Grafi

CORDA – Informatica

A. Ferrari

Testi da

Marco Bernardo Edoardo Bontà

Dispense del Corso di

Algoritmi e Strutture Dati

Grafo – definizioni (1)

- Si dice grafo diretto o orientato una coppia $G = (V, \mathcal{E})$ dove V è un insieme di vertice ed $\mathcal{E}$ è una relazione binaria su V.

- Sia $G = (V, \mathcal{E})$ un grafo diretto:
  - Se $(v, v') \in \mathcal{E}$, allora si dice che v' è adiacente a v o, equivalentemente, che c'è un arco da v a v'.
  - Si dice grado uscente di $v \in V$ il numero di vertici adiacenti a v:
  $$d_o(v) = |\{v' \in V \mid (v, v') \in \mathcal{E}\}|$$
  - Si dice grado entrante di $v \in V$ il numero di vertici ai quali v è adiacente:
  $$d_i(v) = |\{v' \in V \mid (v', v) \in \mathcal{E}\}|$$
  - Si dice grado di $v \in V$ il numero di archi in cui v è coinvolto:
  $$d(v) = d_o(v) + d_i(v)$$

Grafo – definizioni (2)

- Se $v \in V$ è tale che:
  - $d_o(v) = 0$ e $d_i(v) > 0$ allora si dice che v è un vertice terminale;
  - $d_i(v) = 0$ e $d_o(v) > 0$ allora si dice che v è un vertice iniziale;
  - $d(v) = 0$ allora si dice che v è un vertice isolato.
- Si dice grado di G il massimo grado di un vertice di G:
  $$d(G) = \max\{d(v) \mid v \in V\}$$
- Si dice che G è completo se $\mathcal{E} = V \times V$.

Grafo – definizioni (3)

- Sia $G = (V, \mathcal{E})$ un grafo diretto:
  - Siano $v_1, v_2 \in V$. Si dice che v_2 è raggiungibile da v_1 se esiste un percorso da v_1 a v_2, cioè se esistono $v'_1, \ldots, v'_k \in V$ con $k \geq 2$ tali che $v'_1 = v_1$, $(v'_i, v'_{i+1}) \in \mathcal{E}$ per ogni $1 \leq i \leq k-1$, e $v'_k = v_2$.
  - Si dice che un percorso è semplice se tutti i vertici che lo compongono sono distinti, eccetto al più il primo e l'ultimo vertice.
  - Si dice che un percorso è un ciclo se il suo primo vertice coincide con il suo ultimo vertice.
  - Si dice che G è connesso se per ogni $v_1, v_2 \in V$ esiste un percorso da v_1 a v_2 o da v_2 a v_1.
  - Si dice che G è fortemente connesso se per ogni $v_1, v_2 \in V$ esistono un percorso da v_1 a v_2 e un percorso da v_2 a v_1.

- Si dice grafo pesato una tripla $G = (V, \mathcal{E}, w)$ dove $G = (V, \mathcal{E})$ è un grafo e $w : \mathcal{E} \longrightarrow \mathbb{R}$ è una funzione detta peso. Il peso associato ad un arco rappresenta di solito un tempo, una distanza, una capacità o un guadagno/perdita.

Problema dei ponti di Königsberg

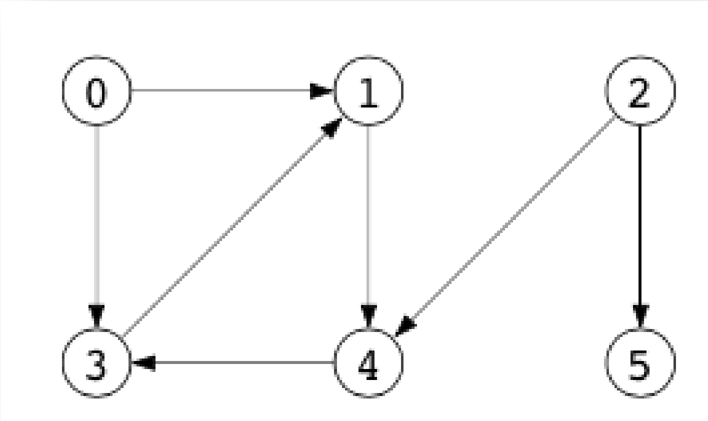
- Il problema dei sette ponti di Königsberg è un problema ispirato da una città reale e da una situazione concreta
- Königsberg è percorsa dal fiume Pregel e da suoi affluenti e presenta due estese isole che sono connesse tra di loro e con le due aree principali della città da sette ponti
- Nel corso dei secoli è stata più volte proposta la questione se sia possibile con una passeggiata seguire un percorso che attraversi ogni ponte una e una volta soltanto e tornare al punto di partenza
- Nel 1736 Leonhard Euler affrontò tale problema, dimostrando che la passeggiata ipotizzata non era possibile

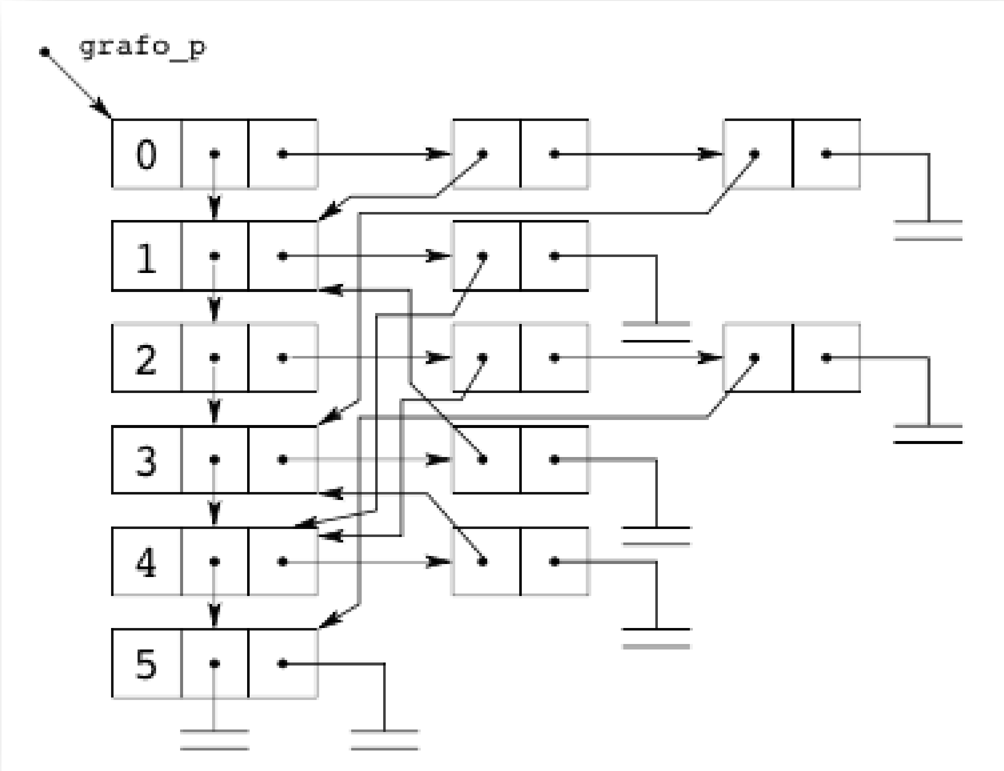
Astrazione

Astrazione - modellizzazione

- Eulero ha il merito di aver formulato il problema in termini di teoria dei grafi, astraendo dalla situazione specifica di Königsberg
- Innanzitutto eliminò tutti gli aspetti contingenti ad esclusione delle aree urbane delimitate dai bracci fluviali e dai ponti che le collegano
- Secondariamente rimpiazzò
  - ogni area urbana con un punto, (vertice o nodo)
  - e ogni ponte con un segmento di linea, chiamato spigolo, arco o collegamento.

Lista di adiacenza





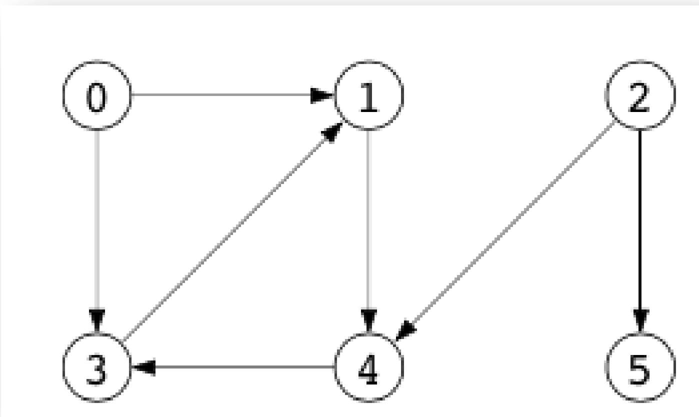
Implementazione lista di adiacenza

- Il grafo viene rappresentato come una struttura dati dinamica reticolare detta lista di adiacenza, formata da una lista primaria dei vertici e più liste secondarie degli archi
- La lista primaria contiene un elemento per ciascun vertice del grafo, il quale contiene a sua volta la testa della relativa lista secondaria.
- La lista secondaria associata ad un vertice descrive tutti gli archi uscenti da quel vertice

Problema 1

- Definire una struttura dati che permetta di implementare un grafo orientato mediante lista di adiacenza

Matrice di adiacenza



$$\begin{vmatrix} 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{vmatrix}$$

Implementazione matrice di adiacenza

- Se la struttura di un grafo non cambia oppure è importante fare accesso rapidamente alle informazioni contenute nel grafo, allora conviene ricorrere ad una rappresentazione a matrice di adiacenza
- Questa matrice ha tante righe e tante colonne quanti sono i vertici
- L'elemento di indici i e j vale 1 se esiste un arco dal vertice i al vertice j, 0 altrimenti
- Per i grafi pesati si può sostituire il valore 1 con il peso del grafo

Problema 2

- Definire una struttura dati che permetta di implementare un grafo orientato mediante matrice di adiacenza