

# *Grafi: rappresentazioni in memoria*

corso di laurea in **Matematica**

*Informatica Generale*, **Ivano Salvo**

Lezione **21(b)** [5/12/23]



**SAPIENZA**  
UNIVERSITÀ DI ROMA

# Rappresentazioni in memoria

Usualmente i **nodi** di un grafo finito di  $n$  nodi vengono rappresentati **implicitamente** con i numeri in  $[1, n]$  (o  $[0, n)$ ).

Altre informazioni sui nodi (ad esempio il **colore**) possono essere rappresentate da un **vettore indicizzato sui nodi**.

Ci sono numerosi modi di rappresentare gli **archi di un grafo** in memoria. Tra questi, noi vedremo:

- liste di adiacenza
- matrice di adiacenza
- matrice di incidenza
- liste di archi

Al solito, la struttura dati può essere **più o meno buona** a seconda delle **operazioni** che sono richieste per risolvere un determinato problema.

Sono listate in **ordine di diffusione** e in particolare le **prime due** sono di gran lunga **le più usate**, a meno di validi motivi per scegliere altre rappresentazioni.

# Liste di adiacenza

Questa rappresentazione consiste in un **vettore di liste**.

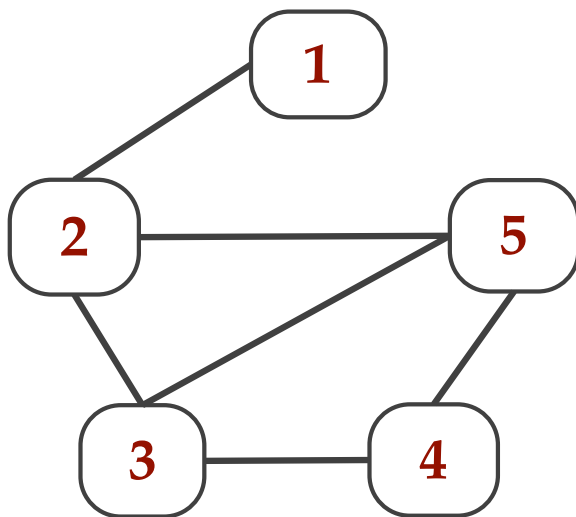
Il vettore  $adj$ , per ogni nodo  $v$ , contiene un puntatore alla testa della lista dei nodi adiacenti a  $v$ .

Rappresentando un grafo **non orientato**, ogni **arco**  $(u, v)$  **appare due volte**: nella lista  $adj[u]$  troveremo il nodo  $v$ , e nella lista  $adj[v]$  troveremo il nodo  $u$ .

Nei **grafi orientati** ogni **arco** appare **una sola volta**.

Altre informazioni (esempio: peso dell'arco) possono essere codificate anch'esse nella lista.

L'occupazione in memoria è  $\Theta(n+m)$



**Esempio:** Il grafo a sinistra può essere rappresentato dal seguente vettore  $adj$ :

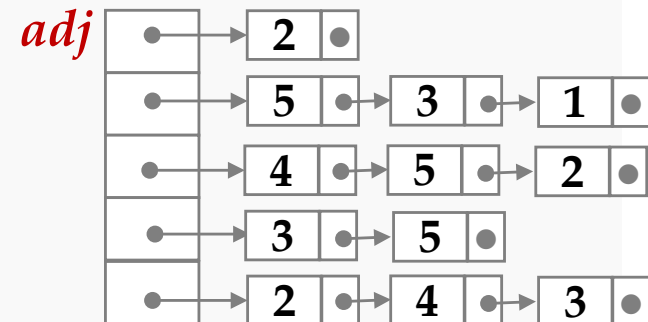
$adj[1] = [2]$

$adj[2] = [5, 3, 1]$

$adj[3] = [4, 5, 2]$

$adj[4] = [3, 5]$

$adj[5] = [2, 4, 3]$



# Matrice di adiacenza

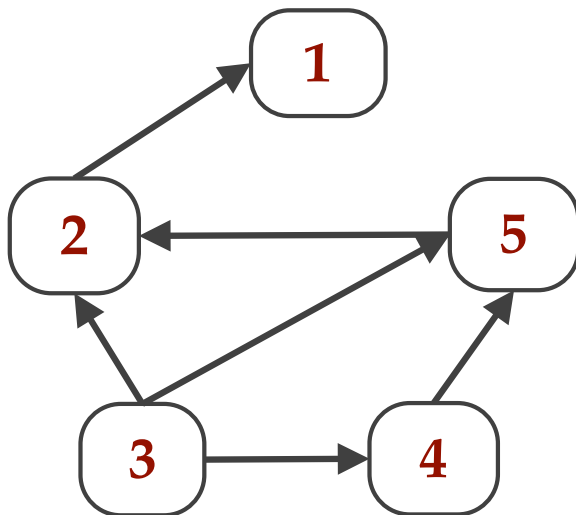
Questa rappresentazione consiste di una matrice **A** di dimensione  $n \times n$  indicizzata sui nodi del grafo.

L'arco  $(u, v)$  è presente se  $a[u][v]=1$ .

Nei grafi **non orientati**, la matrice è **simmetrica**, cioè  $a[u][v]=a[v][u]$ .

Anche qui, informazioni come il peso possono essere rappresentate nella matrice, sostituendo 1 con  $w(u, v)$ .

L'occupazione di memoria è  $\Theta(n^2)$ .



**Esempio:** La matrice rappresenta (**numeri verdi**) il grafo **diretto** a sinistra. **Tutti i numeri** la versione **non orientata**.

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

# Matrice di incidenza

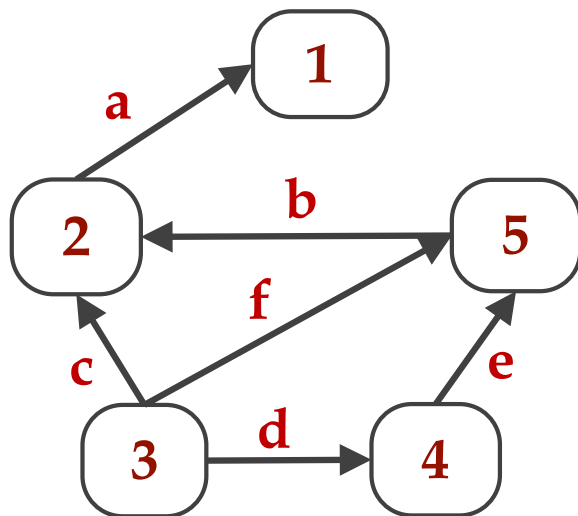
Questa rappresentazione consiste di una matrice **I** di dimensione  $n \times m$ , **righe** indicizzate sui **nodi** e **colonne** indicizzate sugli **archi**.

Si mette un 1 in  $I[u][e]$  se  $u$  è un estremo dell'arco  $e$ .

Nei **grafi orientati**, è possibile mettere **-1** nel nodo di **partenza** e **1** in quello di **arrivo**, cioè se  $e = u \rightarrow v$ ,  $I[u][e] = -1$  e  $I[v][e] = 1$ .

Anche qui, informazioni come il peso possono essere rappresentate nella matrice, sostituendo 1 con  $w(u, e)$ .

L'occupazione di memoria è  $\theta(n \cdot m)$ .



**Esempio:** Ricordando che nella versione **non orientata** ho tutti 1...

|   | a  | b  | c  | d  | e  | f  |
|---|----|----|----|----|----|----|
| 1 | -1 |    |    |    |    |    |
| 2 | 1  | -1 | -1 |    |    |    |
| 3 |    |    | 1  | 1  |    | 1  |
| 4 |    |    |    | -1 | 1  |    |
| 5 |    | 1  |    |    | -1 | -1 |

# Liste di archi

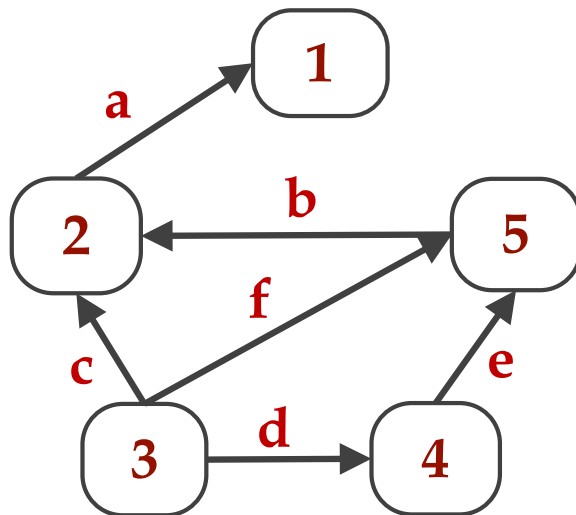
A dispetto del nome, in questa rappresentazione ho un **vettore**  $L$  di lunghezza  $m$  che contiene **per ogni arco la coppia di vertici**.

Equivalentemente posso usare due vettori di vertici.

I grafi orientati **si rappresentano nello stesso modo**, solo che si interpreta uno dei due vettori come il nodo da cui esce e l'altro come il nodo in cui entra l'arco.

Il peso può essere rappresentato aggiungendo un campo o in un terzo vettore.

L'occupazione di memoria è  $\theta(m)$ .



**Esempio:** La lista di archi è identica per versione orientata/non orientata.

| a | b | c | d | e | f |
|---|---|---|---|---|---|
| 2 | 5 | 3 | 3 | 4 | 3 |
| 1 | 2 | 2 | 4 | 5 | 5 |

# Confronto tra rappresentazioni

Vediamo ora di riassumere l'occupazione in memoria di un grafo e la complessità di alcune operazioni semplici: vedremo che le 4 rappresentazioni viste possono avere prestazioni molto diverse.

| Operazione<br>Strutt. Dati | Occupaz.<br>memoria | calcolare<br>$\deg(v)$                   | $(u, v) \in E?$                                                        |
|----------------------------|---------------------|------------------------------------------|------------------------------------------------------------------------|
| Liste di Adiacenza         | $\theta(n+m)$       | $\theta(\deg(v))$<br>si scorre $adj[v]$  | $\theta(\deg(u))$<br>scorre $adj[u]$                                   |
| Matrice di Adiacenza       | $\theta(n^2)$       | $\theta(n)$<br>si scorre <i>riga</i> $v$ | $\theta(1)$<br>legge $a[u][v]$                                         |
| Matrice di Incidenza       | $\theta(n \cdot m)$ | $\theta(m)$<br>si scorre <i>riga</i> $v$ | $\theta(m)$<br>si scorre <i>riga</i> $u$<br>poi stessa <i>col.</i> $v$ |
| Lista di Archi             | $\theta(m)$         | $\theta(m)$<br>si scorre vettore         | $\theta(m)$<br>si scorre vettore                                       |



# *Alberi come grafi*

# Alberi come grafi

I disegni con pallocchi e linee che congiungono i pallocchi dovrebbero suggerire che anche **gli alberi sono grafi**. E così è.

In realtà, abbiamo visto sempre un particolare tipo di albero, detto **albero radicato**, con un nodo speciale detto **radice** e **archi orientati**.

Gli **alberi radicati** sono **LA struttura induttiva**, mentre i grafi spesso non si mostrano molto maneggevoli per induzione.

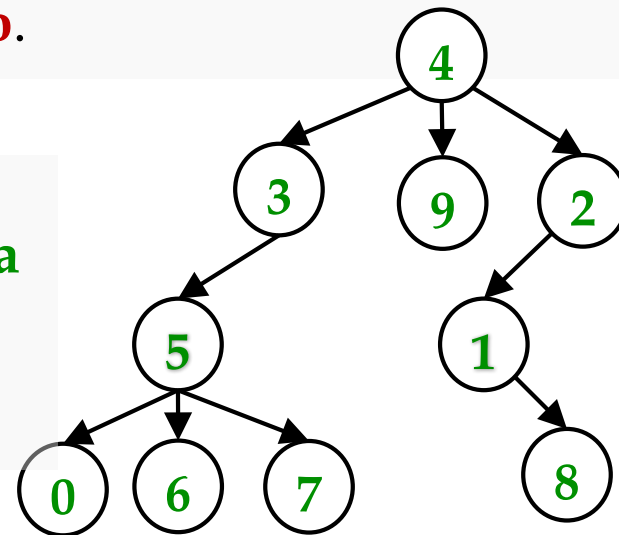
Ma visti come grafi, gli alberi che grafi sono?

Tutti i nodi sono legati alla radice e c'è un **unico cammino** tra radice-foglia e, in generale, **un unico cammino** tra **due nodi**..

Un **albero** è un grafo **connesso** e **aciclico**.

Notate che la **direzione delle frecce** è completamente determinata dalla **scelta della radice**.

Come se appendete una catenella...



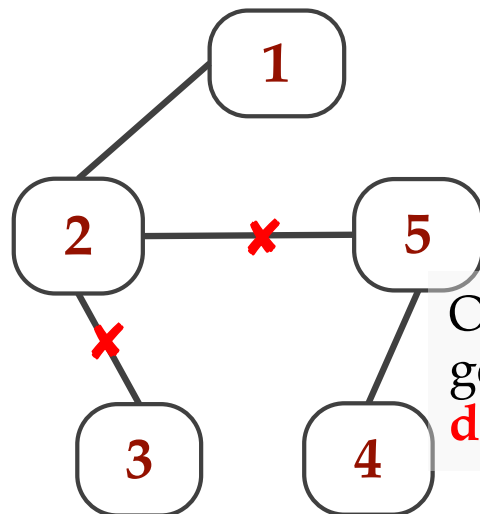
# Caratterizzazioni degli Alberi (1)

**Proposizione:** Sia  $A$  un albero  $(V, E)$ .

1. togliendo un arco ad  $A$ , si ottiene un grafo sconnesso.
2. aggiungendo un arco ad  $A$ , si genera un ciclo;

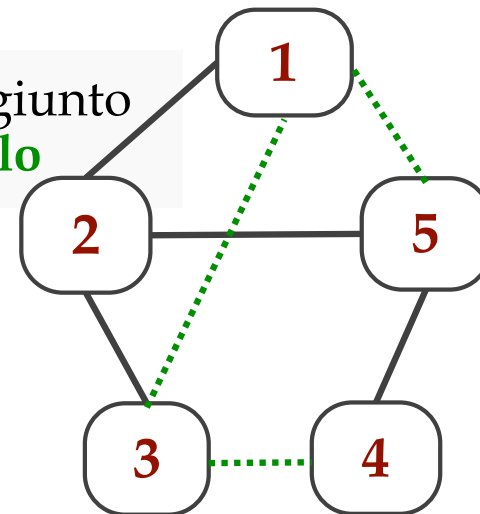
**Dim:** (1) Rimuoviamo l'arco  $(u, v)$ . Essendo aciclico, c'è un unico cammino tra ciascuna coppia di nodi  $u$  a  $v$ . In particolare,  $u$  e  $v$  **dopo la rimozione** saranno in due **componenti connesse diverse**, altrimenti il cammino da  $u$  a  $v$  più l'arco  $(u, v)$  genererebbe un ciclo.

(2) Aggiungiamo l'arco  $(u, v)$ . L'albero è un grafo connesso e quindi **prima dell'aggiunta**, c'è già **un cammino da  $u$  a  $v$** . E quindi l'arco  $(u, v)$  **genera un ciclo**.  $\square$



Ogni arco rimosso genera una **disconnessione**

Ogni arco aggiunto genera un **ciclo**



## Caratterizzazione degli Alberi (2)

**Proposizione:** Sia  $G = (V, E)$  un grafo.

Allora, le seguenti affermazioni sono equivalenti:

1.  $G$  è un albero;
2.  $G$  è connesso e  $|E| = |V| - 1$ .

**Dim: (1  $\Rightarrow$  2, Induz.)** La proposizione è banale per  $|V| = 1$  o  $|V| = 2$ .  
Togliendo un arco, otteniamo due componenti connesse  $(E_1, V_1)$  e  $(E_2, V_2)$  e per ipotesi induttiva hanno  $|E_1| = |V_1| - 1$  e  $|E_2| = |V_2| - 1$ .  
Il grafo completo ha invece  $1 + |E_1| + |E_2|$  archi, cioè  $1 + |V_1| - 1 + |V_2| - 1 = |V_1| + |V_2| - 1 = |V| - 1$  archi.

**(2  $\Rightarrow$  1)** supponiamo ci sia un ciclo  $u_1, u_2, \dots, u_k, u_1$  ( $k > 1$ ).

Il **ciclo ha  $k$  archi** e quindi non contiene tutti i nodi, altrimenti  $k = |V|$ , contro le ipotesi.

Visto che il grafo è connesso, costruiamo una sequenza di insiemi in cui  $C_0 = \{u_1, u_2, \dots, u_k\}$  è il ciclo e  $C_{i+1} = C_i \cup \{z\}$ , con  $(u, z) \in E$ ,  $u \in C_i$  e  $z \in V \setminus C_i$  in modo che  $C_{n-k} = V$ .

Ma ciò è **assurdo**, perché servono  $n - k$  archi (gli  $z$  sono tutti distinti) e quindi avrei  $n$  archi nel grafo, contro l'ipotesi che siano  $n - 1$ .  $\square$

***That's all Folks!***

corso di laurea in **Matematica**

*Informatica Generale*, **Ivano Salvo**

Esercizi lez. **21(b)** [5/12/23]



**SAPIENZA**  
UNIVERSITÀ DI ROMA