18. AVL Trees

Balanced Trees [Ottman/Widmayer, Kap. 5.2-5.2.1, Cormen et al, Kap. Problem 13-3]

Background

- Search tree: Search, insertion and removal of a key in average in $\mathcal{O}(\log n)$ steps (given n keys in the tree)
- Worst case, though: $\Theta(n)$ (degenerated tree)

Goal: Avoid degeneration, by balancing the tree after each update operation.

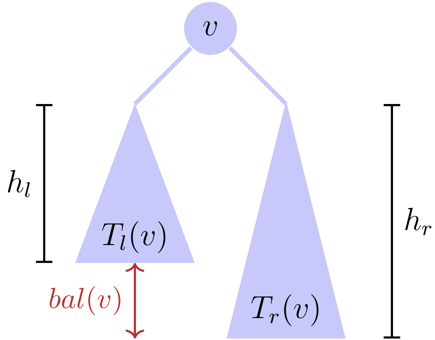
Balancing: guarantee that a tree with n nodes always has a height of $\mathcal{O}(\log n)$.

Adelson-Velsky and Landis (1962): AVL-Trees

Balance of a node

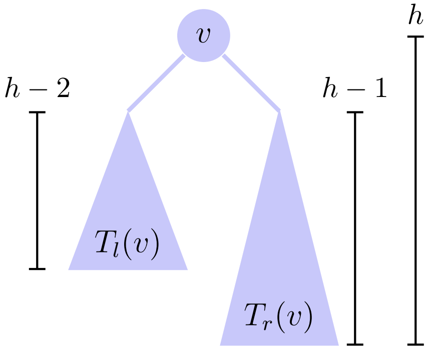
The *balance* of a node v is defined as the height difference of its sub-trees $T_l(v)$ and $T_r(v)$

$$\mathrm{bal}(v) := h(T_r(v)) - h(T_l(v))$$

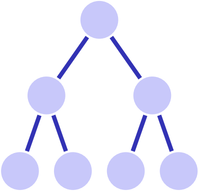


AVL Condition

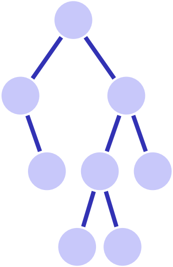
AVL Condition: for each node v of a tree $\mathrm{bal}(v) \in \{-1, 0, 1\}$



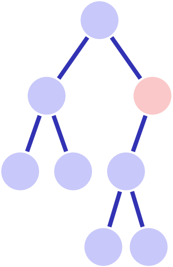
(Counter-)Examples



AVL tree with height 3



AVL tree with height 4

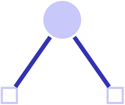


No AVL tree

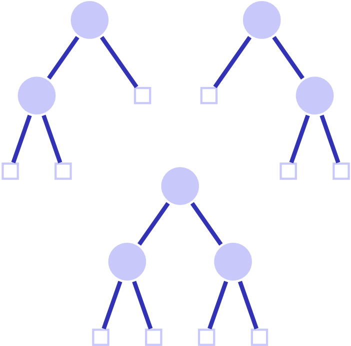
Number of Leaves

- 1. observation: a binary tree with n keys provides exactly $n+1$ leaves. Simple induction argument.
  - The binary tree with $n = 0$ keys has $m = 1$ leaves
  - When a key is added ($n \to n+1$), then it replaces a leaf and adds two new leafs ($m \to m - 1 + 2 = m + 1$).
- 2. observation: a lower bound of the number of leaves in a binary tree with given height implies an upper bound of the height of a binary tree with given number of keys.

Lower bound of the leaves



AVL tree with height 1 has
$N(1) := 2$ leaves.



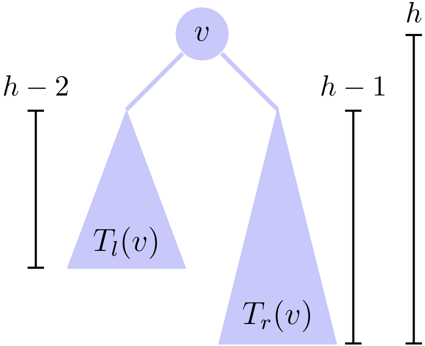
AVL tree with height 2 has at
least $N(2) := 3$ leaves.

Lower bound on the leaves for $h > 2$ in AVL trees

- Height of one subtree $\geq h - 1$.
- Height of the other subtree $\geq h - 2$.

Minimal number of leaves $N(h)$ is

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



Overal we have $N(h) = F_{h+2}$ with **Fibonacci-numbers** $F_0 := 0$, $F_1 := 1$, $F_n := F_{n-1} + F_{n-2}$ for $n > 1$.

Fibonacci Numbers, closed Form

It holds that[20]

$$F_i = \frac{1}{\sqrt{5}}(\phi^i - \hat{\phi}^i)$$

with the roots $\phi, \hat{\phi}$ of the golden ratio equation $x^2 - x - 1 = 0$:

$$\phi = \frac{1+\sqrt{5}}{2} \approx 1.618$$
$$\hat{\phi} = \frac{1-\sqrt{5}}{2} \approx -0.618$$

[20] Derivation using generating functions (power series) in the appendix.

Tree Height

Because $|\hat{\phi}| < 1$, overal we have

$$N(h) \in \Theta\left(\left(\frac{1+\sqrt{5}}{2}\right)^h\right) \subseteq \Omega(1.618^h)$$

and thus

$$N(h) \geq c \cdot 1.618^h \quad \Rightarrow \quad h \leq 1.44 \log_2 n + c'.$$

- I.e. an AVL tree has, as desired, a height of $\mathcal{O}(\log n)$
- and is asymptotically not more than 44% higher than a perfectly balanced tree (height $\lceil \log_2 n + 1 \rceil$)

Insertion and Balancing

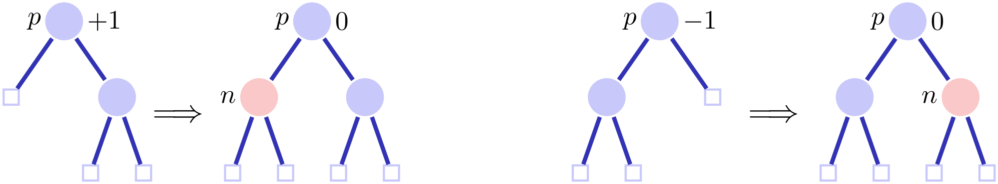
Balance:

- Insertion potentially violates AVL condition $\rightarrow$ balancing
- For that, we store the balance in each node

Insert:

- Insert new node n, as done for search trees
- Check, and potentially restore, balance of all nodes from n upwards to the root

Balance at Insertion Point

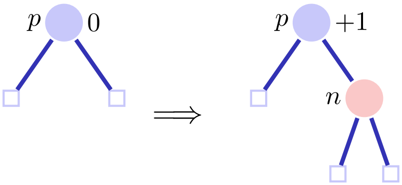


case 1: $\mathrm{bal}(p) = +1$

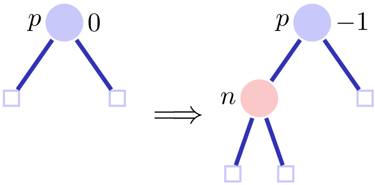
case 2: $\mathrm{bal}(p) = -1$

Directly done in both cases because the height of subtree p did not change. Balance of parent node thus also unchanged.

Balance at Insertion Point



case 3.1: $\mathrm{bal}(p) = 0$ right



case 3.2: $\mathrm{bal}(p) = 0$, left

Not yet done in both case, since parent node potentially no longer balanced $\rightarrow$ Invocation of function **`upin(p)`** (upwards + insert)

upin(p): Recursive Invocation Requirement

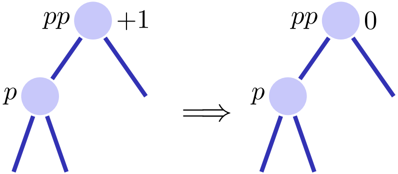
For every call **`upin(p)`** it must hold that

- the subtree p grew and thereby
- changed $\mathrm{bal}(p)$ from 0 to $\in \{-1, +1\}$.

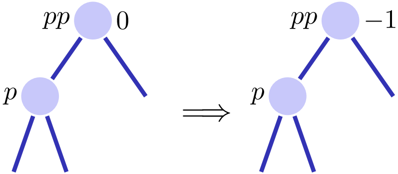
Because only in this situation can the newly developed imbalance of p $(\mathrm{bal}(p) \neq 0)$ affect the tree structure above.

upin(p)

Assumption: p is left son of pp[21]



case 1: $\text{bal}(pp) = +1$, done.



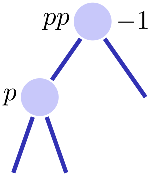
case 2: $\text{bal}(pp) = 0$, **`upin(pp)`**

In both cases the AVL-Condition holds for the subtree from pp

[21] If p is a right son: symmetric cases with exchange of $+1$ and -1

upin(p)

Assumption: p is left son of pp



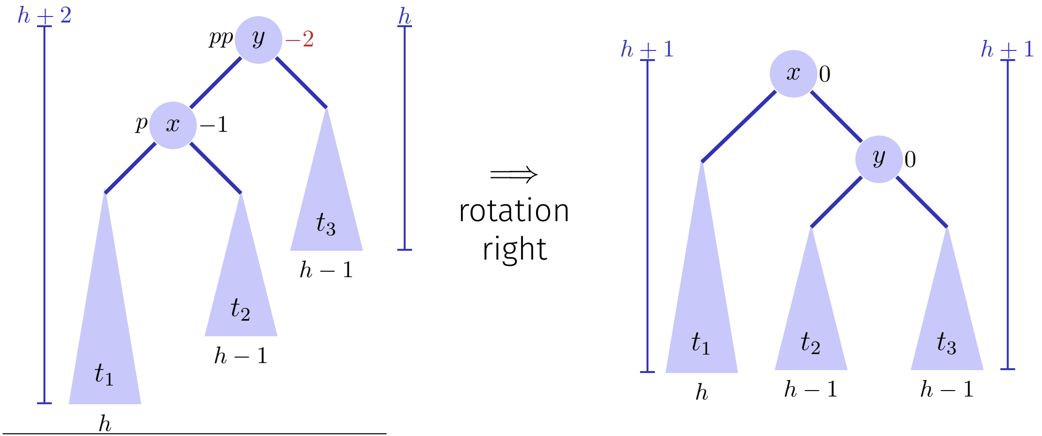
case 3: $\mathrm{bal}(pp) = -1$,

This case is problematic: adding n to the subtree from pp has violated the AVL-condition. Re-balance!

Two cases $\mathrm{bal}(p) = -1$, $\mathrm{bal}(p) = +1$

Rotations

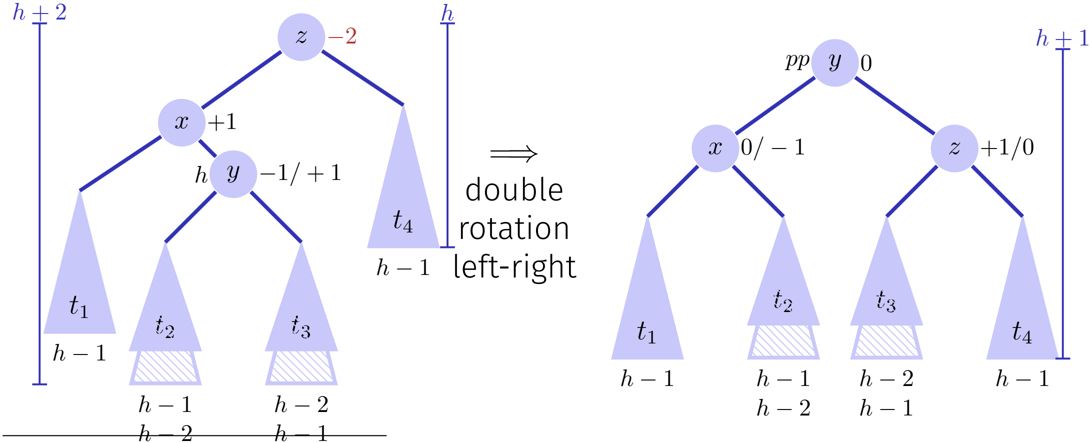
case 1.1 $\text{bal}(p) = -1.$ [22]



[22] p right son: $\Rightarrow \text{bal}(pp) = \text{bal}(p) = +1$, left rotation

Rotations

case 1.2 $\text{bal}(p) = +1$. [23]



[23] p right son $\Rightarrow \text{bal}(pp) = +1$, $\text{bal}(p) = -1$, double rotation right left

Analysis

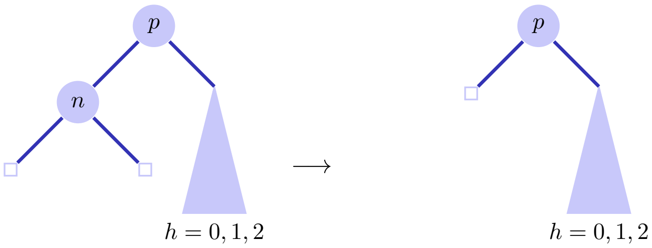
- Tree height: $\mathcal{O}(\log n)$.
- Insertion like in binary search tree.
- Balancing via recursion from node to the root (during recursive ascend). Maximal path lenght $\mathcal{O}(\log n)$.

Insertion in an AVL-tree provides run time costs of $\mathcal{O}(\log n)$.

Deletion

Case 1: Children of node n are both leaves.
Let p be parent node of $n \Rightarrow$ Other subtree has height $h' = 0$, 1 or 2.

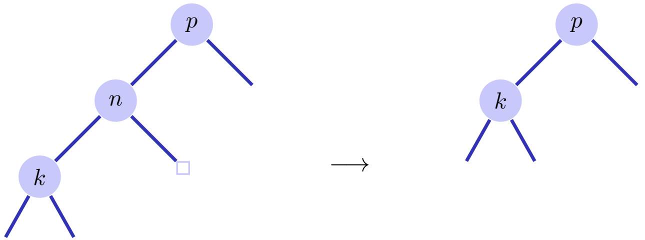
- $h' = 1$: Adapt $\text{bal}(p)$.
- $h' = 0$: Adapt $\text{bal}(p)$. Call **`upout(p)`**.
- $h' = 2$: Rebalanciere des Teilbaumes. Call **`upout(p)`**.



Deletion

Case 2: one child k of node n is an inner node

- Replace n by k. **`upout(k)`**



Deletion

Case 3: both children of node n are inner nodes

- Replace n by symmetric successor $\Rightarrow$ **`upout(k)`**
- Deletion of the symmetric successor is as in case 1 or 2.

`upout(p)`

Let pp be the parent node of p.

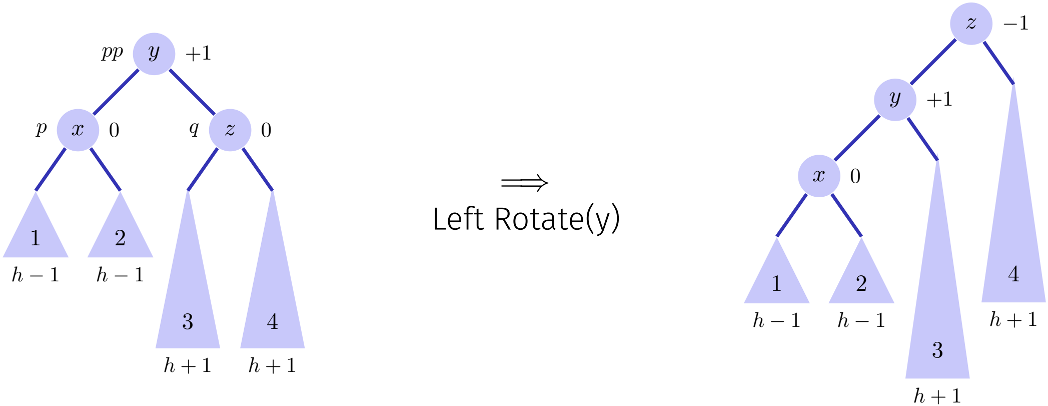
(a) p left child of pp

1. $\text{bal}(pp) = -1 \;\Rightarrow\; \text{bal}(pp) \leftarrow 0$. **`upout(pp)`**
2. $\text{bal}(pp) = 0 \;\Rightarrow\; \text{bal}(pp) \leftarrow +1$.
3. $\text{bal}(pp) = +1 \Rightarrow$ next slides.

(b) p right child of pp: Symmetric cases exchanging $+1$ and -1.

upout(p)

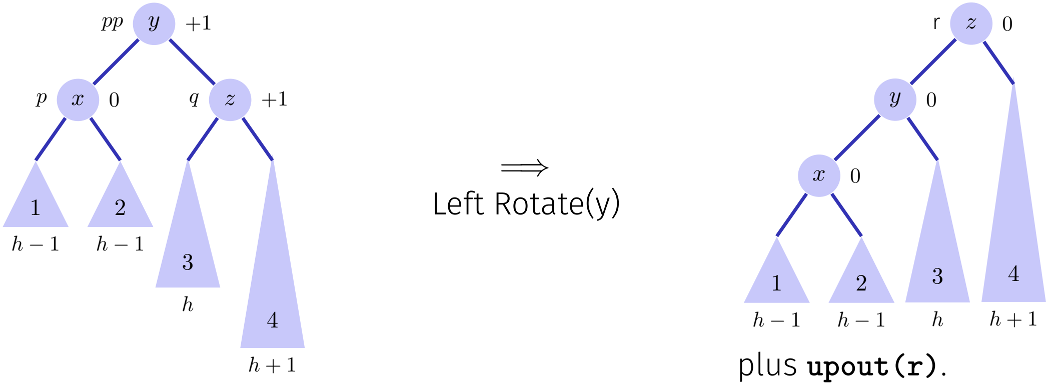
Case (a).3: $\text{bal}(pp) = +1$. Let q be brother of p
(a).3.1: $\text{bal}(q) = 0$.[24]



[24] (b).3.1: $\text{bal}(pp) = -1$, $\text{bal}(q) = 0$, Right rotation

upout(p)

Case (a).3: $\mathrm{bal}(pp) = +1$. (a).3.2: $\mathrm{bal}(q) = +1$.[25]



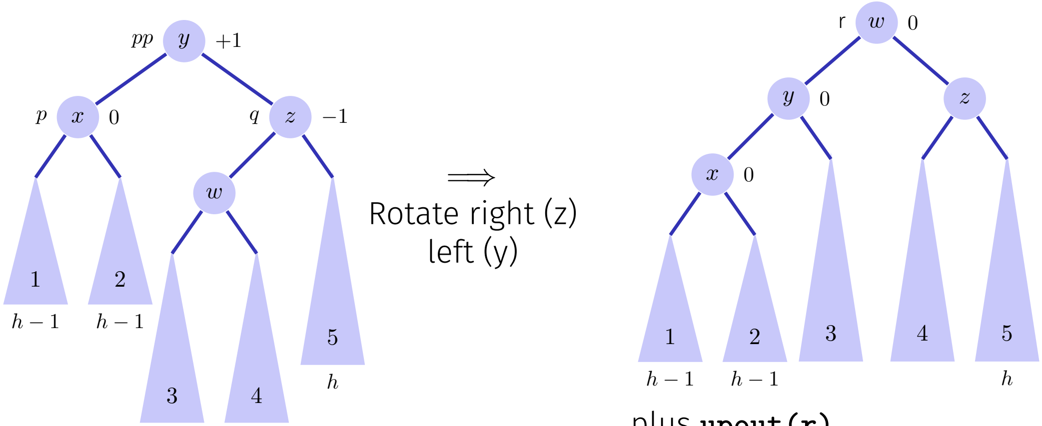
plus **upout(r)**.

---

[25] (b).3.2: $\mathrm{bal}(pp) = -1$, $\mathrm{bal}(q) = +1$, Right rotation+upout

upout(p)

Case (a).3: $\text{bal}(pp) = +1$. (a).3.3: $\text{bal}(q) = -1$.[26]



plus **upout(r)**.

[26] (b).3.3: $\text{bal}(pp) = -1$, $\text{bal}(q) = -1$, left-right rotation + upout

Conclusion

- AVL trees have worst-case asymptotic runtimes of $\mathcal{O}(\log n)$ for searching, insertion and deletion of keys.
- Insertion and deletion is relatively involved. For small trees (key sets), the costs of balancing outweighs the gain of $\mathcal{O}(\log n)$ height.
- Several other balanced trees exist: Red-Black tree (**`std::map`** in $\mathrm{C++}$), B-tree (**`std::collections::BTreeMap`** in Rust), Splay tree; Treap (balanced with high probability)

18.6 Appendix

Derivation of some mathemmatical formulas

Fibonacci Numbers, Inductive Proof

$F_i \stackrel{!}{=} \frac{1}{\sqrt{5}}(\phi^i - \hat{\phi}^i) \qquad [*]$ $\quad \left(\phi = \frac{1+\sqrt{5}}{2}, \hat{\phi} = \frac{1-\sqrt{5}}{2}\right).$

1. Immediate for $i = 0, i = 1$.
2. Let $i > 2$ and claim $[*]$ true for all F_j, $j < i$.

$$F_i \stackrel{def}{=} F_{i-1} + F_{i-2} \stackrel{[*]}{=} \frac{1}{\sqrt{5}}(\phi^{i-1} - \hat{\phi}^{i-1}) + \frac{1}{\sqrt{5}}(\phi^{i-2} - \hat{\phi}^{i-2})$$

$$= \frac{1}{\sqrt{5}}(\phi^{i-1} + \phi^{i-2}) - \frac{1}{\sqrt{5}}(\hat{\phi}^{i-1} + \hat{\phi}^{i-2}) = \frac{1}{\sqrt{5}}\phi^{i-2}(\phi + 1) - \frac{1}{\sqrt{5}}\hat{\phi}^{i-2}(\hat{\phi} + 1)$$

($\phi, \hat{\phi}$ fulfil $x + 1 = x^2$)

$$= \frac{1}{\sqrt{5}}\phi^{i-2}(\phi^2) - \frac{1}{\sqrt{5}}\hat{\phi}^{i-2}(\hat{\phi}^2) = \frac{1}{\sqrt{5}}(\phi^i - \hat{\phi}^i).$$

[Fibonacci Numbers: closed form]

Closed form of the Fibonacci numbers: computation via generation functions:

1. Power series approach

$$f(x) := \sum_{i=0}^{\infty} F_i \cdot x^i$$

[Fibonacci Numbers: closed form]

2. For Fibonacci Numbers it holds that $F_0 = 0$, $F_1 = 1$, $F_i = F_{i-1} + F_{i-2} \; \forall\, i > 1$. Therefore:

$$\begin{aligned}
f(x) &= x + \sum_{i=2}^{\infty} F_i \cdot x^i = x + \sum_{i=2}^{\infty} F_{i-1} \cdot x^i + \sum_{i=2}^{\infty} F_{i-2} \cdot x^i \\
&= x + x \sum_{i=2}^{\infty} F_{i-1} \cdot x^{i-1} + x^2 \sum_{i=2}^{\infty} F_{i-2} \cdot x^{i-2} \\
&= x + x \sum_{i=0}^{\infty} F_i \cdot x^i + x^2 \sum_{i=0}^{\infty} F_i \cdot x^i \\
&= x + x \cdot f(x) + x^2 \cdot f(x).
\end{aligned}$$

[Fibonacci Numbers: closed form]

3. Thus:

$$f(x) \cdot (1 - x - x^2) = x.$$

$$\Leftrightarrow \quad f(x) = \frac{x}{1 - x - x^2} = -\frac{x}{x^2 + x - 1}$$

with the roots $-\phi$ and $-\hat{\phi}$ of $x^2 + x - 1$,

$$\phi = \frac{1 + \sqrt{5}}{2} \approx 1.6, \qquad \hat{\phi} = \frac{1 - \sqrt{5}}{2} \approx -0.6.$$

it holds that $\phi \cdot \hat{\phi} = -1$ and thus

$$f(x) = -\frac{x}{(x + \phi) \cdot (x + \hat{\phi})} = \frac{x}{(1 - \phi x) \cdot (1 - \hat{\phi} x)}$$

[Fibonacci Numbers: closed form]

4. It holds that:

$$(1-\hat{\phi}x)-(1-\phi x)=\sqrt{5}\cdot x.$$

Damit:

$$\begin{aligned} f(x) &= \frac{1}{\sqrt{5}}\frac{(1-\hat{\phi}x)-(1-\phi x)}{(1-\phi x)\cdot(1-\hat{\phi}x)} \\ &= \frac{1}{\sqrt{5}}\left(\frac{1}{1-\phi x}-\frac{1}{1-\hat{\phi}x}\right) \end{aligned}$$

[Fibonacci Numbers: closed form]

5. Power series of $g_a(x) = \frac{1}{1-a\cdot x}$ ($a \in \mathbb{R}$):

$$\frac{1}{1-a\cdot x} = \sum_{i=0}^{\infty} a^i \cdot x^i.$$

E.g. Taylor series of $g_a(x)$ at $x = 0$ or like this: Let $\sum_{i=0}^{\infty} G_i \cdot x^i$ a power series of g. By the identity $g_a(x)(1 - a\cdot x) = 1$ it holds that for all x (within the radius of convergence)

$$1 = \sum_{i=0}^{\infty} G_i \cdot x^i - a \cdot \sum_{i=0}^{\infty} G_i \cdot x^{i+1} = G_0 + \sum_{i=1}^{\infty} (G_i - a \cdot G_{i-1}) \cdot x^i$$

For $x = 0$ it follows $G_0 = 1$ and for $x \neq 0$ it follows then that $G_i = a \cdot G_{i-1} \Rightarrow G_i = a^i$.

[Fibonacci Numbers: closed form]

6. Fill in the power series:

$$f(x) = \frac{1}{\sqrt{5}}\left(\frac{1}{1-\phi x} - \frac{1}{1-\hat{\phi}x}\right) = \frac{1}{\sqrt{5}}\left(\sum_{i=0}^{\infty} \phi^i x^i - \sum_{i=0}^{\infty} \hat{\phi}^i x^i\right)$$
$$= \sum_{i=0}^{\infty} \frac{1}{\sqrt{5}}(\phi^i - \hat{\phi}^i)x^i$$

Comparison of the coefficients with $f(x) = \sum_{i=0}^{\infty} F_i \cdot x^i$ yields

$$F_i = \frac{1}{\sqrt{5}}(\phi^i - \hat{\phi}^i).$$