

# Tutorato 29 Novembre

## 1 Esercizi

### Esercizio 1 - ABR e max-heap

Rispondere alle seguenti domande:

- Qual è la differenza tra alberi binari di ricerca e max-heap?
- In un albero binario di ricerca è possibile stampare i valori dei nodi in ordine utilizzando un algoritmo di visita in-order di costo  $O(n)$ . E' possibile fare lo stesso in un max-heap?

### Esercizio 2

Considerare la seguente frase e discutere se è vera o falsa:

*In un albero binario di ricerca in cui tutti gli elementi sono distinti, se il nodo  $x$  ha sotto-albero destro vuoto e ha un successore  $y$ , allora  $y$  è il nodo antenato di  $x$  più basso (i.e. con altezza minore) che abbia un figlio sinistro anch'esso antenato di  $x$ .*

Notare che ogni nodo è antenato di se stesso.

## 2 Soluzioni

### Esercizio 1

- La proprietà degli alberi binari di ricerca è che ogni nodo sia maggiore di tutti i nodi nel sottoalbero sinistro e minore di tutti i nodi nel sottoalbero destro. La proprietà di max-heap invece prevede che ogni nodo sia maggiore dei due figli.

Per vedere che le strutture sono in generale diverse si possono fare le seguenti osservazioni (questi sono alcuni esempi, se ne potrebbero fare anche altri):

- in un max-heap la radice dell'albero è sempre il massimo; ciò non è vero negli alberi binari di ricerca;
- in un max-heap se un nodo ha un figlio destro questo è minore del nodo; ciò non è vero negli alberi binari di ricerca.

- Non è possibile, perchè l'algoritmo in-order sfrutta la proprietà che il sottoalbero sinistro contiene valori minori del resto dell'albero. Ciò non è vero nel max-heap.

### **Esercizio 2**

La frase è corretta. Per dimostrarlo devo dimostrare tre cose:

- $y$  è un antenato di  $x$ ;
- $y$  ha un figlio sinistro antenato di  $x$ ;
- $y$  è il più basso nodo con queste proprietà.

Intanto facciamo vedere che  $y$  è un antenato di  $x$ . Se non lo fosse, potremmo prendere il più basso antenato comune di  $x$  e  $y$ , chiamiamolo  $z$ .  $x$  e  $y$  devono stare necessariamente in sottoalberi diversi di  $z$  (altrimenti l'antenato comune sarebbe un figlio di  $z$ ), ma questo implicherebbe  $x \leq z \leq y$ , quindi  $y$  non sarebbe il successore di  $x$ . Dunque  $y$  è un antenato di  $x$ .

Mostriamo che  $y$  ha un figlio sinistro antenato di  $x$ . Sappiamo che  $x$  si deve trovare nel sottoalbero sinistro di  $y$ , altrimenti  $x \geq y$ . Dunque se  $y$  ha un figlio sinistro diverso da  $x$ , chiamiamolo  $z$ , questo è ancora antenato di  $x$ .

Infine facciamo vedere che  $y$  è il nodo più basso con le prime due proprietà. Supponiamo che nel cammino dalla radice a  $x$  ci siano altri due nodi,  $y'$  e  $z'$  con altezza minore e tali che  $z'$  è figlio sinistro di  $y'$ . Allora  $x \leq y'$  perchè sta nel suo sottoalbero sinistro, e  $y' < y$  per lo stesso motivo. Dunque  $y$  è il più basso nodo il cui figlio sinistro sia un antenato di  $x$ .