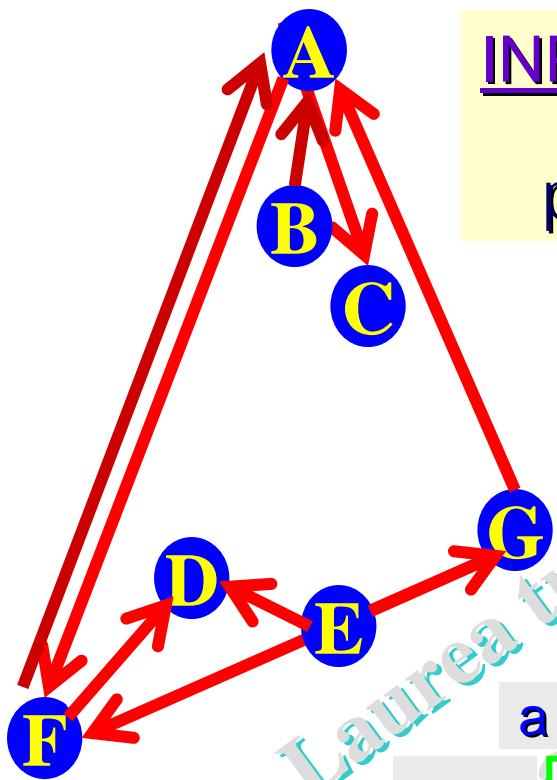
(ragg.) **F, C**

(nodo) **F**

(ragg.) **A, D**

...

Costruzione della matrice di adiacenze di un grafo orientato



INPUT

per ogni nodo sono elencati i nodi raggiungibili

Esempio:

		a	A	B	C	D	E	F	G
da	A	0	0	1	0	0	0	1	0
	B	1	0	0	0	0	0	0	0
	C	0	0	0	0	0	0	0	0
	D	0	0	0	0	0	0	0	0
	E	0	0	0	1	0	1	1	0
	F	1	0	0	1	0	0	0	0
	G	1	0	0	0	0	0	0	0

(nodo) **A**

(ragg.) F,C

(nodo) **B**

(ragg.) A

(nodo) **C**

(ragg.) -

(nodo) **D**

(ragg.) -

(nodo) **E**

(ragg.) F,D,G

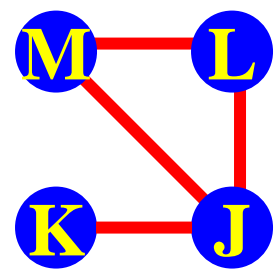
(nodo) **F**

(ragg.) A,D

(nodo) **G**

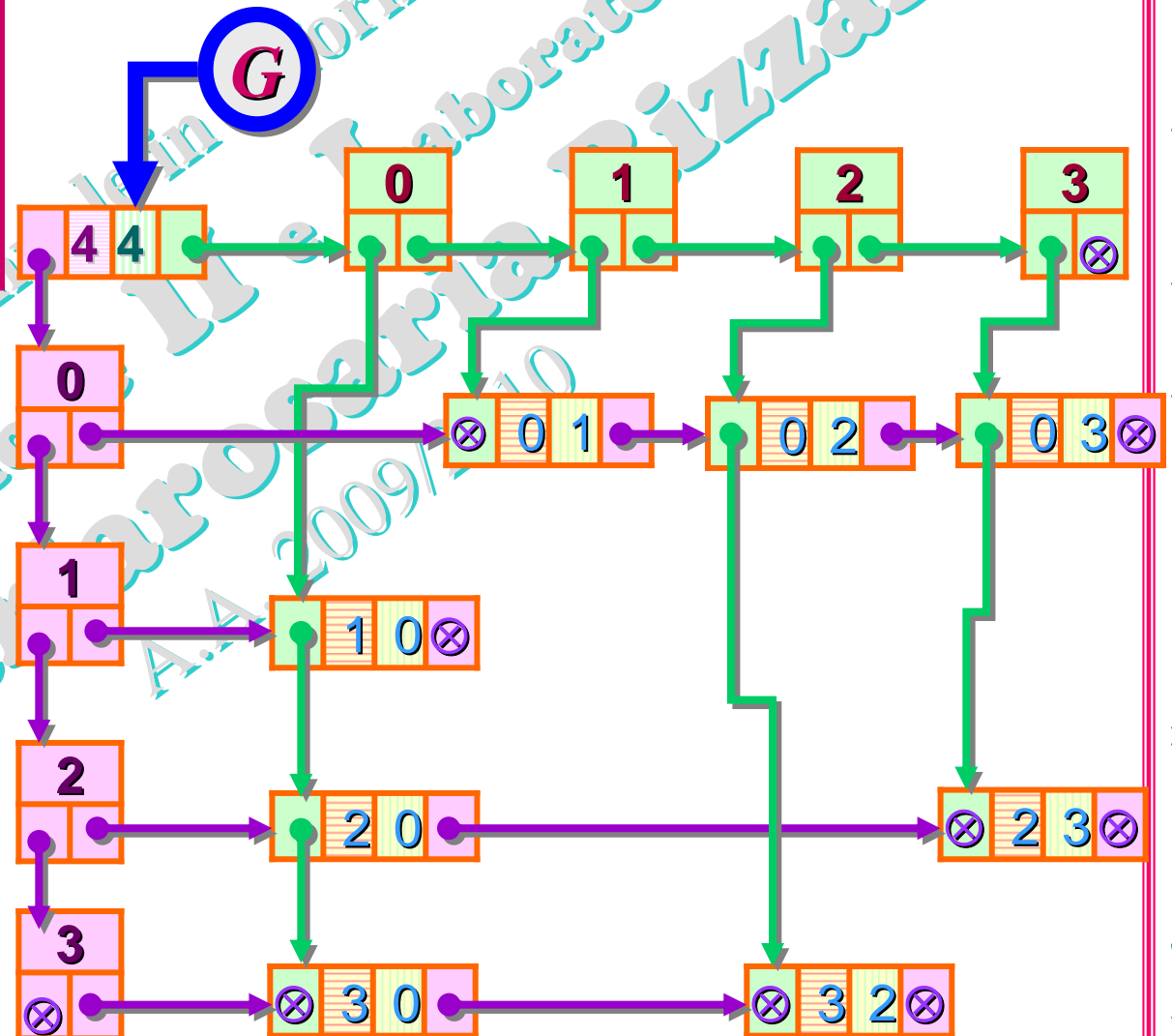
(ragg.) A

matrice di adiacenze di un grafo mediante lista multipla

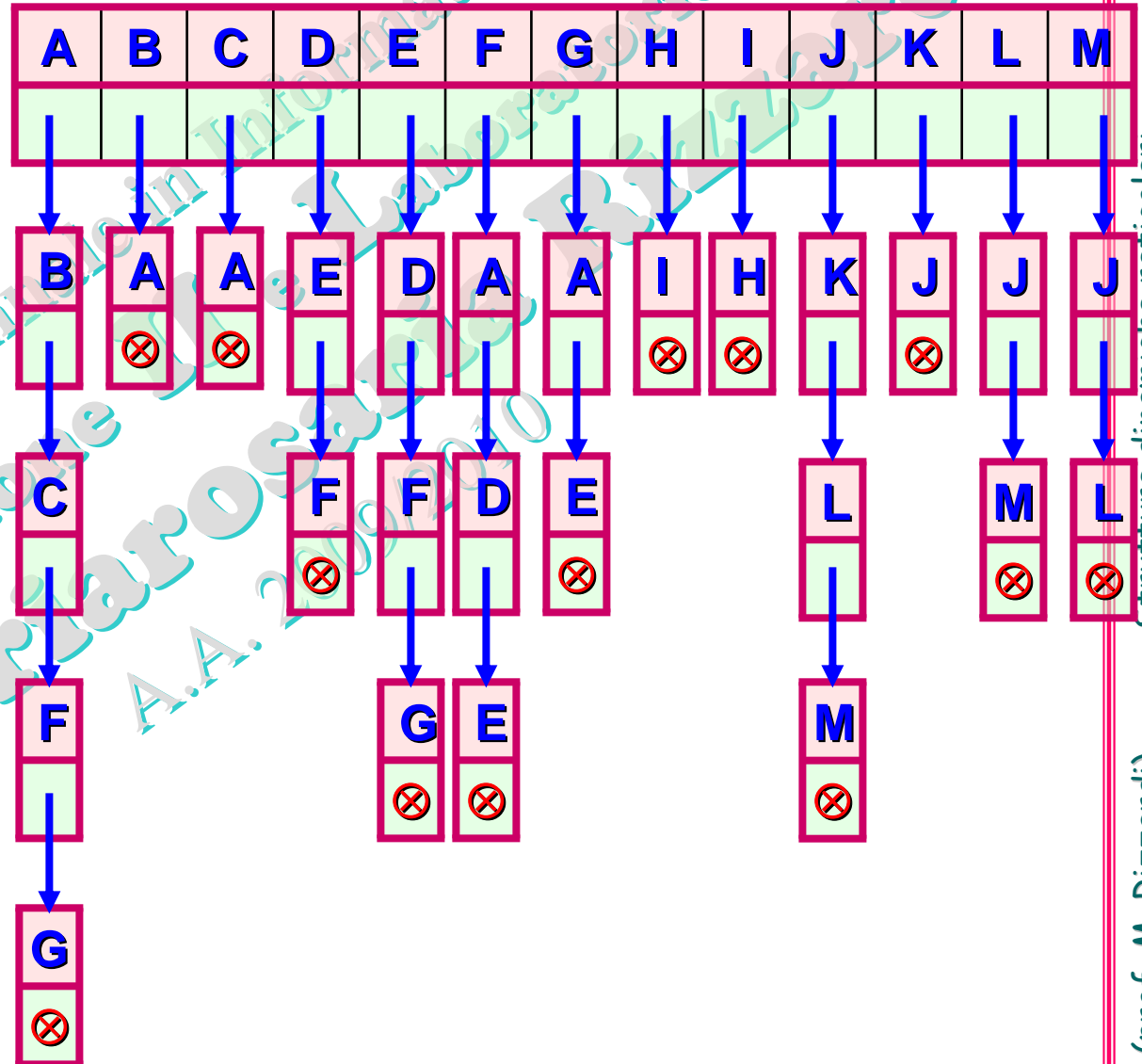
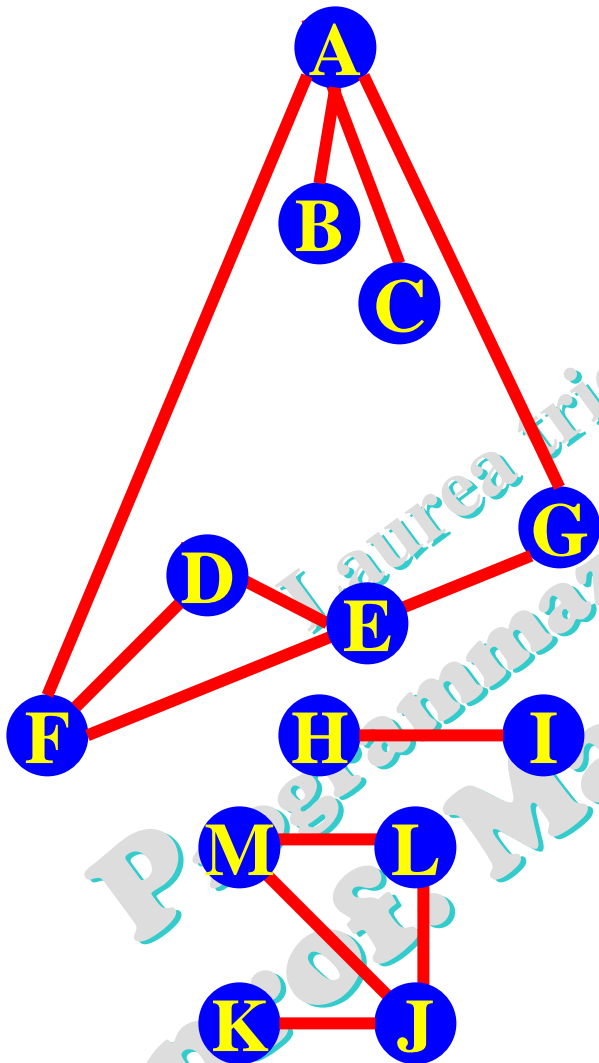


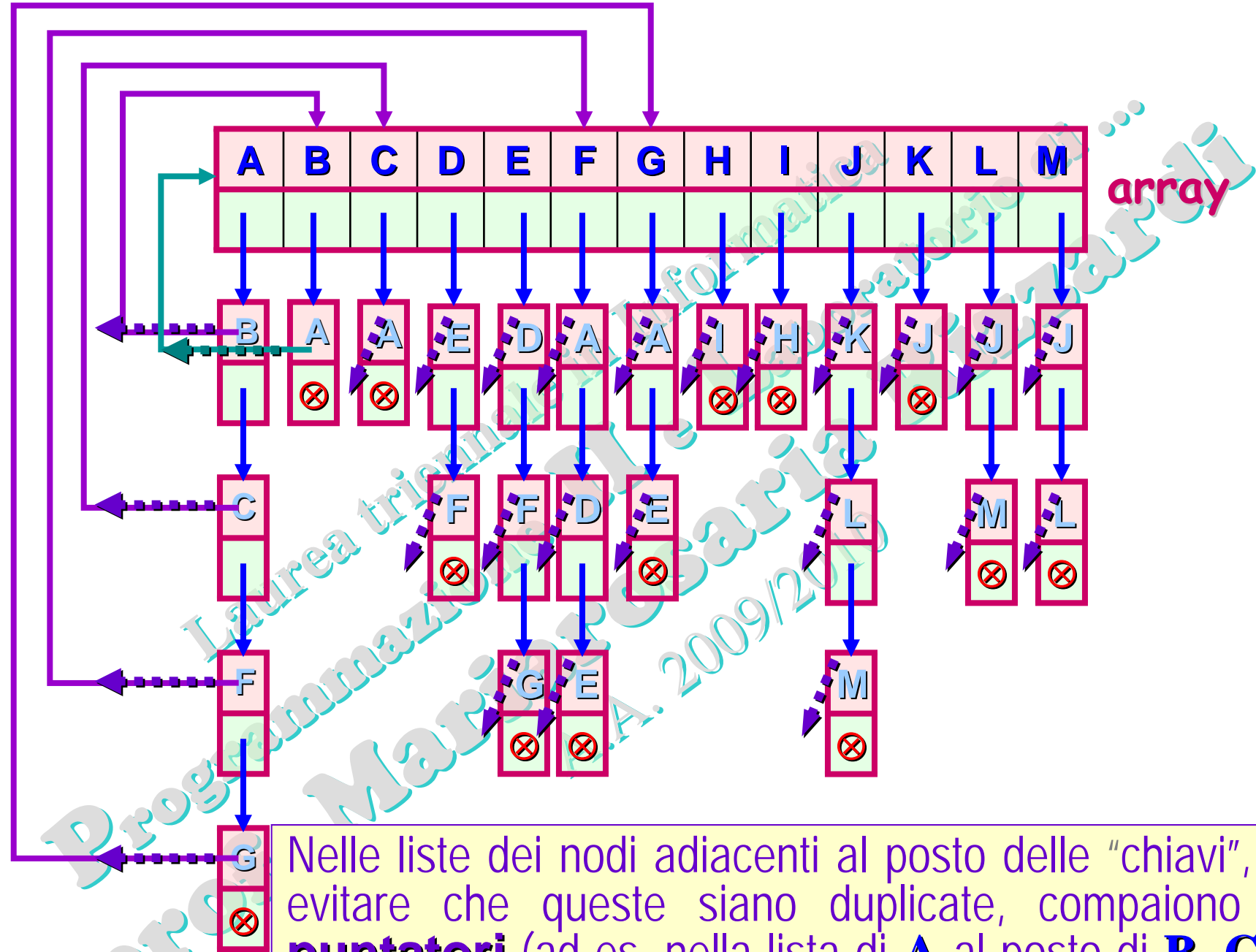
	0	1	2	3
J	0	0	1	1
K	1	1	0	0
L	2	1	0	1
M	3	1	0	0

Poiché spesso la matrice di adiacenze è sparsa, la si può rappresentare mediante una **lista multipla**.



Esempio: grafo mediante liste di adiancenze





Nelle liste dei nodi adiacenti al posto delle "chiavi", per evitare che queste siano duplicate, compaiono dei **puntatori** (ad es. nella lista di **A** al posto di **B**, **C**, **F**, ... ci sono dei **puntatori** rispettivamente a **B**, **C**, **F**, ...).

Esercizi

1 Scrivere *function C* per la costruzione di un **grafo non orientato** mediante matrice di adiacenze: in input per ogni nodo sono specificati quelli adiacenti. Scegliendo in input un nodo, scrivere *function C* che restituisca il suo grado.

2 Scrivere *function C* per la costruzione di un **grafo orientato** mediante matrice di adiacenze: in input per ogni nodo sono specificati quelli raggiungibili. Scegliendo in input un nodo, scrivere *function C* che restituisca il numero degli archi uscenti e quello degli archi entranti.

3 Scrivere *function C* per la costruzione di un grafo mediante liste di adiacenze. In input per ogni nodo sono specificati quelli adiacenti.
[liv. 2]