1 Circuiti

Definizione 1 (Trail Aperto):
Un *trail aperto* su G è un *cammino* in cui gli spigoli non si ripetono

Definizione 2 (Trail Chiuso/Circuito):
Un trail su G i cui estremi coincidono è detto *trial chiuso* o *circuito*.

Definizione 3 (Trail Euleriano/Circuito Euleriano):
Un *Trail Euleriano* (tour di Eulero), detto anche **Circuito Euleriano**, è un trail chiuso che contiene ogni arco di G **esattamente una volta**. Un grafo è chiamato *Euleriano* se contiene un tour di Eulero.

Definizione 4 (Path):
Un *path* su G è un trail in cui **i vertici non si ripetono** con la sola eccezione, possibilmente, dei vertici estremi.

Definizione 5 (Ciclo):
Un path i cui estremi coincidono è detto *ciclo*.

Definizione 6 (Ciclo Hamiltoniano):
Un ciclo che passa per tutti i vertici del grafo è detto *Hamiltoniano*.

Definizione 7 (Condizione di esistenza di un *trail Euleriano aperto*):
Sia $G(V, E)$ un grafo connesso. G ammette un trail Euleriano *aperto* (cioè un trail aperto che contiene ogni arco di G esattamente una volta) se e solo se i vertici di grado dispari di G sono esattamente due.

Definizione 8 (Condizione di esistenza di un *trail Euleriano chiuso*):
Sia $G(V, E)$ un grafo connesso. G ammette un trail Euleriano *chiuso* se ogni vertice di G ha grado *pari* oppure se l'insieme degli spigoli di G può essere partizionato in circuiti disgiunti sugli archi,

2 Grafi

Definizione 9 (Handshaking):
Per un qualunque grafo $G(V, E)$ vale che:

$$\sum_{v \in V} deg(v) = 2|E| \tag{1}$$

Definizione 10 (Grafo Complemento):
Dato un qualunque grafo $G(V, E)$, il *grafo complemento* di G è $\overline{G} = (V, \binom{V}{2} \setminus E)$

Definizione 11 (Grafo completo):
Un grafo $G(V, E)$ si definisce *completo* se ogni vertice $v \in V$ è adiacente a tutti gli $n-1$ vertici restanti. Se un grafo completo ha n vertici esso viene indicato con K_n. Ogni grafo completo K_n ha un numero di spigoli pari a $C(n, 2) = \binom{n}{2}$.

Definizione 12 (Grafo bipartito):
Un grafo $G(V, E)$ si definisce *bipartito* se e soltanto se **non** contiene *cicli dispari*; in generale tutti i grafi che contengano *cicli pari* sono bipartiti. Il *numero cromatico* nei grafi bipartiti è sempre pari a **due**, cioè $\chi(G) = 2$.

Definizione 13 (Grafo a intervallo):
Un grafo $G(V, E)$ si definisce a *intervallo* se dato un insieme di intervalli $I = \{I_1, I_2, \ldots, I_n\}$ risulta che:

- $V = \{I_1, I_2, \ldots, I_n\} = I$ cioè ha un solo vertice per ciascun intervallo dell'insieme I;
- $\{I_a, I_b\} \in E$ se e soltanto se $I_a \cap I_b \neq \emptyset$: in altri termini G possiede uno spigolo per ogni coppia di vertici corrispondenti agli intervalli che si intersecano.

Gli unici cicli presenti in un grafo a intervallo sono di *lunghezza 3*.

Definizione 14 (Grafo planare):
Un grafo $G(V, E)$ si definisce *planare* se ha un disegno senza intersezioni. Nei grafi planari il numero cromatico è sempre minore o uguale a 4, cioè $\chi(G) \leq 4$.

Definizione 15 (Connessione):
Sia $G(V, E)$ un grafo. Due vertici u e v sono *connessi* se esiste un path di G con estremi u e v.

Definizione 16 (Grafo connesso):
Un grafo $G(V, E)$ è detto *connesso* se per ogni coppia di vertici distinti $u, v \in V$, u e v sono connessi.

Definizione 17 (Condizione necessaria di aciclicità):
Condizione necessaria purché un grafo sia **aciclico** è che il numero degli spigoli sia al più pari al numero dei vertici meno uno. Formalmente:

$$|E(G)| \leqslant |V(G)| - 1 \tag{2}$$

Definizione 18 (Condizione necessaria di connessione):
Condizione necessaria purché un grafo sia **connesso** è che il numero degli spigoli sia almeno pari al numero dei vertici meno uno. Formalmente:

$$|E(G)| \geqslant |V(G)| - 1 \tag{3}$$

Definizione 19 (Numero componenti connesse nei grafi complemento):
Sia $G(V, E)$ un grafo connesso e sia $v \in V$ un suo vertice. Il grafo $\overline{G}$ ha al più $deg(v)$ componenti connesse.

Definizione 20 (Numero vertici nei grafi connessi):
Il numero di vertici di grado dispari in un grafo connesso è *sempre pari.*

Definizione 21:
Sia $G(V, E)$ un grafo non orientato con n vertici e $n - k$ spigoli. Allora G avrà **almeno** k componenti connesse.

Definizione 22:
Sia G un grafo con n vertici. Allora **due** qualunque delle seguenti affermazioni implicano la **terza**:

1. Il grafo G ha **esattamente** k componenti connesse.
2. Il grafo G ha $n - k$ spigoli.
3. Il grafo G è aciclico.

Definizione 23:
Sia $G(V, E)$ un grafo e sia $e \in E$ un suo spigolo. Valgono:

2. Se G è *connesso*, allora il grafo $\overline{G}$ ha al più due componenti connesse.

3. Se G è *aciclico*, allora il grafo $\overline{G}$ è ancora aciclico e il suo numero di componenti connesse è pari al numero di componenti connesse di G più 1.

Definizione 24:
Si definisce **albero** un grafo aciclico e connesso. Inoltre gli alberi sono *particolari* grafi bipartiti (*più in generale tutti i grafi non orientati aciclici sono bipartiti*). Se l'albero in questione ha n vertici, allora esso ha esattamente $n-1$ spigoli.

Definizione 25:
Per ogni coppia di vertici di un albero, esiste **uno e un solo** cammino con estremi i due vertici.

Definizione 26 (Prüfer code):
Il *prüfer code* è una sequenza **univoca** associata ad un albero etichettato. La lunghezza del *prüfer code* associato ad un albero di n vertici è pari a $n-2$.

Procedura 1 (Calcolo Prüfer code):
Allo step i-esimo si deve rimuovere la foglia con il valore dell'etichetta più basso (**escludendo la radice!**) e impostare l'i-esimo elemento del prüfer code uguale all'etichetta del *padre* della foglia eliminata. L'algoritmo si arresta quando rimangono solo 2 vertici.

Definizione 27 (Matching):
Dato un grafo non orientato $G(V, E)$, un matching $M \subseteq E$ è un insieme di spigoli che **non** hanno estremi in comune.

Teorema 1 (Teorema di Cayley):
I diversi alberi con insieme dei vertici $V = \{0, 1, 2..., n-2, n-1\}$ sono n^{n-2}.

Teorema 2 (Teorema di Hall):
Un grafo bipartito $G(V_1 \cup V_2, E)$ ammette un matching completo di V_1 in V_2 se e solo se $\forall A \subseteq V_1$ risulta che $|A| \leq |R(A)|$ dove $|R(A)| \subseteq V_2$ rappresentano i vertici adiacenti ai vertici appartenenti ad A.

Procedura 2 (Verificare se un grafo è bipartito o meno):
Se si individua un *ciclo dispari* il grafo **non** è bipartito.
Un altro meccanismo molto semplice consiste **nel valutare la distanza** di ogni vertice da un vertice qualsiasi. Sia $G(V, E)$ un grafo e sia v il vertice da cui si intende valutare la distanza da tutti gli altri $n-1$ vertici. Sia L_i l'insieme contenente tutti i vertici a distanza i da v. *Se non ci sono vertici di una stessa classe L_i che risultano **adiacenti** il grafo è bipartito*

3 Colorazione

Definizione 28:
Dato un grafo $G(V, E)$ il minimo k per cui il grafo è k-colorabile, ovvero il numero minimo di colori utilizzati in un coloring ammissibile di G, si indica con $\chi(G)$ ed è chiamato numero cromatico di G.

Procedura 3 (Algoritmo Greedy):
Sia G un grafo e supponiamo che l'insieme dei disponibili sia $1, 2, 3 \ldots, n$. L'algoritmo Greedy è così definito:

1. Scegliere un qualunque ordinamento $w_1, w_2 \ldots w_n$ dei vertici di G.

2. Per ogni i colora il vertice w_i con un colore $f(w_i)$ pari al più piccolo colore non ancora assegnato a vertici a cui w_i è adiacente.

La qualità della soluzione individuata dall'algoritmo Greedy **dipende tuttavia dall'ordinamento** dei vertici da colorare eventualmente scelto. In particolare valgono le seguenti considerazioni:

1. Per un qualunque grafo **esiste un ordinamento** per cui l'algoritmo Greedy restituisce una colorazione **ottima**, cioè colora G con il minimo numero di colori possibile.

2. L'algoritmo Greedy **non** restituisce tuttavia una colorazione ottima per **qualunque ordinamento** dei vertici.

3. Se G è un *grafo intervallo*, dopo aver ordinato gli intervalli considerando i loro estremi sinistri in modo crescente, una volta assegnato un numero ad ogni intervallo e aver quindi ottenuto un ordinamento dei vertici $w_1, w_2, \ldots, w_n$, allora l'algoritmo Greedy colora G con il minimo numero di colori possibile.

Definizione 29:
Sia G un grafo qualsiasi. Vale sempre $\chi(G) \leq \triangle(G) + 1$, dove $\triangle(G)$ rappresenta il grado massimo di un vertice di G. La quantità $\triangle(G) + 1$ rappresenta un *upper bound* al valore di $\chi(G)$.

Definizione 30:
Una *clique* di un grafo $G(V, E)$ è un insieme di vertici a coppie adiacenti. Il *clique number* di G, ovvero il numero di vertici della *clique* più grande di G, si indica con $\omega(G)$ e rappresenta il *lower bound* del numero cromatico cioè è vero che $\omega(G) \leq \chi(G)$.

Definizione 31:
Se G è un grafo a intervallo il *clique number* è pari al numero cromatico di G, ovvero $\omega(G) = \chi(G)$.

Definizione 32:
Il numero minimo di colori utilizzati da una *edge-coloring* di G si indica con $\chi'(G)$. In particolare vale che $\chi'(G) \geq \triangle(G)$

Definizione 33:
Sia K_n un grafo completo. Allora $\chi'(G) = n$ se n è dispari; $\chi'(G) = n - 1$ se n è pari.

4 Conteggio

Definizione 34 (r**-permutazione):**
Se n e r sono interi tali che $0 \leq r \leq n$ allora una r-permutazione, indicata con $P(n,r)$ è definita come segue:

$$P(n,r) = \frac{n!}{(n-r)!} \tag{4}$$

Definizione 35 (r**-combinazione):**
Se n e r sono interi non negativi tali che $0 \leq r \leq n$ allora una r-combinazione, indicata con $C(n,r)$ è definita come segue:

$$C(n,r) = \binom{n}{r} = \frac{n!}{r!(n-r)!} \tag{5}$$

Inoltre valgono le seguenti proprietà:

$$C(n,r) = C(n,n-r) \tag{6}$$

$$C(n,0) = C(n,n) = 1 \qquad \forall n \geqslant 1 \tag{7}$$

$$C(n,1) = C(n,n-1) = n \qquad \forall n \geqslant 1 \tag{8}$$

Definizione 36 (Identità di Pascal):
Data una r-combinazione è vero che:

$$\binom{n+1}{r} = \binom{n}{r-1} + \binom{n}{r} \tag{9}$$

$$C(n+1,r) = C(n,r-1) + C(n,r) \tag{10}$$

Definizione 37 (Regola della somma):
Se un certo processo richiede lo svolgimento **alternativo** di un primo compito, che può essere svolto in n_1 modi diversi, oppure di un secondo compito, che può essere svolto in n_2 modi diversi, ci sono $n_1 + n_2$ modi di svolgere il processo.

Definizione 38 (Regola del prodotto):
Se un certo processo richiede lo svolgimento di un primo compito, che può essere svolto in n_1 modi diversi, **e poi** di un secondo compito, che può essere svolto in n_2 modi diversi, ci sono $n_1 n_2$ modi di svolgere il processo.

Definizione 39 (Pigeonhole principle):
Se $k+1$ oggetti devono essere collocati in k scatole, almeno due oggetti termineranno nella stessa scatola.

Definizione 40 (Pigeonhole principle generalizzato):
Se n oggetti devono essere collocati in k scatole, una scatola conterrà almeno $\lceil \frac{n}{k} \rceil$ oggetti.

Definizione 41:
Il numero totale di modi in cui è possibile distribuire n elementi **uguali** a k persone **con il vincolo di assegnare a ogni persona almeno un elemento** è pari a $C(n-1,k-1)$. In assenza di quest'ultimo vincolo, il numero totale diventa pari a $C(n+k-1,k-1)$.

Definizione 42:
Il numero totale di modi in cui è possibile distribuire n elementi **diversi** a k persone **con il vincolo di assegnare a ogni persona almeno un elemento** è pari a $P(n,k)$.

Definizione 43:
Dato $V = \{v_1, v_2, \ldots, v_n\}$ un insieme di n vertici, esistono esattamente $2^{C(n,2)}$ differenti grafi G tali che $V(G) = V$.

Teorema 3 (Teorema Binomiale):

$$(x+y)^n = \sum_{k=0}^{n} \binom{n}{k} x^{n-k} y^k \tag{11}$$

Teorema 4:

$$\sum_{k=0}^{n} \binom{n}{k} = 2^n \tag{12}$$

5 Massimo flusso

Definizione 44 (Flusso s-t ammissibile):
Sia dato un grafo orientato $G(V, E)$ con $|E| = m$ e supponiamo che sia anche data una *funzione di capacità* $c : E \rightarrow \mathbb{R}^+$ sugli archi di G; in altri termini per ogni arco $(u, v) \in E$ esiste una quantità $c_{u,v} \in \mathbb{R}^+$ detta *capacità*. Siano infine dati un nodo (di partenza o sorgente) s e un nodo (di arrivo) t. Allora il vettore $f \in \mathbb{R}^m$ si definisce **flusso s-t ammissibile** se e soltanto se sono soddisfatte le seguenti condizioni:

$$0 \leq f_{u,v} \leq c_{u,v} \qquad \forall (u, v) \in E \qquad \textbf{(vincolo di capacità)} \tag{13}$$

$$\sum_{u \in N^+(v)} f_{v,u} + \sum_{u \in N^-(v)} f_{u,v} = 0 \qquad \forall v \in V \setminus \{s, t\} \qquad \textbf{(vincolo di bilancio)} \tag{14}$$

Dove $N^+(v)$ e $N^-(v)$ rappresentano, rispettivamente, i *successori* e i *predecessori* di v. In altri termini il flusso di ogni arco $(u, v) \in E$ deve essere **non negativo** e **non deve eccedere la capacità** associata ad esso; inoltre la somma del flusso uscente e di quello entrante in un nodo $v \in V \setminus \{s, t\}$ deve essere nullo.

Definizione 45 (Valore del flusso):
Il valore del flusso $val(f)$ è pari alla *somma dei flussi uscenti dal nodo sorgente*. Più in generale, il valore del flusso $val(f)$ è dato dalla differenza tra il flusso uscente ed entrante da e verso il nodo sorgente. Formalmente:

$$val(f) = \sum_{u \in N^+(s)} f_{s,u} - \sum_{u \in N^-(s)} f_{u,s} \tag{15}$$

Definizione 46 (Taglio s-t):
Dato un grafo orientato $G(V, E)$ con $s, t \in V$ tali che $s \neq t$ di definisce *taglio* $s - t$ (W_1, W_2) una partizione dei nodi in due insiemi W_1 e W_2 tali che $s \in W_1$ e $t \in W_2$.

Definizione 47 (Capacità di un taglio s-t):
La *capacità* $C(W_1, W_2)$ di un taglio $s - t$ (W_1, W_2) è pari alla somma delle capacità degli archi che hanno il primo estremo in W_1 e il secondo estremo in W_2:

$$C(W_1, W_2) = \sum_{\substack{u \in W_1 \\ v \in W_2}} c_{u,v} \tag{16}$$

Definizione 48 (Cammino s-t aumentante):
Sia dato un grafo $G(V, E)$ capacitato e siano $s, t \in V$. Un cammino P, non necessariamente orientato, tra s e t è detto *s-t aumentante* rispetto ad un vettore di flusso f se:

1. Ogni arco $(u, v) \in E(P)$ che è concorde con la percorrenza da s a t è **non saturo**; cioè
$$c_{u,v} > f_{u,v} \tag{17}$$
2. Ogni arco $(u, v) \in E(P)$ che è discorde con la percorrenza da s a t è **non vuoto**; cioè
$$f_{u,v} > 0 \tag{18}$$

Definizione 49:
Sia un grafo $G(V, E)$. Se P è un *cammino aumentante* rispetto un vettore di flusso f, allora f **non** è un vettore di flusso massimo e possiamo quindi aumentarne il valore della seguente quantità:

$$\varepsilon(P) = min(min_{\text{Archi diretti}}(c_{u,v} - f_{u,v}); min_{\text{Archi inversi}}(f_{u,v})) \tag{19}$$

Il nuovo vettore di flusso f' sarà così definito:

$$f' = \begin{cases} f'_{u,v} = f_{u,v} & \text{per ogni arco}(u, v) \notin E(P, f) \\ f'_{u,v} = f_{u,v} + \varepsilon(P) & \forall (u, v) \in E(P, f) \text{ concorde con la percorrenza da s a t} \\ f'_{u,v} = f_{u,v} - \varepsilon(P) & \forall (u, v) \in E(P, f) \text{ discorde con la percorrenza da s a t} \end{cases} \tag{20}$$

Procedura 4 (Algoritmo dei cammini aumentanti o di Ford e Fulkerson):
Per individuare un flusso s-t di valore massimo è sufficiente utilizzare l'algoritmo dei cammini aumentanti o di Ford e Fulkerson che consiste in:

1. Trovare un cammino s-t aumentante P rispetto un vettore di flusso f;
2. Se P non esiste l'algoritmo termina.
3. Se P esiste calcola il nuovo flusso f' e poni $f = f'$; successivamente ritorna al punto 1.

6 Alcuni appunti sulla risoluzione dei problemi di programmazione lineare

Definizione 50 (Relazione tra Primale e Duale):
Sia P_I un *problema di programmazione lineare intera* e sia P il *rilassamento lineare* di P_I e sia D il *duale*. Allora valgono le seguenti considerazioni:

- Se una soluzione ammissibile x di P_I e una soluzione ammissibile y di D hanno lo stesso valore, allora x e y sono soluzioni ottime per i rispettivi problemi.
- Se una soluzione ammissibile x di P_I e una soluzione ammissibile y di D hanno lo stesso valore, allora x è una soluzione ottima per il problema P.
- Se una soluzione ammissibile x di P e una soluzione ammissibile y di D hanno lo stesso valore, allora x e y sono soluzioni **ottime** per i rispettivi problemi.

Primale \ Duale	**Ottimale**	**Non ammissibile**	**Illimitato**
Ottimale	Possibile	Impossibile	Impossibile
Non ammissibile	Impossibile	Possibile	Possibile
Illimitato	Impossibile	Possibile	Impossibile

Procedura 5 (Guida alla costruzione del duale):
Ricordiamo brevemente le relazioni tra primale e duale:

- il problema duale di un problema di minimizzazione è un problema di massimizzazione e simmetricamente, il problema duale di un problema di massimizzazione è un problema di minimizzazione;
- ad ogni vincolo di uguaglianza del problema primale è associata una variabile nel problema duale non vincolata in segno;
- ad ogni vincolo di disuguaglianza (di maggiore o uguale) del problema primale è associata una variabile nel problema duale vincolata in segno;
- ad ogni variabile vincolata in segno del problema primale è associato un vincolo di disuguaglianza ($\leq$ se il problema 'e di massimizzazione, $\geq$ se il problema è di minimizzazione) del problema duale;
- ad ogni variabile non vincolata in segno del problema primale è associato un vincolo di uguaglianza del problema duale.

	Primale	**Duale**	
	min $c^T x$	max $b^T y$	
Vincoli	$= b_i, i \in I$	$y_i, i \in I, \textit{libere}$	**Variabili**
	$\geq b_i, i \in J$	$y_i, i \in J, y_i \geq 0$	
Variabili	$x_j \geq 0, j \in M$	$\leq c_j, j \in M$	**Vincoli**
	$x_j, j \in N, \textit{libere}$	$= c_j, j \in N$	

Per esempio sia dato il seguente PL:

$$\begin{array}{lccccccc}
max & 2x_1 & +4x_2 & +3x_3 & +x_4 & & \\
 & 3x_1 & +x_2 & +x_3 & +4x_4 & \leq & 12 \\
 & x_1 & -3x_2 & +2x_3 & +3x_4 & \leq & 7 \\
 & 2x_1 & +x_2 & +3x_3 & -x_4 & = & 10 \\
 & x_1 & & & & \geq & 0 \\
 & & & x_3 & & \geq & 0 \\
 & & & & x_4 & \geq & 0
\end{array}$$

Il suo duale diventa:

$$\begin{array}{lcccccc}
min & 12y_1 & +7y_2 & +10y_3 & & \\
 & 3y_1 & +y_2 & +2y_3 & \geq & 2 \\
 & y_1 & -3y_2 & +y_3 & = & 4 \\
 & y_1 & +2y_2 & +3y_3 & \geq & 3 \\
 & 4y_1 & +3y_2 & -y_3 & \geq & 1 \\
 & y_1 & & & \geq & 0 \\
 & & y_2 & & \geq & 0
\end{array}$$

Supponiamo debba studiare la soluzione $(x_1, x_2, \ldots, x_n)$. Qualora sia *ammissibile* allora per ogni $x_j > 0$ si ha che nel duale risulta che $\sum b_j y_i = c$ (cioè lo j-esimo vincolo del duale presenterà un uguaglianza). Se una componente della soluzione $x_j = 0$ il vincolo corrispondente presenterà una disuguaglianza. (cioè se ho $(3, 0)$ allora il secondo vincolo del duale sarà del tipo $y_1 + y_2 + \ldots + y_n \geq v$.) Quando si studia l'ammissibilità di una soluzione nel primale se lo j-esimo vincolo dato da $\sum a_j x_i$ e risulti strettamente minore del vincolo, cioè $\sum a_j x_i < b$, allora la j-esima variabile del duale risulta nulla, cioè $y_j = 0$.

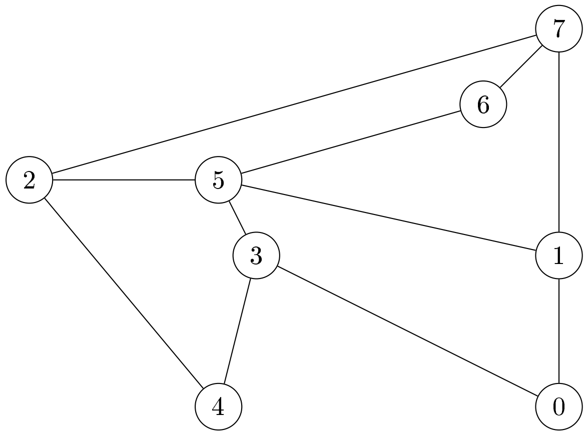
7 Alcuni esercizi d'esame

1. $G(V, E)$ è un grafo non orientato con 16 spigoli, 4 componenti connesse e aciclico. Quanti vertici ha?
   Risposta: 20

2. Quanti sono i diversi alberi con insieme dei vertici V = 0, 1, . . . , 7? (Due alberi sono diversi se esiste almeno una coppia di vertici i, j ∈ V che sono adiacenti per un albero e non adiacenti per l'altro.)
   Risposta: 8^6

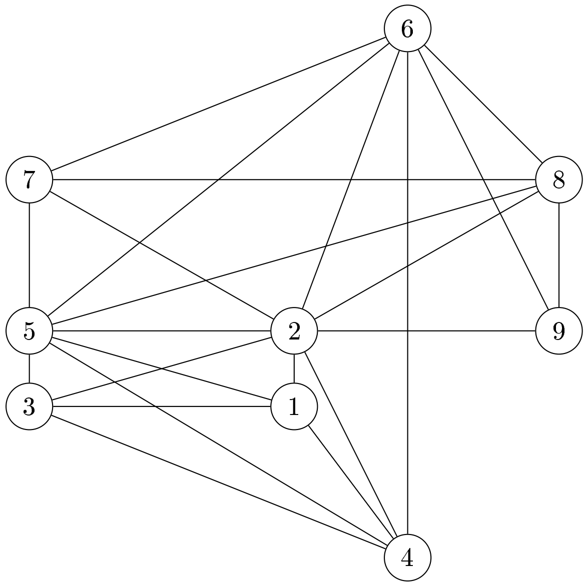
3. Si consideri l'albero T con vertici 0, 1, . . . , e spigoli 06, 14, 15, 16, 23, 38, 68, 78. Calcolane il prufer code.
   Risposta: $3, 8, 1, 1, 6, 8, 6$

4. Si consideri il grafo non orientato con insieme dei vertici 0, 1, . . . , 7 e insieme degli spigoli 01, 03, 35, 25, 56, 24, 27, 15, 17, 67, 34. Valutare la distanza di ogni vertice dal vertice 5. Dire quindi se il grafo è bipartito. Per rispondere è necessario riportare la distanza di ogni vertice da 5 e la bipartizione dei vertici o un certificato che tale bipartizione non esiste.
   Risposta: È bipartito

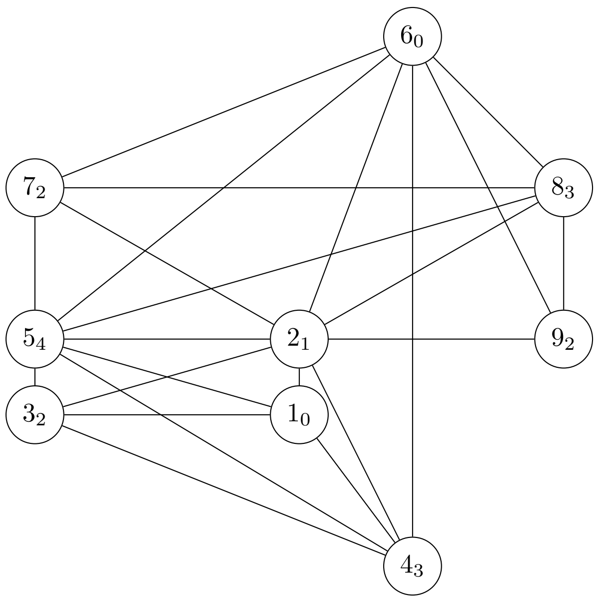
   

   Si valuta la distanza dal nodo 5 e otteniamo: $L_0 = \{5\}$, $L_1 = \{1, 3, 2, 6\}$, $L_2 = \{0, 4, 7\}$. Poiché non ci sono vertici di una stessa classe L_i che risultano adiacenti il grafo è bipartito. Le due classi della bipartizione sono date da $V_1 = \{0, 4, 7, 5\}$ e $V_2 = \{1, 2, 3, 6\}$.

5. Disponete di 4 server $S = \{A, B, C, D\}$ che volete destinare alla esecuzione di un insieme $P = \{1, 2, 3, 4, 5, 6, 7, 8\}$, di processi. Ogni processo $i \in P$ ha un orario di inizio r_i e un orario di completamento d_i per $i = 1..7$, gli intervalli $[r_i, d_i]$ sono rispettivamente: [11, 12]; [13, 20]; [9, 14]; [1, 8]; [11, 18]; [3, 16]; [5, 8]; [7, 12]; [7, 10]. Ogni server di S è in grado di svolgere un qualsiasi processo: tuttavia, perché i sia svolto correttamente, nell'intervallo $[r_i, d_i]$ dovete assegnare a i un server di S in maniera esclusiva (ovvero, in quell'intervallo, il server scelto può processare solo i). Fornite un assegnamento dei server ai processi che consenta lo svolgimento corretto di tutti i processi oppure un certificato (ovvero un'evidenza breve e semplice) che dimostri che tale assegnamento non può esistere.
   Risposta: 4 server non bastano.
   Ordiniamo gli intervalli considerando i loro **estremi sinistri** (ed, eventualmente, quello destro) in modo **crescente** e otteniamo:
   $[1, 8], [3, 16], [5, 8], [7, 10], [7, 12], [9, 14], [11, 12], [11, 18], [13, 20]$
   Costruiamo il grafo a intervallo in accordo alla definizione:



Adesso applichiamo **l'algoritmo Greedy** e otteniamo che:



Da cui risulta che $\chi(G) = 5$ e poiché per il grafo è a intervallo risulta che il *clique number* è uguale al numero cromatico. Per cui la *clique* è pari a $[1, 8], [3, 16], [5, 8], [7, 10], [7, 12]$. Per cui 4 server non bastano.

6. In quanti modi diversi 7 buste possono essere assegnate a 7 persone, se ognuna di esse riceve esattamente una busta?
Risposta: 7!

7. In quanti modi diversi 7 buste identiche possono essere assegnate a 7 persone, se non è richiesto che ogni persona riceva una busta?
Risposta: $C(7 + 7 - 1, 7 - 1) = C(13, 6)$

8. 16 Giocatori di tennis decidono di giocare un doppio. Quante coppie distinte si possono formare?
Risposta: $\binom{16}{2} \cdot \binom{14}{2} \cdot \binom{12}{2} \cdot ... \cdot \binom{2}{2}$

9. Una volta formate le 8 coppie, quante distinte partite (coppia contro coppia) si possono giocare?
Risposta: $C(8, 2)$

10. Avete un insieme X di oggetti e volete assegnare a ciascun oggetto di X un codice. Per semplicità ogni codice sarà dato da una sequenza ordinata di 3 caratteri, dove ogni carattere è preso dall'insieme di caratteri A, B,C, D, E, 1, 2, 3, 4, 5 e nessun carattere si ripete (quindi, per esempio: 1D2 e DAC sono codice ammissibili, 1DD non è un codice ammissibile e infine 1D2 e 2D1 sono codici ammissibili e diversi tra loro). Quanti oggetti devono essere presenti al più in X perché ogni oggetto riceva un codice diverso?
Risposta: $P(10, 3)$

11. Sia K_{11} il grafo non orientato, con 11 vertici e completo (ovvero tale che tutti i vertici sono adiacenti l'un l'altro). Siano u e v due particolari vertici di K_{11}. Quanti sono i diversi cammini da u a v con 4 spigoli? (Ricordiamo che in un cammino tutti i vertici sono distinti. Due cammini sono diversi se esiste almeno uno spigolo che appartiene ad un

cammino e non all'altro.)
Risposta: $P(9,3)$

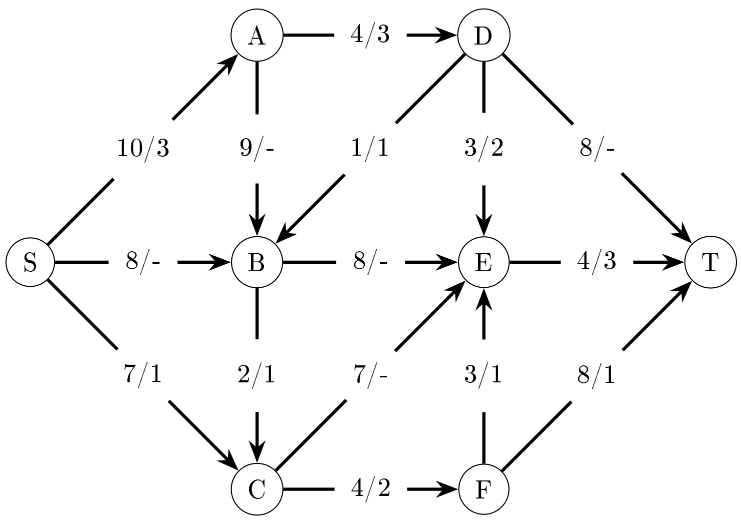
12. Considerate nuovamente il grafo K_{11} e siano u, v e z tre particolari vertici di $K_1 1$. Quanti sono i diversi cammini da u a v con 4 spigoli e che non passano per il vertice z?
Risposta: $P(8,3)$

13. Considerate nuovamente il grafo K_{11} e siano u, v e z tre particolari vertici di K 11. Quanti sono i diversi cammini da u a v con 4 spigoli e tali che il terzo vertice del cammino sia z?
Risposta: $P(8,2)$

14. In quanti modi diversi 8 frittelle di gusti diversi possono essere assegnate a 8 bambini, con il vincolo che ogni bambino riceva una frittella?
Risposta: 8!

15. In quanti modi diversi 8 frittelle uguali possono essere assegnate a 8 bambini, assumendo che non necessariamente ogni bambino riceva una frittella?
Risposta: $C(15,7)$

16. Individuare un flusso $s-t$ di valore massimo per la rete disegnata in figura, utilizzando l'algoritmo dei cammini aumentanti e partendo dal flusso iniziale dato, e certificane l'ottimalità. Per illustrare lo svolgimento dell'algoritmo, è sufficiente indicare tutti i cammini aumentanti scelti con l'indicazione per ogni arco di quanto è aumentato o diminuito il valore del flusso.



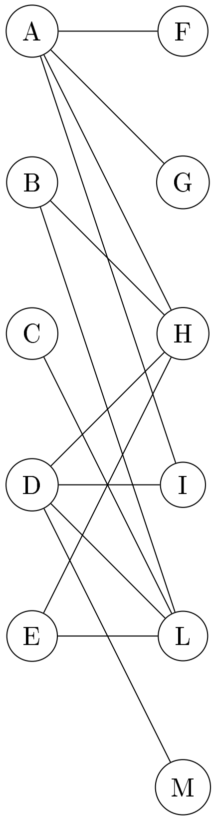
Allora i cammini aumentati sono:

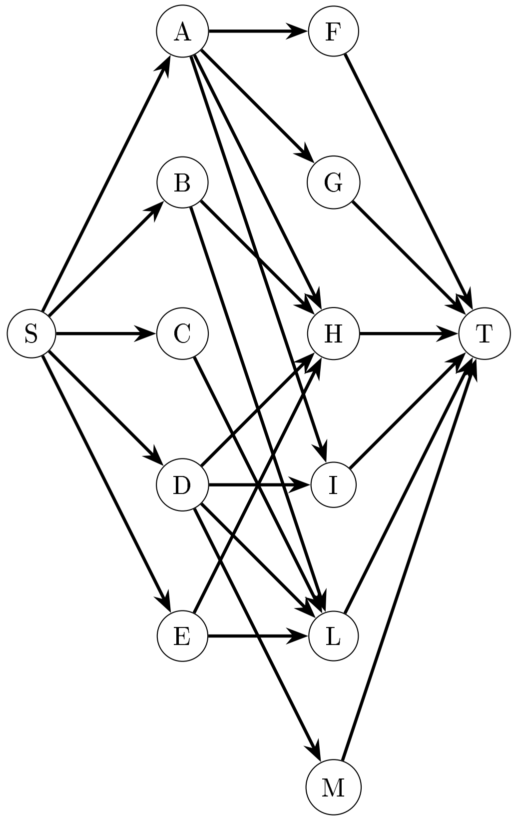
(a) $P_1 = \{S, C, F, T\}$ dove $\varepsilon(P_1) = 2$ e abbiamo che $f_{S,C} = 3$,$f_{C,F} = 4$,$f_{F,T} = 3$
(b) $P_2 = \{S, A, D, T\}$ dove $\varepsilon(P_2) = 1$ e abbiamo che $f_{S,A} = 4$,$f_{A,D} = 4$,$f_{D,T} = 1$
(c) $P_3 = \{S, B, E, T\}$ dove $\varepsilon(P_3) = 1$ e abbiamo che $f_{S,B} = 1$,$f_{B,E} = 1$,$f_{E,T} = 4$
(d) $P_4 = \{S, B, E, D, T\}$ dove $\varepsilon(P_4) = 2$ e abbiamo che $f_{S,B} = 3$,$f_{B,E} = 3$,$f_{D,E} = 0$, $f_{D,T} = 3$
(e) $P_5 = \{S, B, D, T\}$ dove $\varepsilon(P_5) = 1$ e abbiamo che $f_{S,B} = 4$,$f_{B,D} = 0$,$f_{D,T} = 4$
(f) $P_6 = \{S, B, E, F, T\}$ dove $\varepsilon(P_6) = 1$ e abbiamo che $f_{S,B} = 5$,$f_{B,E} = 4$,$f_{E,F} = 0$, $f_{F,T} = 4$

Il flusso così ottenuto ha valore 12 e la sua massimalità è certificata dal taglio $S_1 = \{S, A, B, C, E\}$, $S_2 = \{F, D, T\}$

17. Si consideri il grafo bipartito G con insieme dei vertici a, b, c, d, e, f , g, h, i, l, m e adiacenze definite dalle seguenti liste: ad j[a] = f , g, h, i; ad j[b] = h, l; ad j[c] = l; ad j[d] = h, i, l, m; ad j[e] = h, l; ad j[f] = a; ad j[g] = a; ad j[h] = a, b, d, e; ad j[i] = a, d; ad j[l] = b, c, e; ad j[m] = d. Si consideri il matching M = ag, bh, cl, di. Si certifichi l'ottimalità di tale matching esibendo un minimo taglio per un problema di massimo flusso su una rete ausiliaria oppure se ne certifichi la non l'ottimalità esibendo un cammino aumentante sulla stessa rete. Per rispondere all'esercizio, esibire un taglio minimo della rete ausiliaria oppure un matching di cardinalità maggiore: non è necessario disegnare la rete ausiliaria, ma se preferite disegnarla va bene. 3.1. Sia quindi X la classe della bipartizione che contiene il vertice c. Dire quindi se G ammette un matching X-completo e in caso contrario fornire un insieme che viola la condizione di Hall.

Disegniamo il grafo:





Possiamo quindi individuare un massimo matching **risolvendo un problema di massimo flusso** da cui risulta che il matching dato è di cardinalità massima in quanto il nodo t non è raggiungibile tramite cammini aumentanti da s. Il certificato di ottimalità di tale matching è dato dal taglio (S_1, S_2) con $S_1 = \{s, b, c, e, h, l\}$ e $S_2 = \{a, d, f, g, i, m, t\}$ di capacità 4.

Infine, G non ammette un matching X-completo, e l'insieme di vertici $Q \subseteq X$ che **viola** la condizione di Hall è $Q = \{b, c, e\}$, infatti $R(Q) = \{h, l\}$, da cui $|R(Q)| < |Q| = 3$.