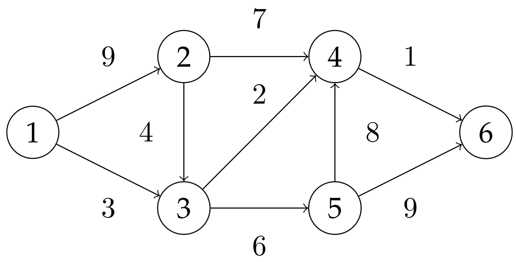
1 Lezione del 20-11-24

1.1 Algoritmo di Dijkstra

Torniamo sul problema dei **cammini di costo minimo**. Prendiamo il grafo:



Un algoritmo celebre per il problema è quello di **Dijkstra**.

Algoritmo 1 di Dijkstra

Input:
Output:
Etichetta ogni nodo come $\pi = (0, +\infty, ..., +\infty)$, assunto il primo nodo come partenza
Associa un predecessore ad ogni nodo come $P = (1, 0, ..., 0)$
Prendi il nodo $n \in N$ di etichetta π minima, rimuovilo da N, e prendi la stella uscente (FS(n)) di n
for $\forall j \in \mathrm{FS}(n)$ **do**
  if $\pi_j \geq \pi_i + c_{ij}$ **then**
    Poni il predecessore $P_j = i$
    Poni l'etichetta $\pi_j = \pi_i + c_{ij}$
  end if
end for

Le etichette π rappresenteranno la **distanza minima** dal nodo di partenza a ogni nodo. Ad algoritmo compiuto (quando N è vuoto) la catena dei predecessori da ogni nodo a quello di partenza forma i cammini minimi.

Sull'esempio riportato sopra, il primo passaggio è:

$$i = 1, \quad N = \{2, 3, 4, 5, 6\}$$

Da cui si calcola quindi la stella uscente $FS(1) = \{2, 3\}$ Prendiamo entrambi gli indici:

- $j = 2$: $\pi_2 \geq \pi_1 + c_{12} \rightarrow \begin{cases} \pi_2 = 8 \\ p_2 = 1 \end{cases}$
- $j = 3$: $\pi_3 \geq \pi_1 + c_{13} \rightarrow \begin{cases} \pi_3 = 3 \\ p_2 = 1 \end{cases}$

Da cui, all'inizio del secondo passaggio, etichette e predecessori sono:

$$\pi = (0, 8, 3, +\infty, +\infty, +\infty)$$

$$p = (1, 1, 1, 0, 0, 0)$$

L'algoritmo, con complessità linearitmica $O(n \log n + m \log n)$ su n nodi a ramificazione m, risulta più efficiente della risoluzione diretta dei problemi dei cammini minimi come problemi di flusso.