

---

# Palestra di Algoritmi



*Liceo Galilei - Trento*

*#8 - 03/02/2022*

---



# o. Calendario

→ Olimpiadi di Informatica  
martedì 8 febbraio 2022

—  
**Siete pronti? Partiamo!**





# Piano

## Piano di studi

[https://training.olinfo.it/#/task/luiss\\_piano/statement](https://training.olinfo.it/#/task/luiss_piano/statement)

Qual è il minimo numero di ore necessarie per laurearsi?

# Pensa alla strategia / logica del programma

1. *Che tipo di grafo è? Orientato o non orientato? Pesato o non pesato?*
2. *Come memorizziamo il grafo?*
3. *Che tipo di visita devo fare? Da che nodo parto?*
4. *Mi basta solo una visita?*

# Pensa alla strategia / logica del programma

1. *Che tipo di grafo è?* **Orientato** e **Pesato**
2. *Come memorizziamo il grafo?* **Matrice** (ce la danno già)
3. *Che tipo di visita devo fare? Da che nodo parto?* **BFS**: sto cercando dei cammini minimi che rispettano delle condizioni
4. *Mi basta solo una visita?* No, devo fare **N visite**: parto da ciascun nodo

## INPUT

8 (K)

4 (N)

2 5 3 4 (crediti)

0 8 0 5

0 0 0 3

0 4 0 0

0 0 2 0

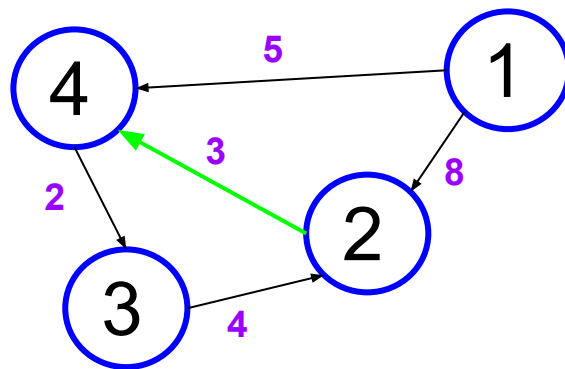
0	8	0	5
0	0	0	3
0	4	0	0
0	0	2	0

K = numero di crediti da ottenere

N = numero di esami (nodi del grafo)

# Pensa alla strategia / logica del programma

0	8	0	5
0	0	0	3
0	4	0	0
0	0	2	0



**Soluzione dell'esempio:** 3 ore richieste. Parto con l'esame 2, per il primo esame non si contano le ore; svolgo poi l'esame 4 con costo di 3 ore. La somma dei loro crediti è  $5+4 = 9$ , che è  $\geq K$ , perfetto!

Anche se ottengo più crediti di  $K$ , va bene comunque. Devono essere **almeno**  $K$ .



# Waterslide

## Scivoli d'acqua

[https://training.olinfo.it/#/task/ois\\_waterslide/statement](https://training.olinfo.it/#/task/ois_waterslide/statement)

In quale piscina arrivano più persone?



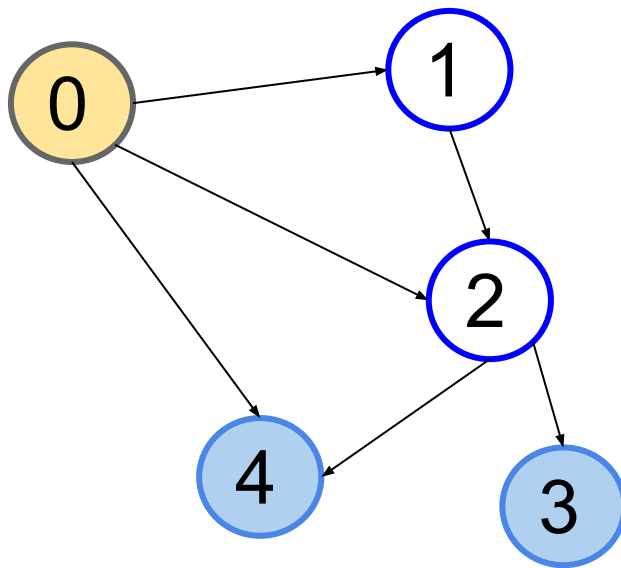


# Test Case #1

Piscine: 3, 4

Il nodo di partenza,  
come in tutti i casi, è 0.

Soluzione: piscina 4



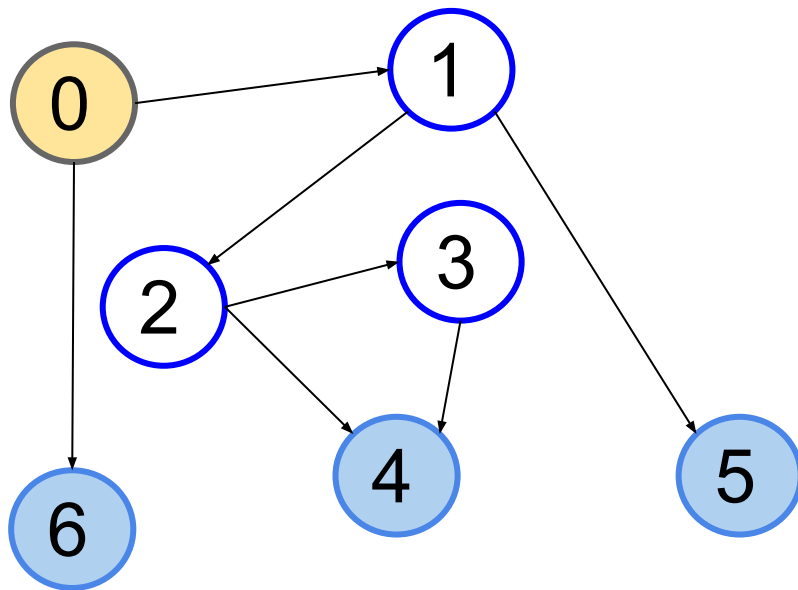
**Spiegazione:** assumendo che, ad ogni “svincolo”, e quindi ad ogni nodo, le persone si dividano equamente tra i possibili scivoli,  $\frac{2}{3}$  di esse finiscono nella piscina 4;  $\frac{1}{3}$  nella piscina 3

# Test Case #2

Piscine: 4, 5, 6

Il nodo di partenza,  
come in tutti i casi, è 0.

Soluzione: piscina 6

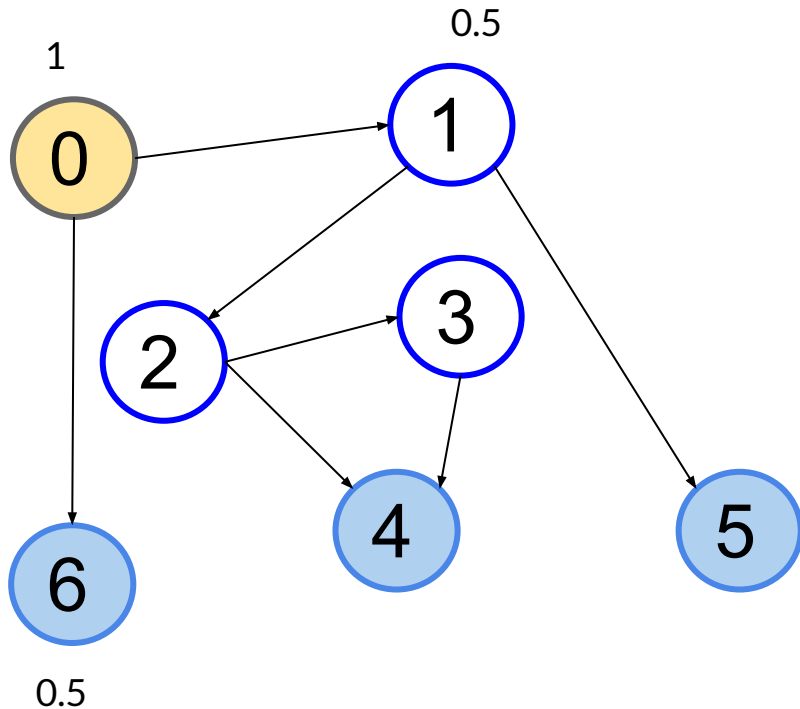


**Spiegazione:** assumendo che, ad ogni “svincolo”, e quindi ad ogni nodo, le persone si dividano equamente tra i possibili scivoli,  $\frac{1}{2}$  finiscono nella piscina 6; nella piscina 4 ne finiscono  $\frac{1}{4}$ , e nella piscina 5 sempre  $\frac{1}{4}$

# Un aiutino?

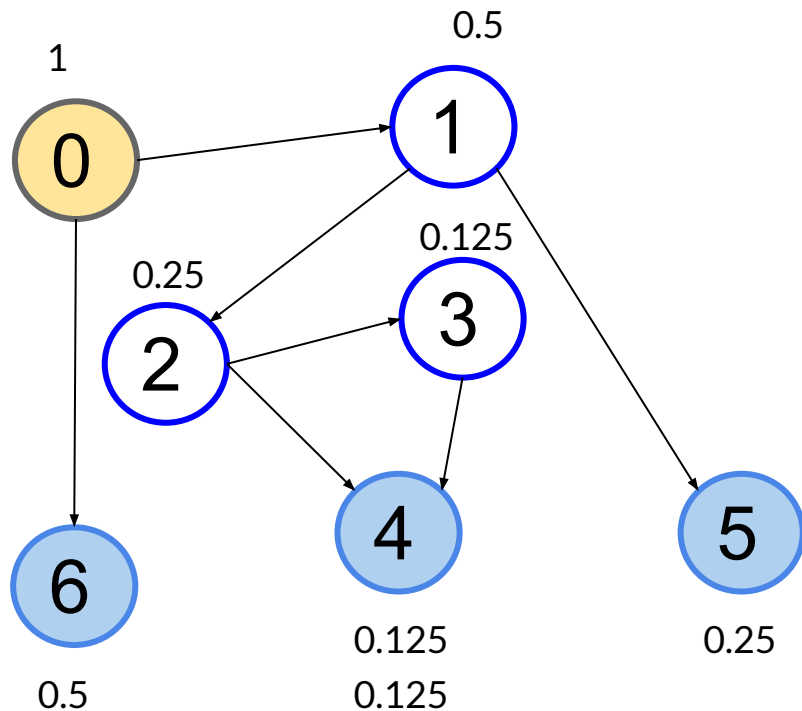
Immaginiamo che il totale di persone che partono dal nodo 0 sia 1.

Assumiamo che vengano divise equamente tra gli scivoli, quindi metà andranno al nodo 1 e metà al 6.



# Un aiutino?

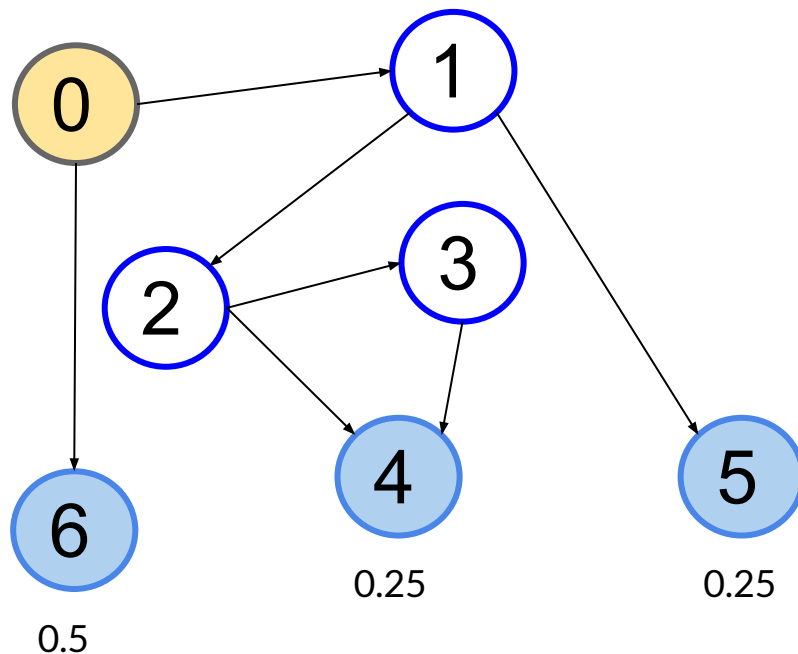
Andiamo avanti con questa strategia, finché arriviamo alla fine del percorso (una piscina), per tutti i percorsi possibili (visita DFS, in profondità)



# Un aiutino?

Inizializziamo ogni piscina  
ad un valore float 0.0

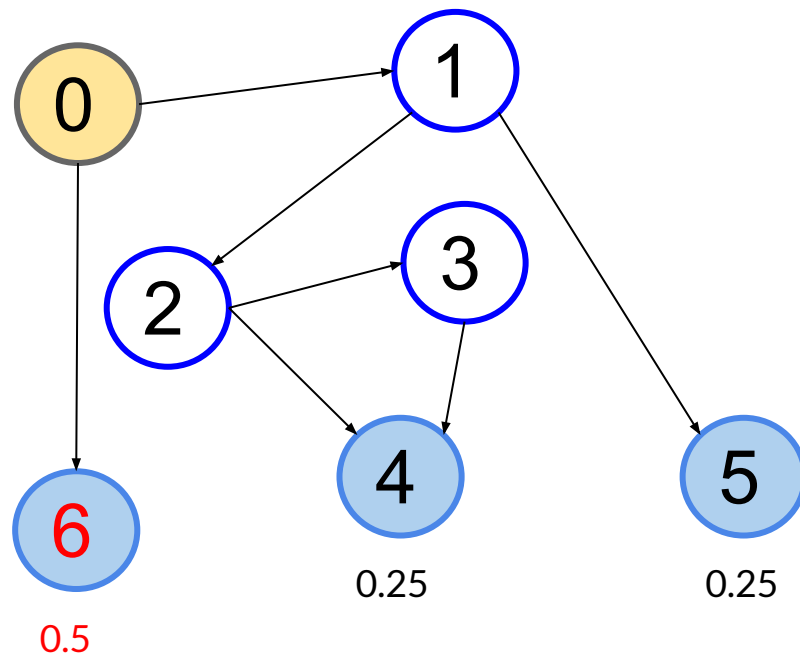
Ogni volta che la  
raggiungiamo con un  
percorso, gli sommiamo il  
valore in quel momento



# Un aiutino?

La piscina dove arrivano  
più persone è quella con il  
valore più alto.

In questo caso, la **6**



# Non arrivo a 100/100, uffa!



L'algoritmo visto funziona, la piattaforma dove inviamo gli esercizi non ci dice che l'output non è corretto. Siamo fuori con il tempo, in alcuni casi superiamo il mezzo secondo (tempo massimo previsto).

Per i curiosi, la soluzione da 100/100 utilizza un ***ordinamento topologico*** su un **DAG** (grafo orientato senza cicli), concetti e algoritmi che non abbiamo avuto modo di vedere, ma che restano comunque interessanti e hanno svariate applicazioni...

—

**in bocca  
al lupo!**