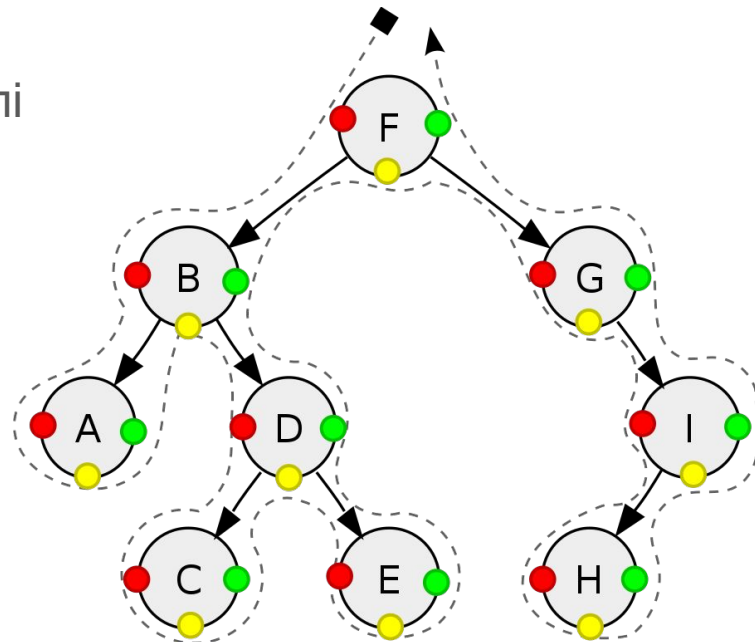


Збалансовані дерева. AVL-дерево

Обхід дерева в порядку зростання

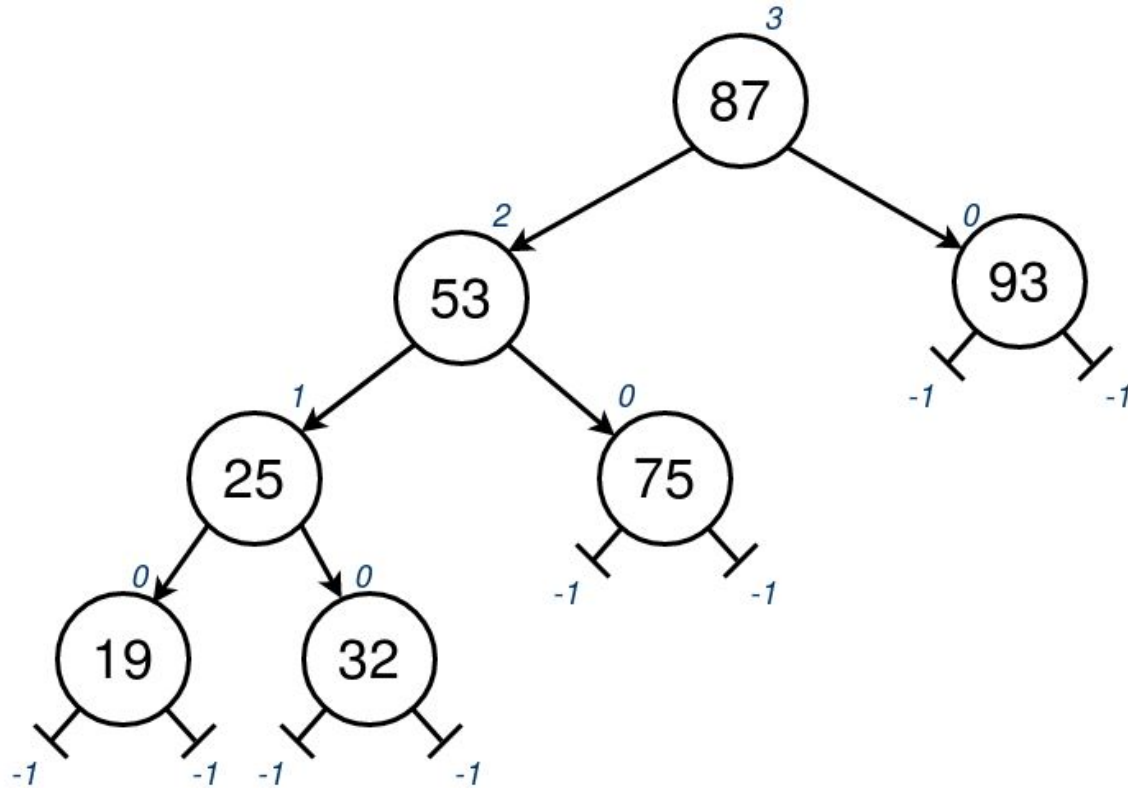
- Рекурсивний алгоритм для обходу бінарного дерева пошуку в порядку зростання
 - Обійти ліве піддерево
 - Розглянути значення у поточному вузлі
 - Обійти праву частину



Висота дерева, висота вузла

- Висота дерева - це довжина найдовшого шляху від кореня цього дерева до одного з його листків
- Висота вузла - це довжина найдовшого шляху від цього вузла до одного з його листків
- Висота листка дорівнює нулю
- Для зручності, вважається, що висота лівого і правого піддерев листка дорівнює мінус одиниці
- Висоту вузла можна порахувати рекурсивно:
 - $\text{height}(\text{node}) = \max(\text{height}(\text{node.left}), \text{height}(\text{node.right})) + 1$

Висота дерева, висота вузла



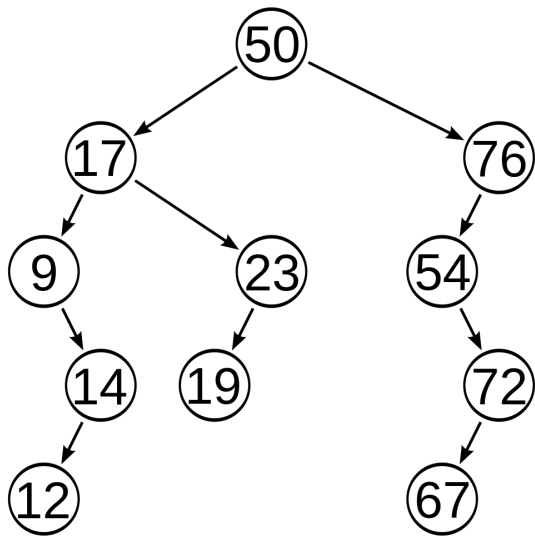
Збалансоване дерево

- Збалансованим (самозбалансованим) вважається таке дерево, в якому висота є пропорційною логарифму від кількості елементів у дереві
- Так як операції пошуку, вставки і видалення пропорційні висоті дерева, то в збалансованому дереві вони будуть пропорційні логарифму
- Найбільш поширені види самозбалансованих дерев:
 - AVL-дерево
 - Червоно-чорне дерево
 - B-дерево

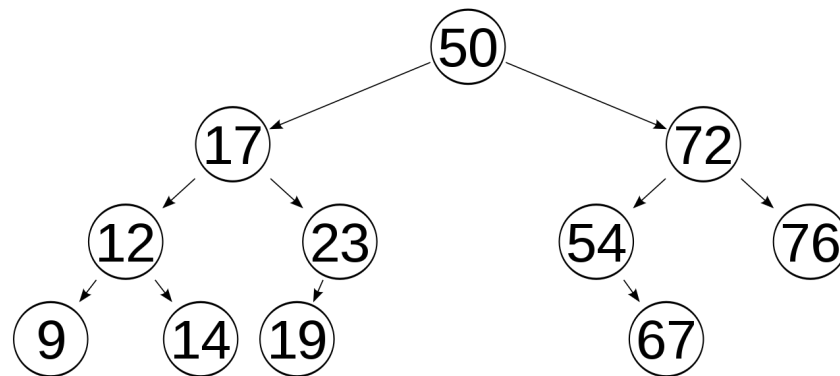
AVL-дерево

- Один із видів самозбалансованих дерев
- Назва походить від перших букв вчених-творців (Адельсон-Вельського Георгія Максимовича і Ландіса Євгена Михайловича)
- Основна властивість AVL-дерева - для кожного вузла висота лівого і правого піддерева відрізняється не більше ніж на одиницю

AVL-дерево



Незбалансоване дерево

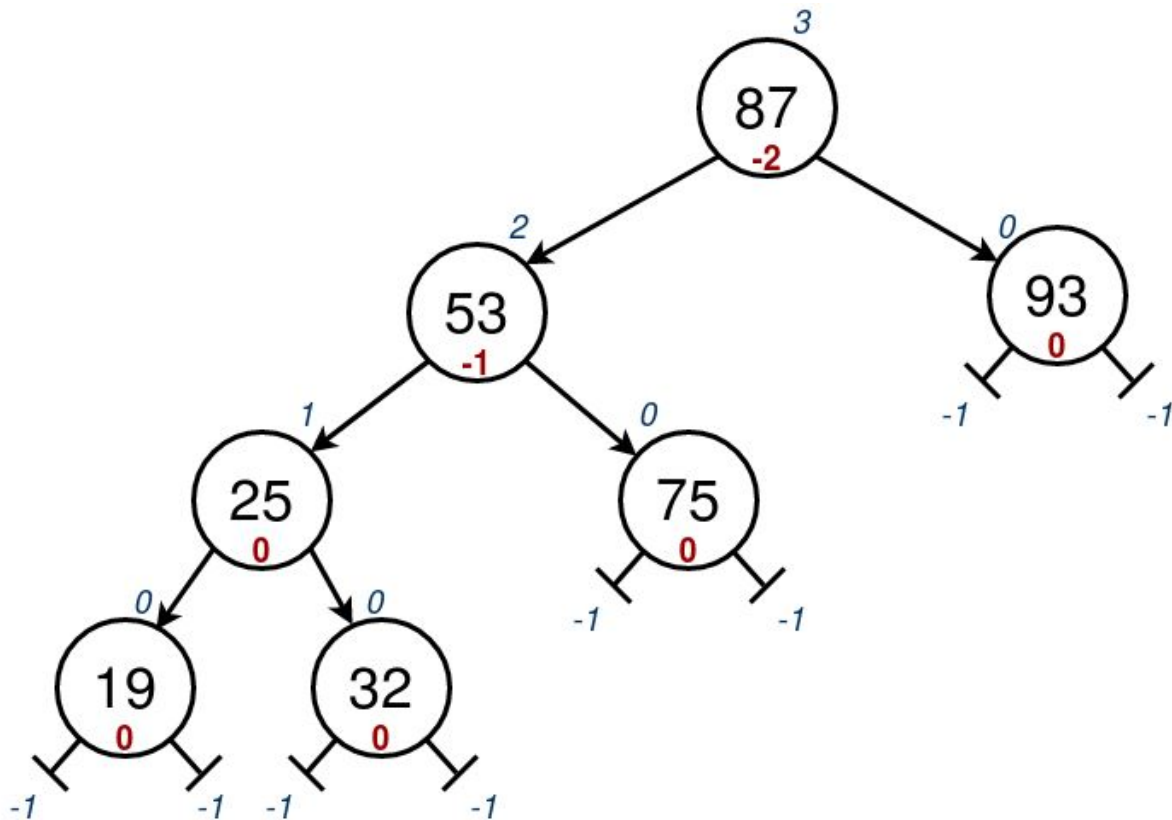


Збалансоване дерево, в
якому виконується основна
властивість AVL

Баланс вузла дерева

- Баланс вузла дерева - це різниця висот його правого і лівого піддерев
- $\text{balance} = \text{node.right.height} - \text{node.left.height}$
- В AVL-дерева баланс кожного вузла має бути в межах від $[-1; 1]$
- Якщо баланс від'ємний, то вважається, що вузол перехилений ліворуч; якщо додатній - вузол перехилений праворуч

Баланс узла дерева



Висота AVL-дерева

- Доведемо, що висота AVL-дерева буде пропорційна логарифму
- Для цього знайдемо спочатку мінімальну можливу кількість вузлів, що може бути в дереві висотою h :

$$N(h) = N(h - 1) + N(h - 2) + 1$$

$N(h) > F(h)$, де $F(h)$ - це число Фібоначчі з індексом h

$$F(h) \approx \frac{\phi^h}{\sqrt{5}}$$

$$N(h) > \frac{\phi^h}{\sqrt{5}}$$

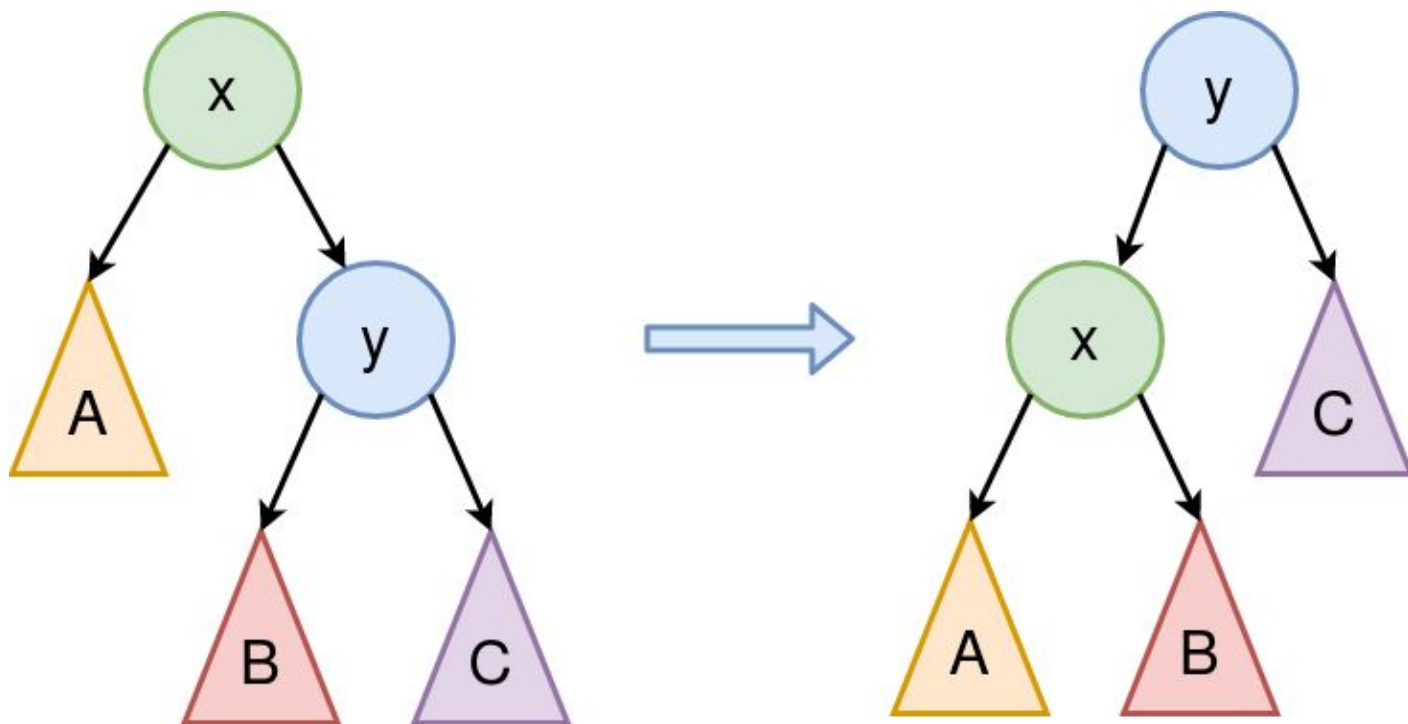
$$h < 1.45 \cdot \log_2(N)$$

Додавання елемента в AVL-дерево

1. Додати новий елемент таким же чином, як і в звичайне дерево
2. Перевірити чи виконується для кожного з вузлів-предків нового листка властивість AVL-дерева. Якщо ні - зробити перебалансування вузла за допомогою поворотів

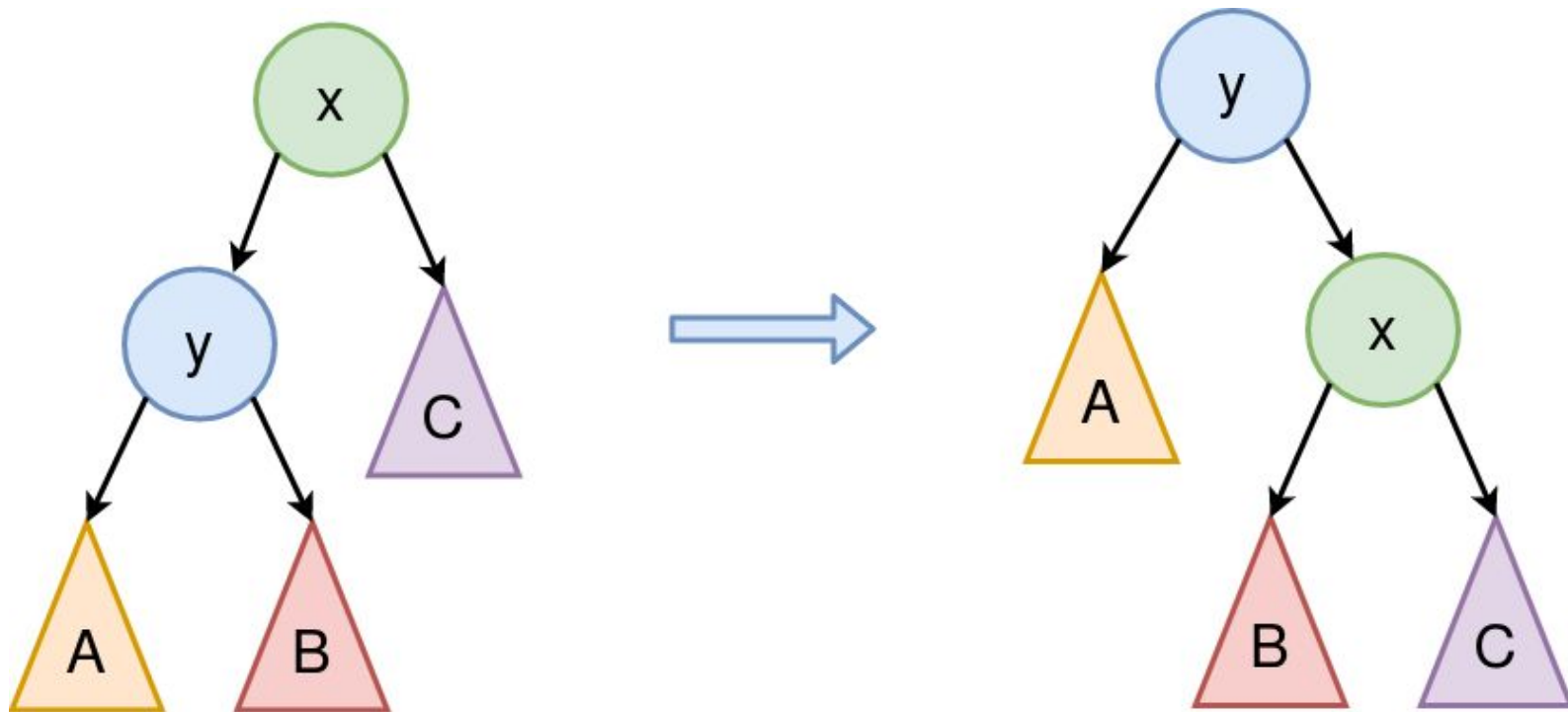
Повороти дерева

- Лівий поворот вузла x :



Повороты дерева

- Правый поворот узла x :



Алгоритм балансування вузла

1. Після вставки нового елемента, баланс кожного з його вузлів-предків може бути у межах $[-2; 2]$. Якщо він знаходиться в межах $[-1; 1]$, то вузол вважається збалансованим, якщо ні, то можливий один із чотирьох випадків
 - a. Баланс поточного вузла дорівнює -2 , баланс лівого піддерева дорівнює -1 => необхідно зробити правий поворот поточного вузла
 - b. Баланс поточного вузла дорівнює -2 , баланс лівого піддерева дорівнює 1 => необхідно зробити лівий поворот лівого піддерева, а потім - правий поворот поточного вузла
 - c. Баланс поточного вузла дорівнює 2 , баланс правого піддерева дорівнює 1 => необхідно зробити лівий поворот поточного вузла
 - d. Баланс поточного вузла дорівнює 2 , баланс правого піддерева дорівнює -1 => необхідно зробити правий поворот правого піддерева, а потім - лівий поворот поточного вузла

Видалення елемента з AVL-дерева

1. Знайти відповідний вузол
2. Далі можливий один з трьох випадків:
 - a. Знайдений вузол є листком - в даному випадку його треба видалити і зробити балансування всіх вузлів-предків
 - b. У знайденого вузла є тільки один дочірній вузол - в цьому випадку потрібно видалити знайдений вузол, розмістити на його місці дочірній вузол і зробити балансування всіх вузлів-предків знайденого
 - c. У знайденого вузла є обидва дочірні вузли - потрібно визначити найбільший елемент лівого піддерева, змінити ці два вузли місцями, видалити знайдений елемент вже на новому місці і зробити балансування всіх вузлів-предків

Посилання на матеріали

1. <https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-006-introduction-to-algorithms-fall-2011/lecture-videos/lecture-6-avl-trees-avl-sort/>
2. <http://www.mathcs.emory.edu/~cheung/Courses/323/Syllabus/Trees/AVL-delete.html>
3. <https://uk.wikipedia.org/wiki/%D0%90%D0%92%D0%9B-%D0%B4%D0%B5%D1%80%D0%B5%D0%B2%D0%BE>
4. <https://www.cs.usfca.edu/~galles/visualization/AVLtree.html>