

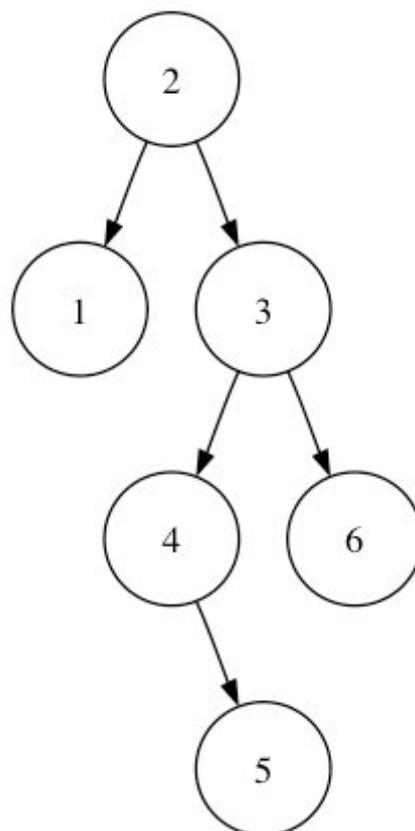
## Esercizio 1

Scrivere le funzioni di lettura e stampa di un albero binario etichettato con interi in base alla seguente rappresentazione con parentesi:

- $()$  corrisponde all'albero vuoto
- se  $n$  è un intero e  $left$  e  $right$  sono le rappresentazioni con parentesi di due alberi, allora  $(n\ left\ right)$  corrisponde all'albero con etichetta  $n$  nella radice che ha  $left$  come sotto-albero sinistro e  $right$  come sotto-albero destro.

Per semplificare la lettura della sequenza di input, le parentesi saranno rappresentate dai numeri negativi -1, per la parentesi aperta, e -2, per la parentesi chiusa. Inoltre, per evitare problemi con la rappresentazione delle parentesi, assumeremo che le etichette dei nodi dell'albero siano non-negative.

Ad esempio, l'albero



nella notazione parentetica precedentemente descritta è rappresentato da

( 2 ( 1 ( ) ( ) ) ( 3 ( 4 ( ) ( 5 ( ) ( ) ) ) ( 6 ( ) ( ) ) ) )

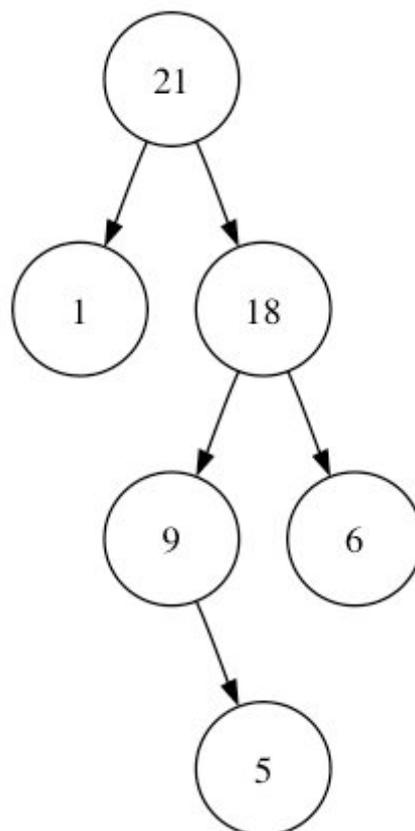
che, rimpiazzando le parentesi con -1 (aperta) e -2 (chiusa) diviene

-1 2 -1 1 -1 -2 -1 -2 -2 -1 3 -1 4 -1 -2 -1 5 -1 -2 -1 -2 -2 -2 -1 6 -1 -2 -1  
-2 -2 -2 -2

## Esercizio 2

Si scriva una funzione che prende come parametro di input un albero binario etichettato con interi  $t$  e crea un nuovo albero  $t'$  con la stessa struttura di  $t$  (ovvero gli alberi  $t$  e  $t'$  hanno la stessa forma e differiscono solo per i valori delle etichette nei nodi), tale che l'etichetta di ogni nodo  $n'$  di  $t'$  è pari alla somma delle etichette del sotto-albero con radice nel nodo  $n$  che corrisponde a  $n'$  in  $t$ .

Ad esempio, se  $t$  è l'albero riportato nell'esercizio 1, l'albero  $t'$  è



Utilizzando le funzioni di input/output dell'esercizio precedente scrivere un main che legge un albero, costruisce e stampa l'albero richiesto

dall'esercizio.

### Esercizio 3

Si scriva una funzione che stampa la sequenza, presa da sinistra a destra, dei nodi di livello  $h$  di un albero  $t$ .

Scrivere una funzione che, utilizzando la precedente funzione, stampa i nodi di un albero per livelli.

Ad esempio, supponendo che si vogliano stampare i livelli dell'albero dell'esercizio 2. Se si vuole stampare solo il livello 2 si ottiene come output "9 6", mentre a livello 3 si ottiene solo "5". La stampa completa per livelli deve invece produrre "21 1 18 9 6 5".

Utilizzando le funzioni di input precedentemente implementate, scrivere un main che legge un albero e richiama la funzione che stampa per livelli.

### Esercizio 4

Si modifichi la funzione dell'esercizio 3 in modo che oltre a stampare le etichette dei nodi di livello  $h$  effettivamente presenti nell'albero, venga stampato un \* al posto di ogni nodo mancante rispetto al caso in cui il livello  $h$  è pieno.

Scrivere la funzione che, utilizzando la precedente funzione, stampa tutti i livelli di un albero, uno per riga.

Prendendo gli stessi esempi riportati nell'esercizio 3, la stampa del livello 2 produce

```
* * 9 6
```

mentre la stampa del livello 3 produce

```
* * * * * 5 * *
```

(si osservi che, contando le etichette vere e proprie e gli \*, ad ogni livello si stampano sempre  $2^n$  valori).

Infine, la stampa per livelli dell'intero albero produce

21

1 18

\* \* 9 6

\* \* \* \* \* 5 \* \*

Utilizzando le funzioni di input precedentemente implementate, scrivere un main che legge un albero e richiama la funzione che stampa per livelli.