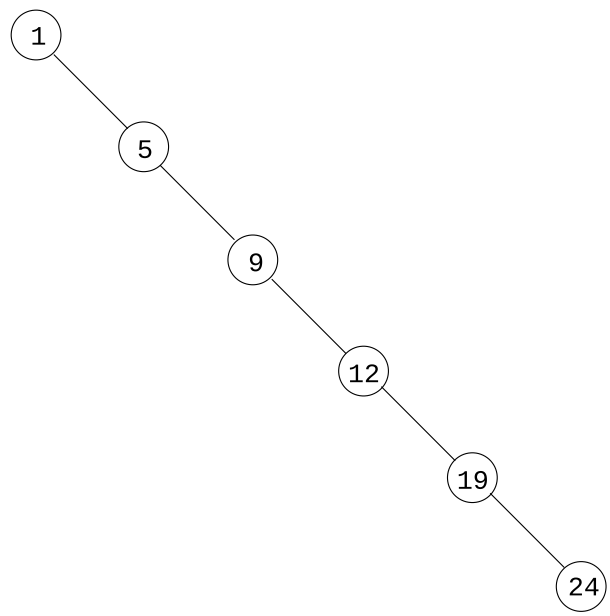
9.3.2 Les arbres équilibrés

Dans un arbre binaire, la racine a au maximum 2^1 enfants, 2^2 petits-enfants, 2^3 arrière-petits-enfants, ... La taille maximale d'un arbre binaire de hauteur `h` est donc $2^0 + 2^1 + 2^2 + \ldots + 2^h = 2^{h+1} - 1$. De ce fait, quand un nombre `n` est tel que $2^{h+1} \leq n < 2^{h+2}$, il est possible de construire un arbre binaire de taille `n` de hauteur `h + 1`, mais pas de hauteur `h`. Autrement dit, la hauteur minimale d'un arbre de taille `n` est $\lfloor \log_2(n) \rfloor$, la partie entière du logarithme binaire de `n`. Un arbre est dit *de hauteur minimale* si sa hauteur est égale à $\lfloor \log_2 n \rfloor$ où `n` est sa taille. La complexité de la recherche, de l'insertion et de la suppression d'un élément dans un arbre de hauteur minimale est donc logarithmique en la taille de l'arbre.

Cependant, la hauteur maximale d'un arbre de taille `n` est `n - 1`.



La complexité de la recherche, de l'insertion et de la suppression d'un élément dans un arbre de hauteur maximale est donc linéaire — et non logarithmique — en la taille de l'arbre.

La hauteur d'un arbre de taille `n` est donc comprise entre $\lfloor \log_2(n) \rfloor$ et `n - 1`. On peut démontrer que si l'on construit un arbre par insertion dans un ordre aléatoire de `n` éléments distincts, alors l'espérance de la hauteur de l'arbre est proportionnelle à $\ln(n)$. L'espérance du temps nécessaire pour rechercher, insérer ou supprimer un élément est donc logarithmique. Mais, ce résultat de complexité en espérance n'empêche pas les cas dégénérés de se produire. En particulier, si l'on insère, comme c'est malheureusement souvent le cas, des