

## Class work: Recurrence example

Consider the following recurrence which is very similar to the one for Mergesort:

$$T(n) = 3T(n/3) + n$$

Solve it (i.e. give a  $\Theta()$ -bound for  $T(n)$ ).

**Answer:**

$$\begin{aligned} T(n) &= 3T(n/3) + n \\ &= 3(3T(n/9) + n/3) + n \\ &= 9T(n/9) + 2n \\ &= 9(3T(n/27) + n/9) + 2n \\ &= 27T(n/27) + 3n \\ &= \dots \\ &= 3^i T(n/3^i) + i \cdot n \end{aligned}$$

- Recursion depth: How long (how many iterations) it takes until the subproblem has constant size?  $i$  iterations, where  $\frac{n}{3^i} = 1 \Rightarrow i = \log_3 n$
- What is the last term?  $3^i T(1) = 3^{\log_3 n} T(1) = n \cdot 1 = n$

So we have:

$$\begin{aligned} T(n) &= 3^{\log_3 n} T(1) + \log_3 n \cdot n \\ &= n + \log_3 n \cdot n \\ &= \Theta(n \lg n) \end{aligned}$$