14

The Master Method

Analiza Algoritmilor

Image: https://www.swiss-knife.com/en/gifts-1/corporate-gifts/victorinox-spartan-black.html

Merge sort complexity

```
merge_sort(A, n):
1.      if (n <= 1)
2.          return A

3.      middle = n / 2
4.      A_left = merge_sort(A, middle)
5.      A_right = merge_sort(A + middle, n - middle)
6.      return merge(A_left, middle, A_right, n - middle)
```

$$T(n) = 2T\left(\frac{n}{2}\right) + \Theta(n)$$

Master method applicability

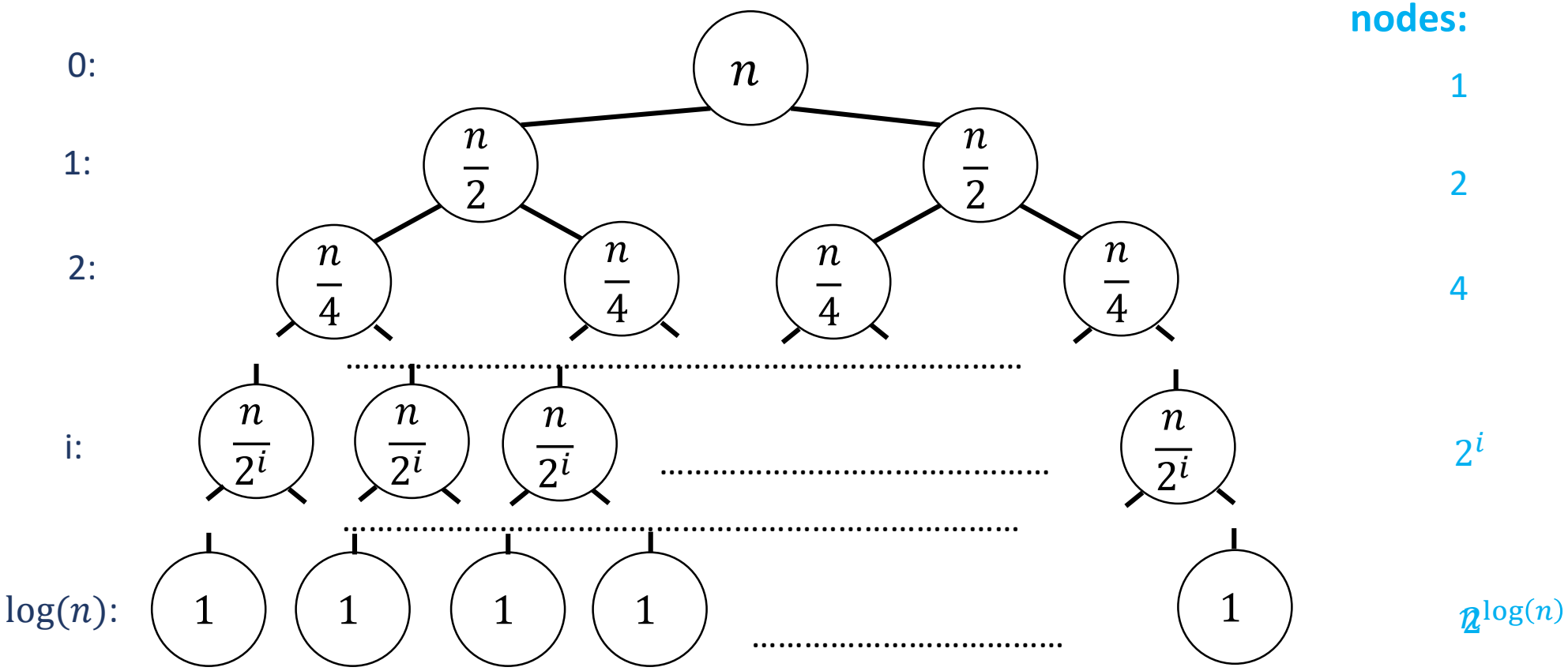
$$T(n) = aT\left(\frac{n}{b}\right) + f(n)$$

a: number of subproblems, $a \geq 1$

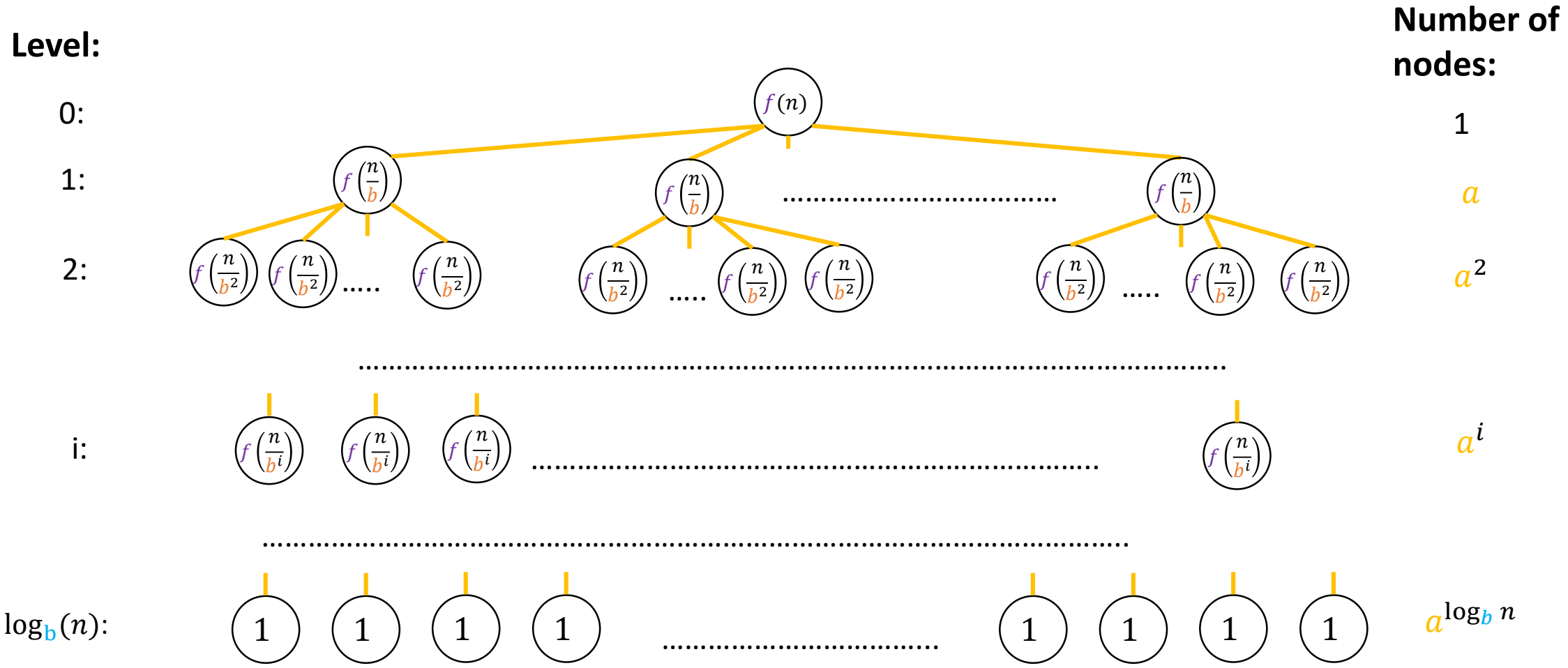
b: inverse of subproblems size, $b > 1$

$f(n)$: amount of work to combine the solutions to subproblems, $f(n) \geq 0$

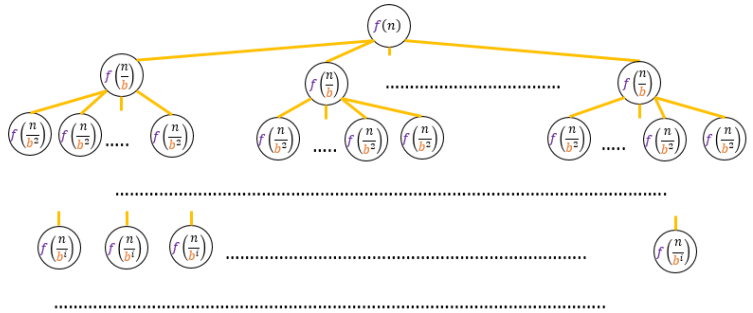
Recursion tree of Merge sort

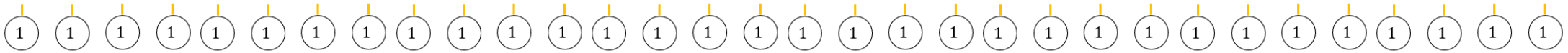


Generalized recursion tree



Leaf-heavy trees

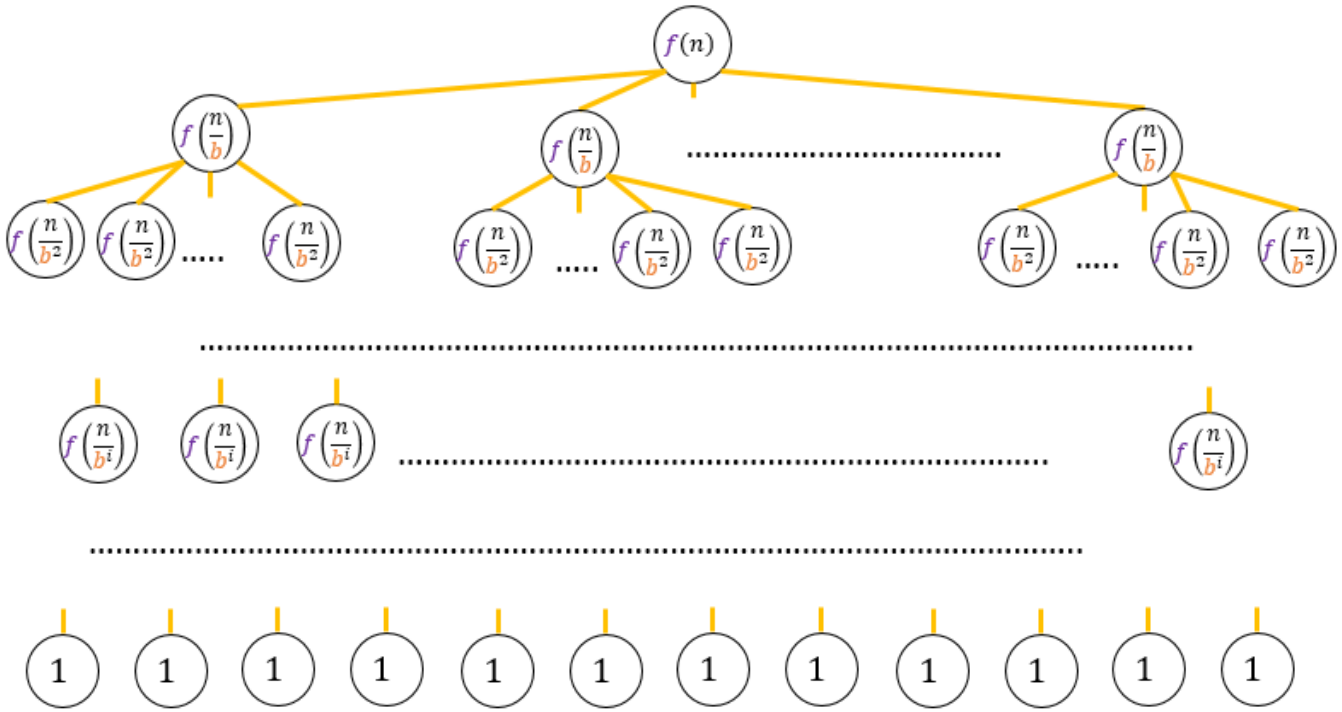




$$f(n) \in O(n^d), \qquad d < \log_b a$$

$$\Rightarrow T(n) \in \Theta(n^{\log_b a})$$

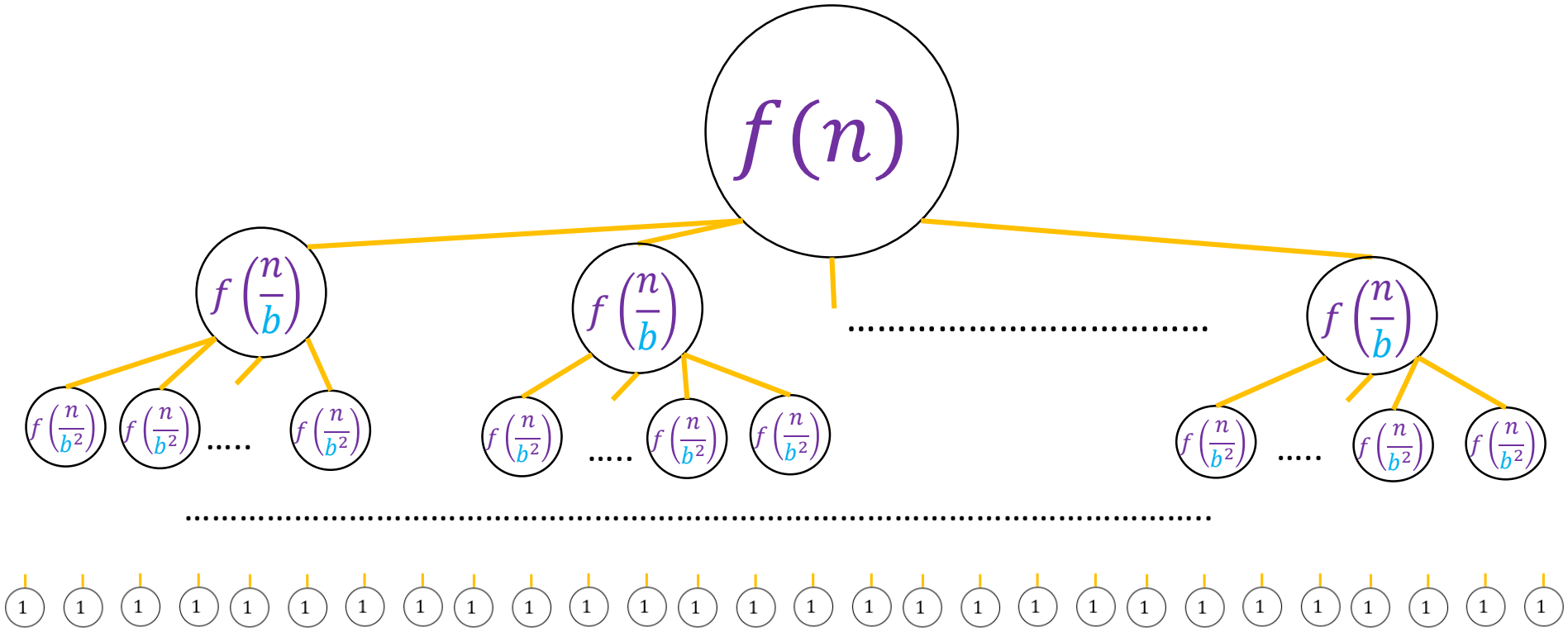
Balanced trees



$$f(n) \in \Theta\left(n^{\log_b a}\right)$$

$$\Rightarrow T(n) \in \Theta(n^{\log_b a} \log_b n)$$

Root-heavy trees



$$\exists k < 1, \qquad af\left(\frac{n}{b}\right) \leq kf(n)$$

$$\Rightarrow T(n) \in \Theta(f(n))$$

The Master Method

Case #	Condition	Result
1	$f(n) \in O(n^d), \qquad d < \log_b a$	$T(n) \in \Theta(n^{\log_b a})$
2	$f(n) \in \Theta(n^{\log_b a})$	$T(n) \in \Theta(n^{\log_b a} \log_b n)$
3	$\exists k < 1, \qquad af\left(\frac{n}{b}\right) \leq kf(n)$	$T(n) \in \Theta(f(n))$