Corso di laurea in Informatica
Introduzione agli Algoritmi
Didattica blended

Strutture dati fondamentali: Alberi

Angelo Monti

SAPIENZA
UNIVERSITÀ DI ROMA

Sulla base delle slides a cura di T. Calamoneri e G. Bongiovanni per il corso di informatica generale AA 2019/2020

Alberi

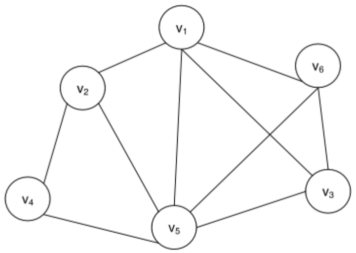
- L'***albero*** è una struttura dati estremamente versatile, utile per modellare una grande quantità di situazioni reali e progettare le relative soluzioni algoritmiche.
- Abbiamo già incontrato la struttura ad albero (in particolare ad albero binario) varie volte, ma l'abbiamo sempre considerata in modo intuitivo.

Alberi

Per dare la definizione formale di albero è necessario prima fornire alcune definizioni relative ad un'altra struttura dati, il ***grafo***:

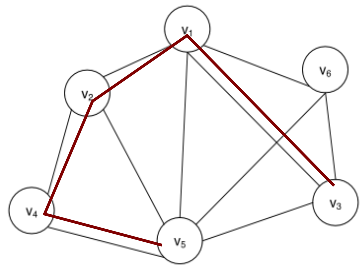
Un ***grafo*** $G = (V, E)$ è costituito da una coppia di insiemi:

- un insieme finito V dei ***nodi***, o ***vertici***;
- un insieme finito $E \subseteq V \times V$ di ***coppie non ordinate di nodi***, dette ***archi*** o ***spigoli***.



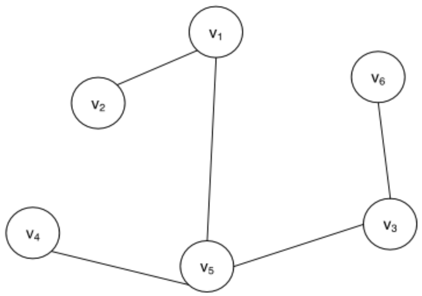
Alberi

- Un ***cammino*** in un grafo $G = (V, E)$ è una sequenza $(v_1, v_2, \ldots, v_k)$ di nodi distinti di V tale che (v_i, v_{i+1}) sia un arco di E per ogni $1 \leq i \leq k-1$.
- Se nel cammino $(v_1, v_2, \ldots, v_k)$ i nodi v_k e v_1 coincidono, si parla di ***ciclo***.
- Un grafo G è ***connesso*** se, per ogni coppia di nodi u e v, esiste un cammino tra u e v. Un grafo G è ***aciclico*** se non contiene cicli.



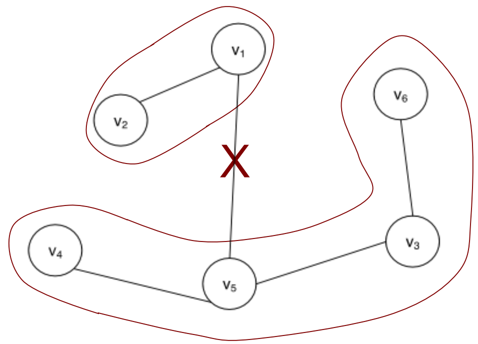
Alberi

Definzione. *Un albero è un grafo $G = (V, E)$ connesso e aciclico.*



Alberi

Lemma. *Sia $G = (V, E)$ un grafo connesso aciclico; eliminando da G un arco qualsiasi, G si disconnette, cioè si suddivide in due grafi $G_1 = (V_1, E_1)$ e $G_2 = (V_2, E_2)$, entrambi connessi e aciclici.*

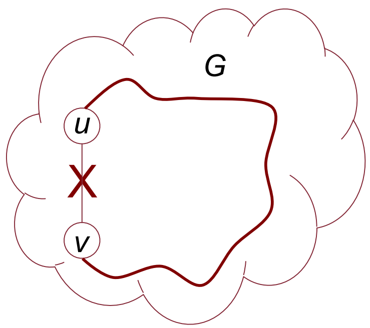


Alberi

Dimostrazione. Per assurdo, dopo l'eliminazione dell'arco $e = (u, v)$ il grafo rimane connesso.

Cioè, nel nuovo grafo esiste un cammino da u a v.

Ma allora, nel grafo originario G, tale cammino, con *e*, forma un ciclo, contro l'ipotesi che G sia aciclico.



Infine, banalmente entrambe le componenti generatesi devono rimanere connesse e acicliche.

Caratterizzazione per gli alberi

Teorema. *Sia $G = (V, E)$ un grafo. Le seguenti due affermazioni sono equivalenti:*

1. G è connesso e aciclico (in altre parole, G è un albero).
2. G è connesso ed $|E| = |V| - 1$.

Dim. **1.** $\Rightarrow$ **2.** Dimostreremo per induzione che, se G è connesso e aciclico, allora $|E| = |V| - 1$.

Passo base: se $|V| = 1$ oppure $|V| = 2$ l'affermazione è banalmente vera.

…

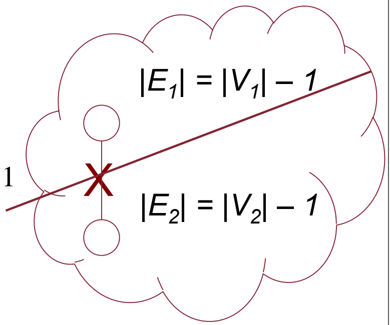
Caratterizzazione per gli alberi

segue dim. *G* è connesso e aciclico ⇒ *G* è connesso ed $|E| = |V| - 1$.

Passo induttivo: rimuovendo un arco qualsiasi, per il lemma provato precedentemente, il grafo *G* si disconnette in due grafi $G_1 = (V_1, E_1)$ e $G_2 = (V_2, E_2)$, entrambi connessi e aciclici.

Per essi vale l'hp induttiva:

- $|V| = |V_1| + |V_2|$
- $|E| = |E_1| + |E_2| + 1$
  $= |V_1| - 1 + |V_2| - 1 + 1 = |V| - 1$

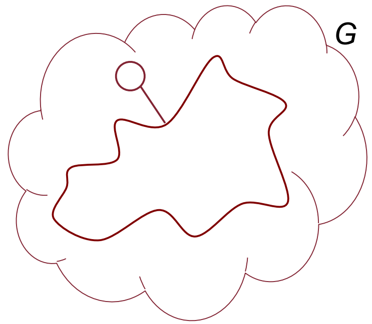


Caratterizzazione per gli alberi

segue dim. *G* è connesso ed $|E| = |V| - 1 \Longrightarrow G$ è connesso e aciclico

2. ⇒ 1. $G = (V, E)$ connesso, con $|V| = n$ e con $|E| = |V| - 1$
per assurdo contenga un ciclo $v_1, v_2, \ldots, v_k, v_1$.
Consideriamo il grafo $G_k = (V_k, E_k)$ costituito dal solo ciclo.
In esso abbiamo: $|E_k| = |V_k|$

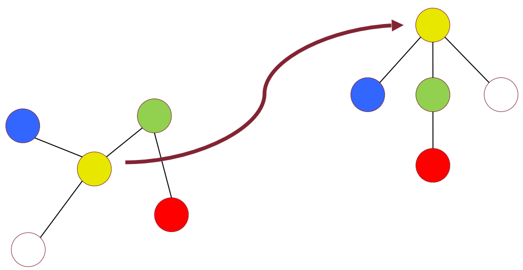
Se $k < n$ esiste un nodo v_{k+1} connesso a G_k tramite un arco, se lo inserisco nel grafo G_k ottengo un nuovo grafo G_{k+1} con
$|E_{k+1}| = |V_{k+1}|$.
Si prosegue fino ad ottenere G_n per cui
$|V_n| = |E_n| \Longrightarrow |E_n| = n$
Ma per ipotesi $|E| = n - 1$, per cui per cui si ha l'assurdo $|E_n| > |E|$



Alberi radicati

Alberi radicati: vi si distingue un nodo particolare tra gli altri, detto ***radice***.

L'albero radicato si può rappresentare in modo tale che i cammini da ogni nodo alla radice seguano un percorso dal basso verso l'alto, come se l'albero venisse, in qualche modo, "appeso" per la radice.



Alberi radicati

In un albero radicato:

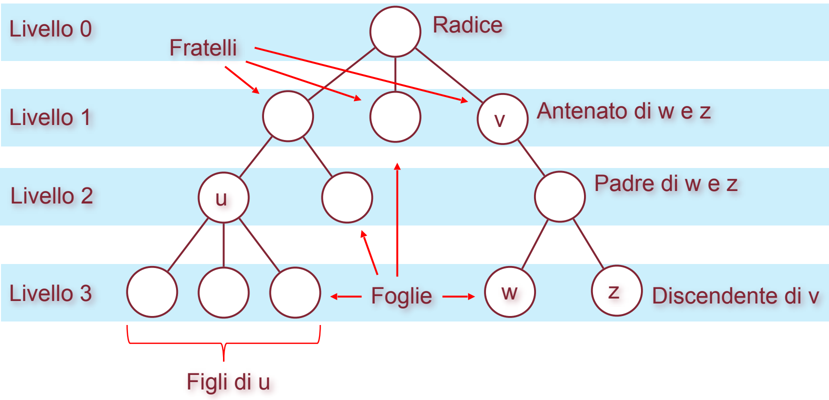
- i nodi sono organizzati in ***livelli***, numerati in ordine crescente allontanandosi dalla radice (di norma la radice è posta a livello zero),
- l'***altezza*** di un albero radicato è la lunghezza del più lungo cammino dalla radice ad una foglia; un albero di altezza h contiene $h + 1$ livelli, di norma numerati *da* 0 *ad* h.

Alberi radicati

In un albero radicato:

- Dato un qualunque nodo v che non sia la radice, il primo nodo che si incontra sul (unico) cammino da v alla radice viene detto ***padre di*** v**,**
- nodi che hanno lo stesso padre sono detti ***fratelli*** e la radice è l'unico nodo che non ha padre,
- ogni nodo sul cammino da v alla radice viene detto ***antenato di*** v**,**
- tutti i nodi che ammettono v come padre sono detti ***figli di*** v, ed i nodi che non hanno figli sono detti ***foglie,***
- tutti i nodi che ammettono v come antenato vengono detti ***discendenti di*** v.

Alberi radicati (esempio, $h = 3$)



Alberi radicati

Un albero radicato si dice ***ordinato*** se attribuiamo un qualche ordine ai figli di ciascun nodo, nel senso che se un nodo ha k figli, allora vi è un figlio che viene considerato primo, uno che viene considerato secondo, …, uno che viene considerato k-esimo.

Una particolare sottoclasse di alberi radicati e ordinati è quella degli ***alberi binari***, che hanno la particolarità che ogni nodo ha al più ***due figli***. Poiché sono alberi ordinati, i due figli di ciascun nodo si distinguono in ***figlio sinistro*** e ***figlio destro.***

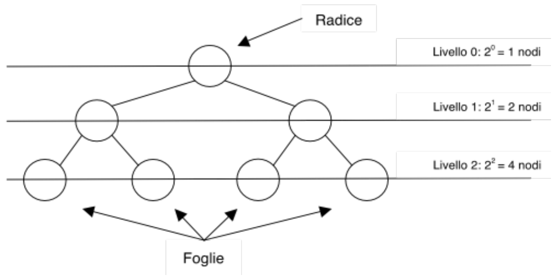
Alberi binari

Un albero binario nel quale tutti i livelli contengono il massimo numero possibile di nodi è chiamato ***albero binario completo***.

Se invece tutti i livelli tranne l'ultimo contengono il massimo numero possibile di nodi mentre l'ultimo livello è riempito completamente da sinistra verso destra solo fino ad un certo punto, l'albero è chiamato ***albero binario quasi completo***.

Alberi binari

In un albero binario completo di altezza h:



- il numero delle foglie è 2^h
- il numero dei nodi interni è $\sum_{i=0}^{h-1} 2^i = \frac{2^h - 1}{2 - 1} = 2^h - 1$
- il numero totale dei nodi è $2^h + 2^h - 1 = 2^{h+1} - 1$.

Alberi binari

Di conseguenza, l'altezza h di un albero binario completo è calcolabile come segue:

- Il numero dei nodi è: $n = 2^{h+1} - 1$
- da cui segue che: $\log_2 (n + 1) = h + 1$
- e quindi: $h = \log_2 (n + 1) - 1 = \log_2 \frac{n+1}{2}$

In sostanza per un albero binario **completo** si ha $h = \Theta(\log n)$

In generale, per un albero binario si ha $\Omega(\log n) = h \leq n - 1$

Rappresentazione di albero binario in memoria.
Descriveremo tre diverse rappresentazioni:

- **Memorizzazione tramite record e puntatori**
- **Rappresentazione posizionale**
- **rappresentazione tramite vettore dei padri**