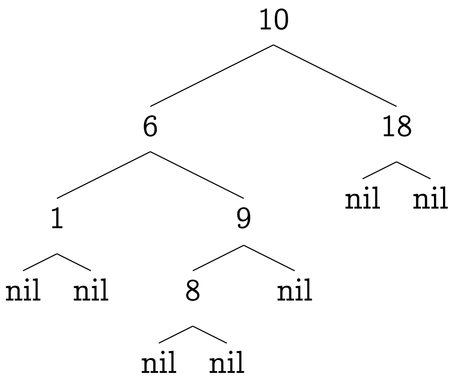
Prof. L. Margara **Scritto del 2/9/2021**

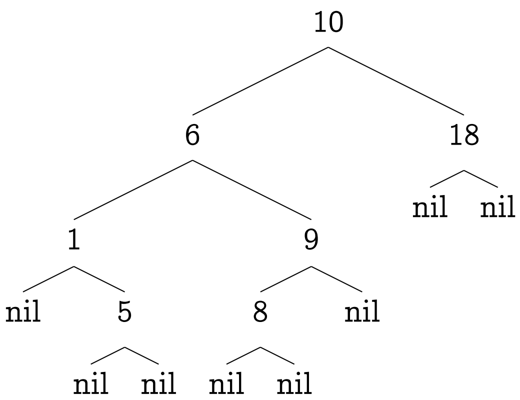
Algoritmi e Strutture Dati

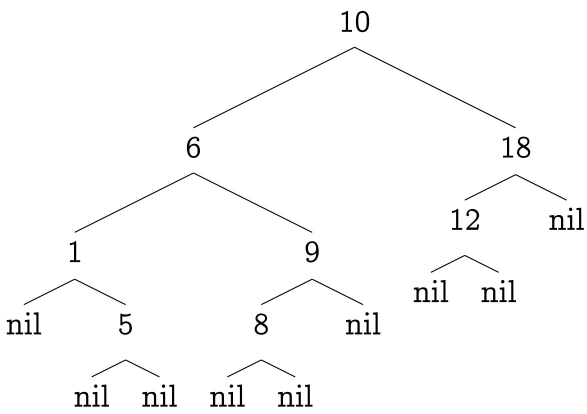
Esercizio 1.[11 punti]
Eseguire le operazioni sotto indicate sul seguente albero binario di ricerca spiegando brevemente la tecnica utilizzata.

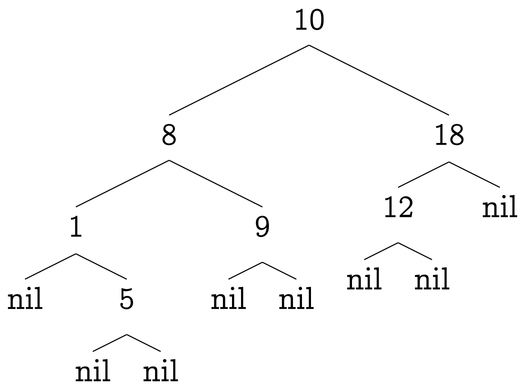


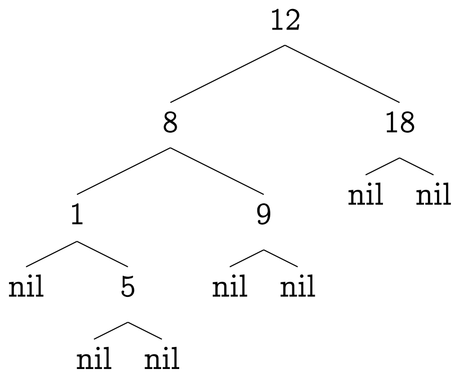
Inserimento 5. Inserimento 12. Cancellazione 6. Cancellazione 10. Inserimento 23. Inserimento 19. Inserimento 6.
Mostrare tutti gli alberi ottenuti dopo ogni operazione (eseguire le operazioni in sequenza senza ripartire dall'albero iniziale, ma da quello appena calcolato al passo precedente)

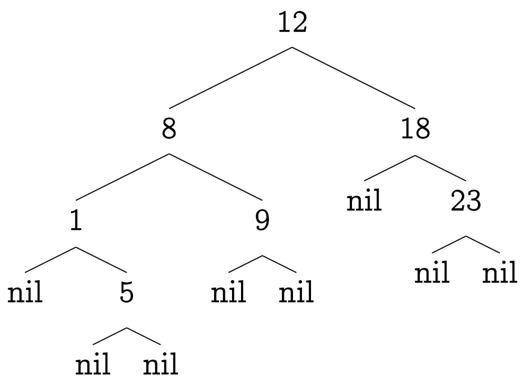
Traccia della soluzione dell'esercizio .

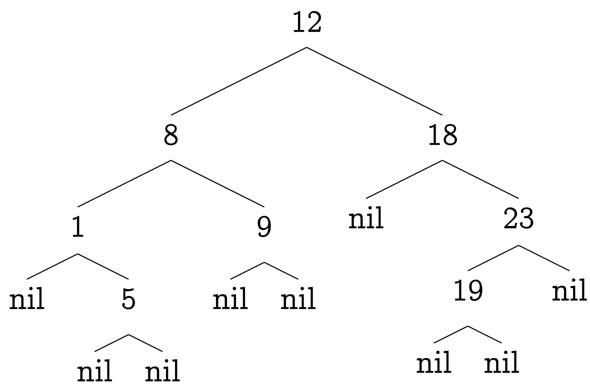


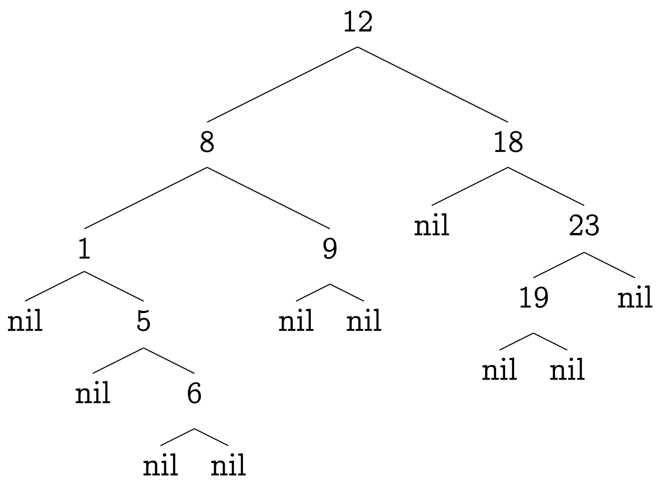



10
8 18
1 9 12 nil
nil 5 nil nil nil nil
nil nil


12
8 18
1 9 nil nil
nil 5 nil nil
nil nil


12
8 18
1 9 nil 23
nil 5 nil nil nil nil
nil nil


12
8 18
1 9 nil 23
nil 5 nil nil 19 nil
nil nil nil nil


12
8 18
1 9 nil 23
nil 5 nil nil 19 nil
nil 6 nil nil
nil nil

Esercizio 2.[11 punti]
Sia $G = (V, E)$ un grafo non orientato pesato. I pesi degli archi possono essere uguali a un certo x oppure a $2x$ dove x è un numero intero maggiore di zero e sconosciuto (non viene dato in input). Sia $s \in V$ (in input). Descrivere accuratamente (non necessariamente in pseudocodice) un algoritmo che calcoli l'albero dei cammini minimi da s a tutti gli altri nodi in tempo $O(|V| + |E|)$.

Traccia Soluzione esercizio .
G' si ottiene sostituendo agli archi $v_1 \to v_2$ di G di peso $2x$ due con due archi $v_1 \to y$ e $y \to v_2$ ciascuno di peso 1 e sostituendo agli archi $v_1 \to v_2$ di G di peso x un arco $v_1 \to v_2$ di peso 1.

Poi si applica a G' la BFS. Il tutto ha costo computazionale $O(|V| + |E|)$.

Esercizio 3.[11 punti]

$$T(n) = \begin{cases} 1 & \text{if } n \leq c, \\ 23\,T(n/600) + 77\,T(n/601) + 1 & \text{if } n > c. \end{cases}$$

Dimostrare che $T(n) = o(n^{0.8})$ e che $T(n) = \omega(n^{0.6})$.

Traccia Soluzione.

$T(n) = 23T(n/600) + 77T(n/601) + 1$
caso: 1.
$T(n) \leq 23T(n/600) + 77T(n/600) + 1 = 100T(n/600) + 1$
caso: 2.
$T(n) \geq 23T(n/601) + 77T(n/601) + 1 = 100T(n/601) + 1$
quindi:
$100T(n/601) + 1 \leq T(n) \leq 100T(n/600) + 1$

visto che:
$100T(n/600) + 1$ è $\Theta(n^{\log_{600}(100)})$ e che
$100T(n/601) + 1$ è $\Theta(n^{\log_{601}(100)})$

concludiamo:
$T(n) = \Omega(n^{\log_{601}(100)})) = \Omega(n^{0.719716})$
$T(n) = O(n^{\log_{600}(100)})) = O(n^{0.719903})$