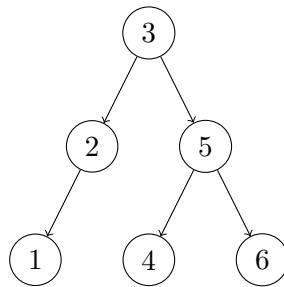


Alberi binari di ricerca

1 Albero binario di ricerca

Un **albero binario di ricerca** (**BST**, **binary search tree**) per un insieme U è un albero binario tale che

- ogni nodo v contiene un elemento $u \in U$;
- se il nodo x appartiene al sottoalbero sinistro di v , allora contiene $s < u$;
- se il nodo y appartiene al sottoalbero destro di v , allora contiene $t > u$.



Gli alberi binari di ricerca costituiscono una possibile implementazione per l'algebra eterogenea *parti di A ordinato*.

Osservazione: La definizione di un BST è diversa da quella di uno heap, nel quale entrambi i figli contengono elementi minori di quello del padre.

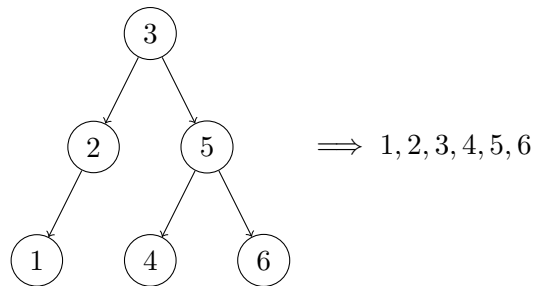
2 Implementazione

Un albero binario di ricerca si implementa solitamente rappresentando ciascun nodo con un record, che contiene:

- un elemento di U ;
- un riferimento al nodo figlio sinistro;
- un riferimento al nodo figlio destro.

3 Visita in ordine simmetrico e costo della costruzione

Visitando in ordine simmetrico un BST, si ottiene la sequenza ordinata dei valori nell'albero.



Di fatto, la costruzione di un BST, seguita dalla visita in ordine simmetrico, costituisce un algoritmo di ordinamento della classe *confronti e scambi*, e allora, per un albero con n nodi, deve avere complessivamente costo $\Omega(n \log n)$ nel caso peggiore (per rispettare il limite inferiore al numero di confronti, il “record imbattibile”).

Siccome la complessità della visita è solo $\Theta(n)$, il costo $\Omega(n \log n)$ nel caso peggiore deve essere quello della costruzione.