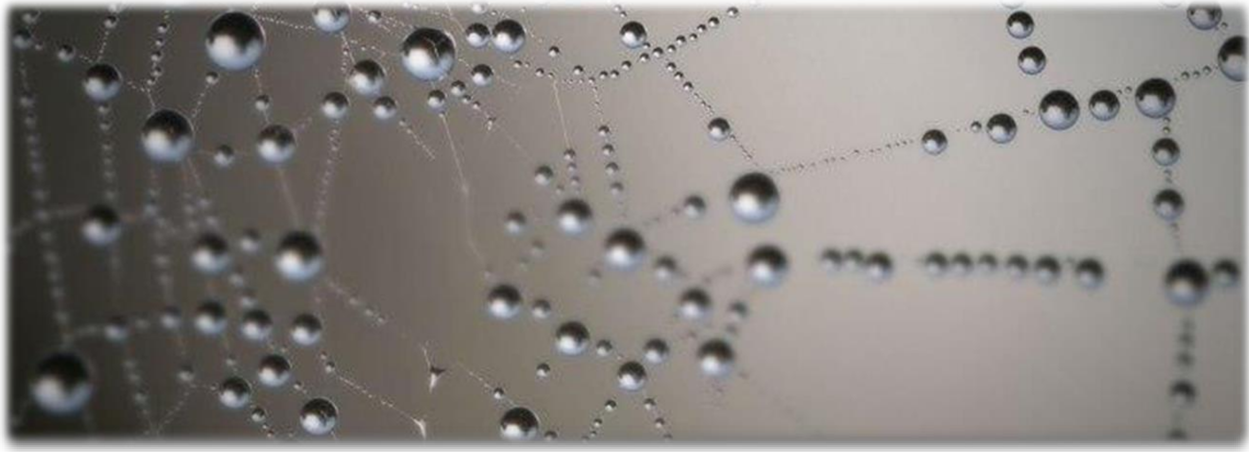
Università Politecnica delle Marche

Introduzione alla Teoria dei Grafi – parte I

ver 2.5.0

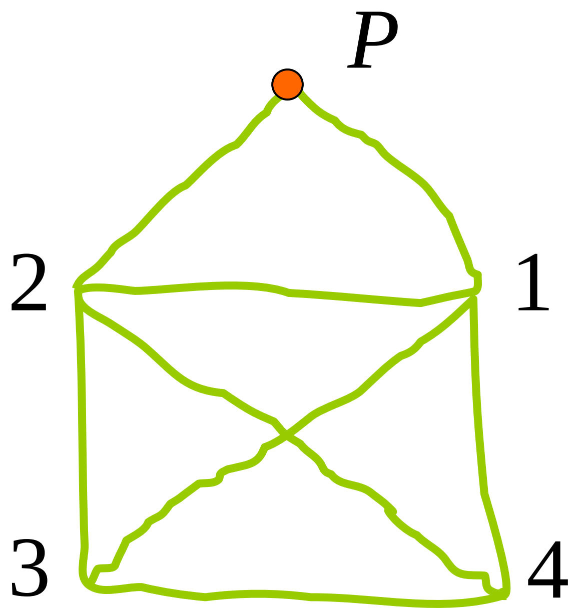


Fabrizio Marinelli
fabrizio.marinelli@staff.univpm.it
tel. 071 - 2204823

Sommario

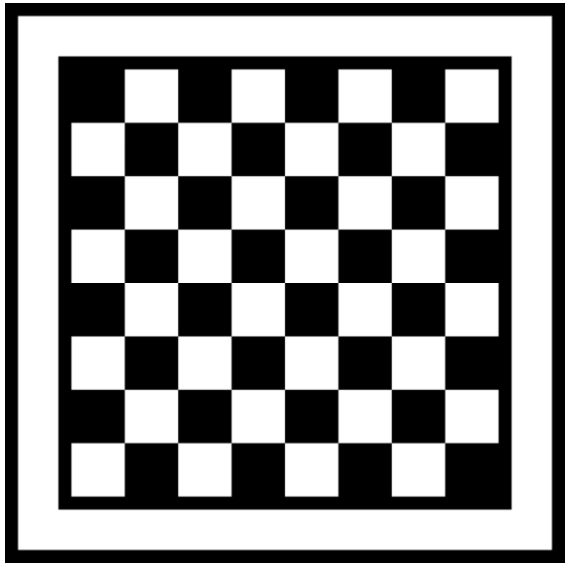
Un po' di enigmistica

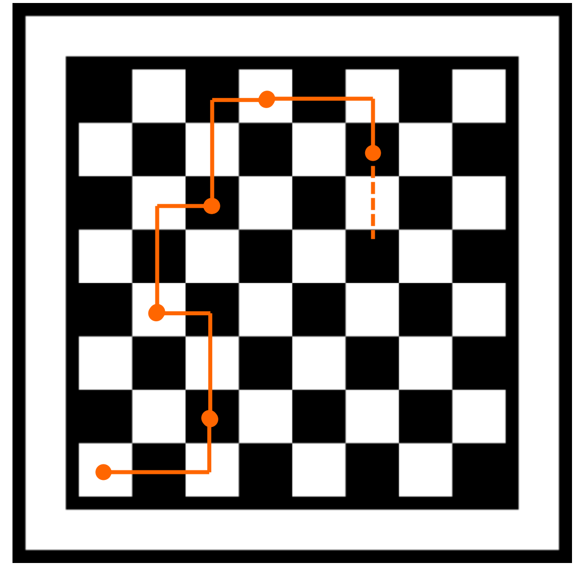
- Fare questo disegno partendo dal punto *P* e **tornando al punto *P*** senza staccare la penna dal foglio e senza ripassare mai su linee già tracciate



…un argomento generale?

Un po' di enigmistica (scacchi… un classico)



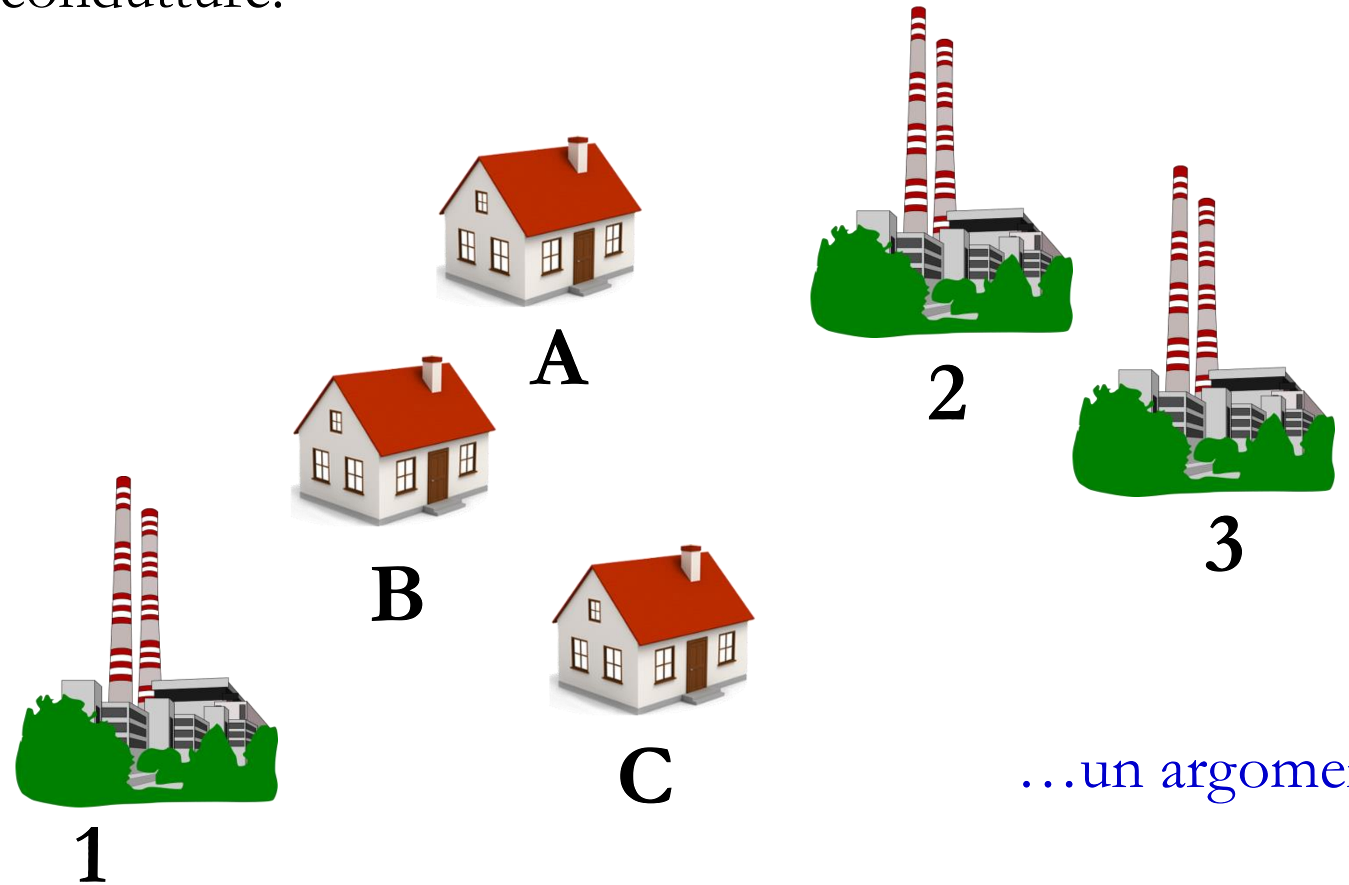


- Un cavallo può toccare tutte la case di una scacchiera, ognuna esattamente una volta, e tornare al punto di partenza?

…un argomento generale?

Un po' di enigmistica

- Tre edifici devono essere collegati alle centrali di energia elettrica, acqua e gas. E' possibile che ciò venga fatto senza intersecare le condutture?



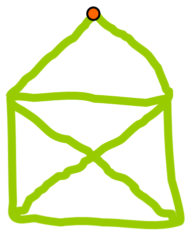
…un argomento generale?

Un po' di (social) enigmistica

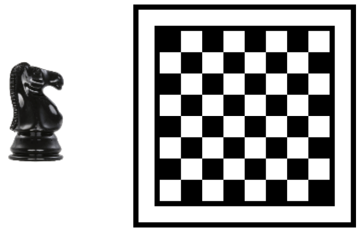
- In un gruppo **arbitrario** di persone qual è la probabilità che ce ne siano due che conoscono lo stesso numero di persone all'interno del gruppo ?

	A	B	C	D	E
A		×		×	
B	×		×		
C		×			
D	×				×
E				×	

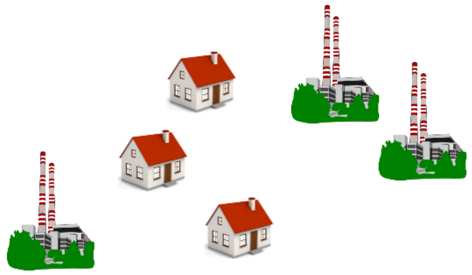
Qui, A e D conoscono entrambi 2 persone… ma anche B e D, e C e E conoscono lo stesso numero di persone



- Il problema può essere risolto ricorrendo all'enumerazione sistematica dei casi



- L'enumerazione è possibile in principio ma è impraticabile data l'ampiezza dello spazio di ricerca. Occorre qualche idea.



- L'enumerazione è impossibile: potenzialmente i casi da verificare sono infiniti. E' necessaria qualche idea…

Un po' di enigmistica

- Ordina un insieme dato di numeri in modo tale che la somma di ogni coppia di numeri adiacenti sia un quadrato. Prova con l'insieme {1, 2,…, 15}

…un metodo generale?

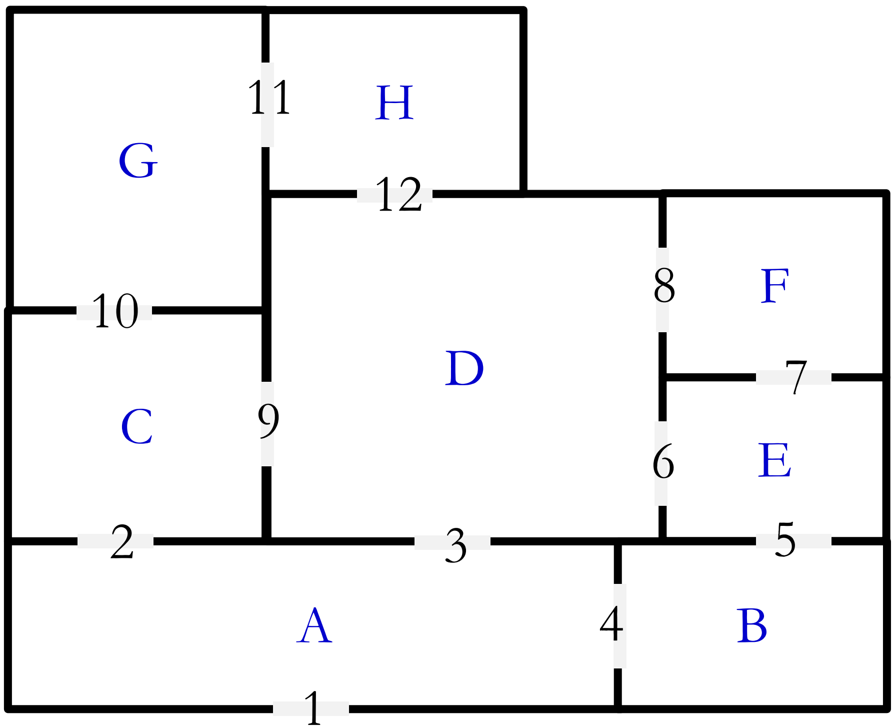
Un po' di enigmistica nello sport

- Un torneo di calcio è formato da due gironi (A e B) ognuno composto da 13 squadre. Ogni squadra deve disputare 14 partite, di cui 11 con squadre del proprio girone e 3 con squadre dell'altro girone. Due squadre non si possono incontrare più di una volta. Sapreste definire un calendario?

…di nuovo: un metodo generale?

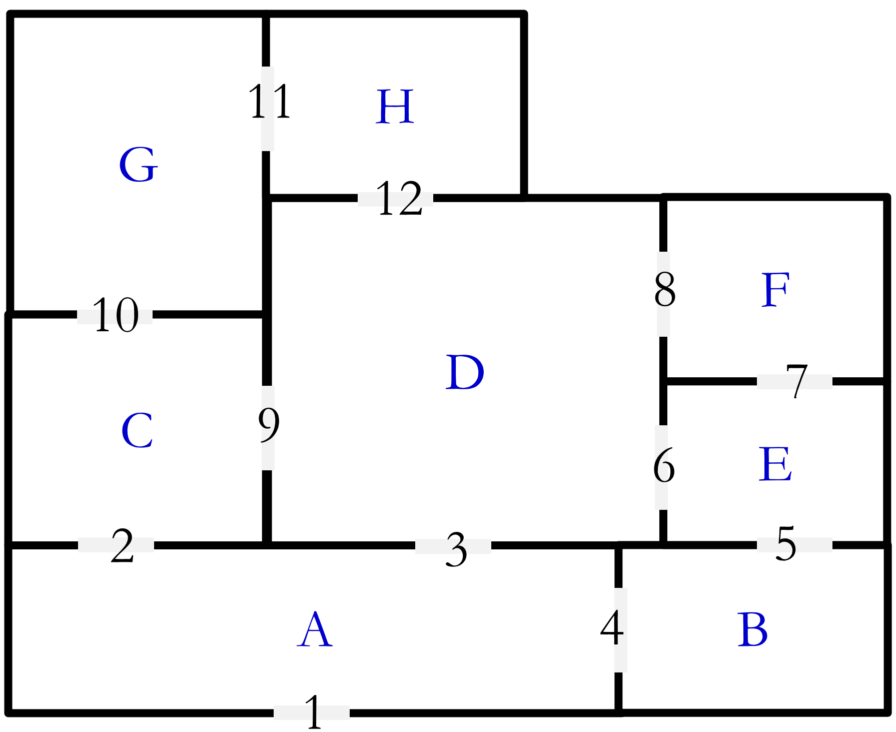
Un po' di enigmistica: una notte al museo

- Di notte tardi, l'ultima guardia deve chiudere tutte le porte. Che giro deve fare se le porte chiuse non si aprono più e se per chiuderle occorre attraversarle?



Un po' di enigmistica: una notte al museo

- Qual è i minimo numero di guardie necessarie per controllare tutte le stanze del museo? (una guardia alla porta controlla le due stanze che la porta collega)

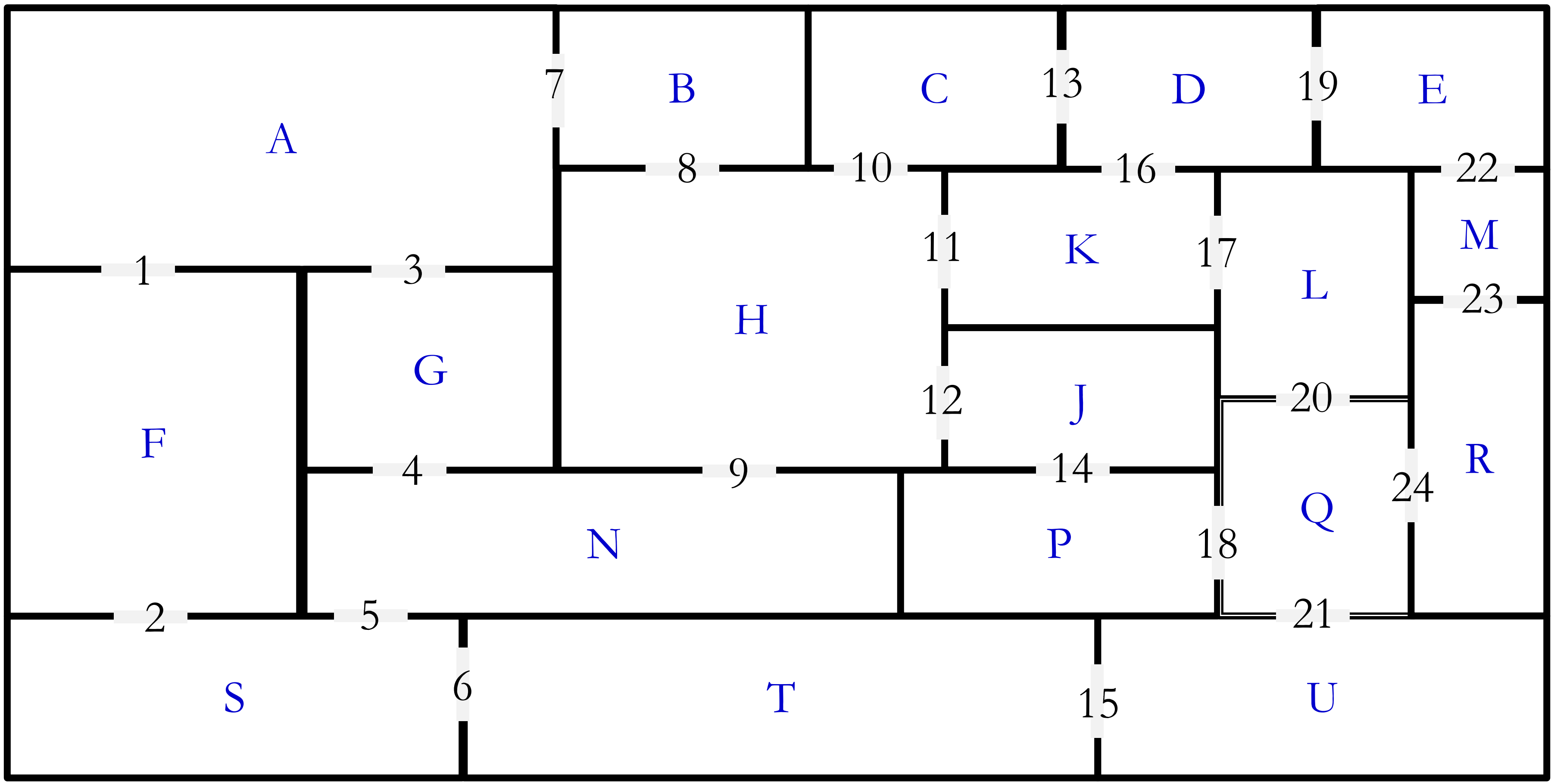


Uno scenario alternativo: qual è il minimo numero di guardie se ognuna deve stare in una stanza e può controllare tutte le stanze «adiacenti»?

…un metodo generale ?

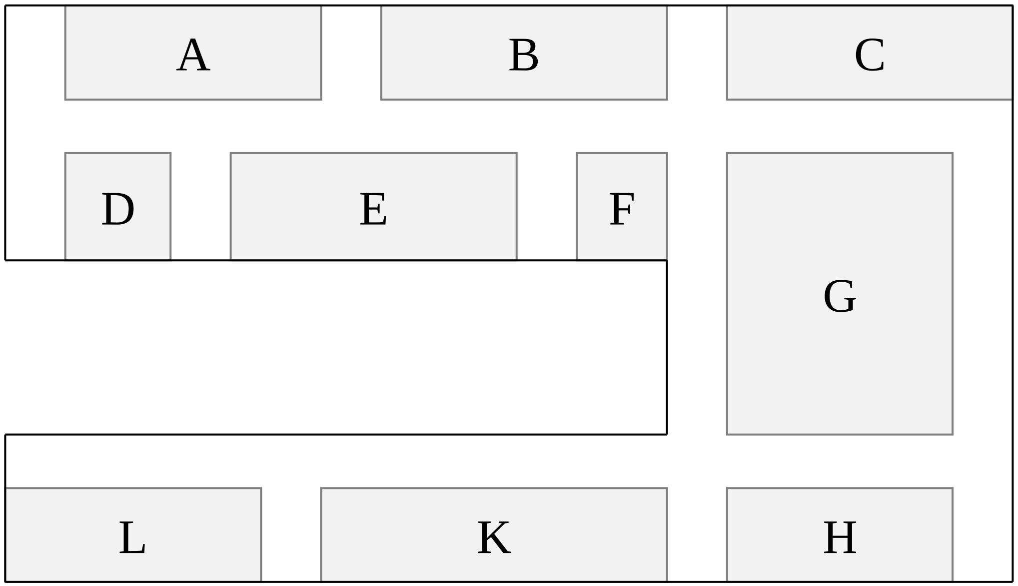
Un po' di enigmistica: una notte al museo

- Cosa riusciamo a dire su questa mappa?



Problemi (semiseri) di un ingegnere

- Si vuole dotare un museo di un sistema di telecamere per la sorveglianza in assenza di personale. Sapendo che una telecamera posta all'incrocio di due corridoi è in grado, con opportune rotazioni, di sorvegliarli entrambi, qual è il minimo numero di telecamere necessarie?



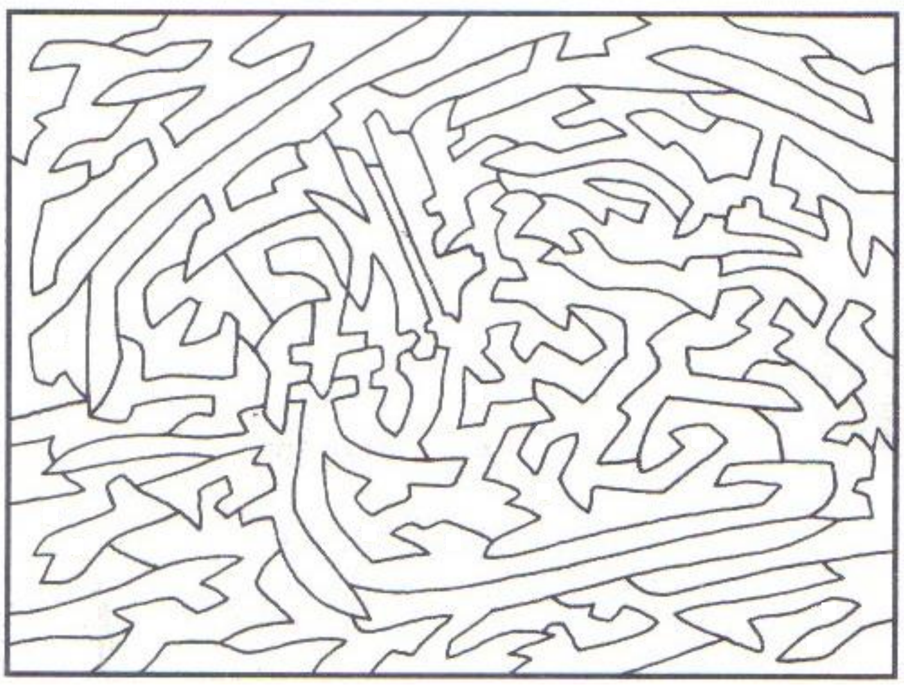
Problemi (semiseri) di un ingegnere

- Una grande potenza (…) vuole creare una grande coalizione per "sconfiggere il male". Per avere successo però deve stare attenta a non coinvolgere stati in conflitto reciproco. Con chi si coalizza?

- In un dato gruppo di persone, come si individua la *cricca* più grande di amici? Cioè come si calcola il massimo numero di persone che non hanno bisogno di presentazione reciproca?

Problemi (semiseri) di un ingegnere

- Di solito, per evitare confusione, le regioni confinanti di una mappa hanno colori diversi. Volendo rispettare questo vincolo, qual è il minimo numero di colori necessari per colorare questa mappa?



Problemi (un po' più seri) di un ingegnere

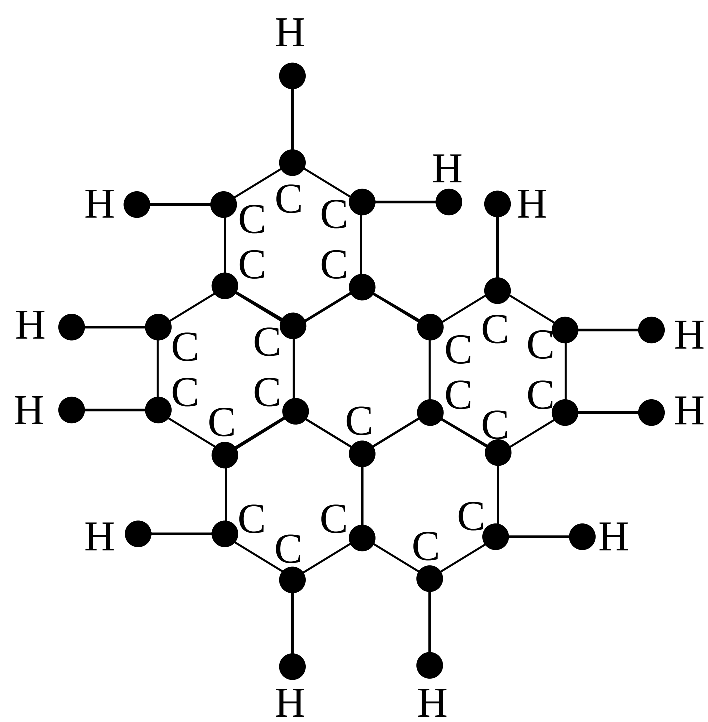
- Un manager deve assegnare un insieme di progetti a un team di ingegneri. In base alle competenze e alle compatibilità attitudinali ogni progetto deve essere svolto da un dato gruppo di ingegneri ma ogni ingegnere può eseguire un solo progetto. Qual è il massimo numero di progetti che possono essere svolti contemporaneamente?

griglia di assegnamento

	p_1	p_2	p_3	p_4	p_5
Claudio	●		●	●	
Gino	●	●		●	●
Luca			●	●	
Andrea		●			

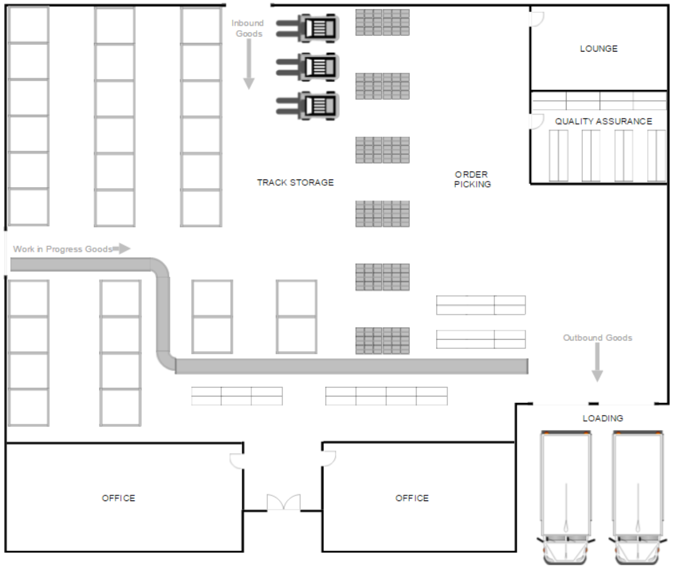
Problemi (un po' più seri) di un ingegnere

- Data la seguente struttura parziale (il carbonio ha valenza 4) di una molecola di un idrocarburo, quale composto può essere sintetizzato?



Problemi (un po' più seri) di un ingegnere

- **Smart factories**: Un magazzino vede operare degli *AGV*s (*automated guided vehicles*) che si muovono su tracciati prestabiliti

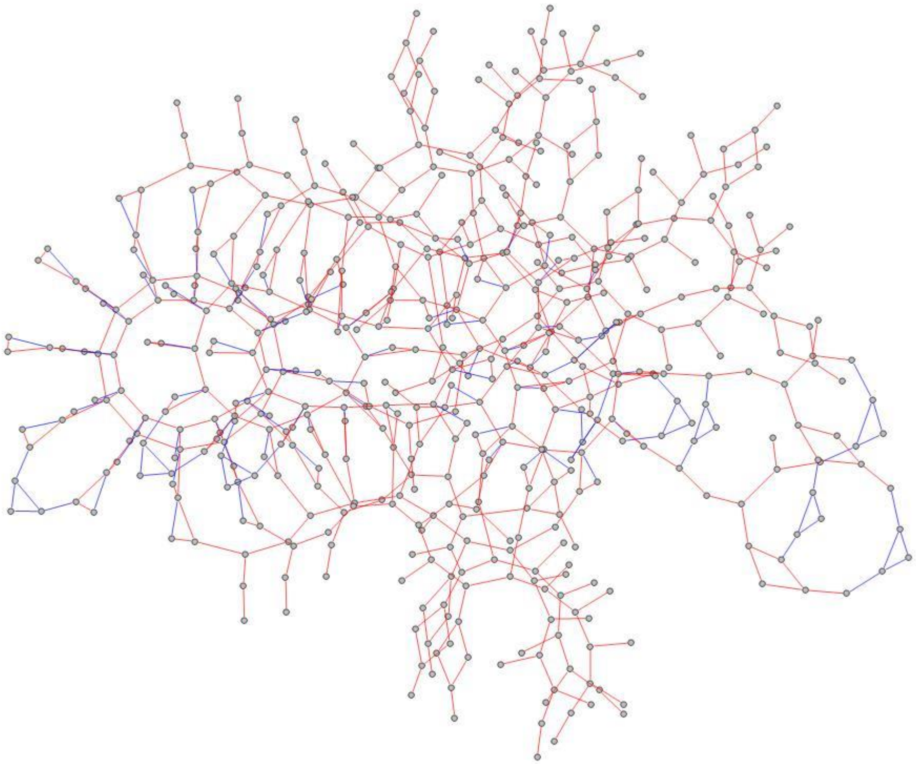


Qual è il massimo numero di AGVs che possono muoversi tra 2 punti dati A e B senza alcun rischio di collisione?

Problemi (un po' più seri) di un ingegnere

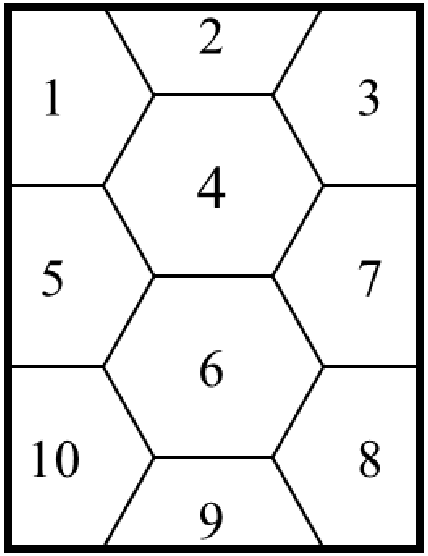
- **Social influencer**: consideriamo un *social network* in cui ogni individuo ha un'opinione su un argomento specifico. Le connessioni in rosso indicano accordo sull'argomento; quelle in blu indicano disaccordo

Qual è potenzialmente il gruppo coeso più numeroso? E a quali individui devo far cambiare opinione per ottenere tale gruppo?



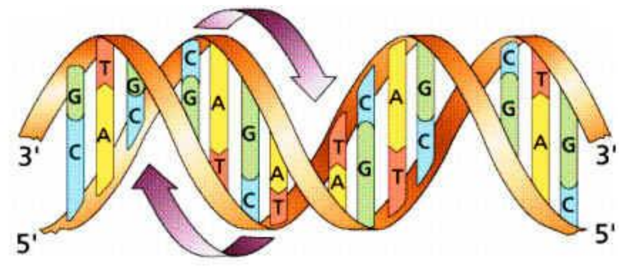
Problemi (un po' più seri) di un ingegnere

- **Assegnamento di frequenze**: Qual è il minimo numero di frequenze e come devono essere assegnare alle celle di una rete di trasmissione in modo che celle adiacenti abbiano frequenze diverse (e quindi non ci siano interferenze)?



Problemi (un po' più seri) di un ingegnere

- Genome assembly problem: Un genoma (semplificato) è una lunga stringa (milioni o anche miliardi) di *simboli* A, C, G e T



- L'intero genoma è in generale sconosciuto ma piccoli frammenti (sottostringhe) possono essere ottenuti con tecniche di sequenziamento dei geni.

> - Come può essere ricostruito un genoma assemblando le sequenze disponibili?

Problemi (un po' più seri) di un ingegnere

Un caso giocattolo

- Sottosequence disponibili di A, C, G e T

CGTC

AACG

CGCT

CTCGTC

GCTC

GCTAT

AATC

CTTC

CTCAA

GGCT

ACGC

GTCGG

TCCTAA

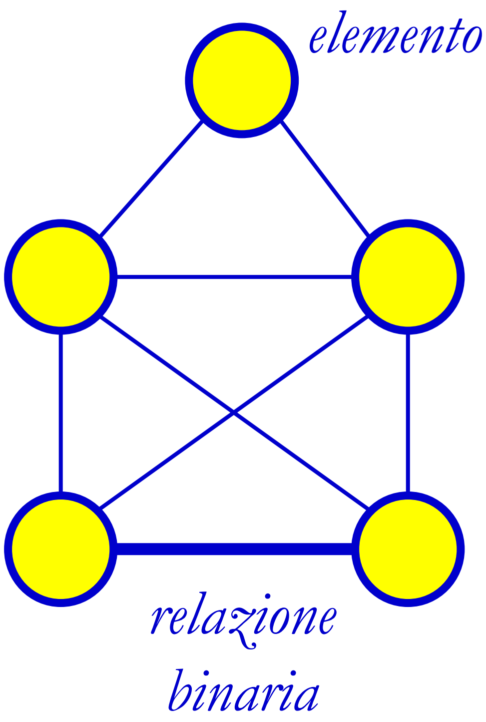
TTGAC

Sommario

- Introduzione
- Motivazioni e origini storiche
- Definizioni e proprietà di base
- Isomorfismi tra grafi
- Grafi di base
- Classi di grafi
- Grafi orientati
- Rappresentazioni
- Appendice

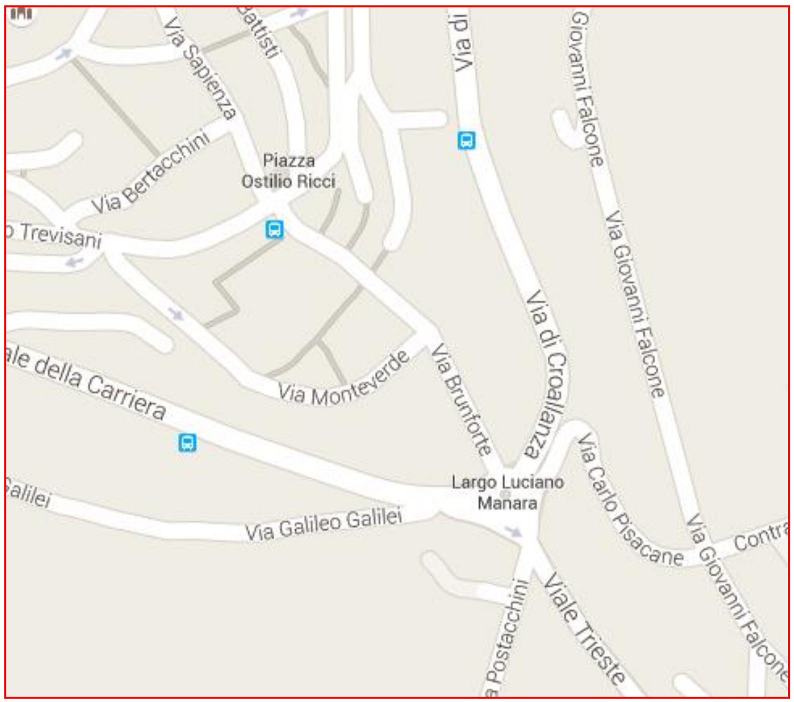
Cos'è un grafo

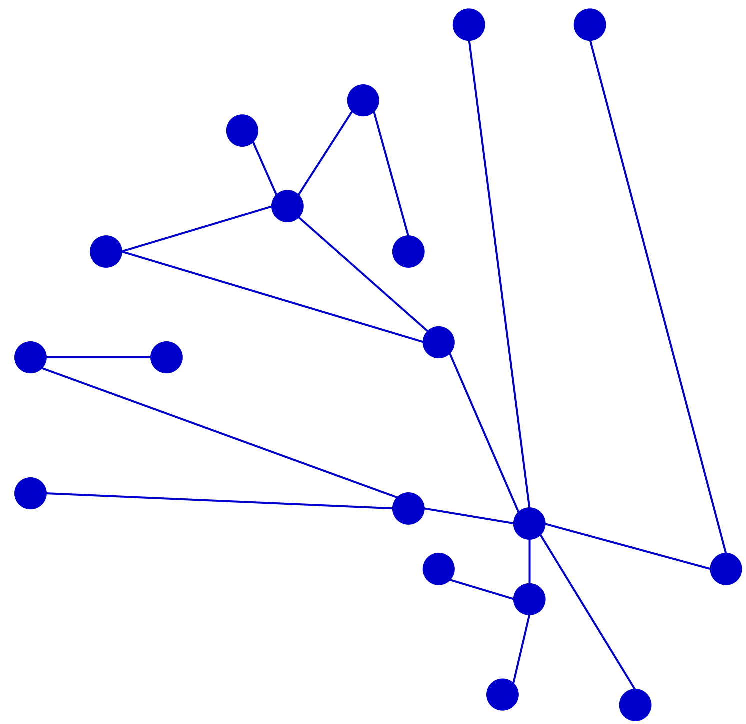
Un *grafo* è un formalismo (grafico) utile per descrivere e rappresentare una *relazione binaria* su una collezione finita e discreta di *elementi*.



- Rete (stradale, di calcolatori,…)
- Circuiti elettrici
- Relazioni tra persone
- Attività di progetto
- Giochi
- Automi
- Mappe
- Algoritmi e strutture dati
- …

Rappresentazione di reti stradali

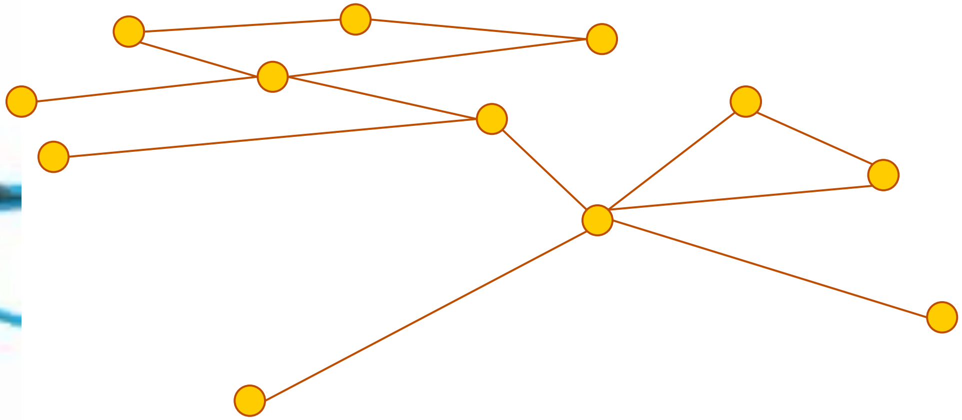




punti: crocevia

segmenti: strade che collegano i crocevia

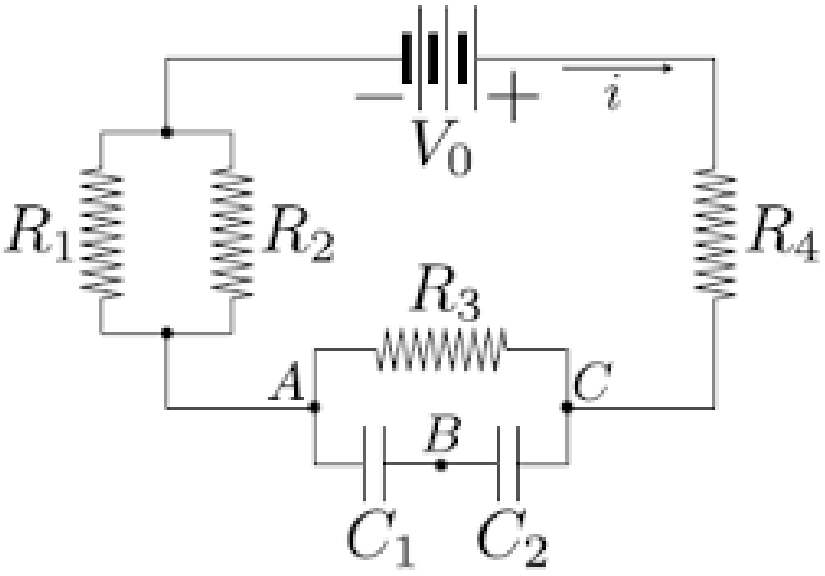
Rappresentazione di reti di calcolatori

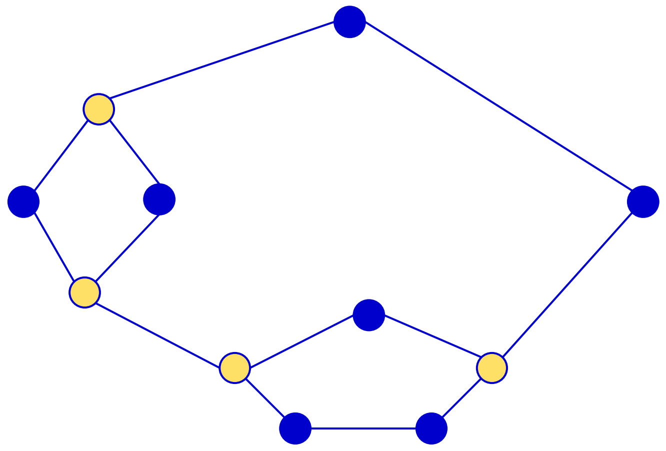


punti: computer e server

segmenti: link e connessioni

Rappresentazione di circuiti elettrici

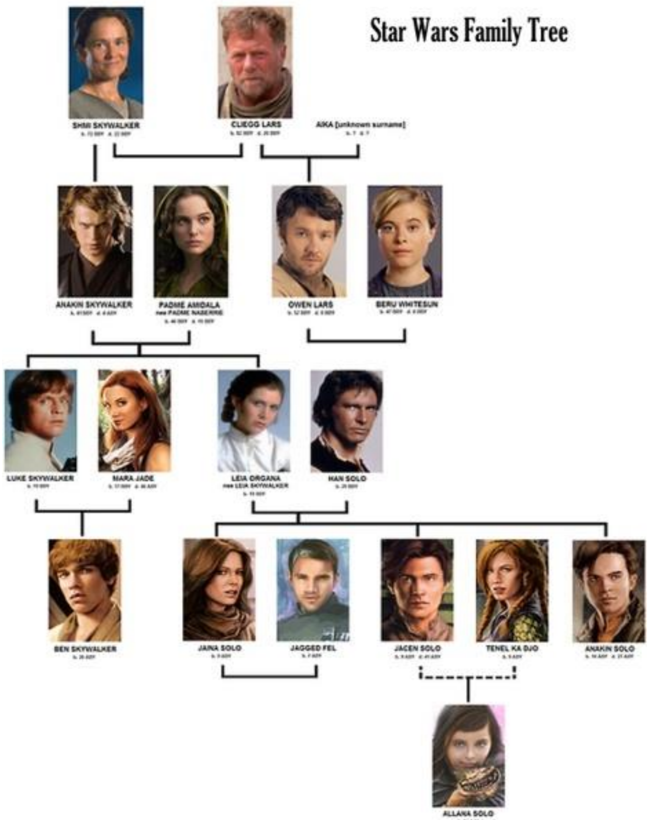


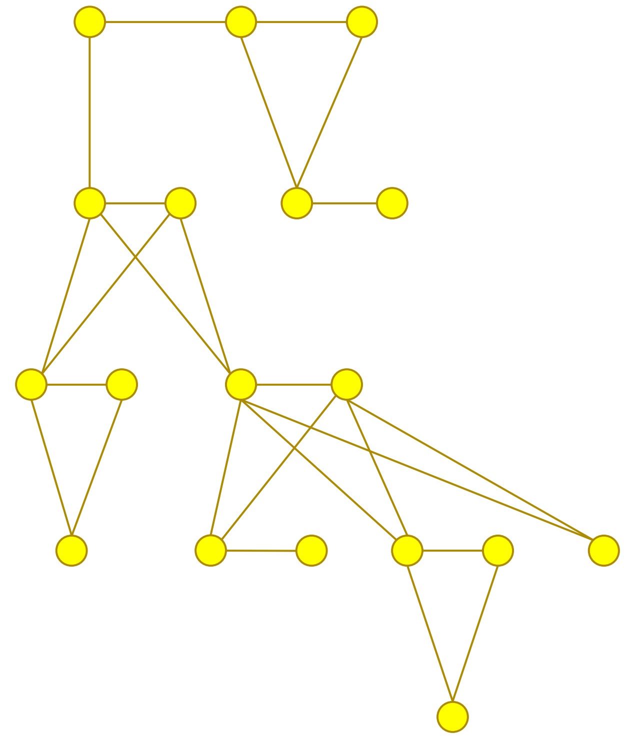


punti: componenti elettrici /giunzioni

segmenti: collegamenti

Rappresentazione di relazioni tra persone

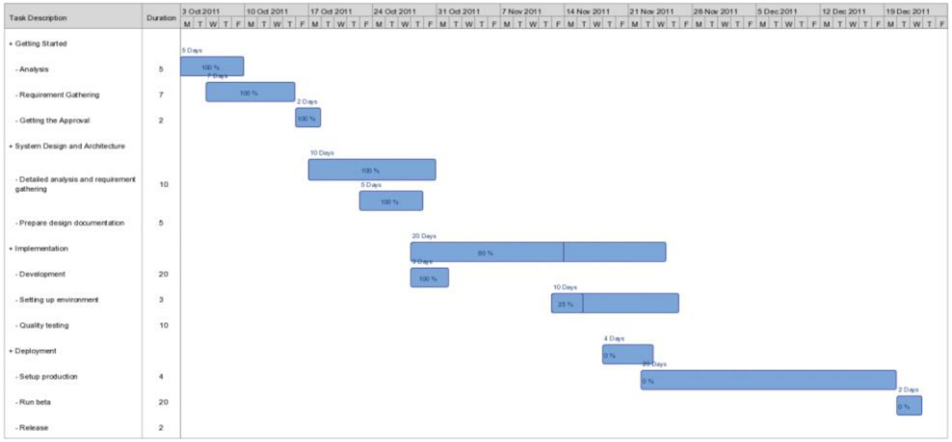


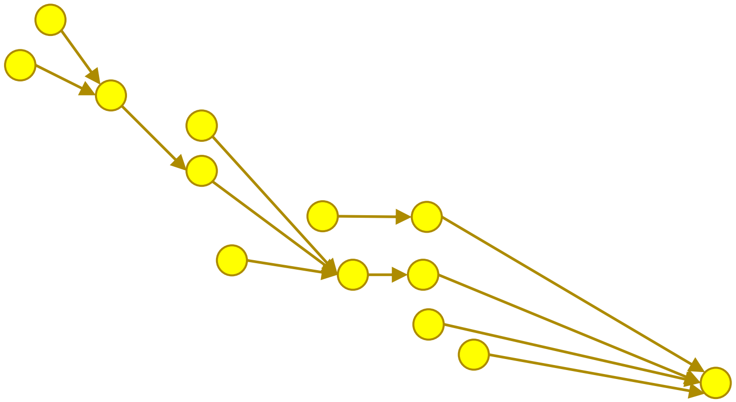


punti: persone

segmenti: relazione (per esempio di parentela)

Rappresentazione di attività di progetto

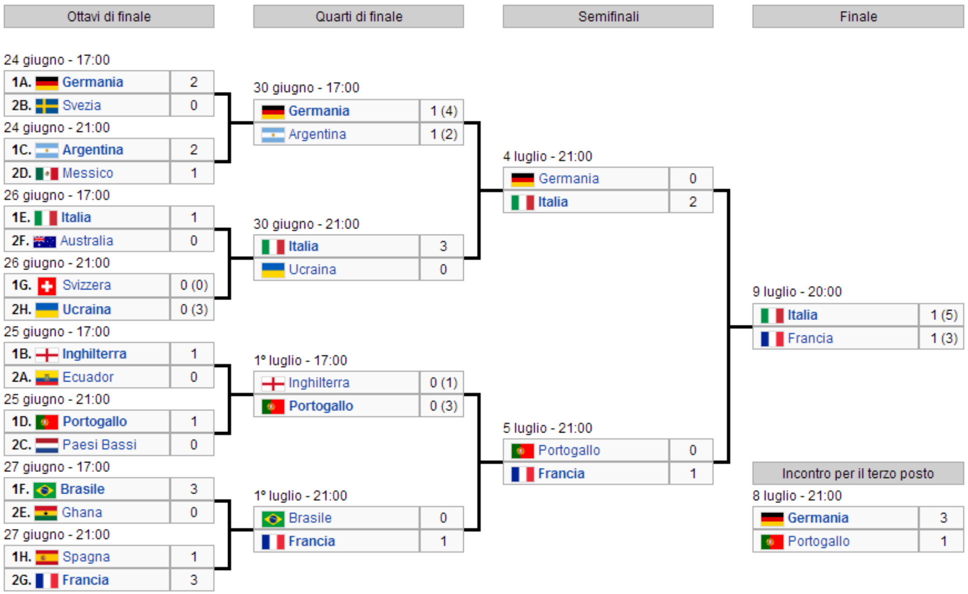


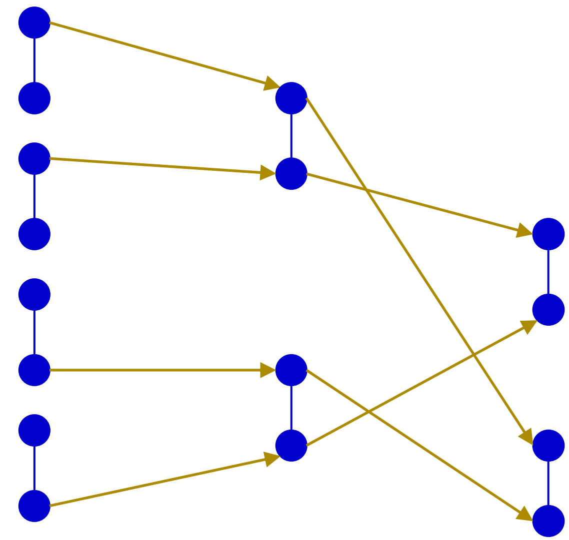


punti: attività

frecce: relazioni di precedenza

Tabellone di un torneo sportivo

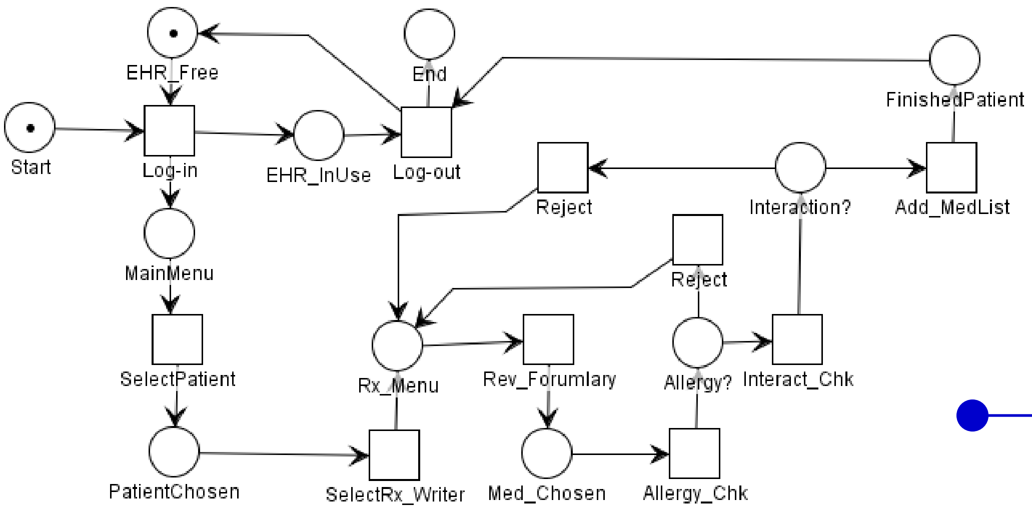


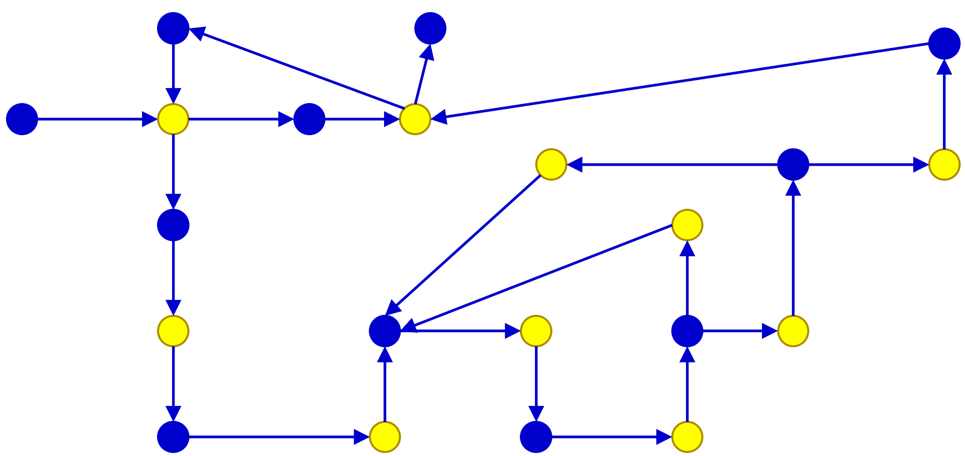


punti: squadre

segmenti/frecce: partite / passaggio di turno

Automa a stati finiti

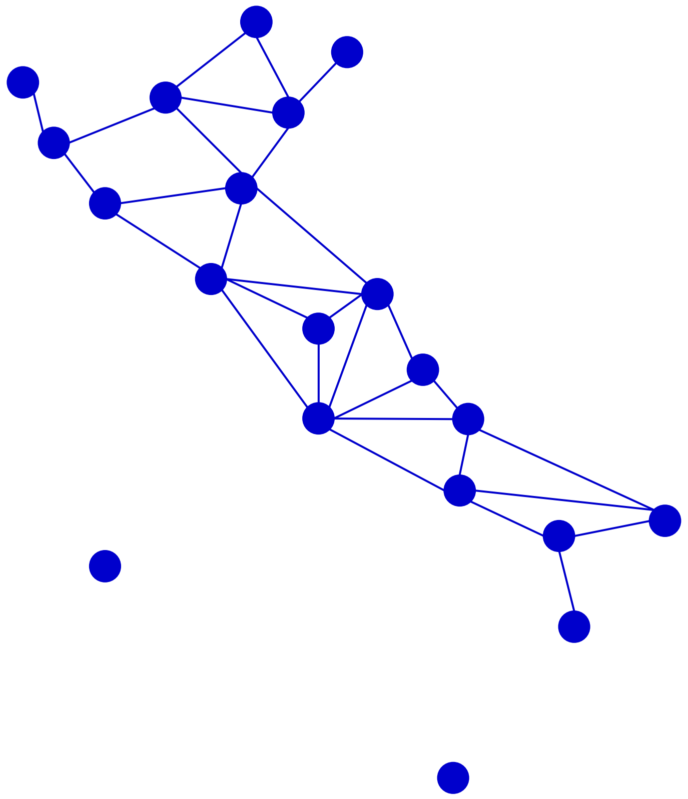




punti: stati / azioni

frecce: transizioni

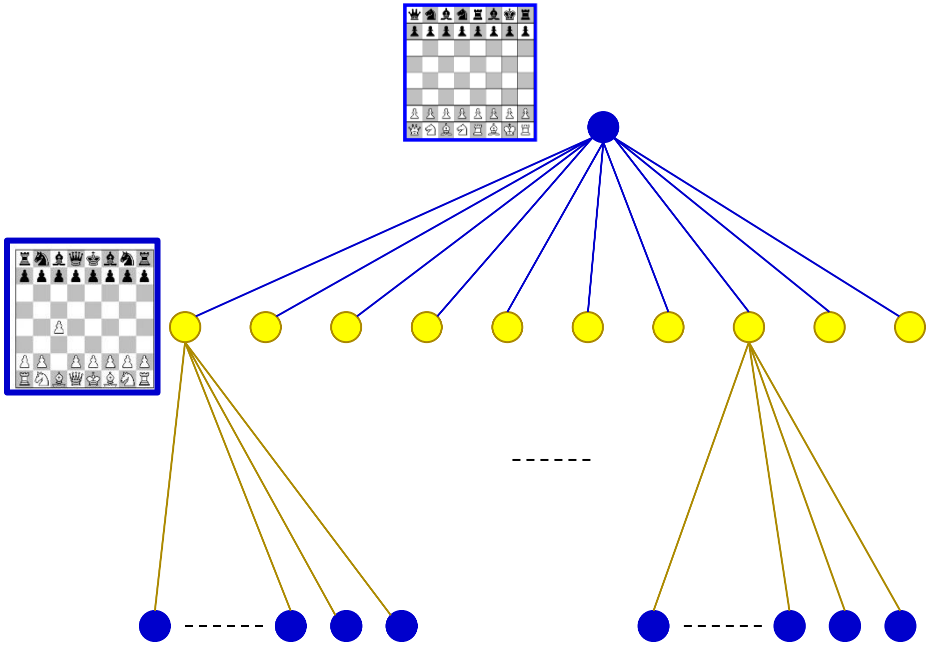
Rappresentazione di mappe



punti: regioni

segmenti: confini

Sequenza di decisioni



punti: stato della scacchiera (disposizione dei pezzi)

frecce: possibili mosse

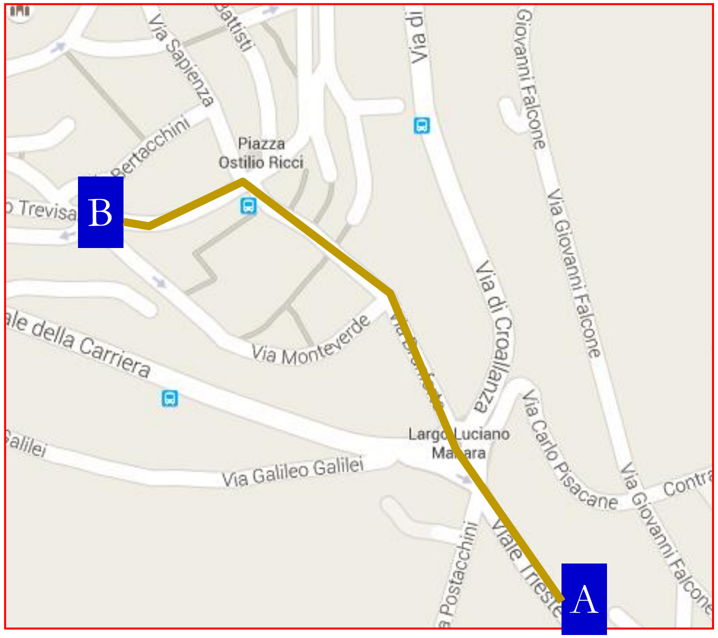
Cos'è un grafo

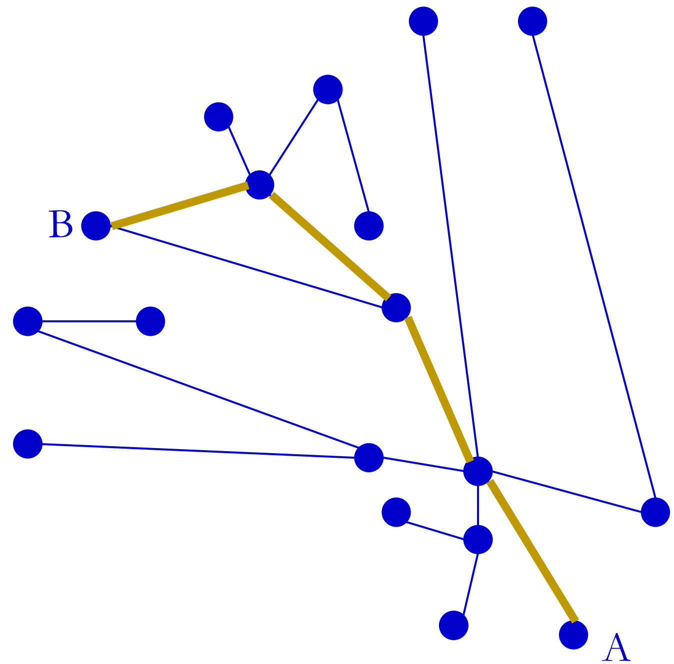
Un *grafo* è un formalismo (grafico) utile per descrivere e rappresentare una *relazione binaria* su una collezione finita e discreta di *elementi*.

… ma non solo!

Un grafo è un mezzo di astrazione: aiuta a comprendere e studiare la *struttura matematica* e la *complessità* di un problema.

Problemi reali e problemi su grafo

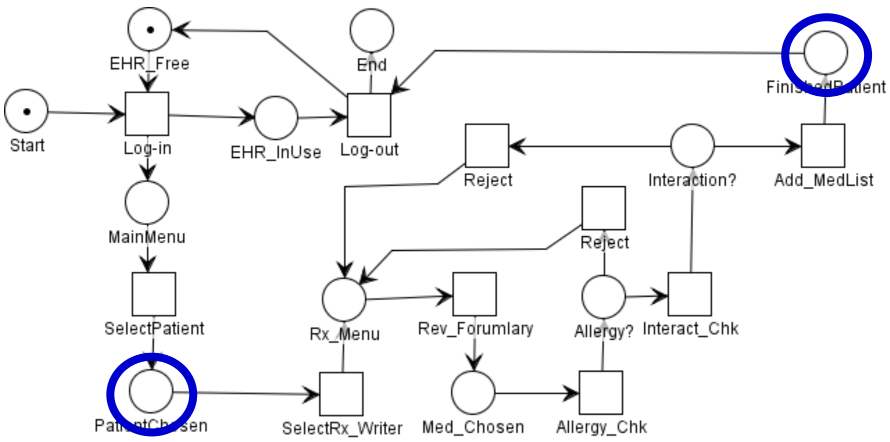


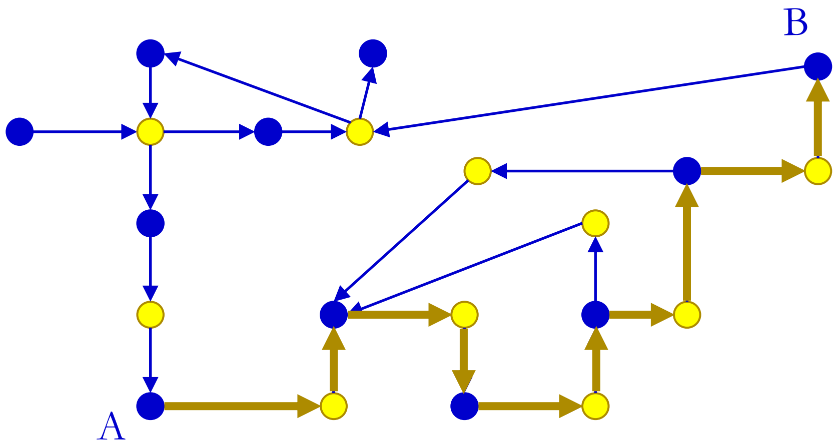


Qual è la strada più breve dall'indirizzo A all'indirizzo B?

Qual è il *percorso minimo* tra i nodi A e B ?

Problemi reali e problemi su grafo

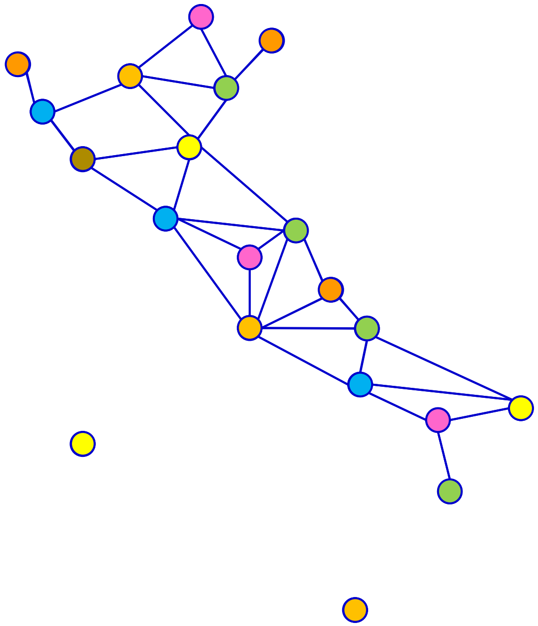




Una volta scelto un paziente, riesco a completare la visita?

Esiste un *cammino diretto* tra i nodi A e B ?

Problemi reali e problemi su grafo

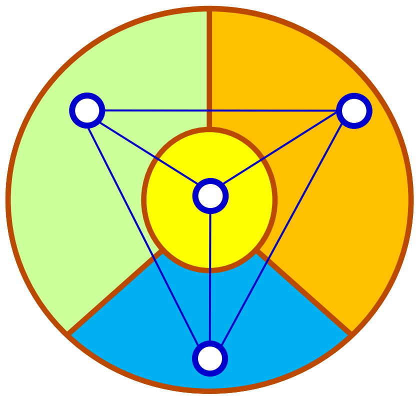


Quanti colori mi servono
per colorare una mappa
«decentemente» ?

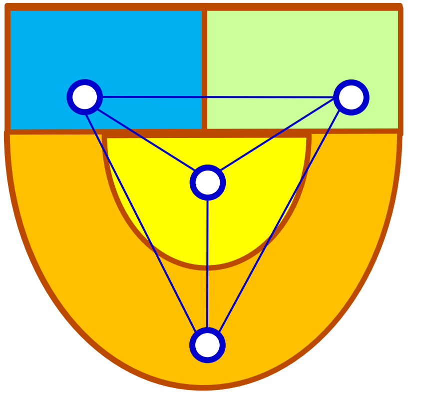
Qual è il *numero cromatico*
del grafo?

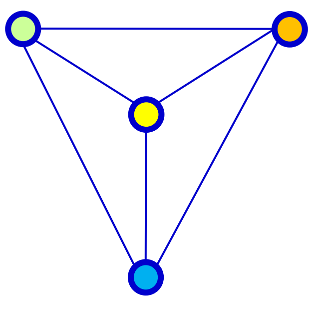
Perché la teoria dei grafi: astrazione

Coloriamo queste mappe

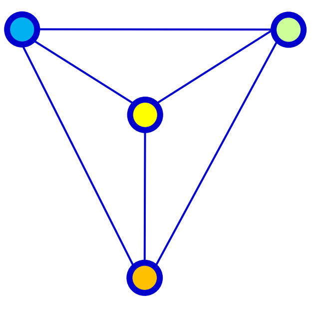


mappe diverse…





…ma grafi identici



Il problema sul grafo individua l'essenza del problema decisionale

Perché la teoria dei grafi: astrazione

… e colorare una mappa «equivale» a

- colorare imballi, manifesti, libri,…
- allocare variabili a registri di una CPU (compilatori)
- testare circuiti elettronici
- assegnare frequenze radio
- determinare tabelle orarie
- …

Perché la teoria dei grafi

Nella teoria dei grafi, il grafo è l'oggetto di studio.

- I problemi di ottimizzazione su grafo costituiscono molto spesso l'essenza di un problema decisionale concreto.
- Lo studio delle proprietà dei grafi aiuta ad analizzare la complessità di problemi/algoritmi in casi particolari.

I padri della teoria dei grafi

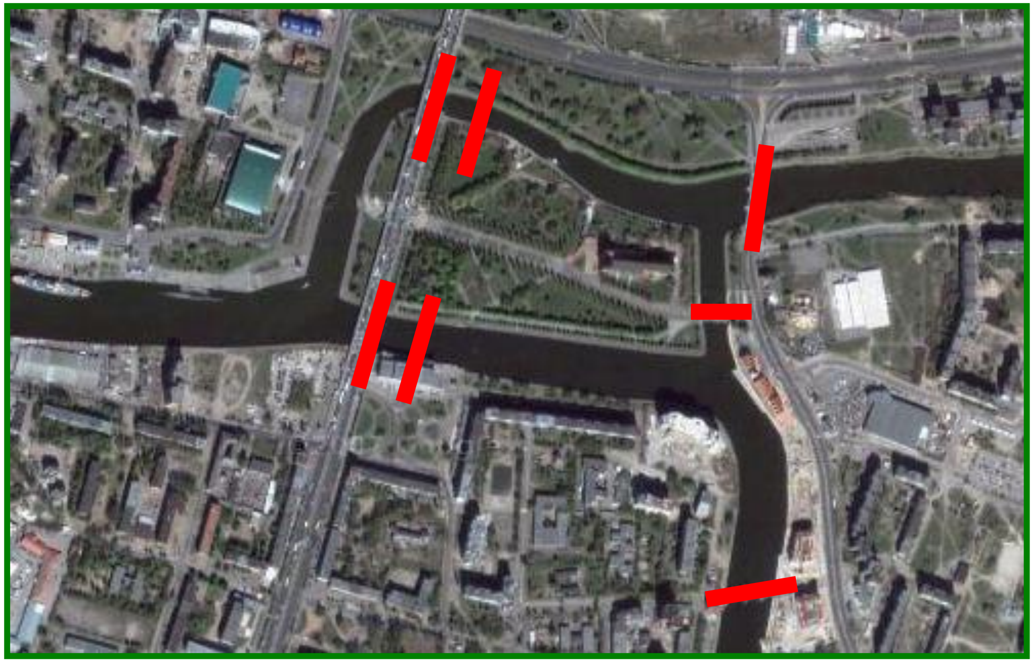
L. Eulero
(1707 – 1783)

W. R. Hamilton
(1805 – 1865)

I padri della teoria dei grafi: Eulero

L. Eulero
(1707 – 1783)

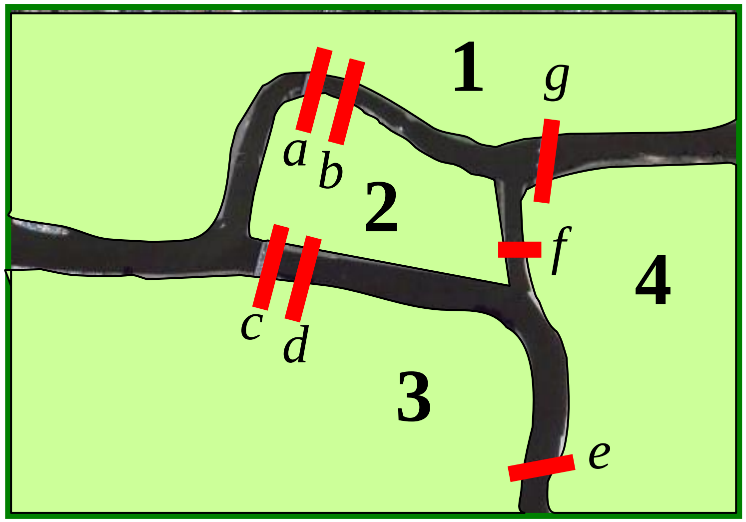
- Il problema dei **ponti di Königsberg** (1736)

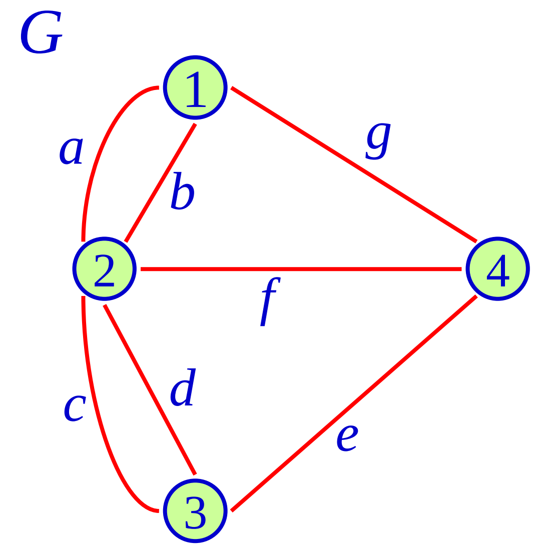


Königsberg (attualmente Kaliningrad)

E' possibile fare una passeggiata attraversando **ogni** ponte **esattamente una volta** e tornare al punto di partenza?

Eulero e i ponti di Königsberg





Il grafo *G* ammette un *ciclo* che attraversa
ogni *arco* **esattamente una volta**?

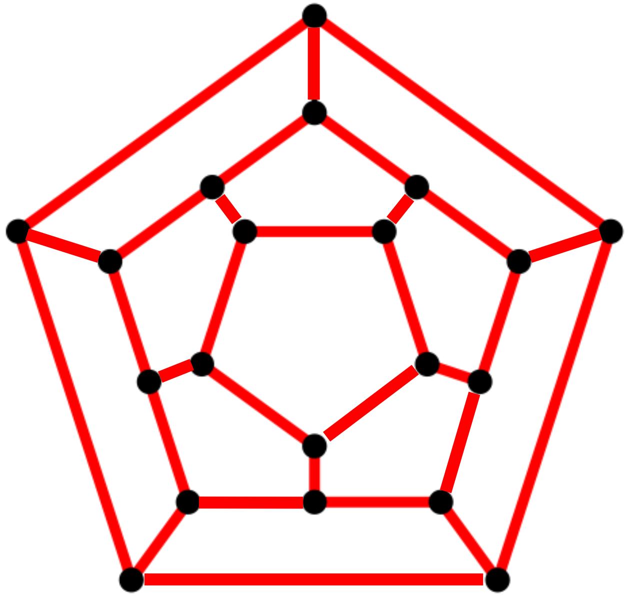
Hamilton e l'*icosian* game

- **Icosian game** (app per Android):

W. R. Hamilton
(1805 – 1865)

E' possibile, passando per gli spigoli, toccare **tutti** i vertici di un dodecaedro **esattamente una volta**?

Hamilton e l'*icosian* game



Il grafo *G* ammette un *ciclo* che tocca
ogni *nodo* **esattamente una volta**?

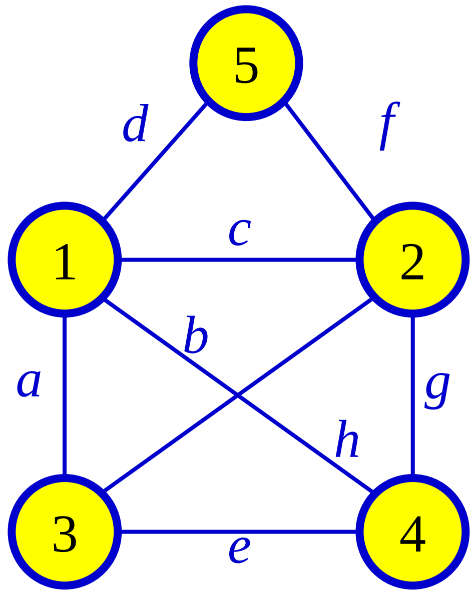
Sommario

- Introduzione
- Motivazioni e origini storiche
- Definizioni e proprietà di base
- Isomorfismi tra grafi
- Grafi di base
- Classi di grafi
- Grafi orientati
- Rappresentazioni
- Appendice

Grafi non orientati (o simmetrici)

Un grafo $G = (V, E)$ è una coppia di insiemi finiti

- nodi o (vertici) $V = \{1, 2, ...\}$
- archi o (spigoli) $E = \{a, b, \ldots\} \subseteq V \times V$



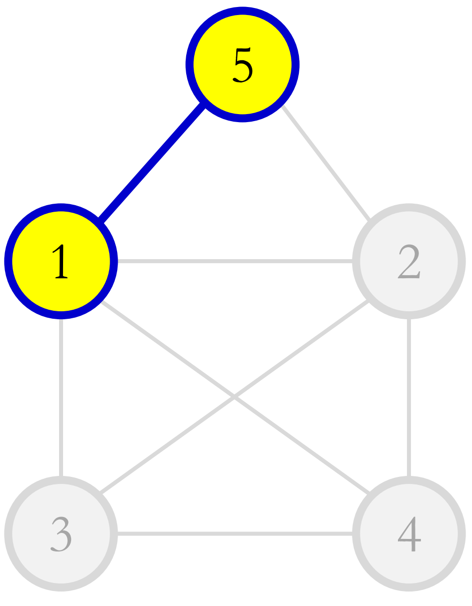
$V = \{1, 2, 3, 4, 5\}$

$E = \{a, b, c, d, e, f, g, h\}$

$$a = \{1, 3\} \equiv \{3, 1\} = a$$

Definizioni: adiacenza e incidenza

I nodi $u,v \in V$ sono detti adiacenti se sono collegati da un arco.
L'arco $\{u,v\}$ si dice incidente sul nodo u e sul nodo v



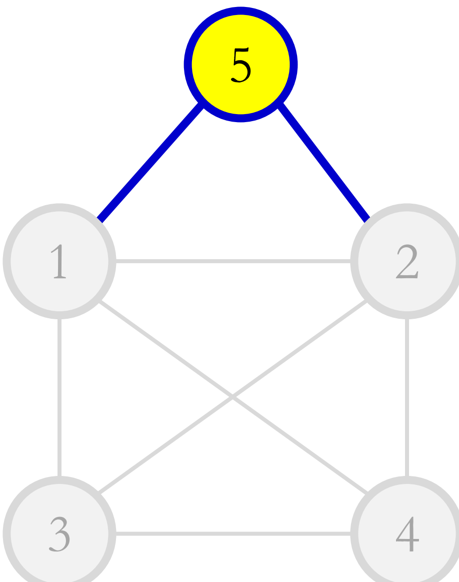
1 e 5 sono nodi adiacenti.

L'arco {1,5} è incidente sui nodi 1 e 2

$\{u,v\} \in E \Leftrightarrow u,v$ nodi adiacenti

Definizioni: adiacenza e incidenza

Gli archi $e, f \in E$ sono detti adiacenti (o consecutivi) se sono incidenti su un nodo comune

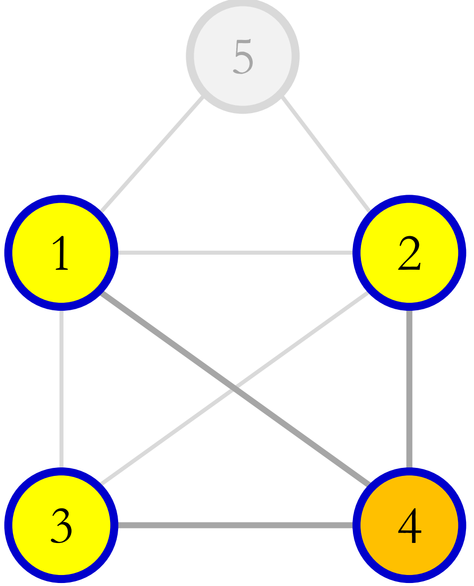


Gli archi {1,5} e {2, 5} sono adiacenti perché entrambi incidenti sul nodo 5

$\{u,v\}$, $\{w,v\}$ sono archi adiacenti

Definizioni: intorno e stella

L' intorno $N(v)$ di un nodo v è l'insieme di <u>nodi</u> adiacenti a v



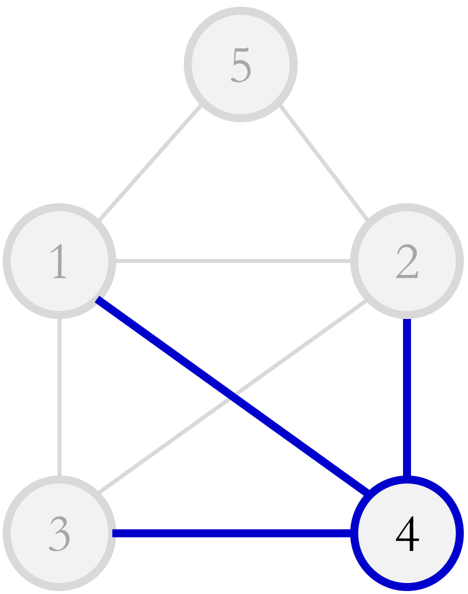
$N(4) = \{1, 2, 3\}$

$$N(v) = \{u \in V \mid \{u,v\} \in E\}$$

Definizioni: intorno e stella

La stella $\delta(v)$ di un nodo v è l'insieme di archi incidenti su v

Il grado $d(v)$ di un nodo v è la cardinalità di $d(v)$



$$\delta(4) = \{\{1, 4\}, \{2, 4\}, \{3, 4\}\}$$

$$d(4) = |\ \delta(4)\ | = 3$$

$$\delta(v) = \{\{u,v\} \in E\}$$

Grafi: proprietà elementari

Sia $G = (V, E)$ un grafo simmetrico con n nodi e m archi

- $d(v) \leq n - 1 \qquad \forall v \in V$
- $m \leq \dfrac{n(n-1)}{2}$
- $\sum_{v \in V} d(v) = 2m$ (la somma dei gradi è pari)

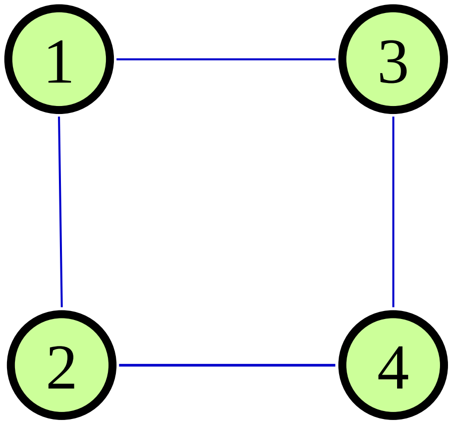
Sommario

- Introduzione
- Motivazioni e origini storiche
- Definizioni e proprietà di base
- Isomorfismi tra grafi
- Grafi di base
- Classi di grafi
- Grafi orientati
- Rappresentazioni
- Appendice

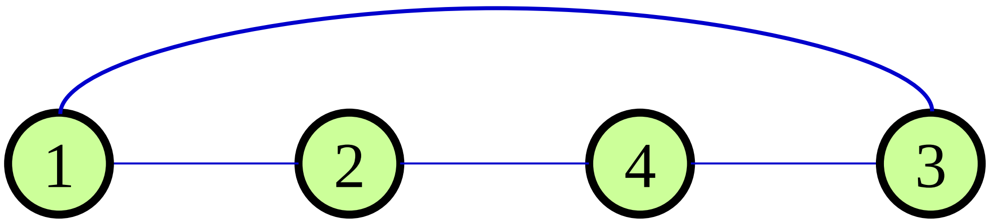
Grafi diversi o uguali?

Questi 2 grafi sono lo stesso grafo ?

$G = (V, E)$



$H = \{V', E'\}$



$V = \{1,2,3,4\}$
$E = \{\{1,2\}, \{2,4\}, \{3,4\}, \{1,3\}\}$

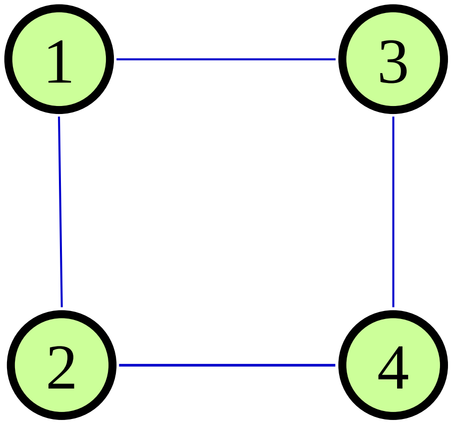
$V' = \{1,2,3,4\}$
$E' = \{\{1,2\}, \{2,4\}, \{3,4\}, \{1,3\}\}$

Stesso grafo ma disegnato (o embedded nel piano) differentemente

Grafi diversi o uguali?

Questi 2 grafi sono lo stesso grafo ?

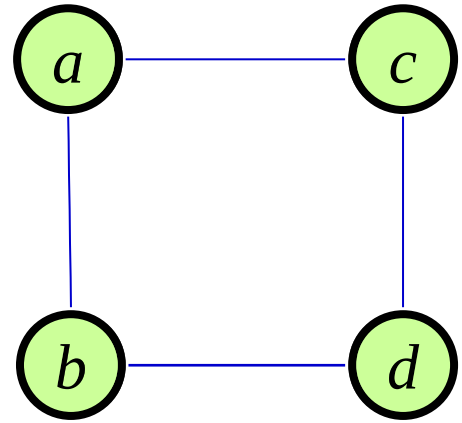
$G = (V, E)$



$V \mapsto V'$

$$1 \mapsto a$$
$$2 \mapsto b$$
$$3 \mapsto c$$
$$4 \mapsto d$$

$K = \{V', E'\}$

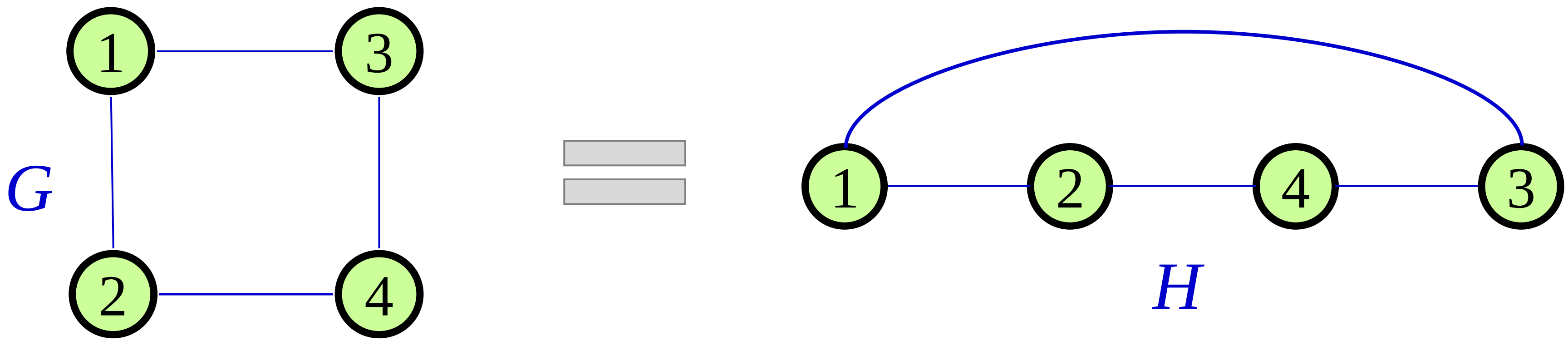


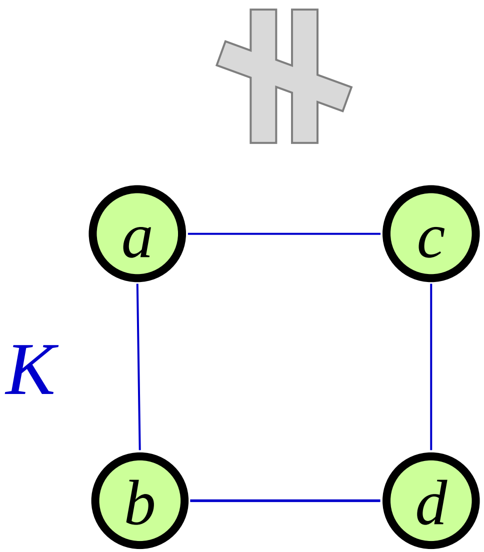
$E = \{\{1,2\}, \{2,4\}, \{3,4\}, \{1,3\}\}$

$E' = \{\{a,b\}, \{b,c\}, \{c,d\}, \{a,c\}\}$

Grafi diversi (insieme di vertici diversi) ma stessa struttura

Grafi diversi o uguali?





Anche se *G* e *K* sono grafi diversi, hanno la stessa struttura

Dal punto di vista strutturale, un grafo *H* ottenuto da *G* rinominando i vertici oppure «disegnando» *G* in modo diverso sono lo stesso grafo.

Isomorfismi

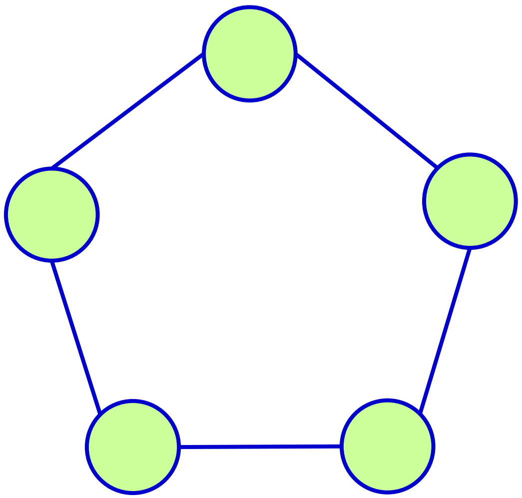
[Definizione] Due grafi $H = (W, F)$ e $G = (V, E)$ sono detti isomorfi ($H \cong G$) se esiste una biiezione $f: W \rightarrow V$ tale che

$$\{u,v\} \in F \quad \Leftrightarrow \quad \{f(u), f(v)\} \in E$$

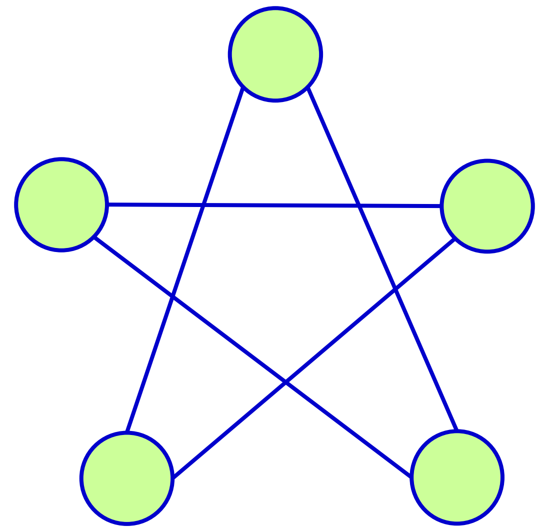
In altri termini, H è isomorfo a G se esiste una «etichettatura» dei nodi di H e G che definisce esattamente lo stesso insieme di archi

Grafi isomorfi, esempio

H e *G* sono isomorfi?



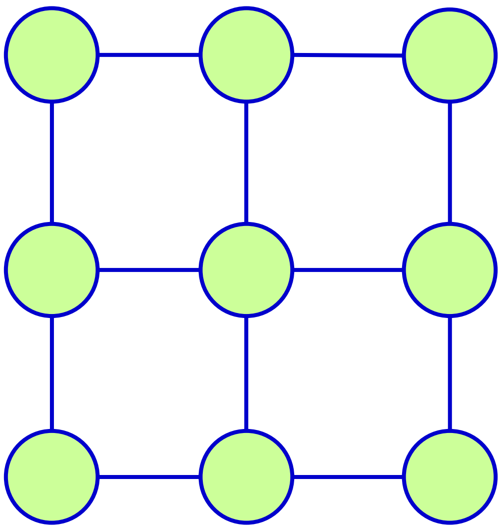
H



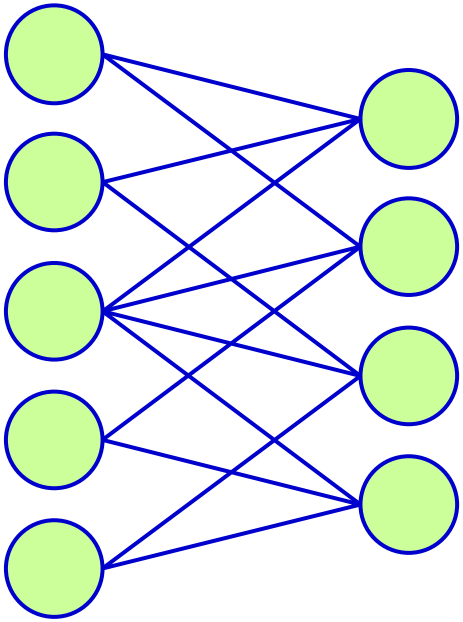
G

Grafi isomorfi, esempio

H e *G* sono isomorfi?



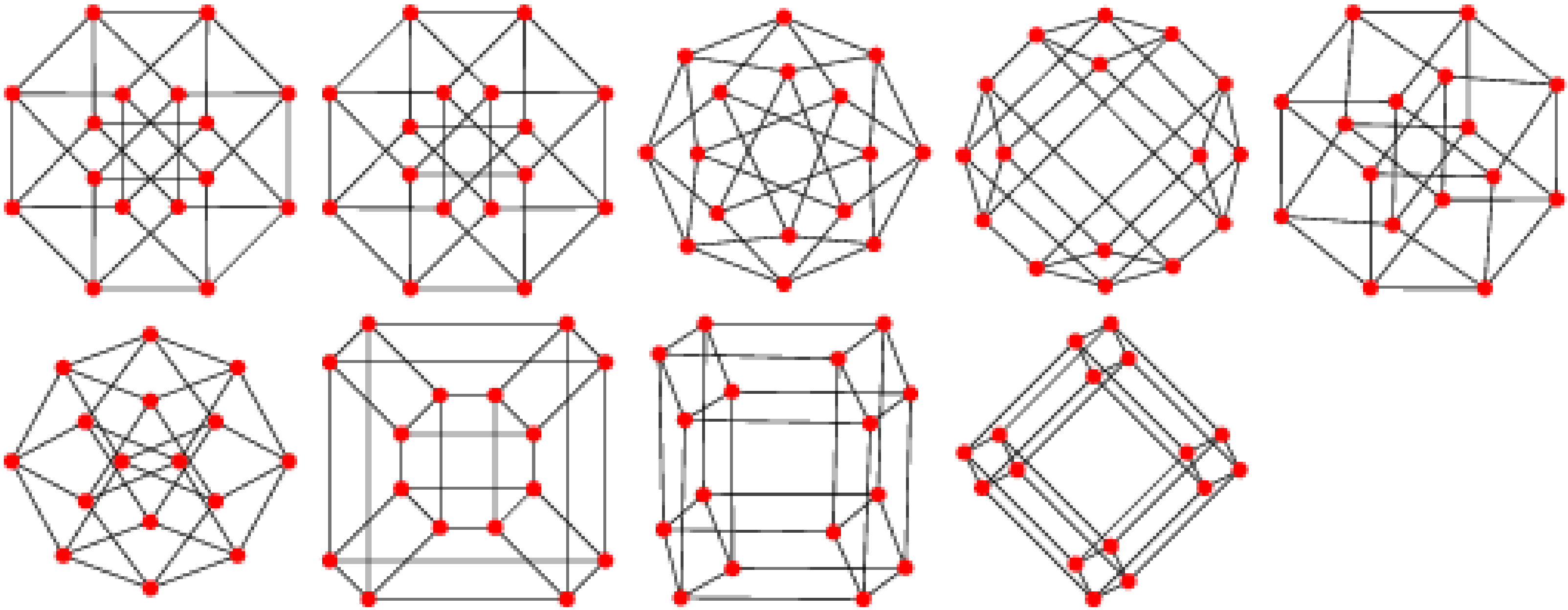
H



G

Isomorfismi

Il *Tesseract*



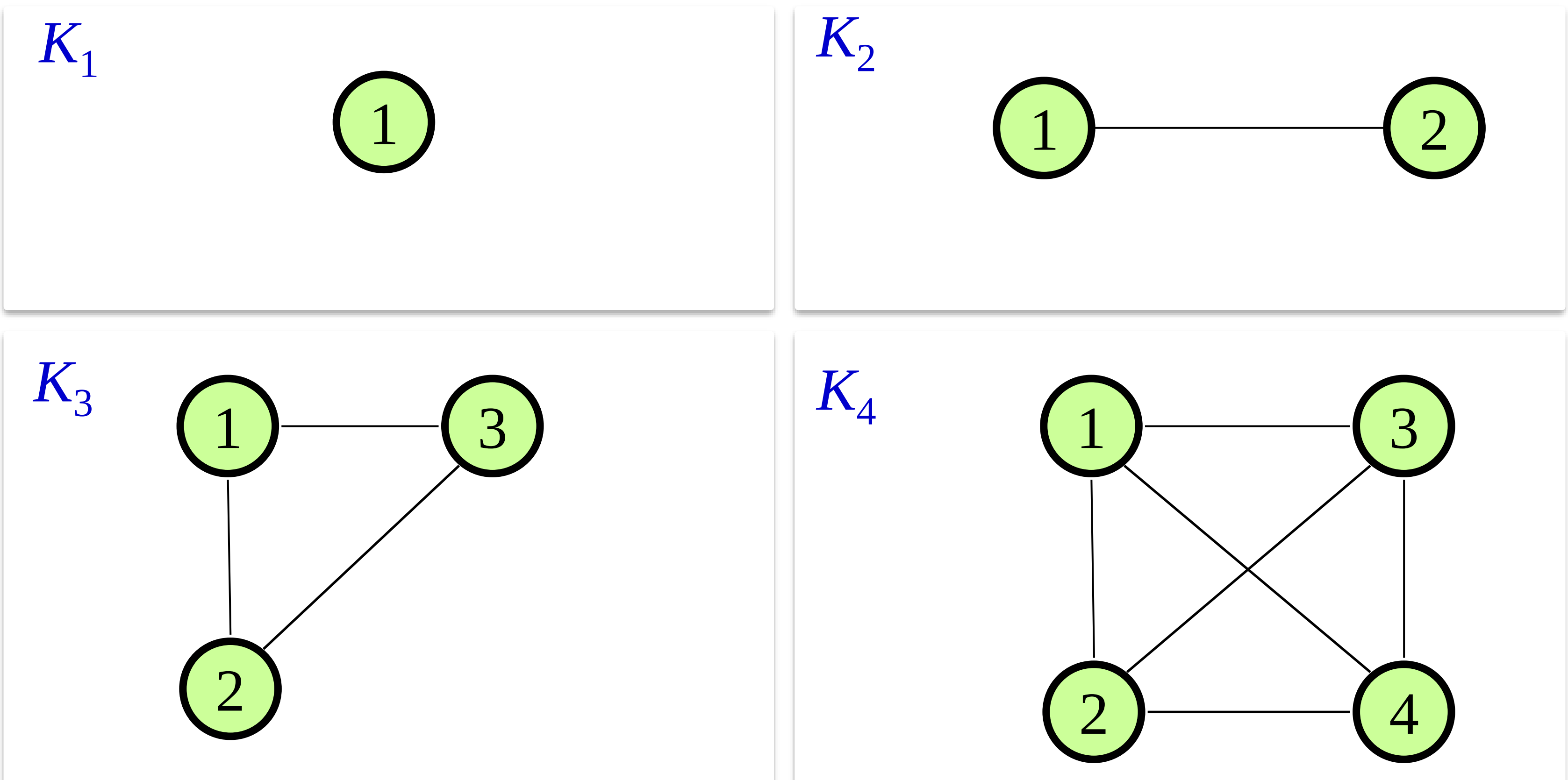
grafi isomorfi rappresentano la stessa relazione
(quindi sono equivalenti dal punto di vista matematico).

Sommario

- Introduzione
- Motivazioni e origini storiche
- Definizioni e proprietà di base
- Isomorfismi tra grafi
- Grafi di base
- Classi di grafi
- Grafi orientati
- Rappresentazioni
- Appendice

Grafo completo

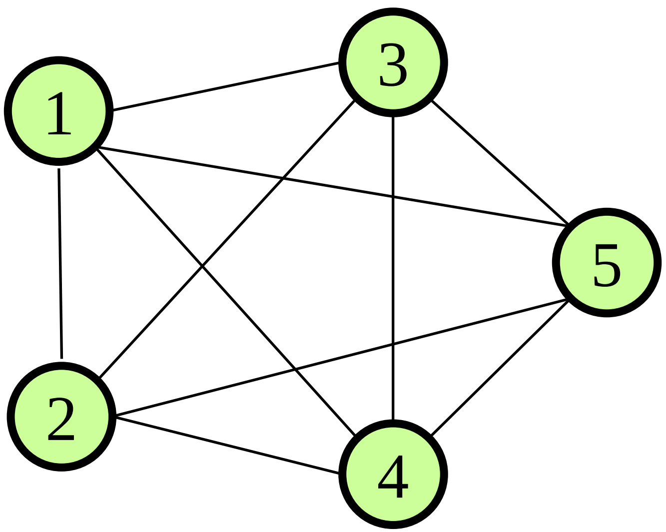
Un grafo con n vertici si dice completo (e si indica con K_n) se ha un arco per ogni coppia di nodi. $K_n = (V, (V \times V))$



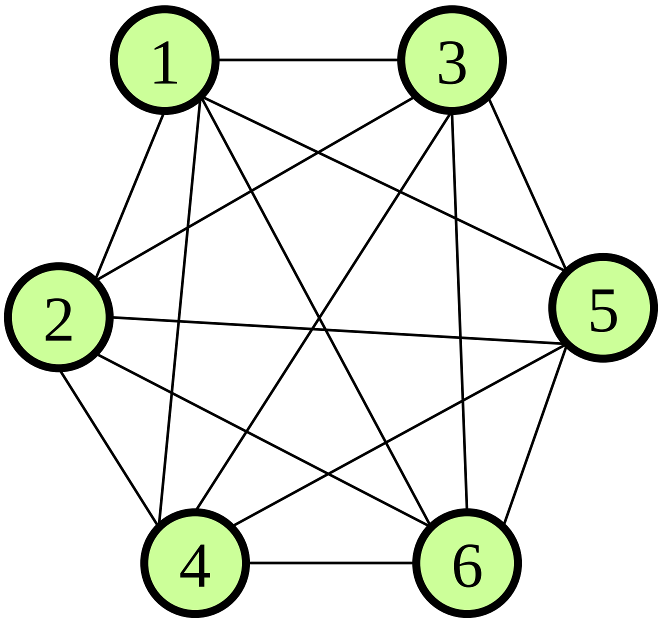
Grafo completo

Un grafo con n vertici si dice completo (e si indica con K_n) se ha un arco per ogni coppia di nodi. $K_n = (V, (V \times V))$

K_5

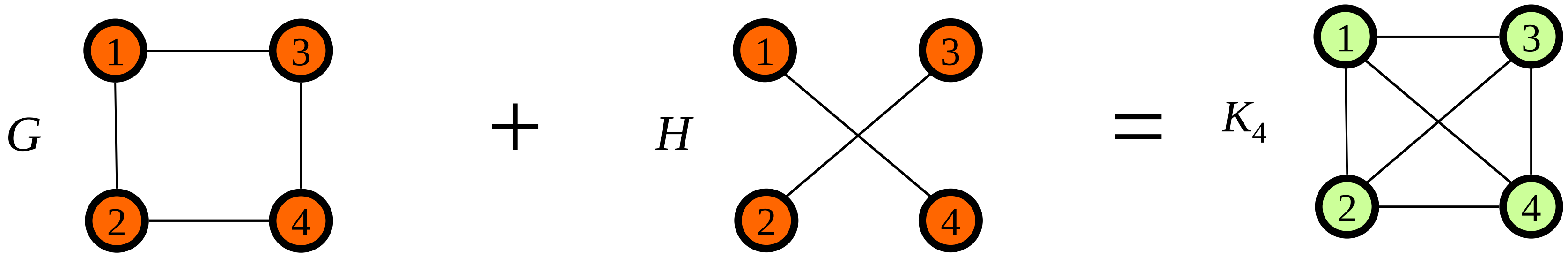


K_6



Grafo complemento, grafo vuoto

Un grafo H è il complemento di un grafo G se ha gli stessi vertici di G e solo gli archi che mancano a G per renderlo completo.



$H = (V, (V \times V) \setminus E)$ è il complemento di $G = (V, E)$

Il complemento K di un grafo completo si dice grafo vuoto.

$K = (V, \varnothing)$

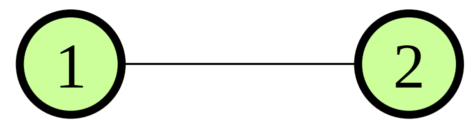
Percorso

Un percorso P è un grafo costituito da una sequenza di archi (o nodi) adiacenti. Il primo e ultimo nodo del percorso si chiamano estremi. La lunghezza $l(P)$ di P è data dal numero di archi che lo compongono.

$P_0 \equiv K_1$

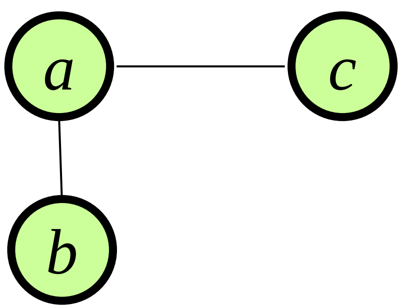
$l(P_0) = 0$

$P_1 \equiv K_2$



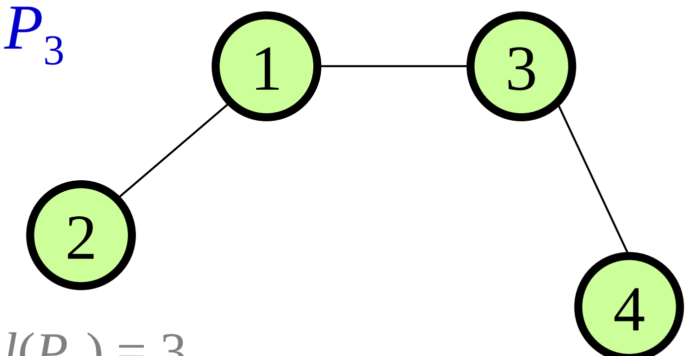
$l(P_1) = 1$

P_2



$l(P_2) = 2$

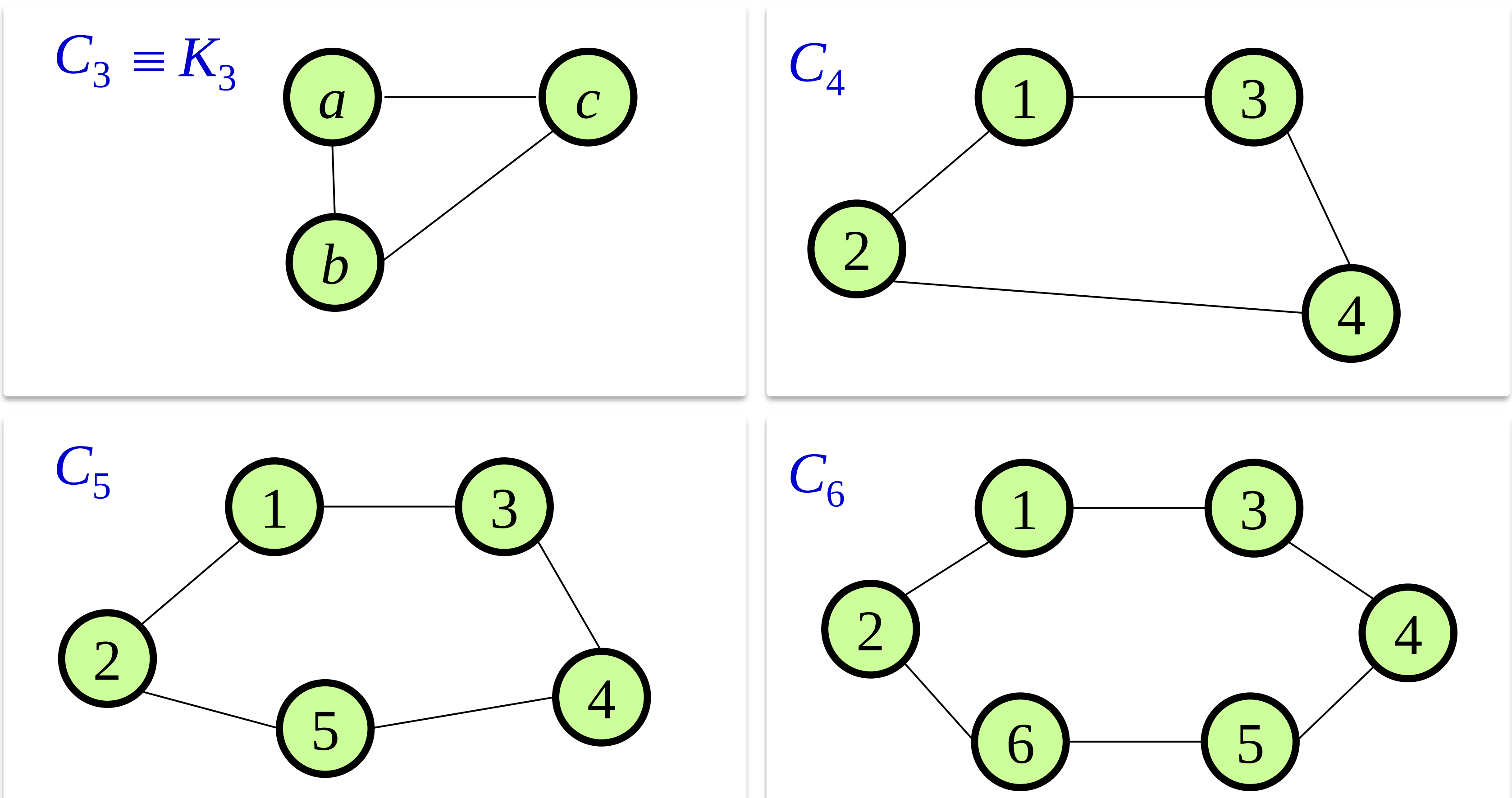
P_3



$l(P_3) = 3$

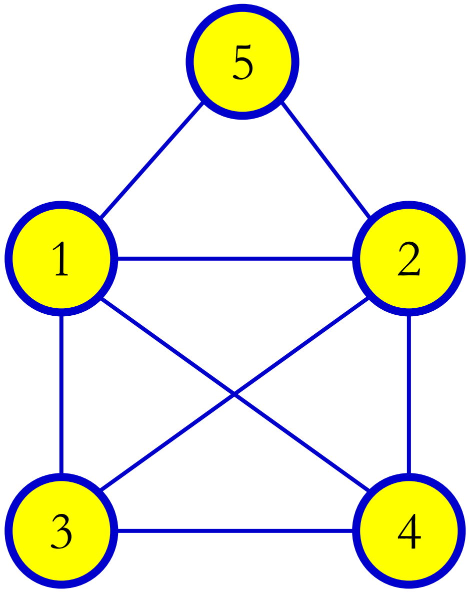
Ciclo

Un ciclo C (o percorso chiuso) è un percorso P con estremi coincidenti.

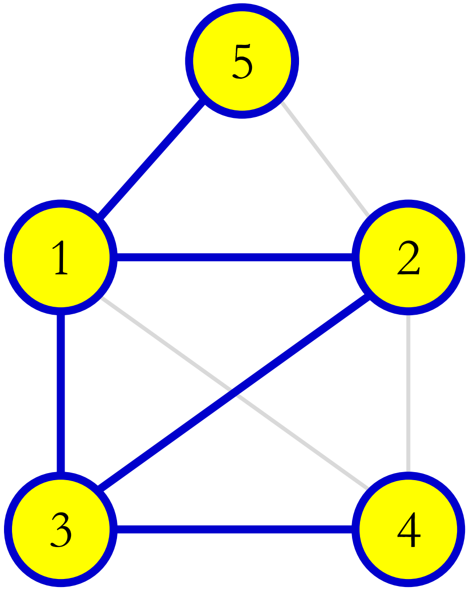


Grafo parziale

$G = (V, E)$



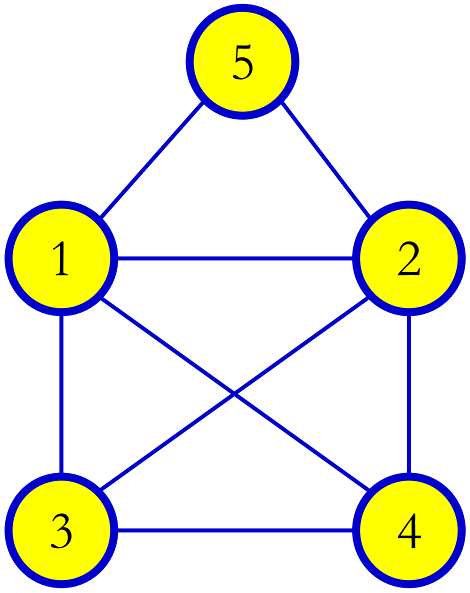
$H = (V, F)$ con $F \subset E$

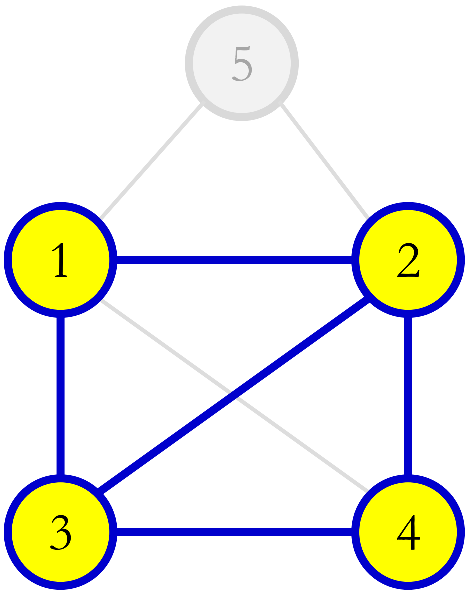


H è un grafo parziale di G se ha gli stessi nodi di G e solo una parte degli archi.

Sottografo

$G = (V, E)$ $H = (W, F)$ con $W \subset V$ e $F \subset E$

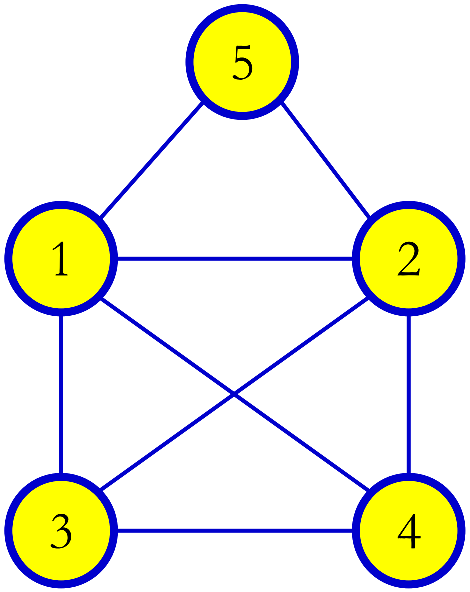


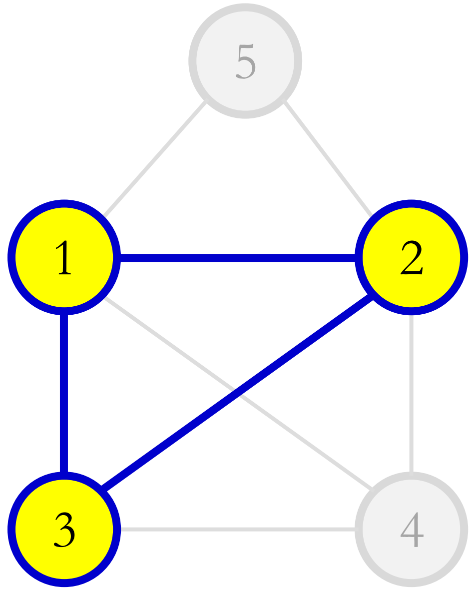


H è un sottografo di G se ha solo una parte dei nodi di G e solo una parte degli archi.

Sottografo indotto (da nodi)

$G = (V, E)$ $\qquad$ $H = (W, F)$ con $F = E \cap \binom{W}{2}$



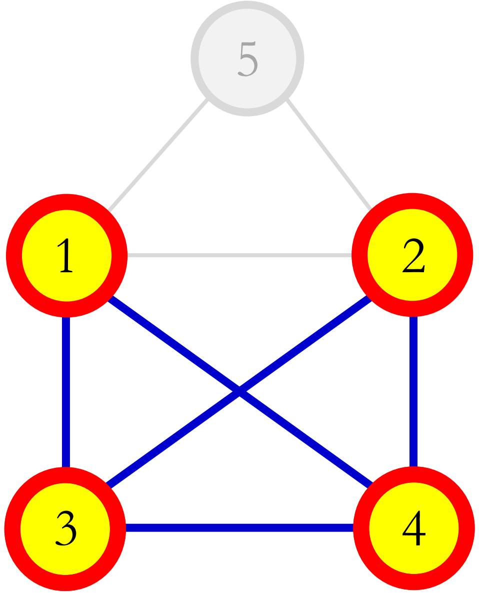


$H = (W, F)$ è un sottografo di $G = (V, E)$ indotto dai nodi $W \subseteq V$ se F contiene solo archi $\{u, v\} \in E$ tali che $u, v \in W$.

H è indicato con $G[W]$

Cammini, passeggiate e percorsi di un grafo

Un cammino (*walk*) di un grafo G è una sequenza finita di archi (o nodi) adiacenti $[\{u,v\}, \{v,w\},\ldots]$.



$P = [\{4,1\}, \{1,3\}, \{3,4\}, \{4,2\}, \{2,3\}]$

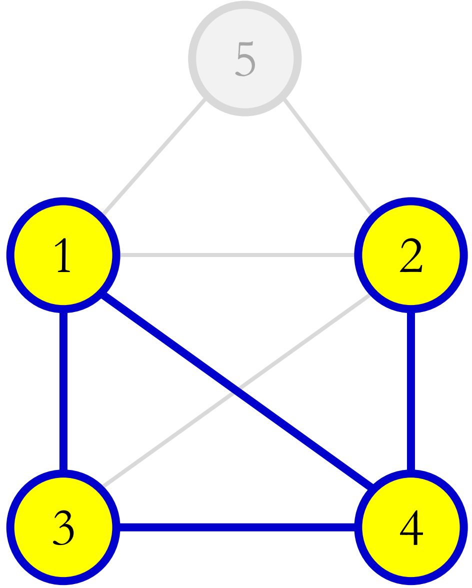
Estremi: nodi 4 e 3

[Nota] il sottografo corrispondente al cammino in generale non è isomorfo a un percorso

Cammini, passeggiate e percorsi di un grafo

Un cammino di un grafo G è
una passeggiata (*trail*) o cammino elementare se
gli archi (ma non necessariamente i nodi) sono
tutti distinti

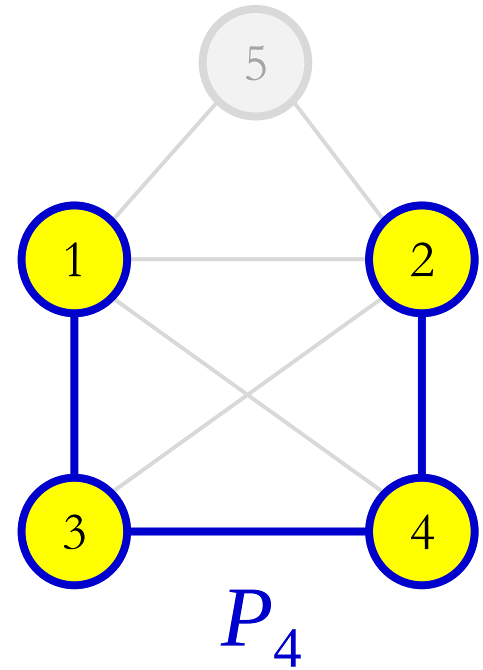
P = [{4,1}, {1,3}, {3,4}, {4,2}]
Estremi: nodi 4 e 2



[Nota] il sottografo corrispondente alla passeggiata in generale non è isomorfo a un percorso

Cammini, passeggiate e percorsi di un grafo

Un cammino di un grafo G è
un percorso (*path*) o cammino semplice
se gli archi e i nodi sono tutti distinti

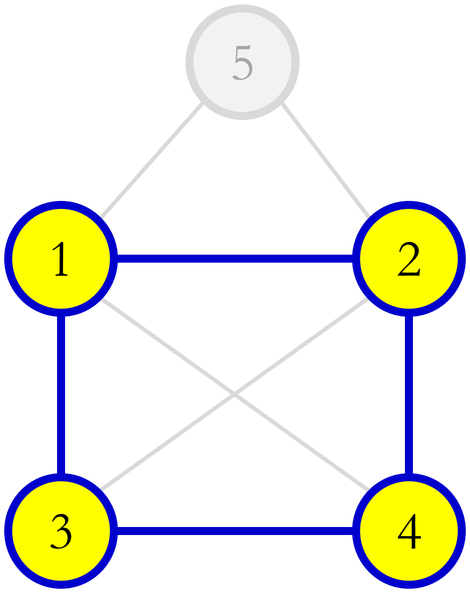


P = [{1,3}, {3,4}, {4,2}]
Estremi: nodi 1 e 2

[Nota] il sottografo corrispondente al percorso è isomorfo a un percorso P_k. Si dice anche che un grafo G contiene un percorso P_k

Cammini, passeggiate e percorsi chiusi

Un cammino, passeggiata o percorso
è chiuso se gli estremi coincidono

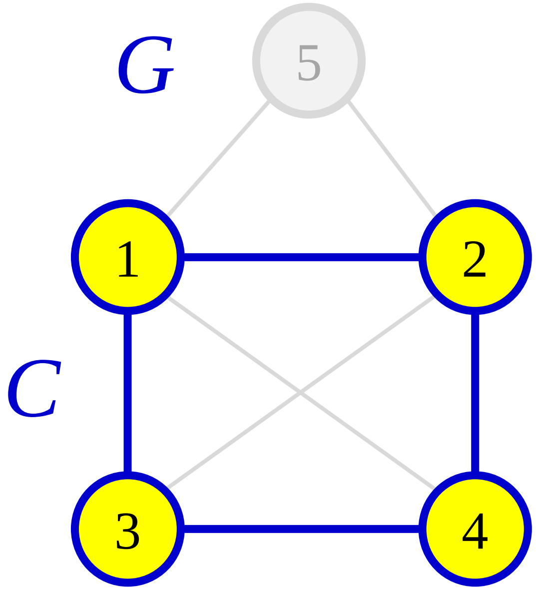


$C = [\{1,3\}, \{3,4\}, \{4,2\}, \{2, 1\}]$

[Nota] il sottografo corrispondente a un cammino (o passeggiata) chiuso/a in generale non è isomorfo a un ciclo

Grafi aciclici

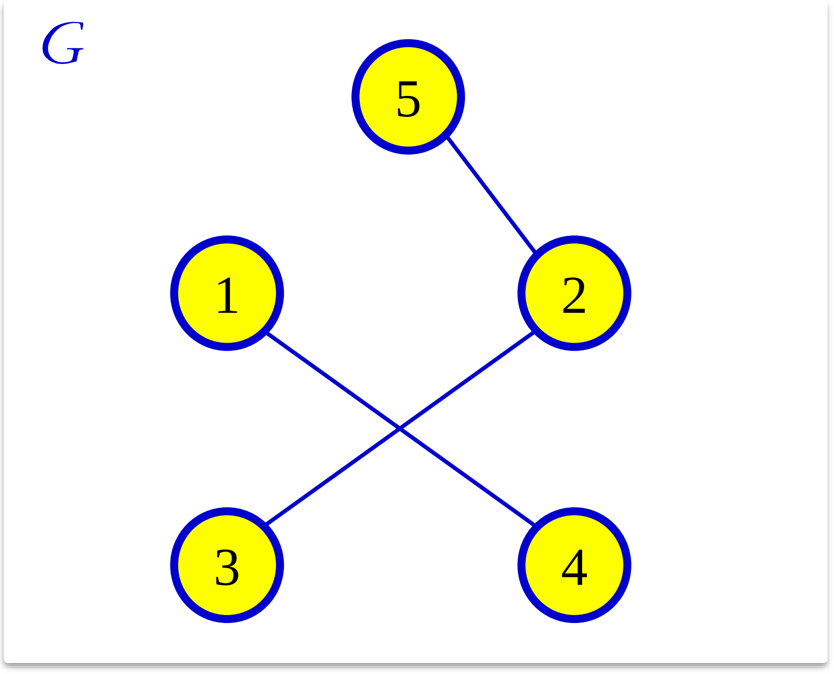
Un grafo G contiene un ciclo C_k se ammette un sottografo isomorfo a C_k



Un grafo G è detto aciclico se non contiene alcun ciclo

Connessione

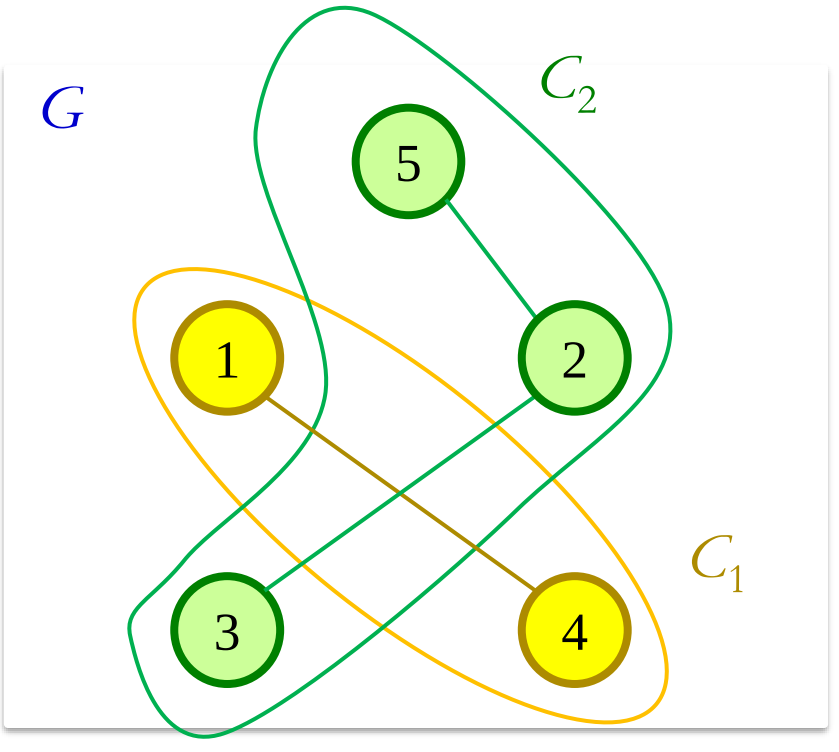
Due nodi $u, v \in V$ di un grafo simmetrico G si dicono connessi se esiste in G un cammino tra u e v



- I nodi 3 e 2 sono connessi
- I nodi 1 e 5 **non** sono connessi

Connessione

La relazione di connessione <u>partiziona</u> G in componenti connesse



G è formato dalle 2 componenti connesse C_1 e C_2

G si dice connesso se è composto da <u>una sola</u> componente connessa

Connessione: ipotesi di lavoro

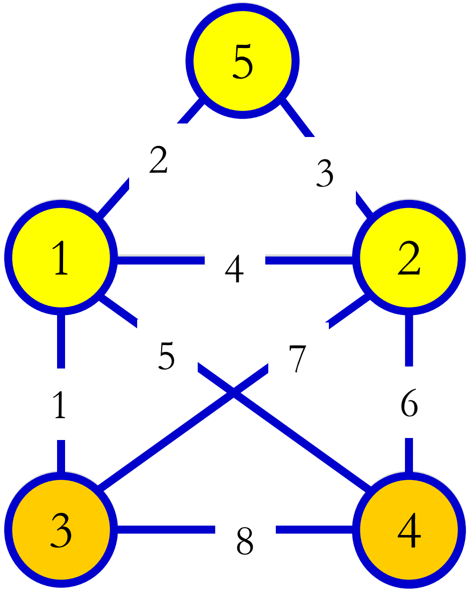
Dato che tutti i problemi su grafo discussi in seguito sono facilmente decomponibili per componenti connesse, se non diversamente specificato

i grafi riportati sono sempre intesi come
grafi connessi

Sommario

Cammino e ciclo *euleriano*

Un cammino (ciclo) è euleriano se e solo se attraversa **tutti** gli archi del grafo **una e una sola** volta

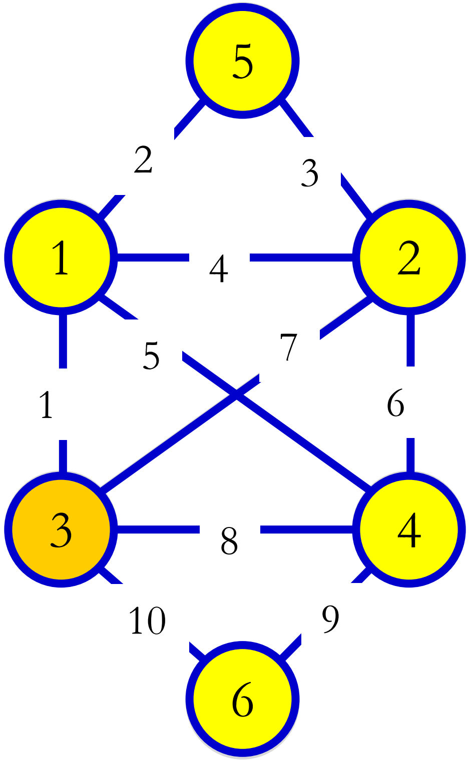


P = [{3, 1}, {1, 5}, {5, 2}, {2, 1} {1, 4}, {4, 2}, {2, 3}, {3, 4}]

è un cammino euleriano

Cammino e ciclo *euleriano*

Un cammino (ciclo) è euleriano se e solo se attraversa **<u>tutti</u>** gli archi del grafo **<u>una e una sola</u>** volta

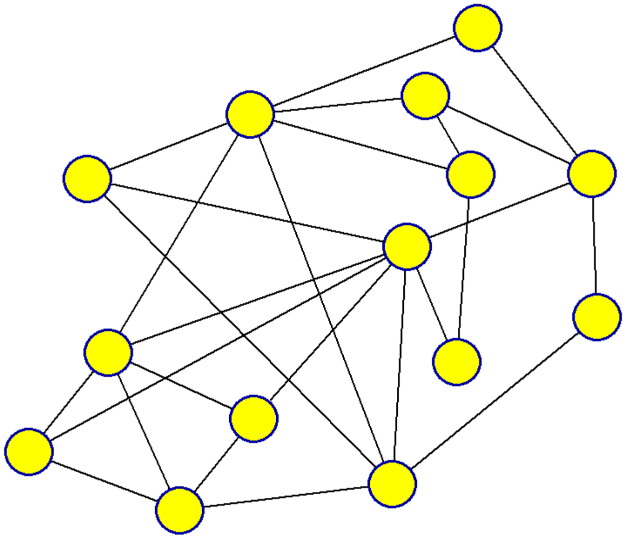


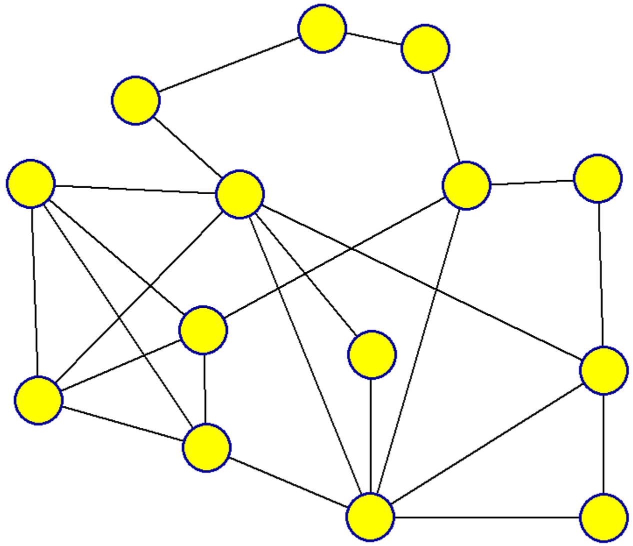
$C = [\{3, 1\}, \{1, 5\}, \{5, 2\}, \{2, 1\}\ \{1, 4\},$
$\{4, 2\}, \{2, 3\}, \{3, 4\}, \{4, 6\}, \{6, 3\}]$

è un ciclo euleriano

Caratterizzazione dei grafi euleriani

Su quali di questi grafi è possibile costruire un ciclo euleriano?



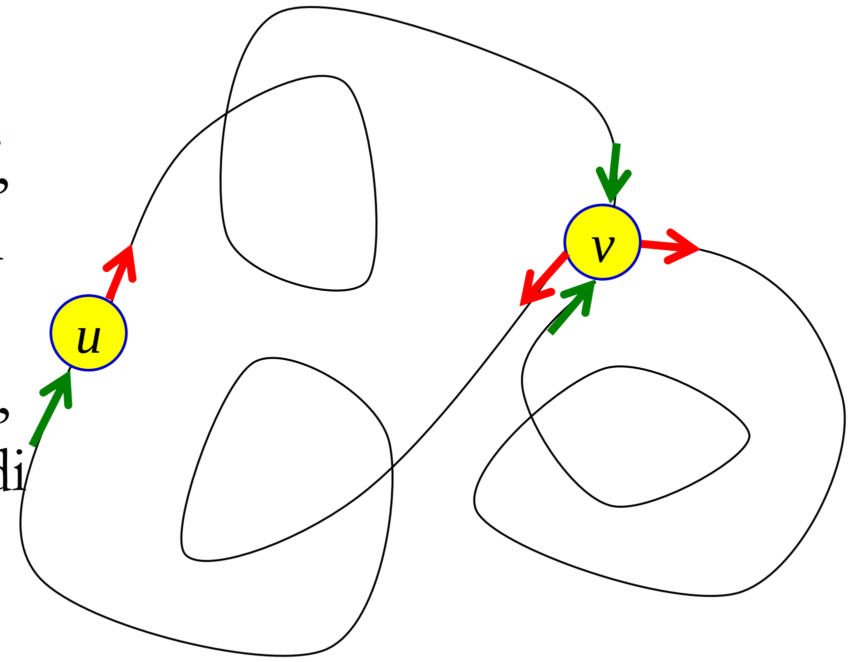


[Teorema di Eulero] Un grafo simmetrico è *euleriano*, cioè ammette un ciclo euleriano, **se e solo se** è connesso e ogni suo nodo ha grado pari.

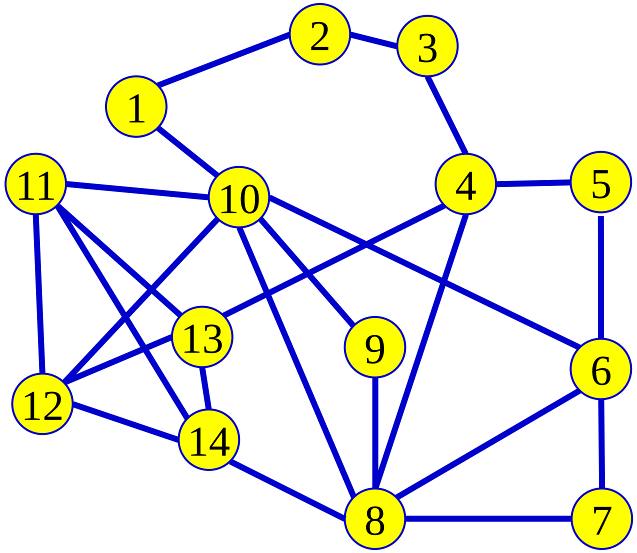
G euleriano $\Rightarrow$ G connesso e con nodi di grado pari.

Sia C un ciclo euleriano di G

- Ogni nodo $v \in V$ è **attraversato** da C, quindi gli archi di $\delta(v)$ sono presenti in coppie C (archi *entranti* e *uscenti*)
- Siccome C contiene tutti gli archi di G, la stella $\delta(v)$ ha un multiplo di coppie di archi, quindi $d(v)$ è pari.

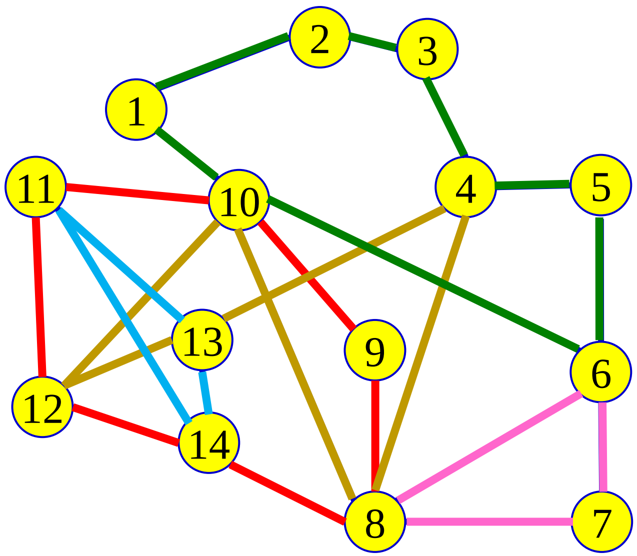


G connesso e con nodi di grado pari $\Rightarrow$ G euleriano



G connesso e con nodi di grado pari $\Rightarrow$ G euleriano

- Dato che ogni nodo è di grado pari, G può essere decomposto in cicli

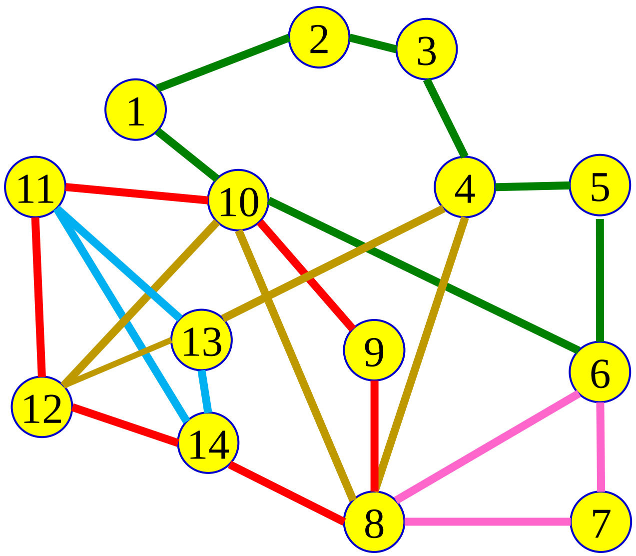


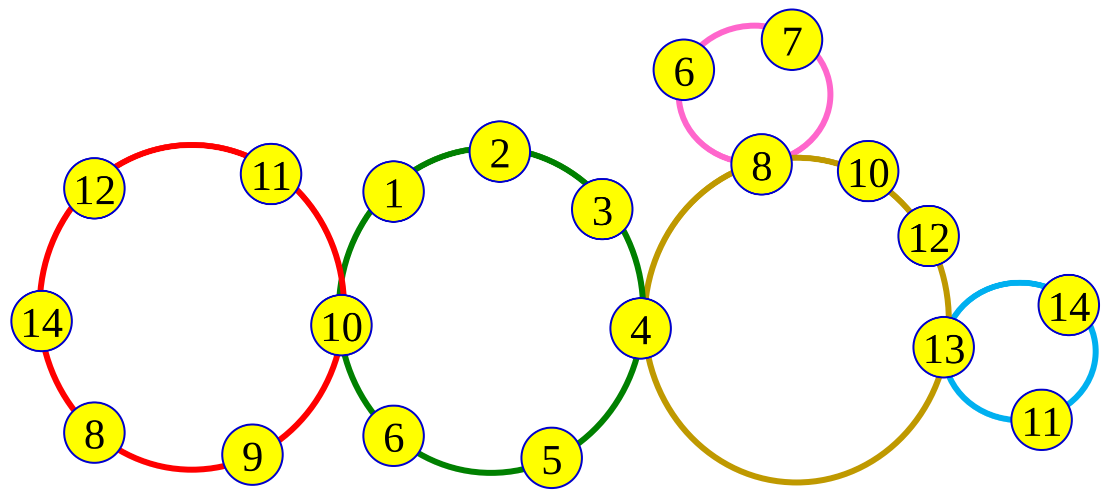
1. (almeno un ciclo esiste sempre: infatti dal grado pari si deduce che $\delta(v) \geq 2 \ \forall v$ e quindi $|E| \geq |V|$. Segue che G **non è** un albero e quindi ammette un ciclo)

2. Una volta rimosso un ciclo, di nuovo ho un grafo con tutti nodi di grado pari

G connesso e con nodi di grado pari $\Rightarrow$ G euleriano

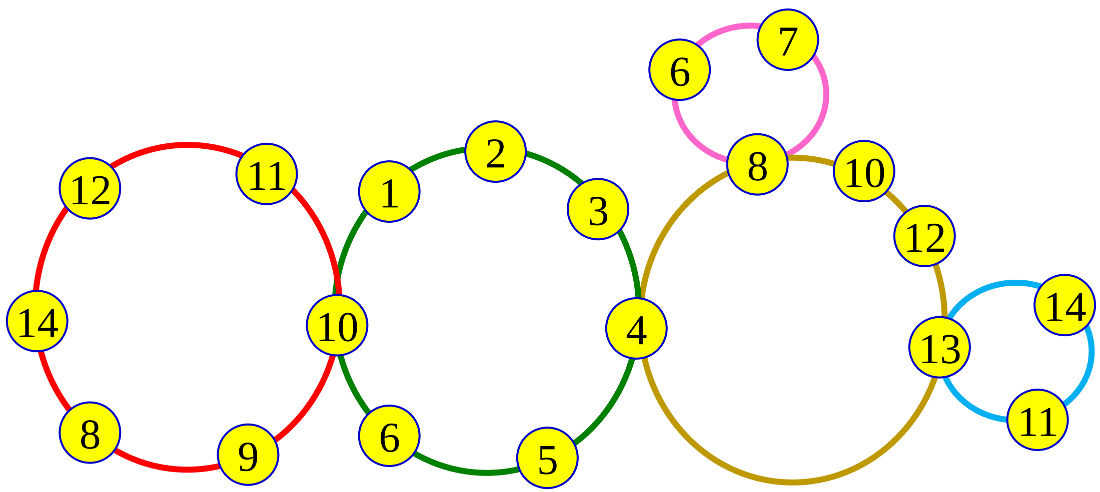
- Dato che ogni nodo è di grado pari, G può essere decomposto in cicli
- I cicli non sono disgiunti perché G è connesso





G connesso e con nodi di grado pari $\Rightarrow$ G euleriano

- Si noti che ogni arco di G è presente una e una sola volta, quindi è facile ottenere un ciclo euleriano di G

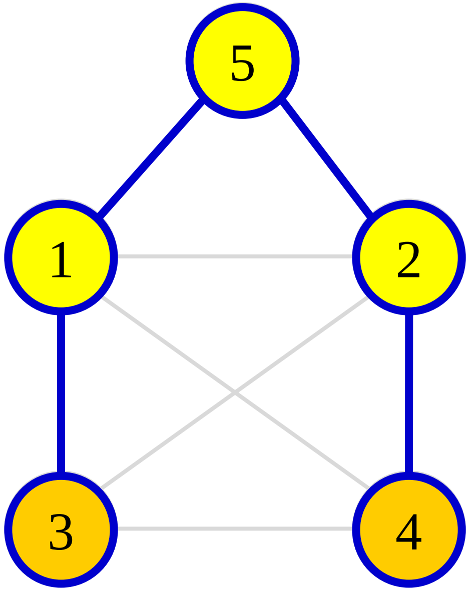


v.l.a.d.

Percorso *hamiltoniano*

Un percorso è hamiltoniano se e solo se attraversa **tutti** i nodi del grafo **una e una sola** volta

Un percorso hamiltoniano è un sottografo isomorfo a $\boldsymbol{P_{n-1}}$



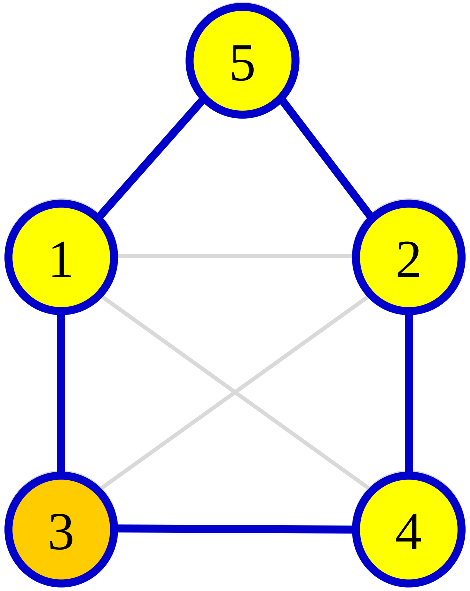
P = [{3, 1}, {1, 5}, {5, 2}, {2, 4}]

è un percorso hamiltoniano

Ciclo *hamiltoniano*

Un ciclo è hamiltoniano se e solo se attraversa **tutti** i nodi del grafo **una e una sola** volta

Un ciclo hamiltoniano è un sottografo isomorfo a C_n

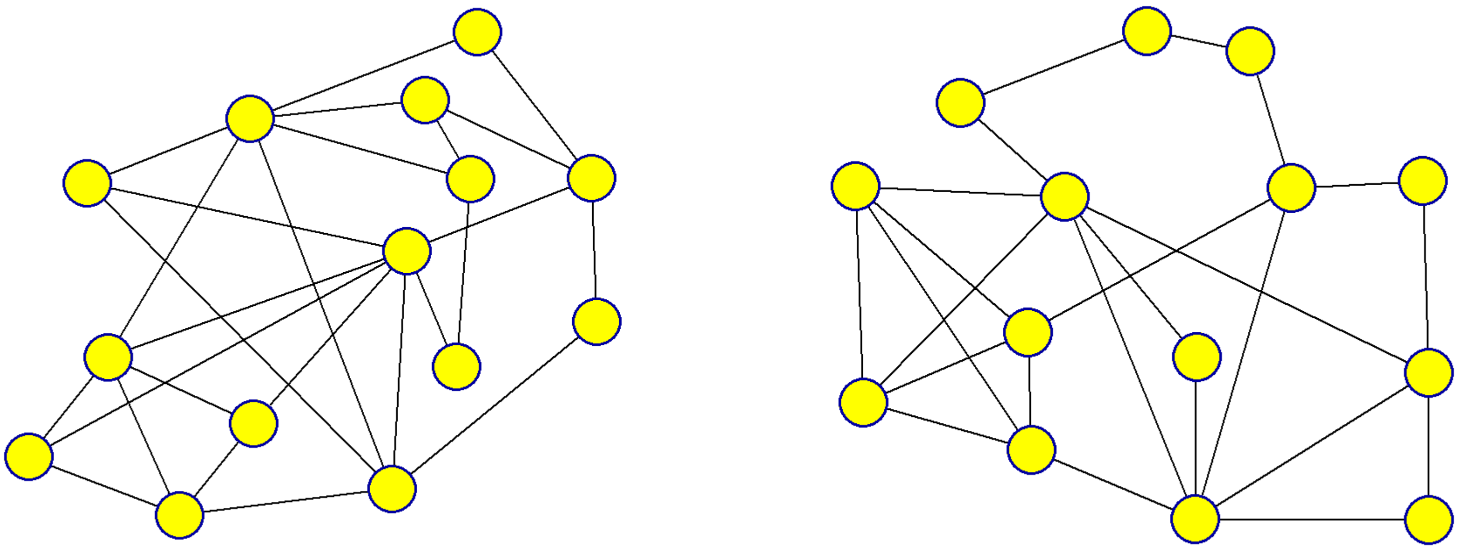


P = [{3, 1}, {1, 5}, {5, 2}, {2, 4}, {4, 1}]

è un ciclo (o *tour*) hamiltoniano

Grafi *hamiltoniani*

un grafo *G* è hamiltoniano se contiene un ciclo hamiltoniano



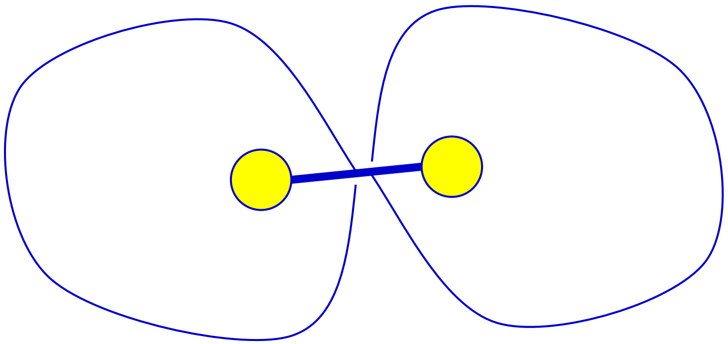
Quali di questi grafi è hamiltoniano?

Grafi *hamiltoniani* : condizioni necessarie

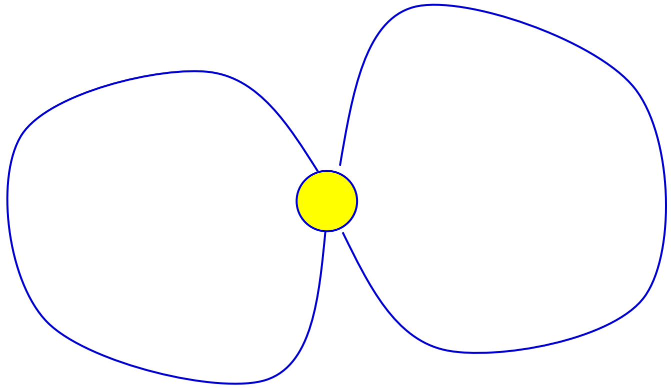
Se $G = (V, E)$ è hamiltoniano allora

- $d(v) \geq 2 \quad \forall v \in V$

- G non ha *cut-edges*



- G non ha *cut-vertices*



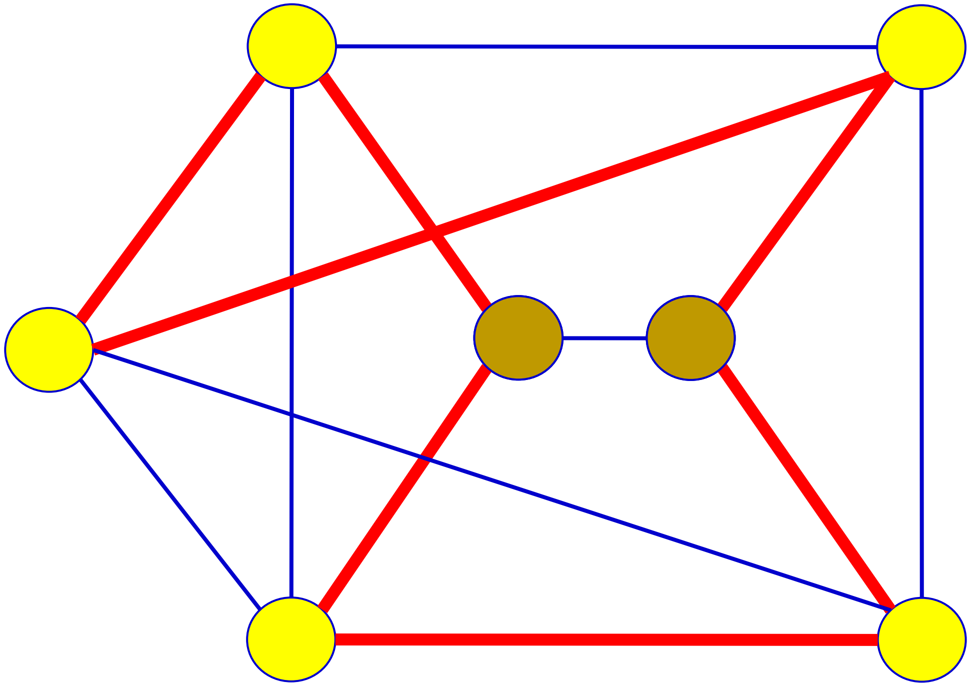
Grafi *hamiltoniani* : condizioni sufficienti

[Teorema di Ore (1960)]
Un grafo $G = (V, E)$ con almeno 3 vertici e $d(u) + d(v) \geq n$ per ogni coppia u, v di vertici non adiacenti è hamiltoniano

Corollario [Teorema di Dirac (1952)]
Un grafo $G = (V, E)$ con almeno 3 vertici e $d(u) \geq n / 2$ per tutti $u \in V$ è hamiltoniano

Il Teorema di Dirac è un caso speciale del teorema di Ore: se $d(u) \geq n / 2$ per tutti i $u \in V$ allora per qualsiasi coppia di vertici vale la condizione del teorema di Ore.

Condizioni di Ore e Dirac

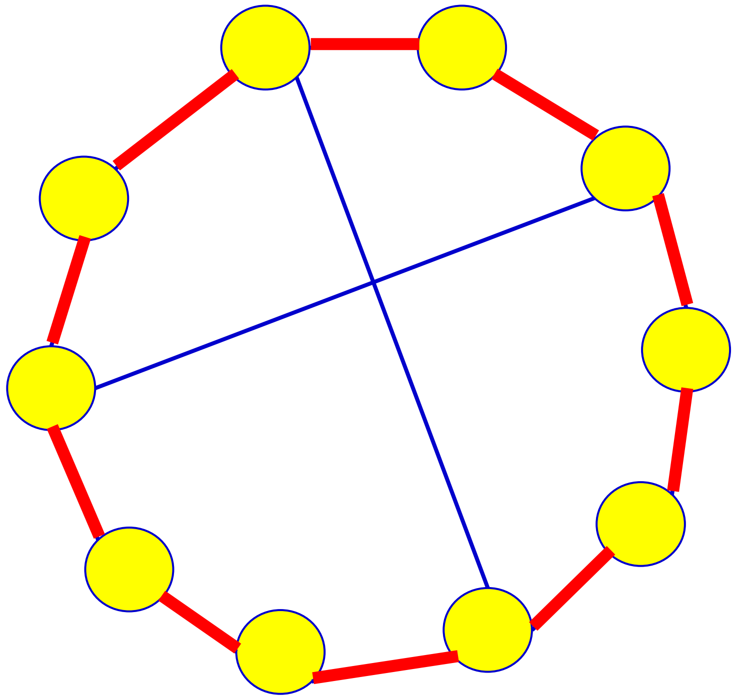


Questo grafo non soddisfa la condizione di Dirac…

Ma soddisfa la condizione di Ore

Infatti è hamiltoniano

Teorema di Ore: solo una condizione sufficiente



Questo grafo non soddisfa la condizione di Ore ma è hamiltoniano!

Non esiste una caratterizzazione dei grafi hamiltoniani !

Problemi (un po' più seri) di un ingegnere

- Genome assembly problem: Un genoma (semplificato) è una lunga stringa (milioni o anche miliardi) di *simboli* A, C, G e T (i nucleotidi Adenina, Citosina, Guanina e Timina)

- L'intero genoma è in generale sconosciuto ma piccoli frammenti (sottostringhe) possono essere ottenuti con tecniche di sequenziamento dei geni.

Qual è una stringa verosimile di genoma che può essere ricostruita a partire dai frammenti disponibili?

Problemi (un po' più seri) di un ingegnere

Un caso giocattolo

- Sottosequence disponibili di A, C, G e T

TCAC

ACCG

CGGT

GTCCCT

AACG

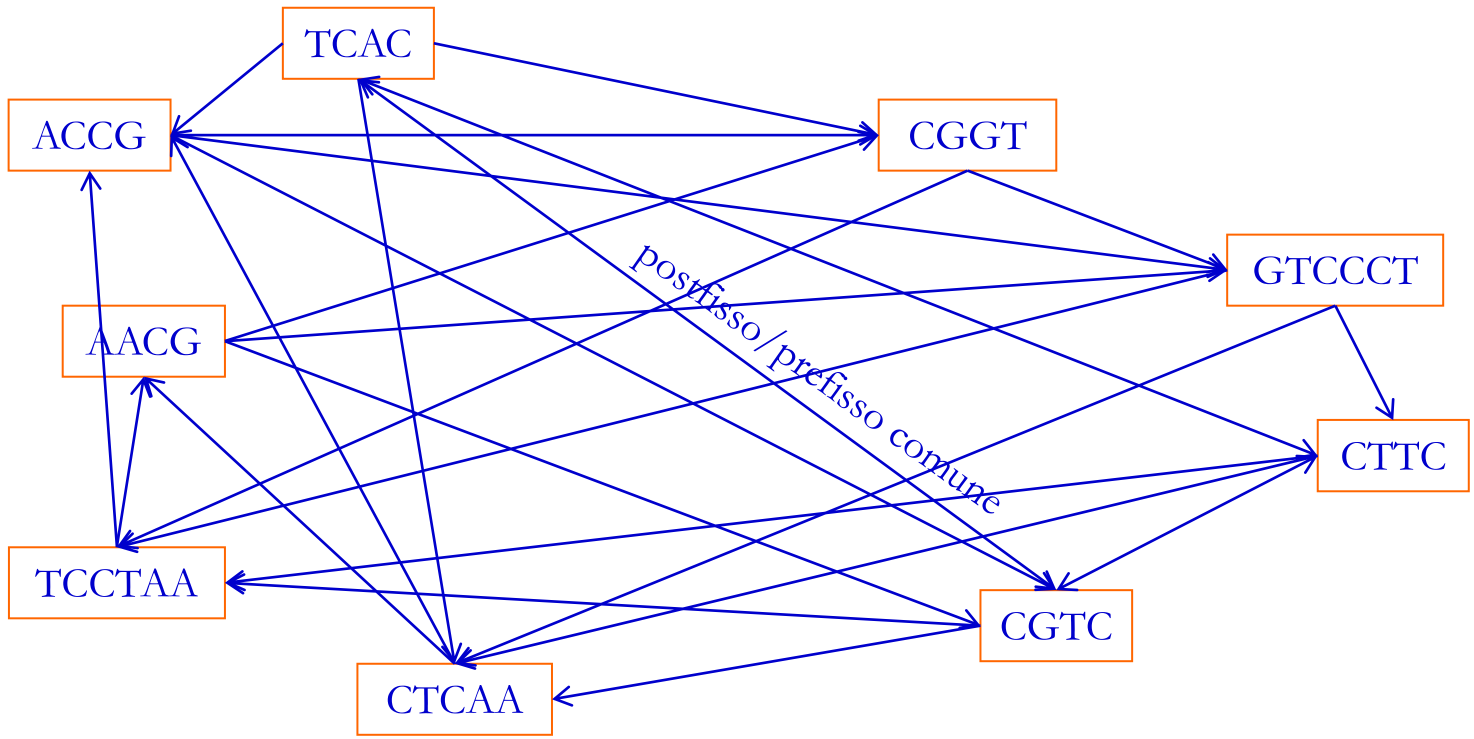
CTTC

TCCTAA

CGTC

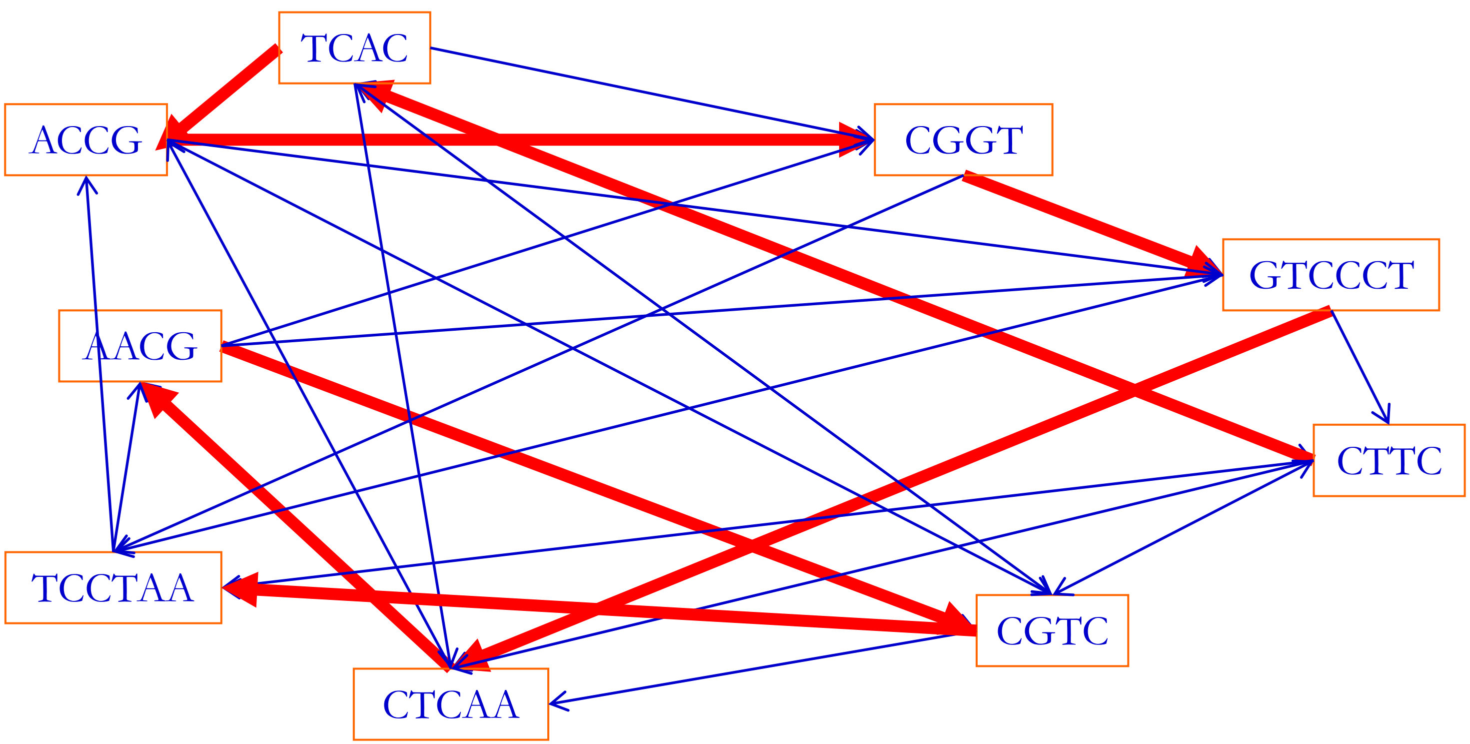
CTCAA

Problemi (un po' più seri) di un ingegnere



- Una ricostruzione complete del genoma è possible se e solo se il grafo ammette un *cammnino diretto hamiltoniano*

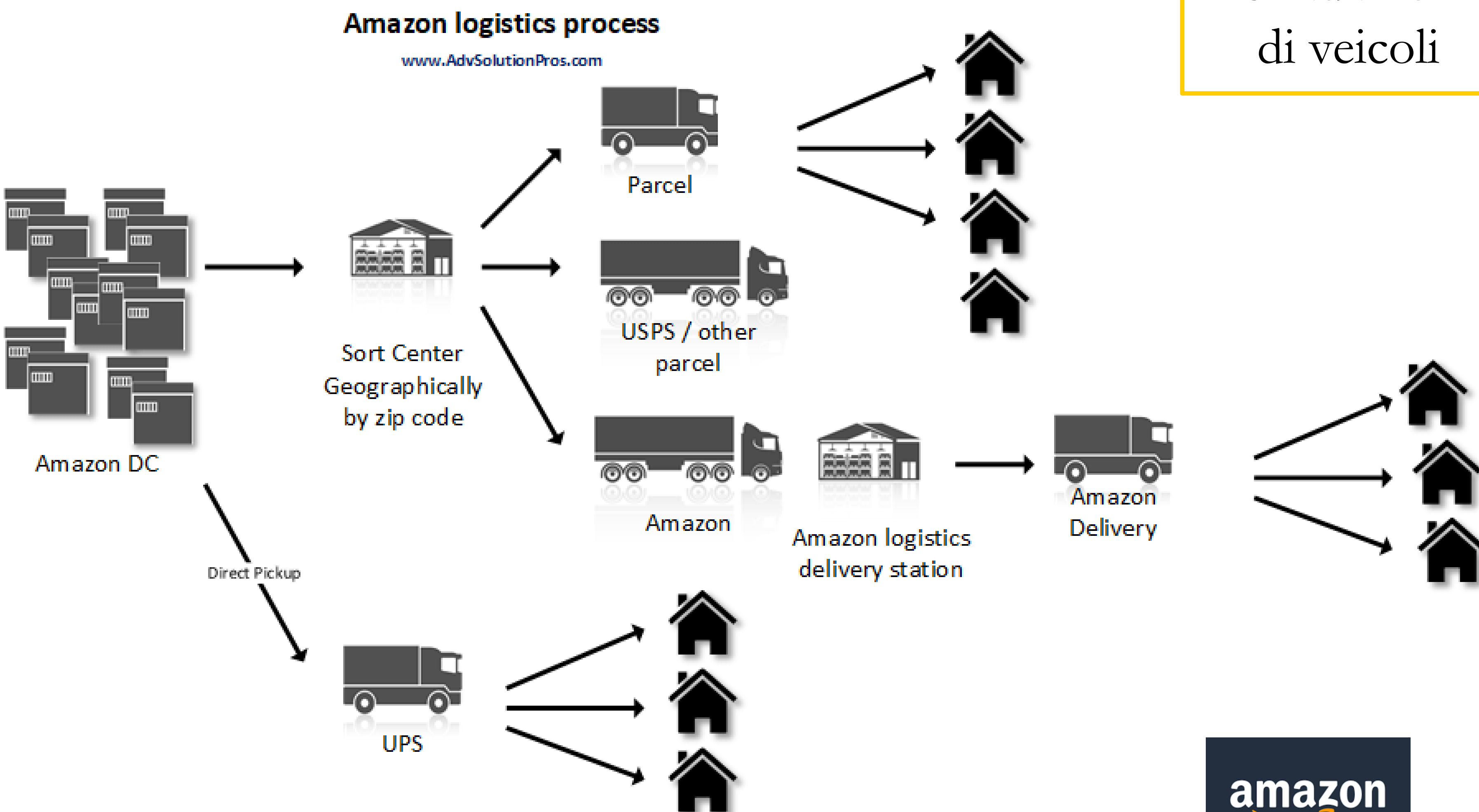
Problemi (un po' più seri) di un ingegnere

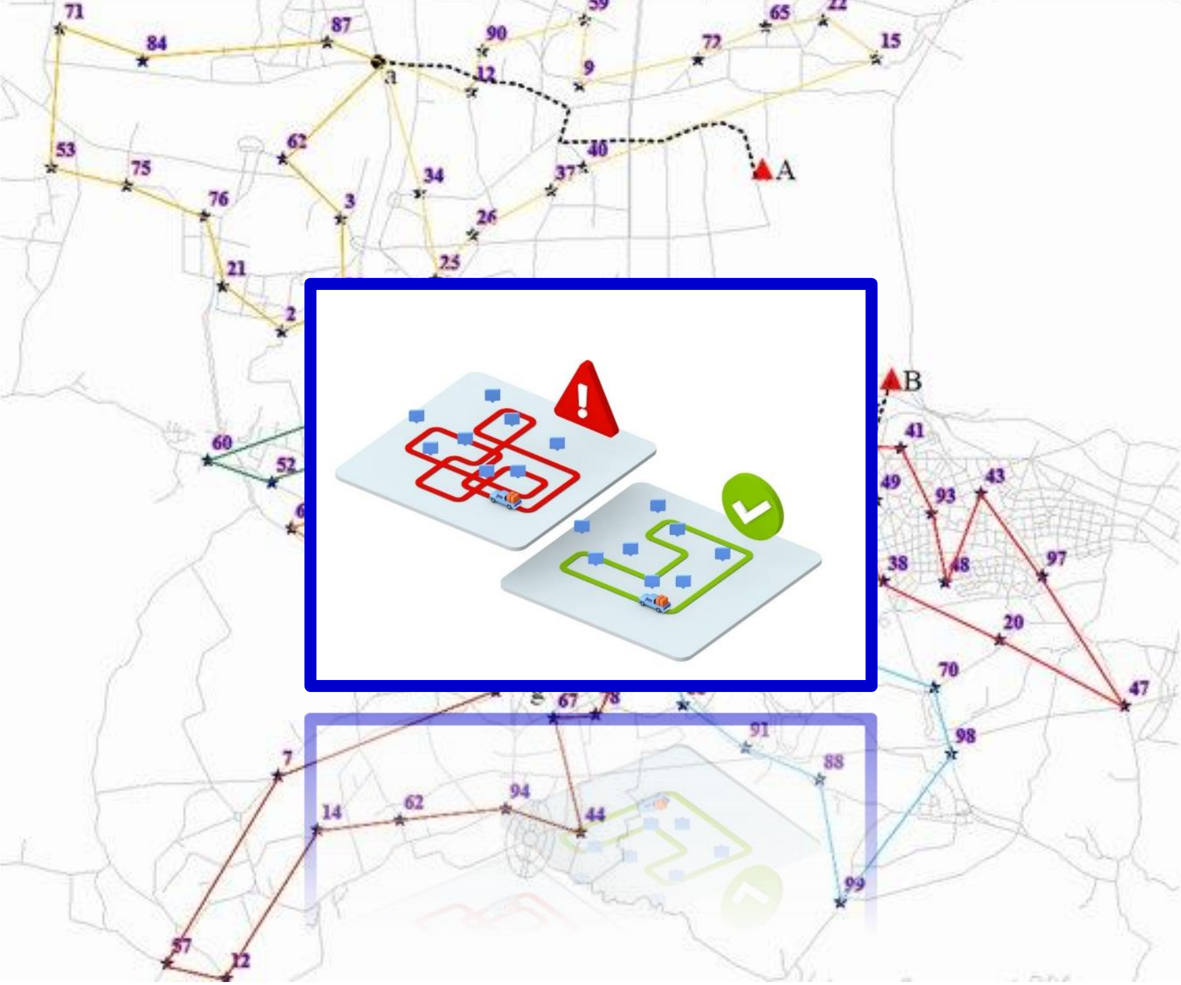


CTTC	TCAC	ACCG	CGGT	GTCCCT	CTCAA	AACG	CGTC	TCCTAA

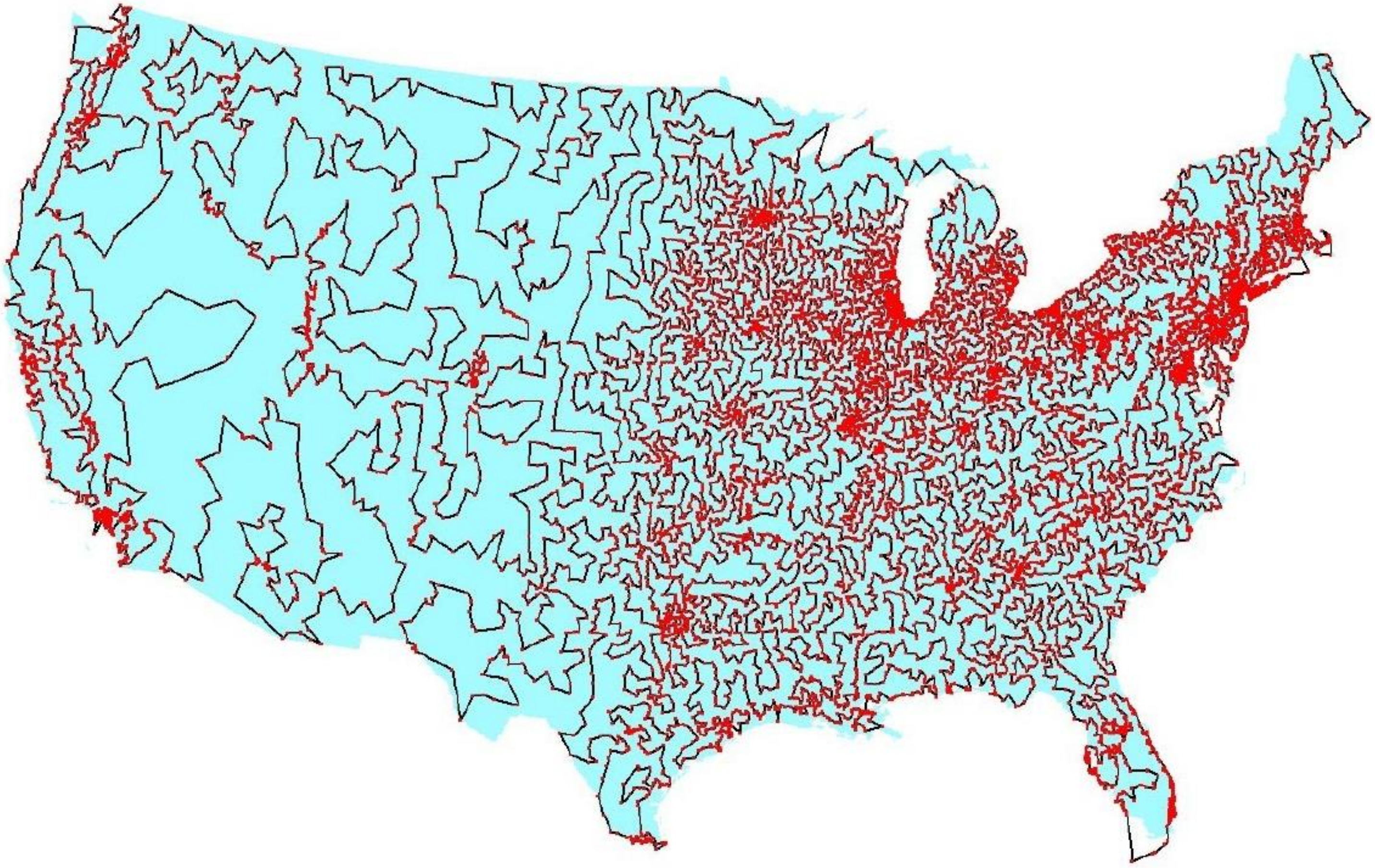
Problemi (un po' più seri) di un ingegnere

Instradamento di veicoli





Quali sono i giri di consegna ottimali?



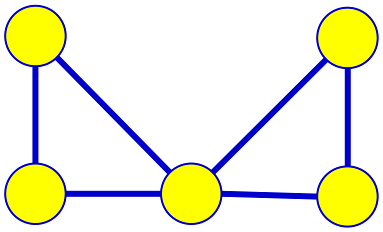
- Questo è il "miglior giro di consegna" che tocca le 13.509 città americane con più di 500 abitanti (calcolato nel 1998)

“Interstellar overdrive” (cit.)

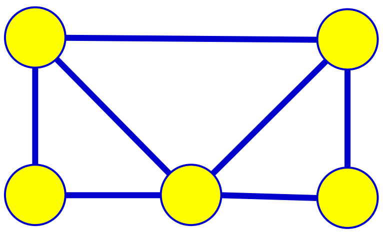
Grafi *hamiltoniani* e grafi *euleriani*

La classe dei grafi hamiltoniani non contiene né è contenuta nella classe dei grafi euleriani

Ci sono grafi euleriani che non sono hamiltoniani

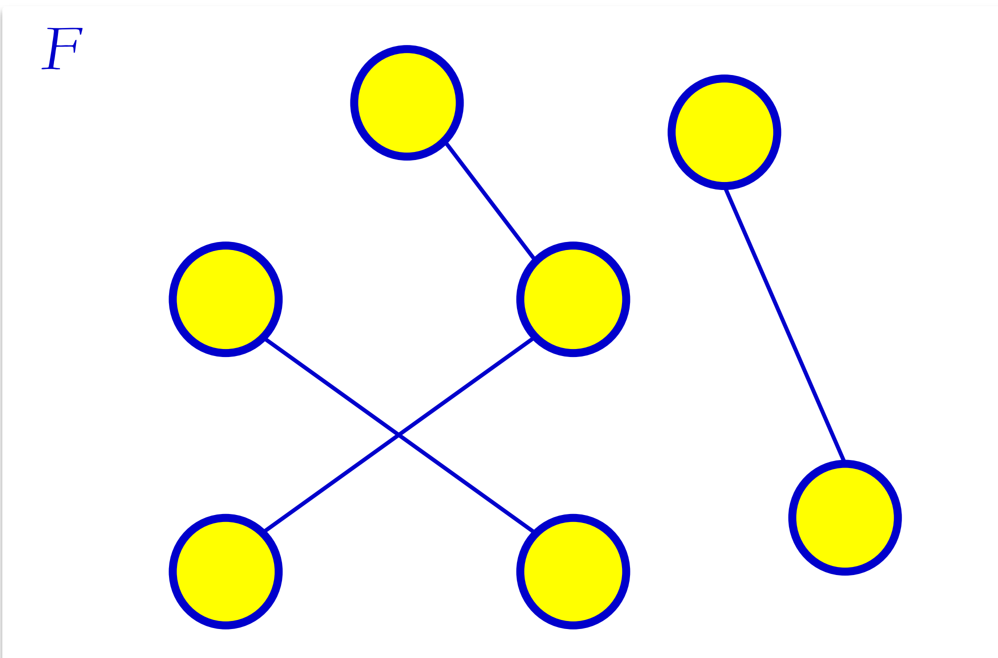


Ci sono grafi hamiltoniani che non sono euleriani



Foreste e alberi

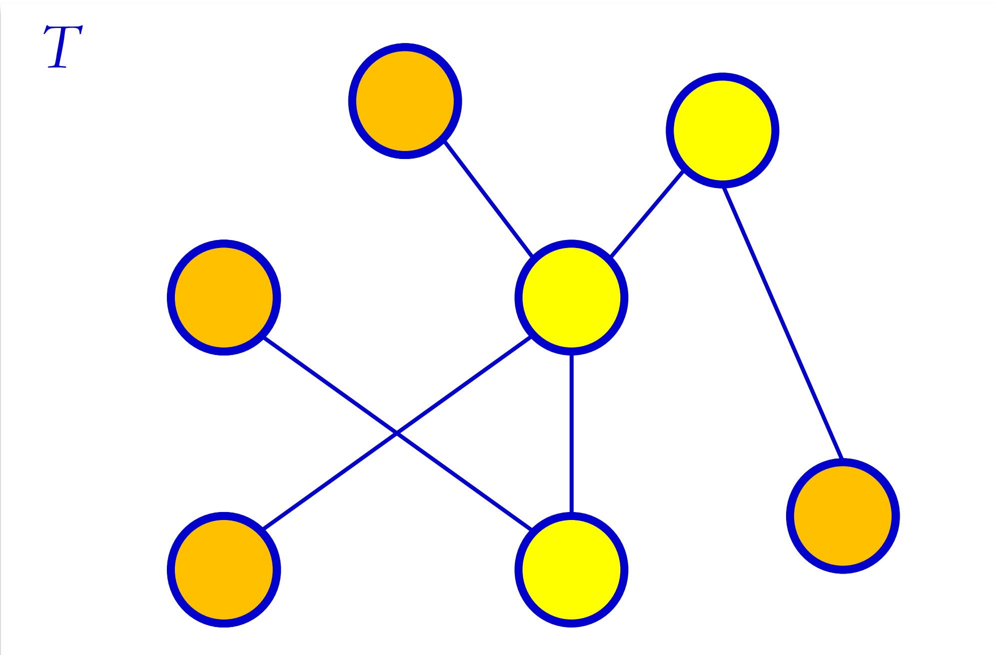
Una foresta F è un grafo non orientato aciclico



una foresta è un grafo non necessariamente connesso

Foreste e alberi

Una albero è una foresta con una unica componente connessa, quindi un albero è un grafo connesso e aciclico



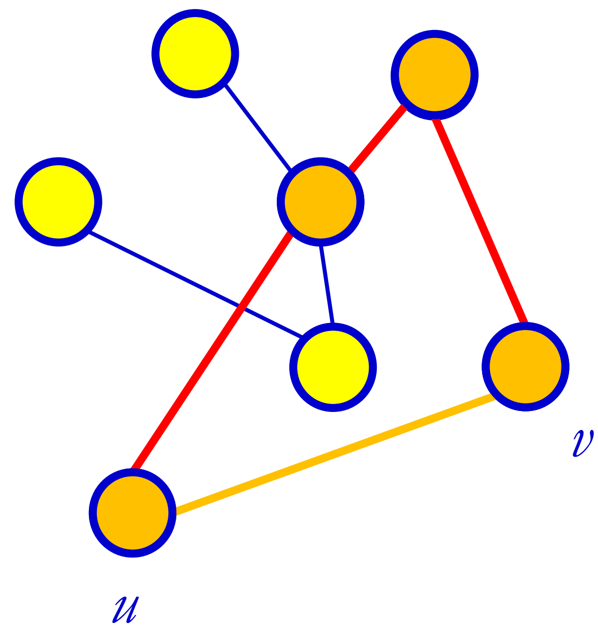
I nodi v di grado 1 sono chiamati foglie

Alberi: caratterizzazione e proprietà

[Caratterizzazione] Un grafo simmetrico $G = (V, E)$ è un albero **se e solo se** è connesso e ha $|V| - 1$ archi.

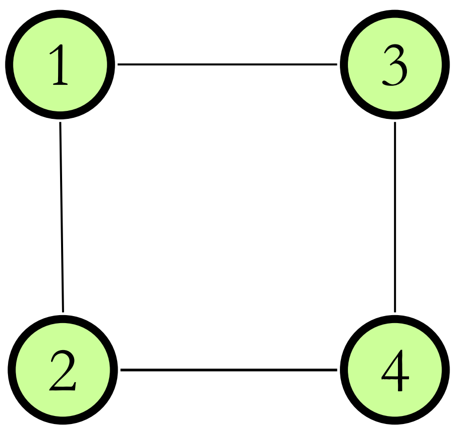
[Proprietà] sia $T = (V, E)$ un albero

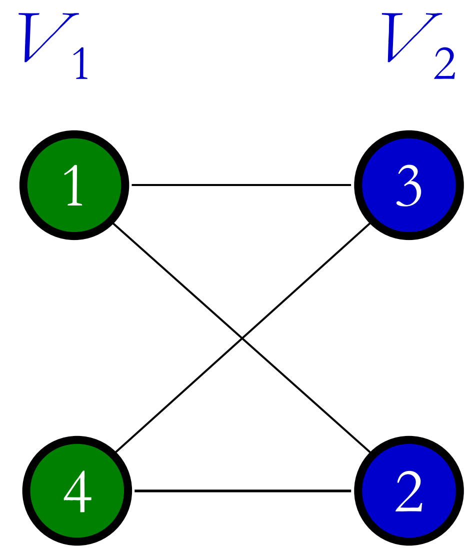
- T ha almeno una foglia.
- esiste un **unico** cammino da u a v, per ogni $u,v \in V$
- Se aggiungiamo un arco a E, il grafo risultante ha **esattamente** un ciclo.
- Se rimuovo una arco E, il grafo non è più connesso.



Grafo bipartito

Un grafo $G = (V, E)$ è bipartito se ogni arco ha un estremo in V_1 e l'altro in V_2 con $V_1 \cup V_2 = V$

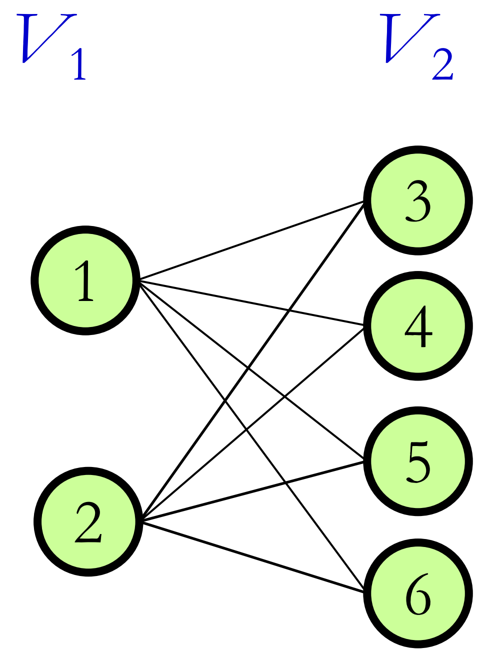




C_4 è un grafo *bipartito* … infatti

Grafo bipartito

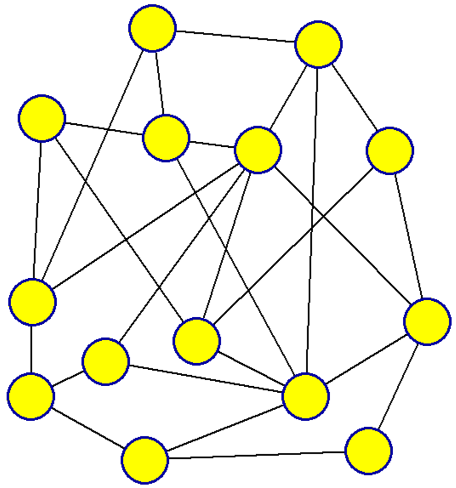
Un grafo bipartito $G = (V, E)$ è completo se ogni nodo in V_1 è adiacente a tutti i nodi di V_2 e viceversa.
Il grafo bipartito completo con $|V_1| = p$ e $|V_2| = q$ si indica con $K_{p,q}$

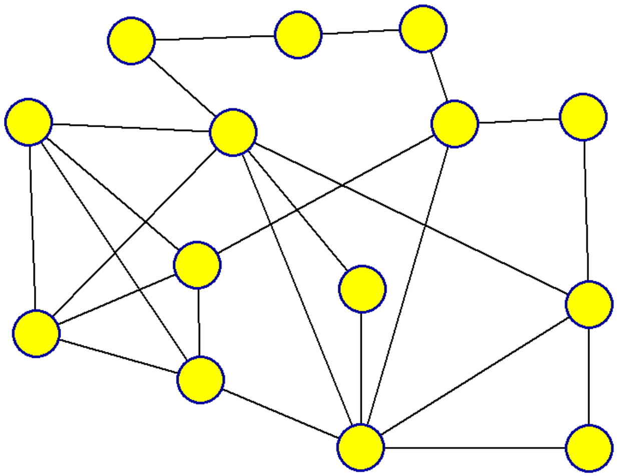


$K_{2,4}$

Caratterizzazione dei grafi bipartiti

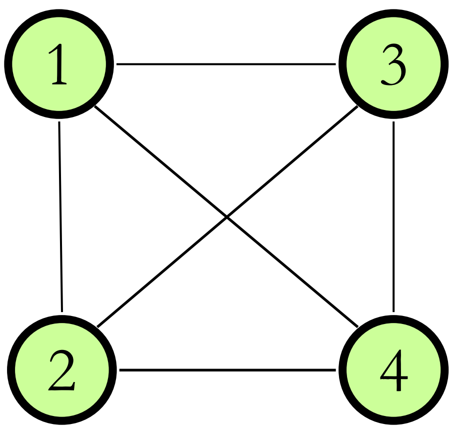
Quali di questi grafi è bipartito?



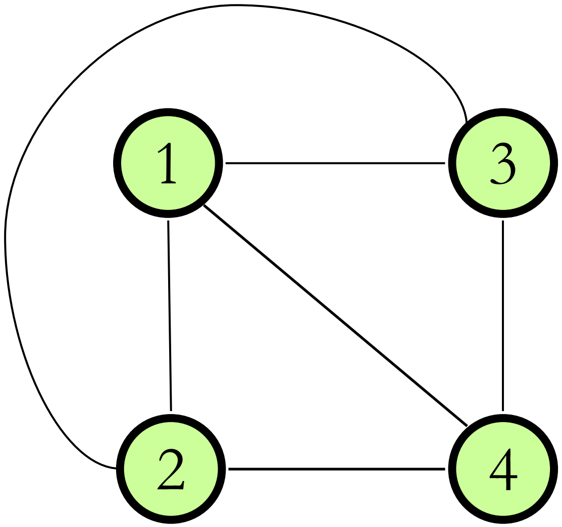


Grafo planare

Un grafo $G = (V, E)$ è planare se può essere disegnato nel piano in modo che nessuna coppia di archi si intersechi.



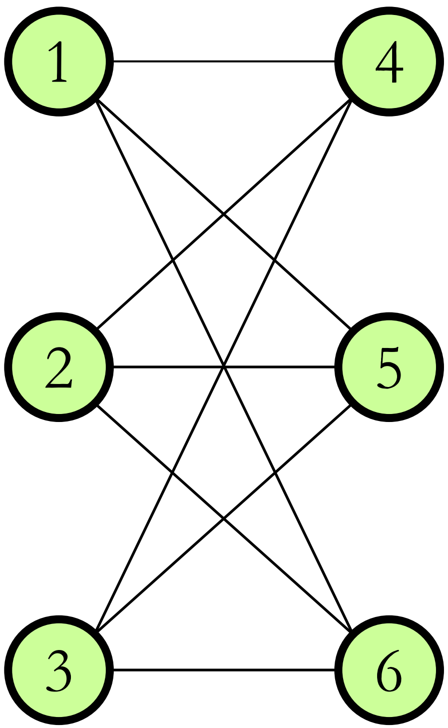
K_4 è un grafo *planare*



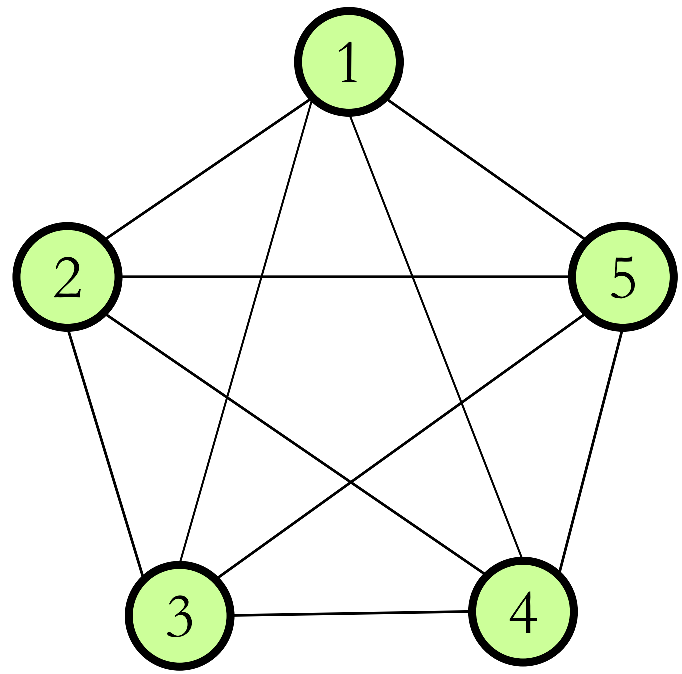
…infatti

Grafo planare

Un grafo $G = (V, E)$ è planare se può essere disegnato nel piano in modo che nessuna coppia di archi si intersechi.



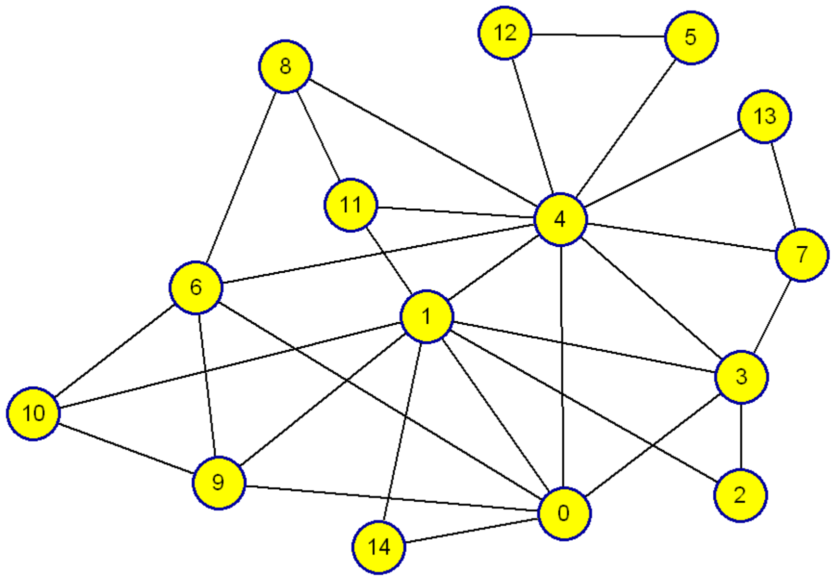
$K_{3,3}$ non è un grafo *planare*

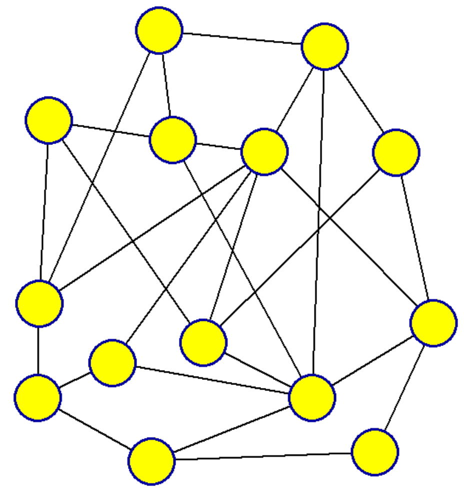


K_5 non è un grafo *planare*

Caratterizzazione dei grafi planari

Quali di questi grafi è planare?





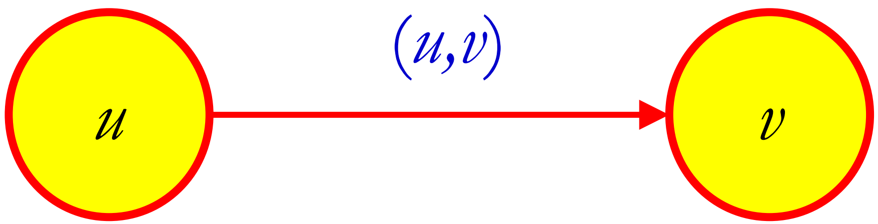
Sommario

- Introduzione
- Motivazioni e origini storiche
- Definizioni e proprietà di base
- Isomorfismi tra grafi
- Grafi di base
- Classi di grafi
- Grafi orientati
- Rappresentazioni
- Appendice

Grafi orientati (o diretti, o asimmetrici)

Un arco orientato (u, v) è una coppia <u>ordinata</u> di nodi

arco uscente da u e entrante in v



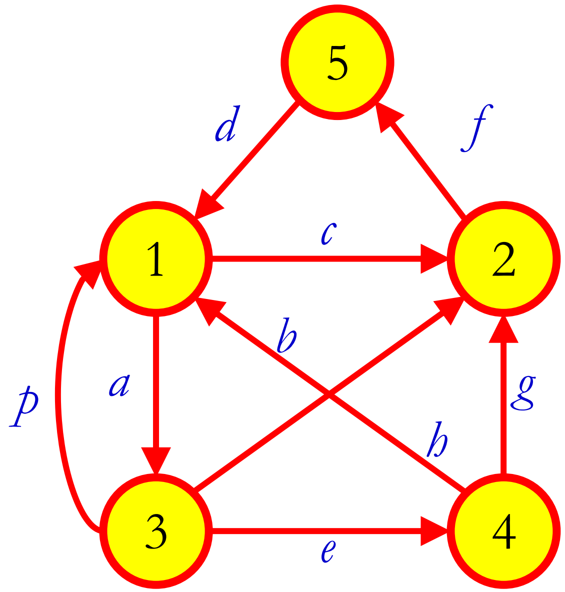
nodo di partenza
(o coda)

nodo di arrivo
(o testa)

$$(u,v) \neq (v, u)$$

Grafi orientati (o diretti o asimmetrici)

Un grafo $G = (V, E)$ è orientato se tutti gli archi sono orientati



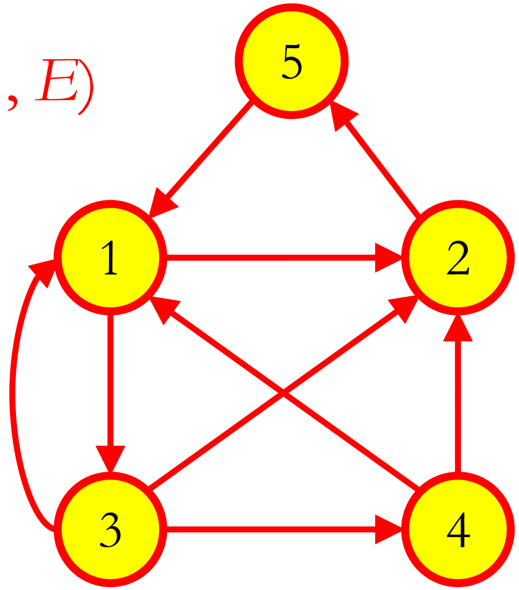
$$V = \{1, 2, 3, 4, 5\}$$

$$E = \{a, b, c, d, e, f, g, h, p\}$$

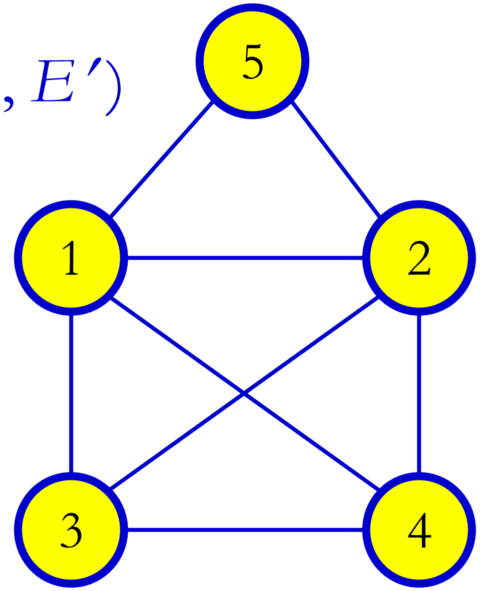
$a = (1, 3) \neq (3, 1) = p$

Grafi orientati e supporto simmetrico

$D = (V, E)$



$G = (V, E')$

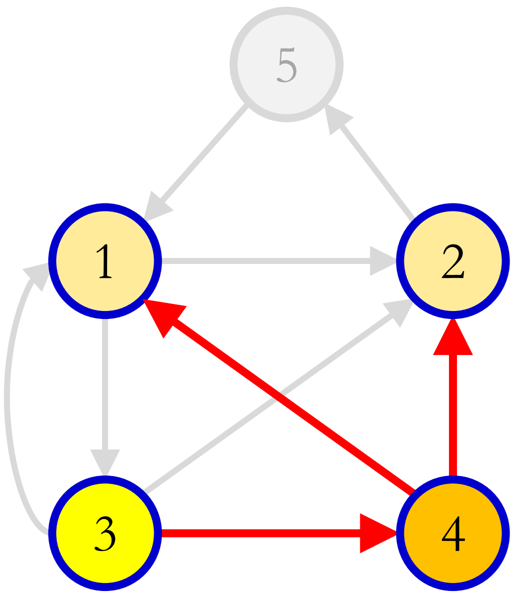


Il supporto simmetrico di un grafo orientato $D = (V, E)$ è il grafo simmetrico $G = (V, E')$ ottenuto ignorando l'orientamento degli archi di D e eliminando gli archi duplicati

Definizioni: intorno e stella di grafi orientati

$N(v) = I(v) \cup O(v)$

- $I(v)$ = nodi di arrivo in v
- $O(v)$ = nodi di partenza da v



$I(4) = \{3\}$

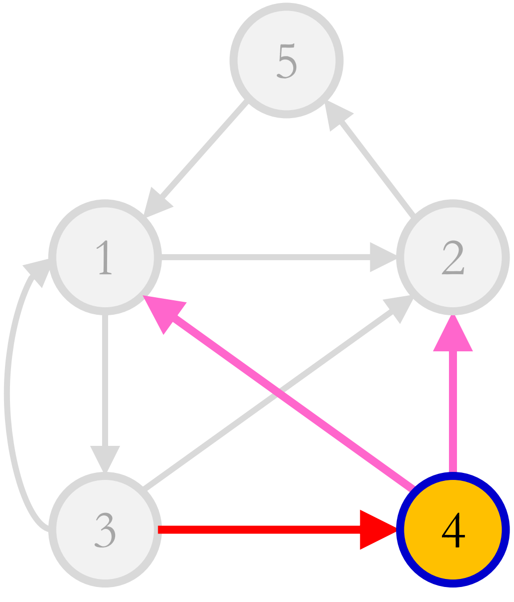
$O(4) = \{1, 2\}$

$$I(v) = \{u \in N \mid (u, v) \in E\}$$
$$O(v) = \{u \in N \mid (v, u) \in E\}$$

Definizioni: intorno e stella di grafi orientati

$\delta(v) = \delta^+(v) \cup \delta^-(v)$

- $\delta^+(v)$ = stella uscente da v
- $\delta^-(v)$ = stella entrante in v



$\delta^+(4) = \{(4, 1), (4, 2)\}$

$\delta^-(4) = \{(3, 4)\}$

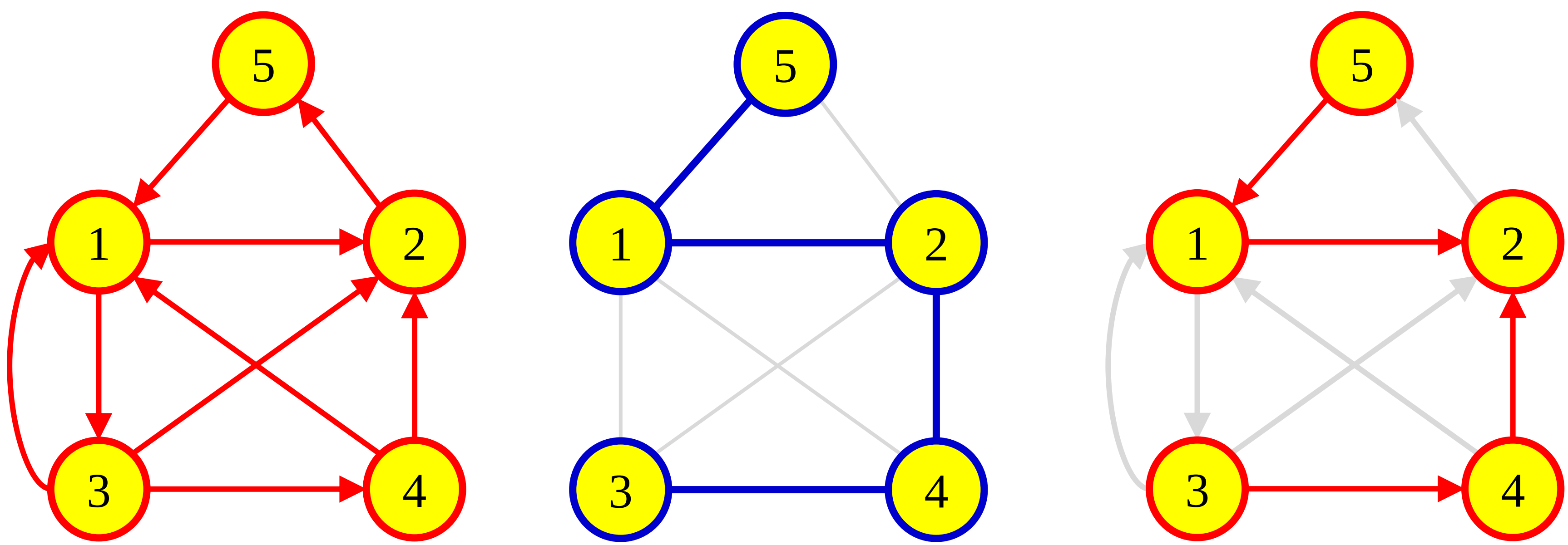
$$\delta^+(v) = \{u \in V \mid (v, u) \in E\}$$

$$\delta^-(v) = \{u \in V \mid (u, v) \in E\}$$

Grafi orientati: cammini e cicli

I cammini, percorsi e cicli di un grafo orientato $D = (V, E)$ sono tutti e soli quelli definiti sul supporto simmetrico di D.

$D = (V, E)$ supporto simm. di D



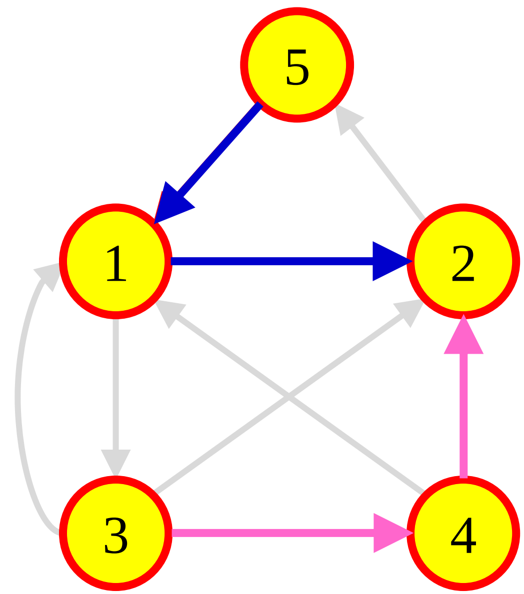
$P = [\{5,1\}, \{1,2\}, \{2,4\}, \{4, 3\}]$

Grafi orientati: cammini e cicli

I cammini, percorsi e cicli di un grafo orientato $D = (V, E)$ sono <u>tutti e soli</u> quelli definiti sul supporto simmetrico di D.

arco in avanti: orientato concordemente a P

arco all'indietro: orientato in direzione opposta a P

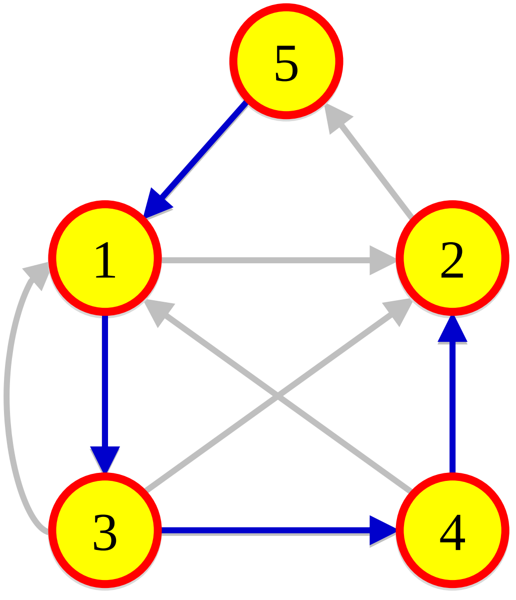


Gli archi diretti (5,1) e (1,2) sono archi in avanti di P
Gli archi diretti (4,2) e (3,4) sono archi all'indietro di P

$P = [\{5,1\}, \{1,2\}, \{2,4\}, \{4, 3\}]$

Grafi orientati: cammini e cicli

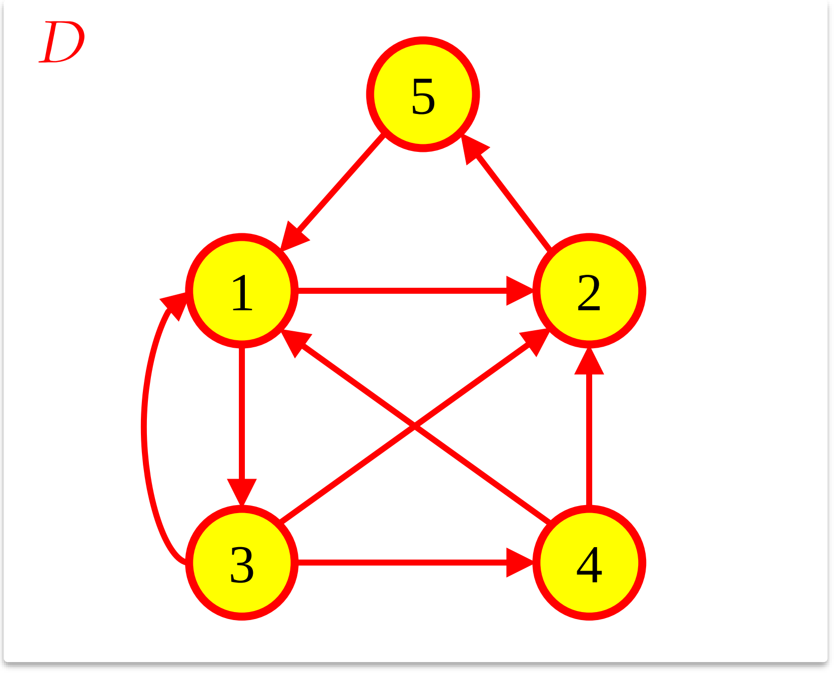
Un cammino, percorso o ciclo di un grafo orientato $D = (V, E)$ è orientato (o diretto) se formato esclusivamente da archi in avanti.



Il percorso $P = [\{5,1\}, \{1,3\}, \{3,4\}, \{4, 2\}]$ è diretto perché formato esclusivamente da archi diretti in avanti.

Forte connessione

Due nodi u, $v \in V$ di un grafo orientato D si dicono fortemente connessi se esiste in D un cammino orientato tra u e v e un cammino orientato tra v e u



In questo grafo tutte le coppie di nodi sono fortemente connesse

Sommario

- Introduzione
- Motivazioni e origini storiche
- Definizioni e proprietà di base
- Isomorfismi tra grafi
- Grafi di base
- Classi di grafi
- Grafi orientati
- Rappresentazioni
- Appendice

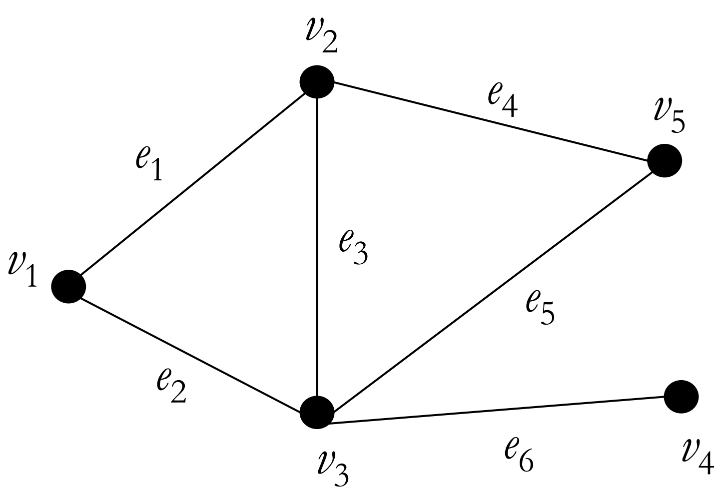
Rappresentazioni

- La rappresentazione grafica non è idonea per la manipolazione algebrica dei grafi né per il loro trattamento automatico mediante calcolatore. Esistono rappresentazioni più adeguate:
  - Matrice di adiacenza
  - Matrice di incidenza
  - Lista di adiacenza

Matrice di adiacenza nodi-nodi

Dato un grafo non orientato $G = (V, E)$, la matrice di adiacenza nodi-nodi è una matrice quadrata $A_G(n \times n)$, con $n = |V|$, tale che

$$a_{ij} = \begin{cases} 1 \text{ se } v_i \text{ e } v_j \text{ sono adiacenti, cioè se } \{v_i, v_j\} \in E \\ 0 \text{ altrimenti} \end{cases}$$

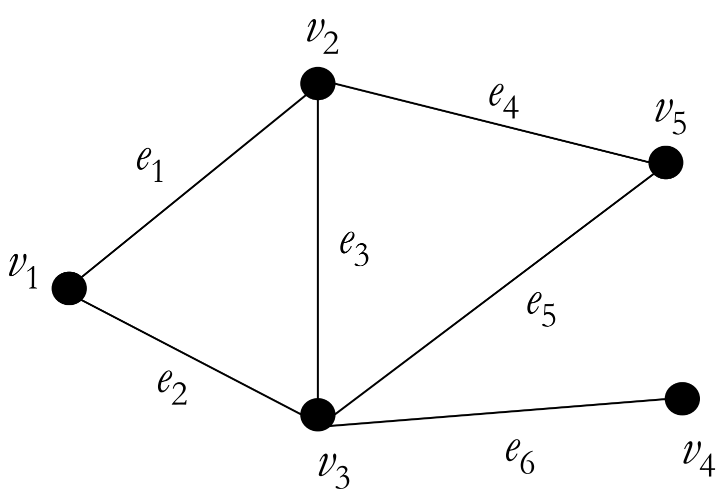


$$\begin{array}{cc} & \begin{array}{ccccc} \mathbf{v_1} & \mathbf{v_2} & \mathbf{v_3} & \mathbf{v_4} & \mathbf{v_5} \end{array} \\ A_G = & \begin{pmatrix} 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 \end{pmatrix} \begin{array}{c} \mathbf{v_1} \\ \mathbf{v_2} \\ \mathbf{v_3} \\ \mathbf{v_4} \\ \mathbf{v_5} \end{array} \end{array}$$

Matrice di adiacenza archi-archi

Dato un grafo non orientato $G = (V, E)$, la matrice di adiacenza archi-archi è una matrice quadrata $A_G(m \times m)$, con $m = |E|$, tale che

$$a_{ij} = \begin{cases} 1 \text{ se } e_i \text{ e } e_j \text{ sono adiacenti} \\ 0 \text{ altrimenti} \end{cases}$$



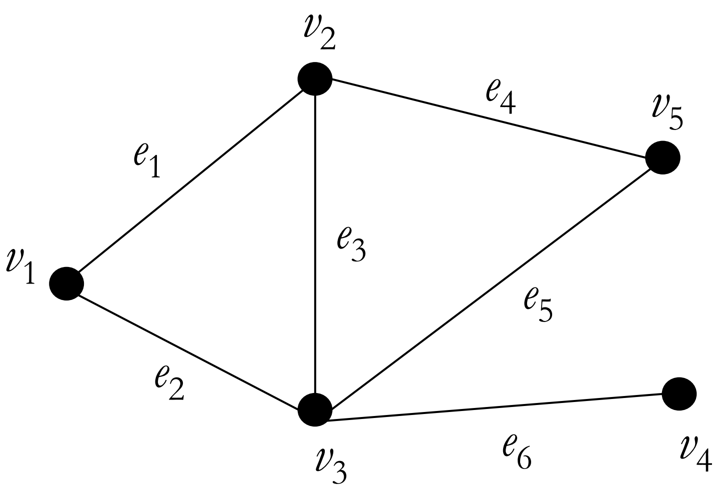
$$A_G = \begin{array}{c} \begin{array}{cccccc} \boldsymbol{e_1} & \boldsymbol{e_2} & \boldsymbol{e_3} & \boldsymbol{e_4} & \boldsymbol{e_5} & \boldsymbol{e_6} \end{array} \\ \begin{pmatrix} 1 & 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 1 & 1 \end{pmatrix} \end{array} \begin{array}{c} \\ \boldsymbol{e_1} \\ \boldsymbol{e_2} \\ \boldsymbol{e_3} \\ \boldsymbol{e_4} \\ \boldsymbol{e_5} \\ \boldsymbol{e_6} \end{array}$$

Matrice di incidenza nodi-archi

Dato un grafo non orientato $G = (V, E)$, la matrice di incidenza nodi-archi

è una matrice $E_G(n \times m)$, con $n = |V|$ e con $m = |E|$, tale che

$$a_{ij} = \begin{cases} 1 \text{ se } e_j \text{ è incidente su } v_i \\ 0 \text{ altrimenti} \end{cases}$$

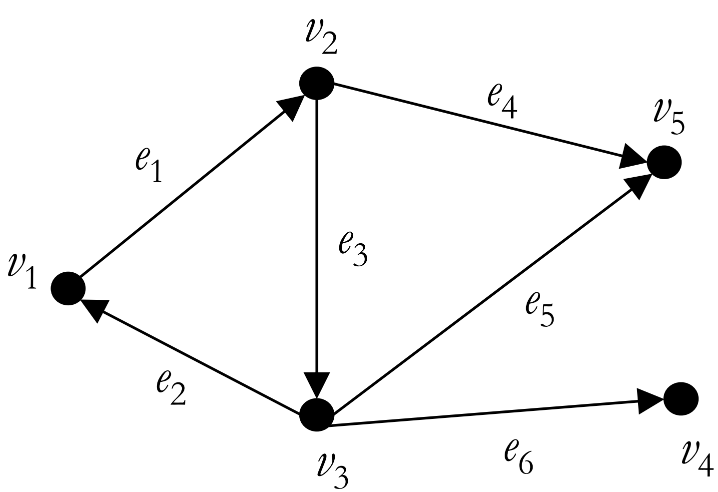


$$E_G = \begin{array}{c c} \begin{array}{cccccc} \boldsymbol{e_1} & \boldsymbol{e_2} & \boldsymbol{e_3} & \boldsymbol{e_4} & \boldsymbol{e_5} & \boldsymbol{e_6} \end{array} & \\ \begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 \end{pmatrix} & \begin{array}{c} \boldsymbol{v_1} \\ \boldsymbol{v_2} \\ \boldsymbol{v_3} \\ \boldsymbol{v_4} \\ \boldsymbol{v_5} \end{array} \end{array}$$

Matrice di incidenza nodi-archi: grafi orientati

Dato un grafo orientato $G = (V, E)$, la matrice di incidenza nodi-archi è una matrice $E_G(n \times m)$, con $n = |V|$ e con $m = |E|$, tale che

$$a_{ij} = \begin{cases} -1 \text{ se } v_i \text{ è la testa di } e_j \\ 1 \text{ se } v_i \text{ è la coda di } e_j \\ 0 \text{ altrimenti} \end{cases}$$



$$E_G = \begin{array}{c} \\ \\ \end{array} \begin{array}{cccccc} \boldsymbol{e_1} & \boldsymbol{e_2} & \boldsymbol{e_3} & \boldsymbol{e_4} & \boldsymbol{e_5} & \boldsymbol{e_6} \end{array}$$

$$E_G = \begin{pmatrix} 1 & -1 & 0 & 0 & 0 & 0 \\ -1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & -1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & -1 & -1 & 0 \end{pmatrix} \begin{array}{c} \boldsymbol{v_1} \\ \boldsymbol{v_2} \\ \boldsymbol{v_3} \\ \boldsymbol{v_4} \\ \boldsymbol{v_5} \end{array}$$

Lista di adiacenza

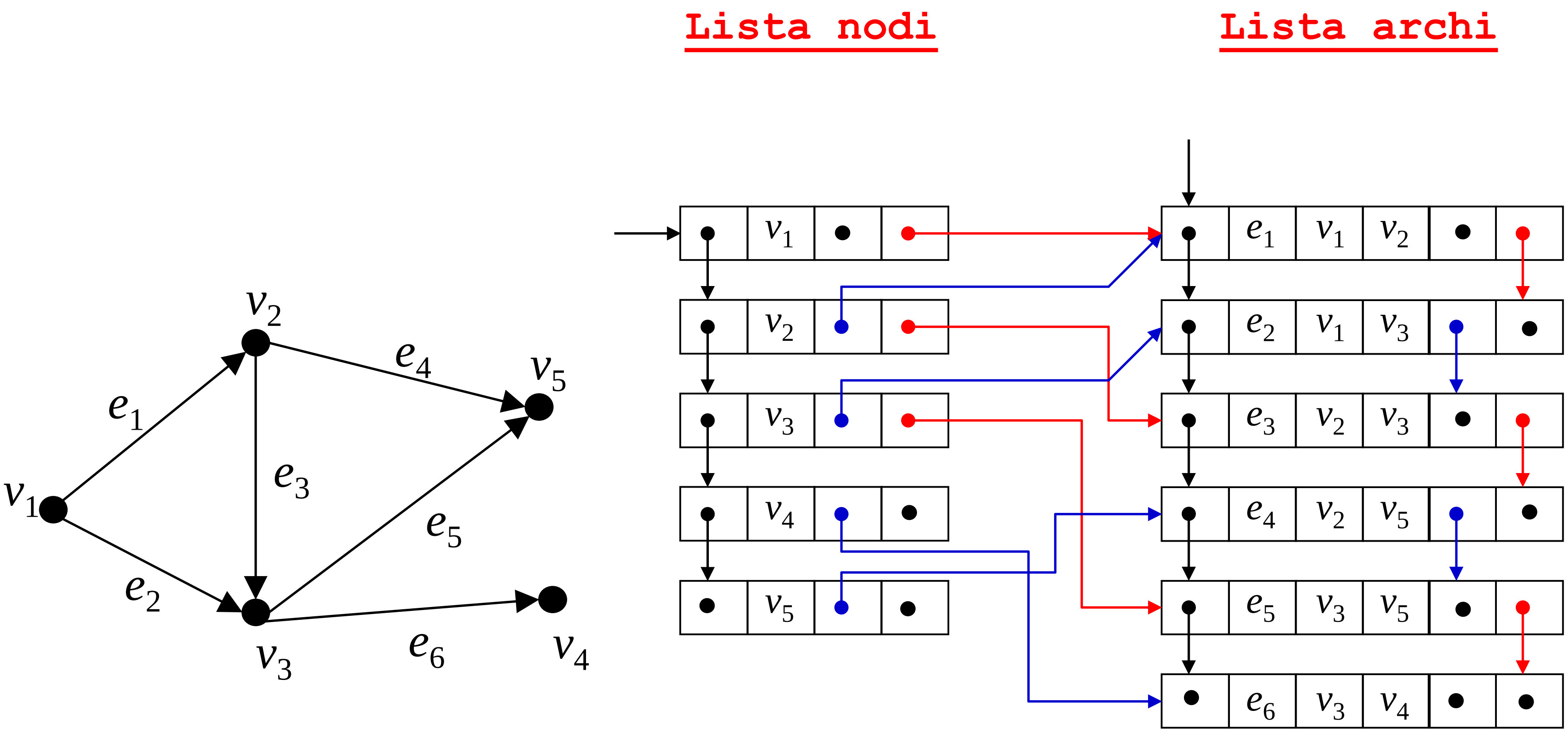
- Solo $2m$ elementi sono non nulli su un totale di n^2 elementi di una matrice di adiacenza nodi-nodi e $n \cdot m$ elementi di una matrice di incidenza.
- La rappresentazione con matrici è poco efficiente dal punto di vista dell'occupazione di memoria

- In alternativa si possono usare liste di adiacenza per stelle uscenti e/o entranti:

```
typedef struct Nodo{
    Nodo      *succ;
    int       nome;
    Arco      *primo_stella_in;
    Arco      *primo_stella_out;
};
```

```
typedef struct Arco{
    Arco      *succ;
    int       nome;
    Nodo      *coda;
    Nodo      *testa;
    Arco      *succ_stella_in;
    Arco      *succ_stella_out;
};
```

Lista di adiacenza, esempio



Esercizi

Scrivere un programma in linguaggio C/C++ che, dato in input un grafo *G*, determini

- se i nodi dati *u*, *v* siano connessi
- tutti i nodi connessi a un dato nodo *u*
- il numero di componenti connesse
- Il grado di un nodo dato *u*
- un cammino, se esiste, tra i nodi dati *u*, *v*

(Utilizzare le liste di adiacenza per rappresentare il grafo)

Letture ricreative e contenuti multimediali

1. P. M. Higgins,
   La matematica dei social network. Una introduzione alla teoria dei grafi
   Dedalo, 2011

2. Brian Eno
   Thursday Afternoon
   1985

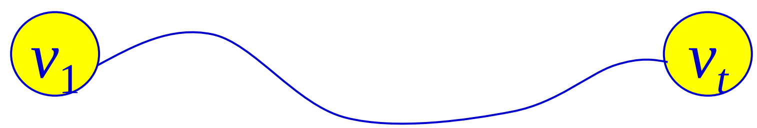
Appendice

grafi hamiltoniani: una condizione sufficiente

[Ore theorem (1960)] A graph $G = (V, E)$ with at least 3 vertices and $d(u) + d(v) \geq n$ for all pairs u,v of non-adjacent vertices is hamiltonian

Let P be a simple path of G of *maximum* length, with say t vertices.

If needed, renumber the vertices so that $P = [v_1, \ldots, v_t]$



Observations

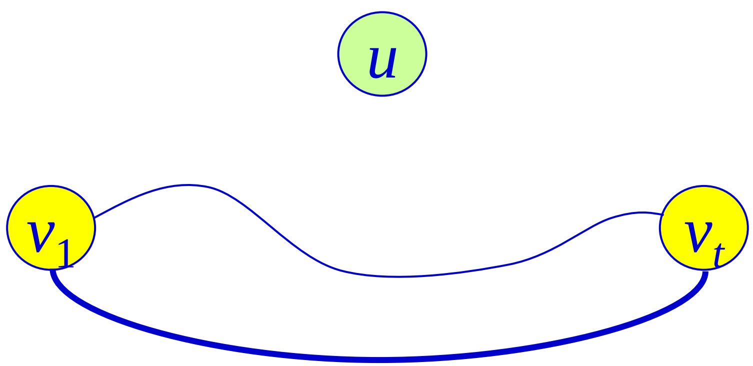
1. All the neighbours of v_1 and v_t are vertices of P, otherwise a longer path exists.
2. If u,v are non-adjacent vertices then their neighbourhoods have at least a vertex in common, otherwise $d(u) + d(v) \leq n - 2$

[Ore theorem (1960)] A graph $G = (V, E)$ with at least 3 vertices and $d(u) + d(v) \geq n$ for all pairs u,v of non-adjacent vertices is hamiltonian

Case a. $\{v_1, v_t\} \in E$

$\Rightarrow C = [v_1, \ldots, v_t, v_1]$ is an hamiltonian cycle, i.e., $t = n$

Indeed, if $t < n$ then there exists a vertex $u \notin P$ which, by obs 1., cannot be adjacent to v_1

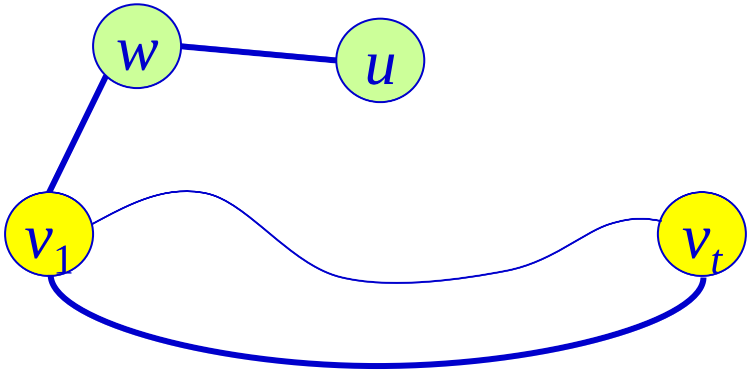


[Ore theorem (1960)] A graph $G = (V, E)$ with at least 3 vertices and $d(u) + d(v) \geq n$ for all pairs u,v of non-adjacent vertices is hamiltonian

Case a. $\{v_1, v_t\} \in E$

$\Rightarrow C = [v_1, \ldots, v_t, v_1]$ is an hamiltonian cycle, i.e., $t = n$

Since u and v_1 are non-adjacent, by obs 2., there must be a vertex w adjacent to both u and v_1.



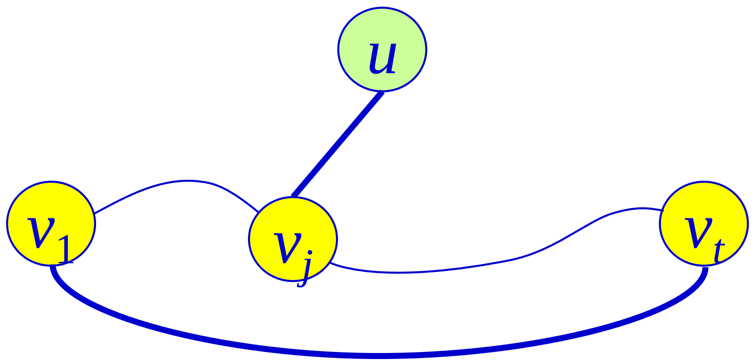
[Ore theorem (1960)] A graph $G = (V, E)$ with at least 3 vertices and $d(u) + d(v) \geq n$ for all pairs u,v of non-adjacent vertices is hamiltonian

Case a. $\{v_1, v_t\} \in E$

$\Rightarrow C = [v_1, \ldots, v_t, v_1]$ is an hamiltonian cycle, i.e., $t = n$

Since w is adjacent to v_1, by obs. 1, it must be part of the path P, say v_j

But now, the path $[u, v_j, v_{j+1}, \ldots, v_t, v_1, \ldots, v_{j-1}]$ is longer than P.

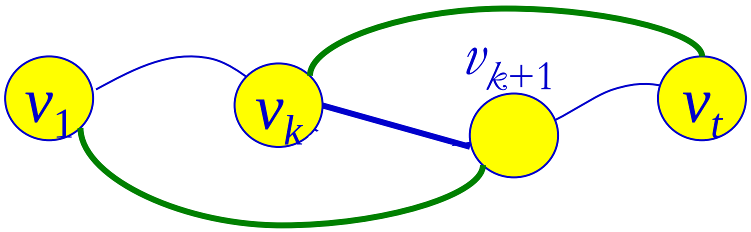


[Ore theorem (1960)] A graph $G = (V, E)$ with at least 3 vertices and $d(u) + d(v) \geq n$ for all pairs u,v of non-adjacent vertices is hamiltonian

Case b. $\{v_1, v_t\} \notin E$

By obs 1. $d(v_1) \leq t - 2$.

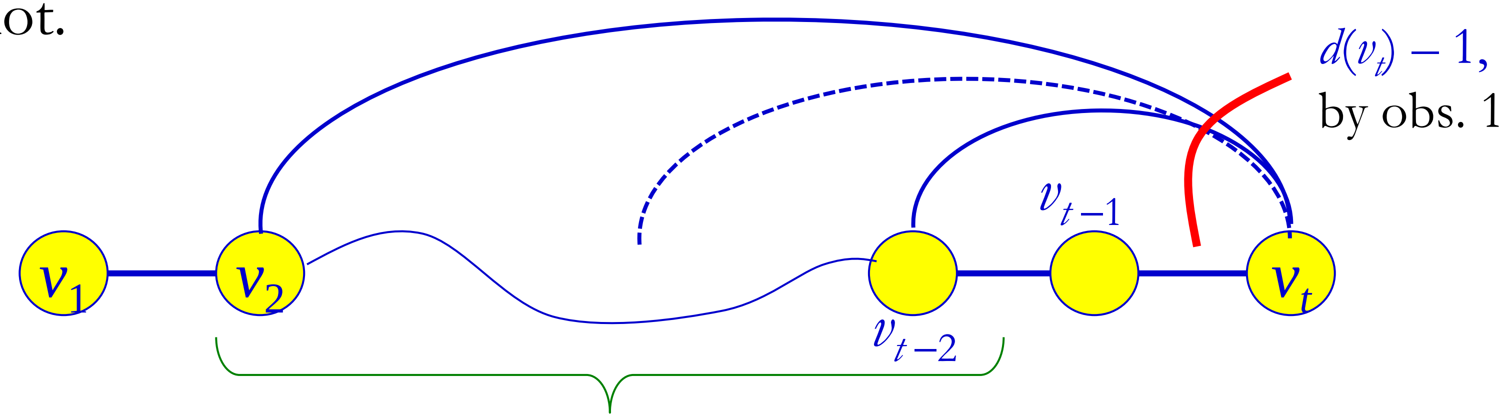
Moreover, we can claim that there is a vertex v_k of P such that v_k is adjacent to v_t and v_{k+1} is adjacent to v_1



[Ore theorem (1960)] A graph $G = (V, E)$ with at least 3 vertices and $d(u) + d(v) \geq n$ for all pairs u,v of non-adjacent vertices is hamiltonian

Case b. $\{v_1, v_t\} \notin E$

Suppose not.



If for any vertex v_k adjacent to v_t the vertex v_{k+1} is non-adjacent to v_1 follows that:

$$d(v_1) \leq t - 2 - (d(v_t) - 1) = t - 1 - d(v_t)$$

[Ore theorem (1960)] A graph $G = (V, E)$ with at least 3 vertices and $d(u) + d(v) \geq n$ for all pairs u,v of non-adjacent vertices is hamiltonian

Case b. $\{v_1, v_t\} \notin E$

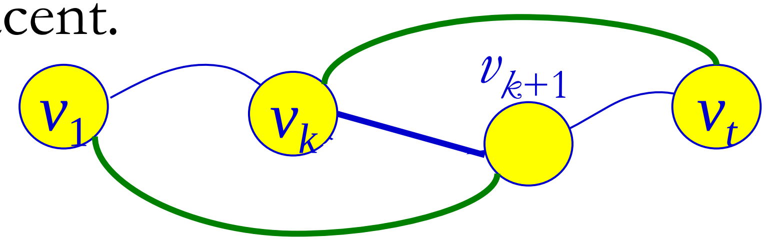
$$d(v_1) \leq t - 1 - d(v_t)$$

$$d(v_t) + d(v_1) \leq t - 1 - d(v_t) + d(v_t)$$

$$d(v_t) + d(v_1) \leq t - 1$$

but v_t and v_1 are non-adjacent and by hp. must be $d(v_t) + d(v_1) \geq n$

Finally, observe that $v_1 - v_k - v_t\, v_{t-1}\, v_{t-2} - v_{k+1}$ is a simple path of maximum length whose endpoints are adjacent.



Caratterizzazione degli alberi

A graph $G = (V, E)$ is a tree
<u>if and only if</u> is connected and has $|V| - 1$ edges.

G tree $\Rightarrow$ G connected and with $|V| - 1$ edges

- By definition G is connected.
- If G is a tree then it has at least a leaf v. If we remove v, the graph G' with vertices in $V \setminus \{v\}$ still is a tree (because we neither disconnect G nor introduce any cycle).
- Again, G' has at least a leaf u and therefore the process can be repeated until a graph with a single vertex is obtained.
- At this point, $|V| - 1$ edges has been removed.

A graph $G = (V, E)$ is a tree
<u>if and only if</u> is connected and has $|V| - 1$ edges.

G connected and with $|V| - 1$ edges $\Rightarrow$ G tree

We have to show that G is acyclic.

Observe that

1. C_n has n vertices and n edges.
2. G has at least a leaf v. Indeed, if not (i.e., if $d(u) \geq 2 \quad \forall u$)

$$\sum_{u \in V} d(u) = 2|E| \geq 2|V| \quad \text{but } |E| < |V| \text{ by hypothesis.}$$

Now we can apply the previous elimination process of leaves; since each time a single vertex and a single edge are removed, we cannot never obtain a subgraph with $|V| = |E|$

Tree properties

Every tree $T = (V, E)$ with $n \geq 2$ vertices has at least two leaves

- Pick a vertex $v \in V$
- Since T is connected and with $n \geq 2$, $\exists\ u \in V$ s.t. $\{v, u\} \in E$.
- If $d(u) = 1$ a leaf has been found, otherwise the same argument can be repeated.
- Since T is acyclic, we reach a leaf w after a finite number of steps.
- Apply the same argument starting from w.