ALGORITMI E STRUTTURE DATI

Prof. Manuela Montangero

A.A. 2022/23

STRUTTURE DATI: grafi

UNIVERSITÀ DEGLI STUDI DI MODENA E REGGIO EMILIA

GRAFI

Qual e' l'ordine giusto con cui indossare gli indumenti?

GRAFI

Qual e' la strada più breve per andare da TORINO a TARANTO?

GRAFI

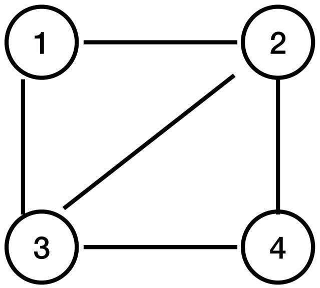
GRAFO G = (V,E)

V = insieme dei vertici (nodi, *vertices*)

E = insieme degli archi (*edges*) $E \subseteq V x V$

GRAFO NON DIRETTO (non orientato):
le coppie di E non sono ordinate

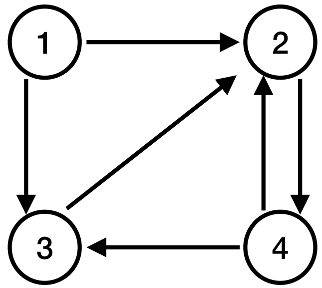
non orientato



V = {1,2,3,4}
E = (1,2),(1,3),(2,3),(2,4),(3,4)}

GRAFO DIRETTO (orientato):
le coppie di E sono ordinate

orientato



V = {1,2,3,4}
E = (1,2),(1,3),(3,2),(2,4),(4,2),(4,3)}

GRAFI

GRAFO G = (V,E)

V = insieme dei vertici (nodi, *vertices*)

E = insieme degli archi (*edges*) $E \subseteq V \times V$

GRAFO NON DIRETTO (non orientato):
le coppie di E non sono ordinate

nodo v **adiacente** a nodo u se
esiste la **coppia** $(v,u) \in E$

arco (u,v) **incidente** in u e v

GRAFO DIRETTO (orientato):
le coppie di E sono ordinate

arco (v,u) **orientato** dal nodo v
al nodo u se
esiste la **coppia** $(v,u) \in E$

arco (u,v) **incidente** in u e v

arco (u,v) **uscente** da u

arco (u,v) **entrante** in v

GRAFI

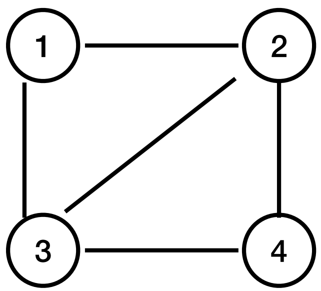
GRAFO G = (V,E)

V = insieme dei vertici (nodi, *vertices*)

E = insieme degli archi (*edges*) $E \subseteq V \times V$

GRAFO NON DIRETTO (non orientato):
le coppie di E non sono ordinate

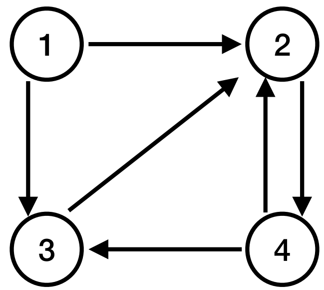
grado del nodo v (deg(v)):
numero di archi $(v,u) \in E$



deg(3) = 3

GRAFO DIRETTO (orientato):
le coppie di E sono ordinate

grado entrante in-deg(v)
(risp. **uscente out-deg(v)**)
in (risp. da) v:
numero di archi $(\mathbf{v},u) \in E$ (risp. $(u,\mathbf{v}) \in E$)



in-deg(3) = 2, out-deg(3) =1

GRAFI

GRAFO G = (V,E)

V = insieme dei vertici (nodi, *vertices*)

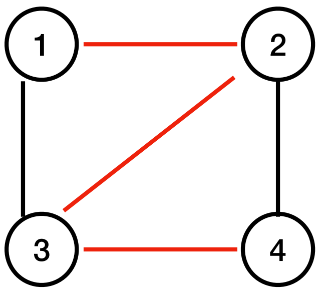
E = insieme degli archi (*edges*) $E \subseteq V \times V$

GRAFO NON DIRETTO (non orientato):
le coppie di E non sono ordinate

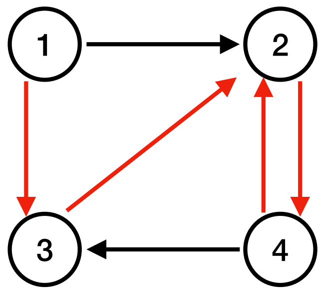
GRAFO DIRETTO (orientato):
le coppie di E sono ordinate

CAMMINO: sequenza di nodi $<v_0,v_1,v_2,v_3,\dots,v_k>$ tale che $(v_i, v_{i+1}) \in E$ per $i=0,..,k-1$

LUNGHEZZA cammino = # archi = k



cammino <1,2,3,4>
lunghezza 3



cammino <1,3,2,4,2>
lunghezza 4

GRAFI

GRAFO $G = (V,E)$

V = insieme dei vertici (nodi, *vertices*)

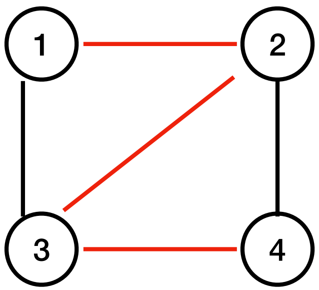
E = insieme degli archi (*edges*) $E \subseteq V \times V$

GRAFO NON DIRETTO (non orientato):
le coppie di E non sono ordinate

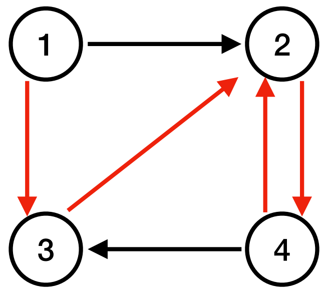
GRAFO DIRETTO (orientato):
le coppie di E sono ordinate

CAMMINO: sequenza di nodi $<v_0,v_1,v_2,v_3,\ldots,v_k>$ tale che $(v_i, v_{i+1}) \in E$ per $i=0,..,k-1$

CAMMINO SEMPLICE: cammino che non contiene nodi ripetuti



cammino semplice



cammino non semplice

GRAFI

GRAFO $G = (V,E)$

V = insieme dei vertici (nodi, *vertices*)

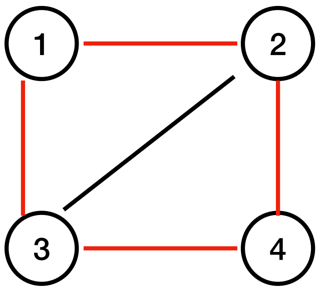
E = insieme degli archi (*edges*) $E \subseteq V x V$

GRAFO NON DIRETTO (non orientato):
le coppie di E non sono ordinate

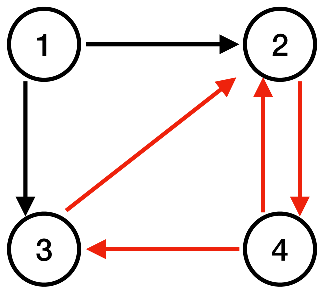
GRAFO DIRETTO (orientato):
le coppie di E sono ordinate

CAMMINO: sequenza di nodi $<v_0,v_1,v_2,v_3,\dots,v_k>$ tale che $(v_i, v_{i+1}) \in E$ per $i=0,..,k-1$

CICLO: sequenza $<v_0,v_1,v_2,v_3,\dots,v_k>$ cammino e $v_0 = v_k$



Ciclo: <1,2,4,3,1>



Ciclo: <3,2,4,2,4,3>

GRAFI

GRAFO G = (V,E)

V = insieme dei vertici (nodi, *vertices*)

E = insieme degli archi (*edges*) $E \subseteq V \times V$

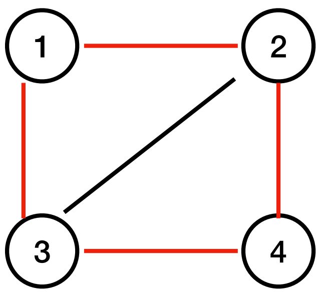
GRAFO NON DIRETTO (non orientato):
le coppie di E non sono ordinate

GRAFO DIRETTO (orientato):
le coppie di E sono ordinate

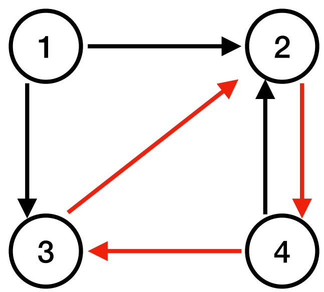
CAMMINO: sequenza di nodi $<v_0,v_1,v_2,v_3,\ldots,v_k>$ tale che $(v_i, v_{i+1}) \in E$ per $i=0,..,k-1$

CICLO: sequenza $<v_0,v_1,v_2,v_3,\ldots,v_k>$ cammino e $v_0 = v_k$

CICLO SEMPLICE: sequenza $<v_0,v_1,v_2,v_3,\ldots,v_k>$ cammino semplice e $v_0 = v_k$



Ciclo semplice: <1,2,4,3,1>



Ciclo semplice: <3,2,4,3>

GRAFI

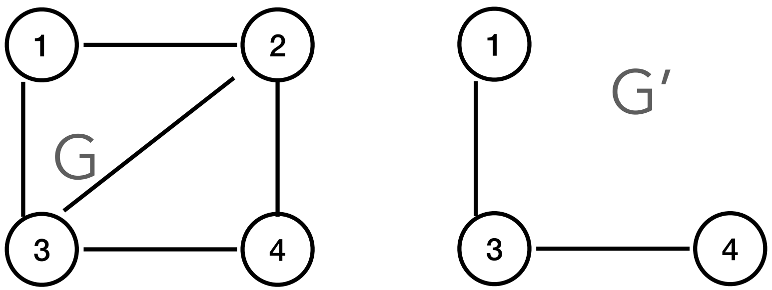
GRAFO G = (V,E)

V = insieme dei vertici (nodi, *vertices*)

E = insieme degli archi (*edges*) $E \subseteq V \times V$

G' = (V'E') è un **SOTTOGRAFO** di *G=(V,E)* se
$V' \subseteq V$ e $E' \subseteq E$

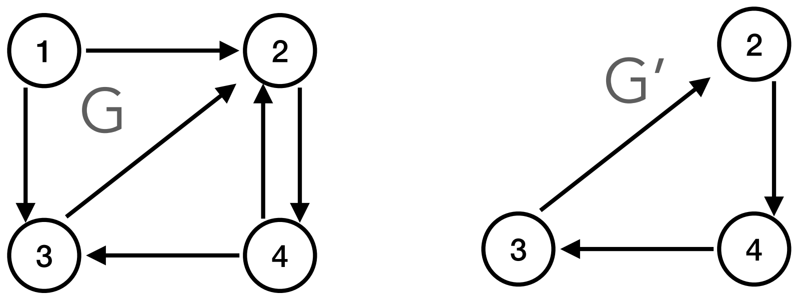
GRAFO NON DIRETTO (non orientato):
le coppie di E non sono ordinate



$V' = \{1,3,4\} \subseteq V$

$E' = \{(1,3),(3,4)\} \subseteq E$

GRAFO DIRETTO (orientato):
le coppie di E sono ordinate



$V' = \{2,3,4\} \subseteq V$

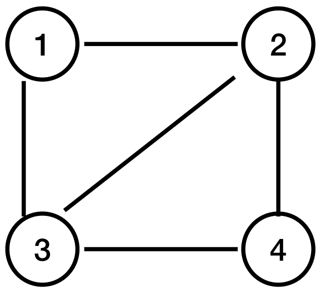
$E' = \{(2,4),(3,2),(4,3)\} \subseteq E$

GRAFI

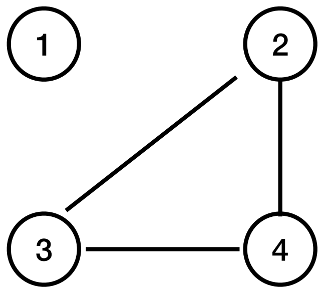
GRAFO G = (V,E)	V = insieme dei vertici (nodi, *vertices*)
	E = insieme degli archi (*edges*) $E \subseteq V \times V$

GRAFO NON DIRETTO (non orientato):
le coppie di E non sono ordinate

GRAFO CONNESSO
esiste un cammino che connette
ogni coppia di nodi del grafo



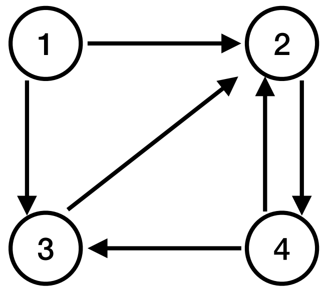
SI



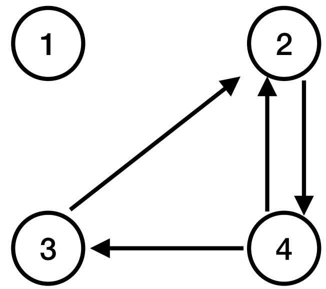
NO

GRAFO DIRETTO (orientato):
le coppie di E sono ordinate

GRAFO CONNESSO
esiste un cammino non necessariamente
diretto che connette
ogni coppia di nodi del grafo



SI



NO

GRAFI

GRAFO $G = (V,E)$

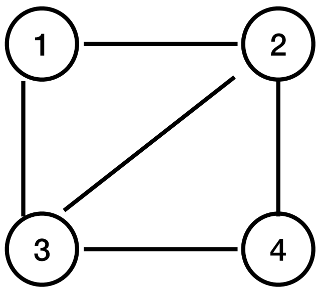
V = insieme dei vertici (nodi, *vertices*)

E = insieme degli archi (*edges*) $E \subseteq V \times V$

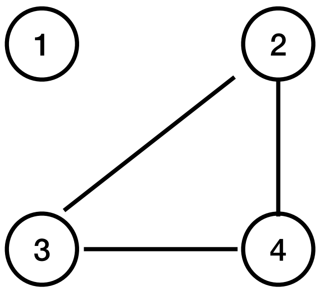
GRAFO NON DIRETTO (non orientato):
le coppie di E non sono ordinate

GRAFO DIRETTO (orientato):
le coppie di E sono ordinate

GRAFO CONNESSO
esiste un cammino che connette ogni coppia di nodi del grafo

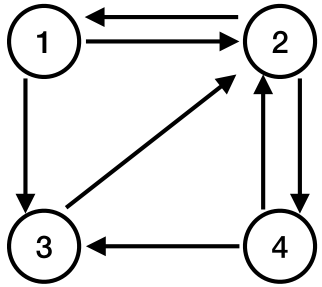


SI

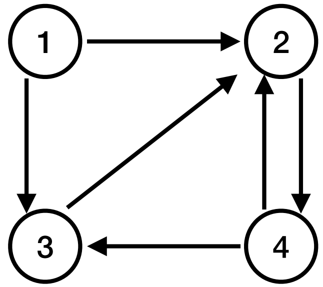


NO

GRAFO FORTEMENTE CONNESSO
esiste un cammino diretto che connette ogni coppia di nodi del grafo



SI



NO
ma connesso

GRAFI

GRAFO G = (V,E)

V = insieme dei vertici (nodi, *vertices*)

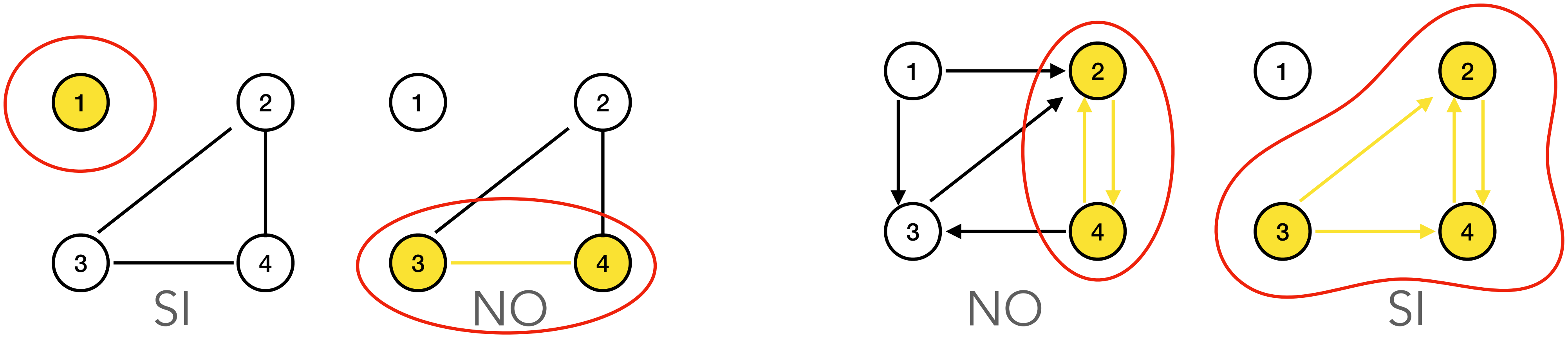
E = insieme degli archi (*edges*) $E \subseteq V \times V$

GRAFO NON DIRETTO (non orientato):
le coppie di E non sono ordinate

GRAFO DIRETTO (orientato):
le coppie di E sono ordinate

COMPONENTE CONNESSA G′ = (V′,E′):

- G′ è un sottografo connesso di G
- *G′* è massimale (cioè non esiste un sottografo connesso di *G″* di *G* che includa tale che $G' \subseteq G'' \subseteq G$)



GRAFI

GRAFO G = (V,E)

V = insieme dei vertici (nodi, *vertices*)

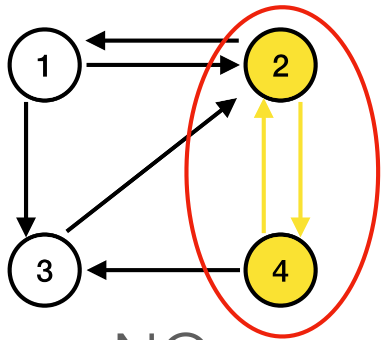
E = insieme degli archi (*edges*) $E \subseteq V \times V$

GRAFO NON DIRETTO (non orientato):
le coppie di E non sono ordinate

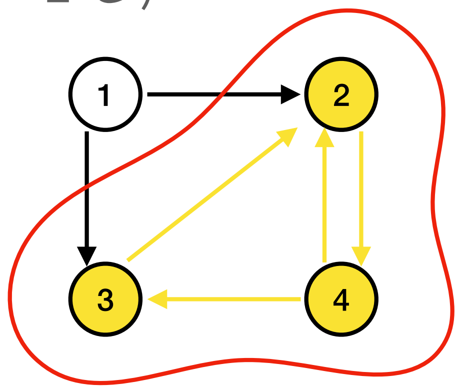
GRAFO DIRETTO (orientato):
le coppie di E sono ordinate

COMPONENTE FORTEMENTE CONNESSA

- G′ e′ un sottografo fortemente connesso di G
- *G′* è massimale (cioè non esiste un sottografo connesso di *G″* di *G* che includa tale che $G' \subseteq G'' \subseteq G$)



NO



SI

GRAFI

DIMENSIONI del GRAFO $G=(V,E)$

- $n = |V|$ numero dei nodi
- $m = |E|$ numero degli archi

Il **COSTO COMPUTAZIONALE** di algoritmi su grafi spesso è espresso in termini di n e m (es, $O(n + m)$)

GRAFI

DIMENSIONI del GRAFO G=(V,E)

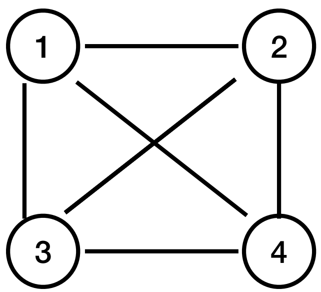
- n = |V| numero dei nodi
- m = |E| numero degli archi

Il **COSTO COMPUTAZIONALE** di algoritmi su grafi spesso e' espresso in termini di n e m (es, O(n + m))

CASI PARTICOLARI:

GRAFO COMPLETO: esiste un arco per ogni coppia di nodi

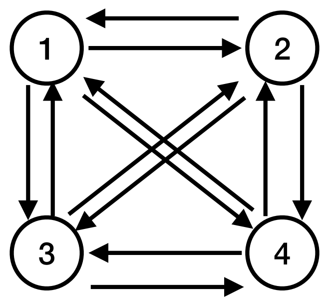
non orientato



n = 4
m = 6

orientato

n = 4
m = 12



In generale:

In generale:

GRAFI

DIMENSIONI del GRAFO G=(V,E)

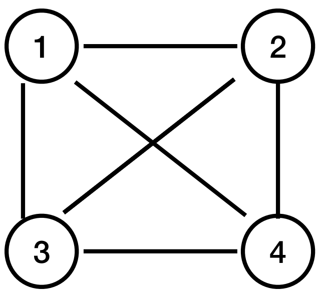
- n = |V| numero dei nodi
- m = |E| numero degli archi

Il **COSTO COMPUTAZIONALE** di algoritmi su grafi spesso e' espresso in termini di n e m (es, O(n + m))

CASI PARTICOLARI:

GRAFO COMPLETO: esiste un arco per ogni coppia di nodi

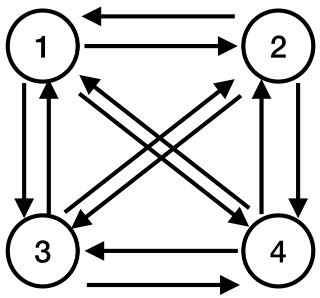
non orientato



n = 4
m = 6

orientato

n = 4
m = 12



In generale:

$$m = (n-1) + (n-2) + \ldots + 1 = \frac{(n-1)n}{2} \in \Theta(n^2)$$

In generale:

$$m = n \cdot (n-1) = n^2 - n \in \Theta(n^2)$$

GRAFI

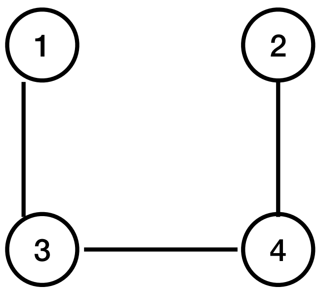
DIMENSIONI del GRAFO G=(V,E)

- n = |V| numero dei nodi
- m = |E| numero degli archi

Il **COSTO COMPUTAZIONALE** di algoritmi su grafi spesso e' espresso in termini di n e m (es, O(n + m))

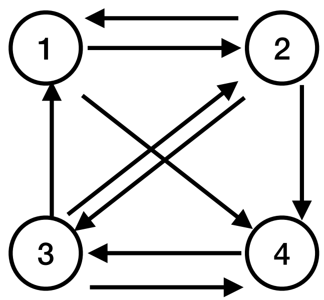
CASI PARTICOLARI:

GRAFO sparso: ha pochi archi (es, $m \in O(n)$)



Conviene usare le liste di adiacenza

GRAFO denso: ha molti archi (es, $m \in \Omega(n^2)$)



Conviene usare le matrici di adiacenza

GRAFI

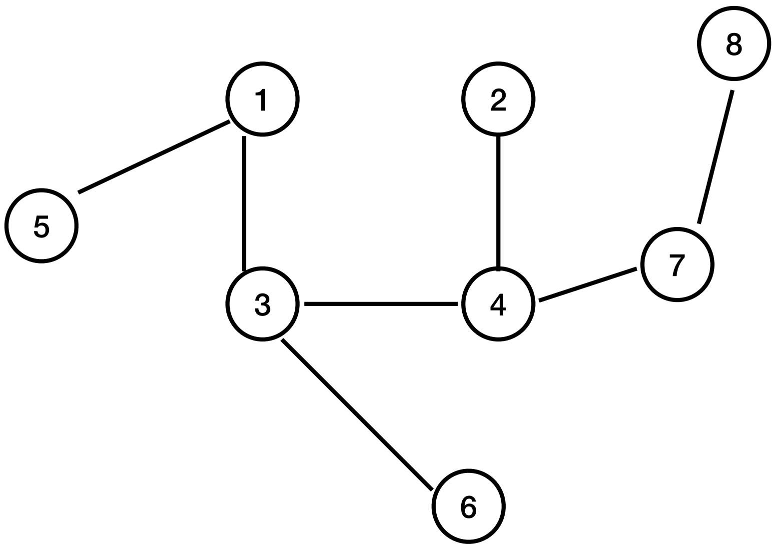
DIMENSIONI del GRAFO G=(V,E)

- n = |V| numero dei nodi
- m = |E| numero degli archi

Il **COSTO COMPUTAZIONALE** di algoritmi su grafi spesso e' espresso in termini di n e m (es, O(n + m))

CASI PARTICOLARI:

ALBERO libero: grafo aciclico connesso senza cicli (o tale che $m = n - 1$)



GRAFI

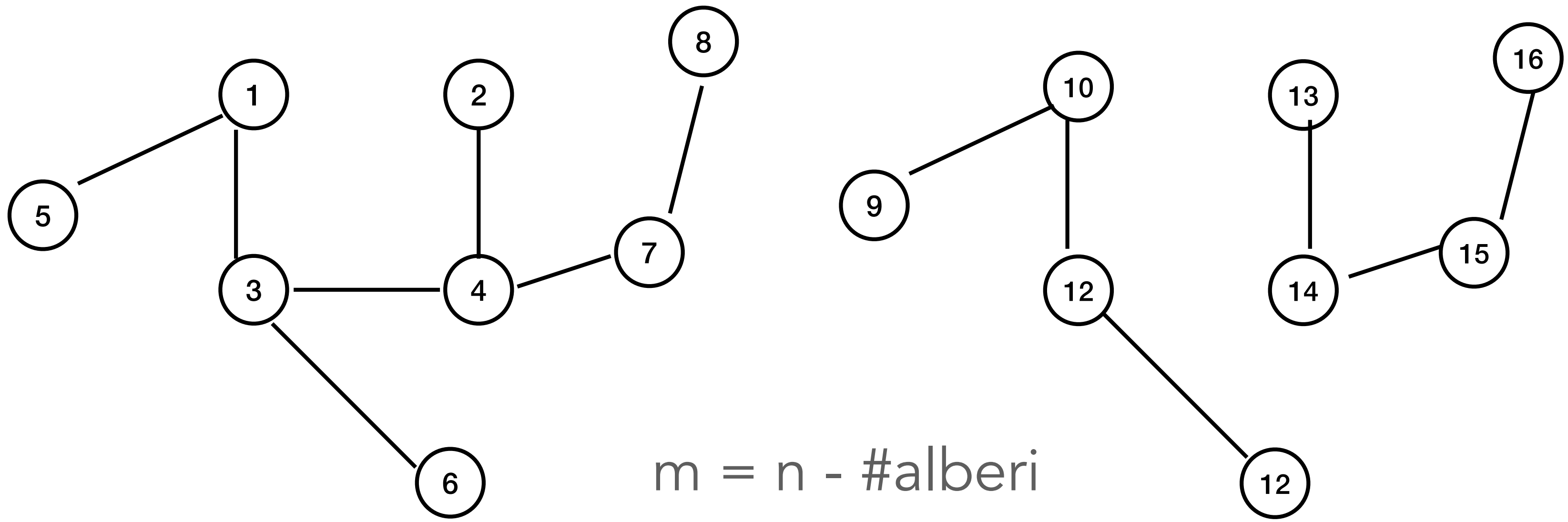
DIMENSIONI del GRAFO G=(V,E)

- n = |V| numero dei nodi
- m = |E| numero degli archi

Il **COSTO COMPUTAZIONALE** di algoritmi su grafi spesso e' espresso in termini di n e m (es, O(n + m))

CASI PARTICOLARI:

FORESTA: insieme di alberi



m = n - #alberi

UNIMORE

GRAFI

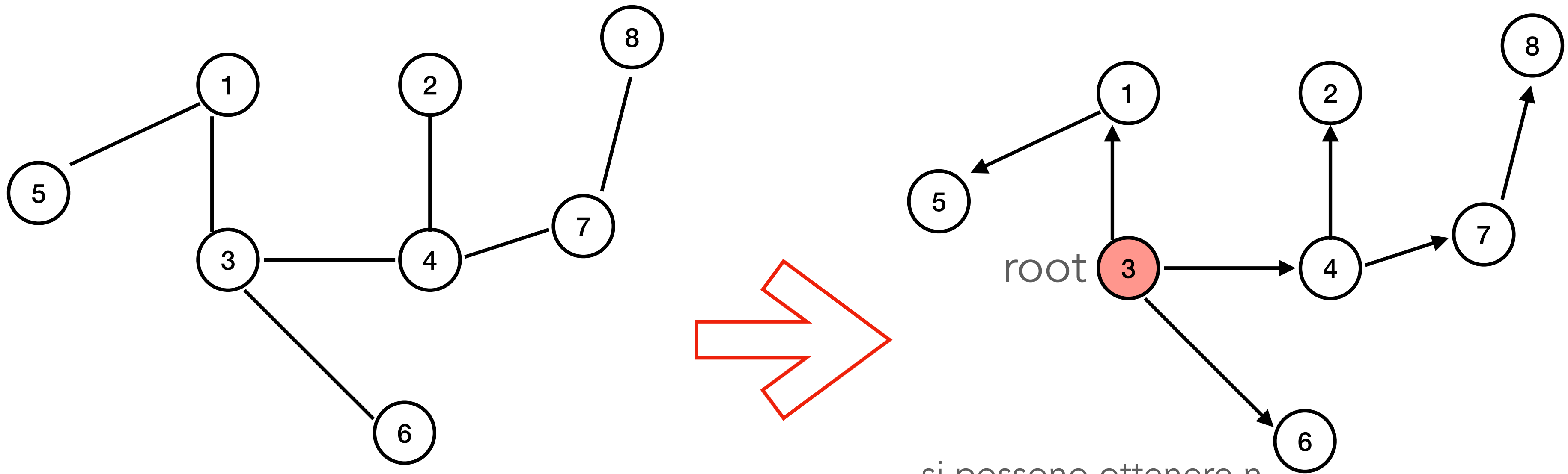
DIMENSIONI del GRAFO G=(V,E)

- n = |V| numero dei nodi
- m = |E| numero degli archi

Il **COSTO COMPUTAZIONALE** di algoritmi su grafi spesso e' espresso in termini di n e m (es, O(n + m))

CASI PARTICOLARI:

ALBERO radicato: albero libero in cui un nodo e' stato designato come radice e gli archi si considerano orientati dai padri ai figli



si possono ottenere n alberi radicati diversi da un albero libero

UNIMORE

GRAFI

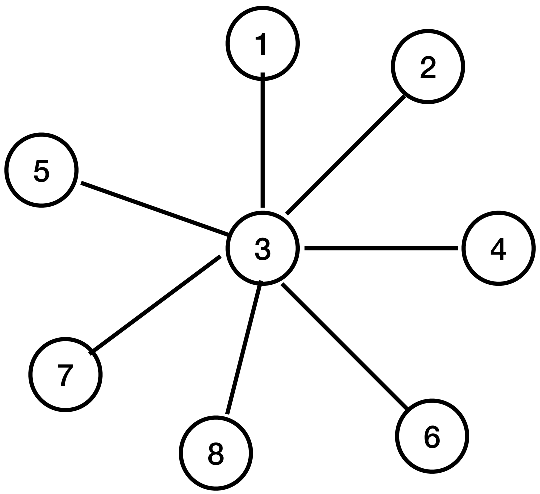
DIMENSIONI del GRAFO G=(V,E)

- n = |V| numero dei nodi
- m = |E| numero degli archi

Il **COSTO COMPUTAZIONALE** di algoritmi su grafi spesso e' espresso in termini di n e m (es, O(n + m))

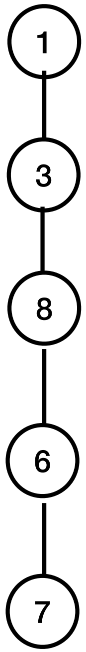
CASI PARTICOLARI:

STELLA



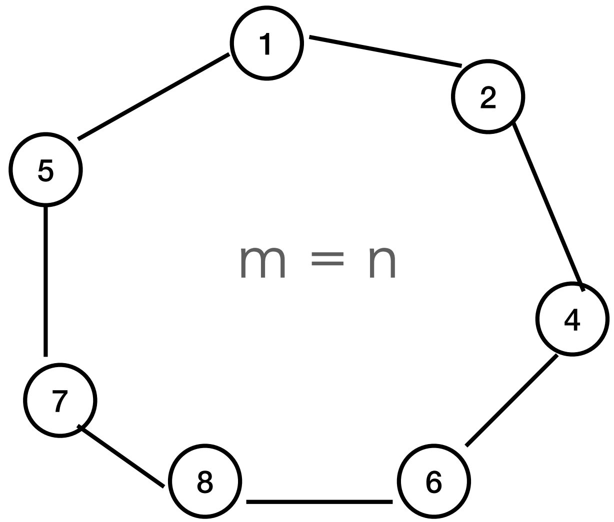
m = n-1

CAMMINO o CATENA



m = n-1

CICLO



GRAFI - rappresentazione

Per convenzione (e semplicità) i nodi del grafo sono rappresentati dai numeri da 1 a n

MATRICE DI ADIACENZA

Matrice G *nxn*

$$G[i,j] = \begin{cases} 1 & \text{se } \exists (i,j) \in E \\ 0 & \text{altrimenti} \end{cases}$$

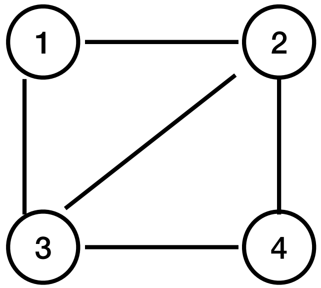
La matrice è simmetrica
se il grafo è non orientato

$G[i,j]$

indice di riga — indice di colonna

ESEMPI

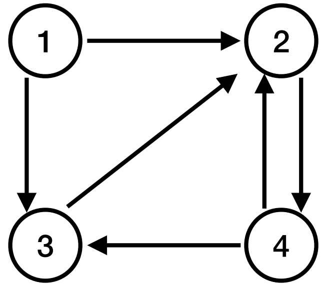
non orientato



$G[i,j]$

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

orientato



$G[i,j]$

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

Spazio $\Theta(n^2)$

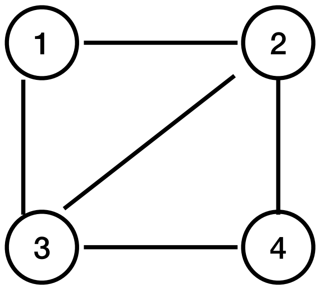
Check presenza arco	$\Theta(1)$
Elenco vicini nodo	$\Theta(n)$

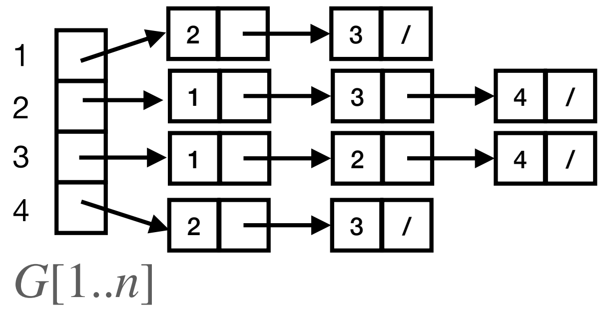
GRAFI - rappresentazione

Per convenzione (e semplicità) i nodi del grafo sono rappresentati dai numeri da 1 a n

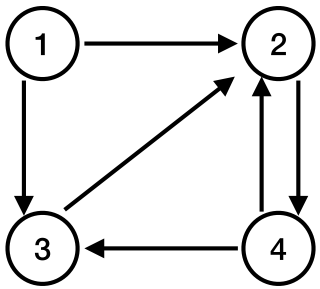
ESEMPI

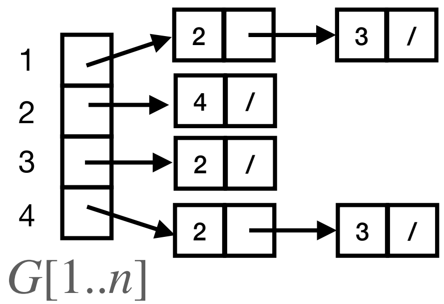
non orientato





orientato





Spazio $\Theta(|V| + |E|)$

LISTE DI ADIACENZA

Array G di |V| liste

G[v] è una lista

(ovvero un puntatore al primo elemento della lista)

che contiene tutti i vicini di v

Check presenza arco

Θ(#vicini nodo)

Elenco vicini nodo

Θ(#vicini nodo)

GRAFI - rappresentazione

AD ALTO LIVELLO:

GRAFO G = (V,E)

V = insieme dei vertici (nodi, *vertices*)

E = insieme degli archi (*edges*) $E \subseteq V \times V$

Per iterare su tutti i nodi del grafo:

```
for all v ∈ V do
 istruzioni per il nodo v
```

Non e' determinato l'ordine con cui si considerano i nodi

Per iterare su tutti gli archi del grafo:

```
for all (u,v)∈ E do
 istruzioni per l'arco (u,v)
```

Non e' determinato l'ordine con cui si considerano gli archi

Si possono eseguire operazioni anche sui nodi u e v

GRAFI - rappresentazione

AD ALTO LIVELLO:

GRAFO G = (V,E)

V = insieme dei vertici (nodi, *vertices*)

E = insieme degli archi (*edges*) $E \subseteq V \times V$

Per iterare su tutti i nodi del grafo e, per ogni nodo, sui suoi archi incidenti:

```
for all v ∈ V do
 istruzioni per il nodo v
 for all (v,u) ∈ E do
  istruzioni sull'arco (v,u)
```

Non e' determinato l'ordine con cui si considerano i nodi

Non e' determinato l'ordine con cui si considerano gli archi

La coppia (v,u) non e' ordinata se il grafo e' indiretto

La coppia (v,u) e' ordinata se il grafo e' diretto e si considerano solo gli archi uscenti