5 Asymptotic Notation

We are usually not interested in exact running times, but only in an asymptotic classification of the running time, that ignores constant factors and constant additive offsets.

- We are usually interested in the running times for large values of n. Then constant additive terms do not play an important role.
- An exact analysis (e.g. *exactly* counting the number of operations in a RAM) may be hard, but wouldn't lead to more precise results as the computational model is already quite a distance from reality.
- A linear speed-up (i.e., by a constant factor) is always possible by e.g. implementing the algorithm on a faster machine.
- Running time should be expressed by simple functions.

```
Ernst Mayr, Harald Räcke
```

Asymptotic Notation

ΠЦΓ

There is an equivalent definition using limes notation (assuming that the respective limes exists). f and g are functions from \mathbb{N}_0 to \mathbb{R}_0^+ .

• $g \in O(f)$: $\lim_{n \to \infty} \frac{1}{f(n)} = 0$ dau notation de	version of the Lan- fined here, we as g are positive func
► $g \in \mathcal{O}(f)$: $0 \le \lim_{n \to \infty} \frac{g(n)}{f(n)} < \infty$ ► $g \in \Omega(f)$: $0 < \lim_{n \to \infty} \frac{g(n)}{f(n)} \le \infty$ g(n)	
g(n)	

Asymptotic Notation

Formal Definition

Let f, g denote functions from \mathbb{N} to \mathbb{R}^+ .

- $\mathcal{O}(f) = \{g \mid \exists c > 0 \ \exists n_0 \in \mathbb{N}_0 \ \forall n \ge n_0 : [g(n) \le c \cdot f(n)]\}$ (set of functions that asymptotically grow not faster than f)
- $\Omega(f) = \{g \mid \exists c > 0 \ \exists n_0 \in \mathbb{N}_0 \ \forall n \ge n_0 \colon [g(n) \ge c \cdot f(n)]\}$ (set of functions that asymptotically grow not slower than f)
- Θ(f) = Ω(f) ∩ O(f)
 (functions that asymptotically have the same growth as f)
- ▶ $o(f) = \{g \mid \forall c > 0 \exists n_0 \in \mathbb{N}_0 \forall n \ge n_0 : [g(n) \le c \cdot f(n)]\}$ (set of functions that asymptotically grow slower than f)
- ► $\omega(f) = \{g \mid \forall c > 0 \exists n_0 \in \mathbb{N}_0 \forall n \ge n_0 : [g(n) \ge c \cdot f(n)]\}$ (set of functions that asymptotically grow faster than f)

החוחר	5 Asymptotic Notation	22. Feb. 2020
UUU Ernst Mayr, Harald Räcke		6/18

Asymptotic Notation

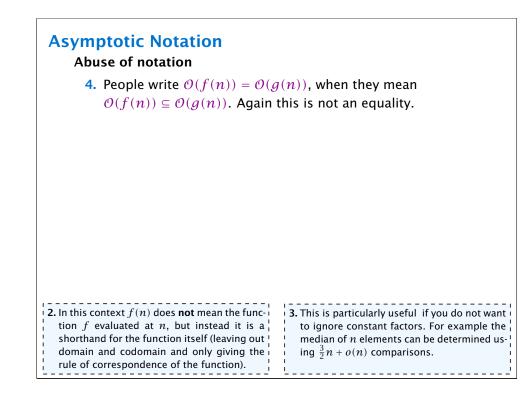
Abuse of notation

5/18

- 1. People write f = O(g), when they mean $f \in O(g)$. This is **not** an equality (how could a function be equal to a set of functions).
- 2. People write $f(n) = \mathcal{O}(g(n))$, when they mean $f \in \mathcal{O}(g)$, with $f : \mathbb{N} \to \mathbb{R}^+, n \mapsto f(n)$, and $g : \mathbb{N} \to \mathbb{R}^+, n \mapsto g(n)$.
- **3.** People write e.g. h(n) = f(n) + o(g(n)) when they mean that there exists a function $z : \mathbb{N} \to \mathbb{R}^+, n \mapsto z(n), z \in o(g)$ such that h(n) = f(n) + z(n).

2. In this context f(n) does **not** mean the function f evaluated at n, but instead it is a shorthand for the function itself (leaving out domain and codomain and only giving the rule of correspondence of the function).

3. This is particularly useful if you do not want to ignore constant factors. For example the median of n elements can be determined using $\frac{3}{2}n + o(n)$ comparisons.



Asymptotic Notation in Equations

How do we interpret an expression like:

 $2n^2 + \mathcal{O}(n) = \Theta(n^2)$

Regardless of how we choose the anonymous function $f(n) \in O(n)$ there is an anonymous function $g(n) \in O(n^2)$ that makes the expression true.

Ernst Mayr, Harald Räcke

5 Asymptotic Notation

22. Feb. 2020 10/18

Asymptotic Notation in Equations

How do we interpret an expression like:

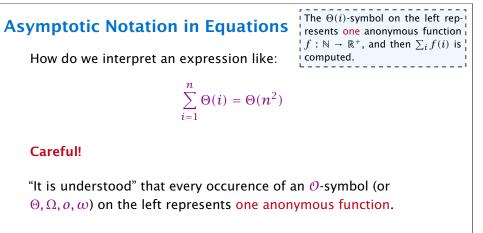
$$2n^2 + 3n + 1 = 2n^2 + \Theta(n)$$

Here, $\Theta(n)$ stands for an anonymous function in the set $\Theta(n)$ that makes the expression true.

Note that $\Theta(n)$ is on the right hand side, otw. this interpretation is wrong.

5 Asymptotic Notation

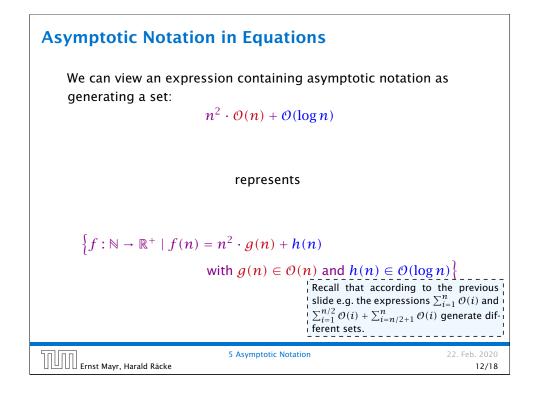
2. Feb. 2020 9/18

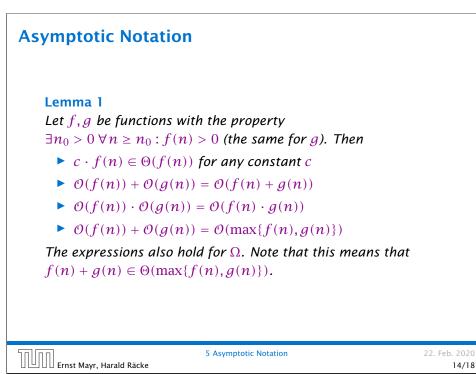


Hence, the left side is not equal to

Ernst May

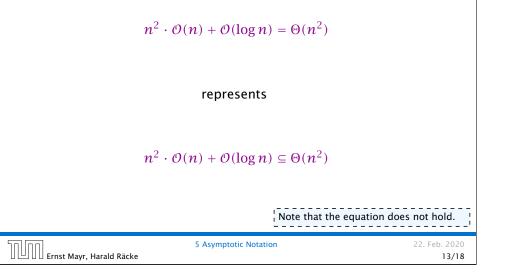
$\Theta(1) + \Theta(2) +$	$\cdots + \Theta(n-1) + \Theta(n)$ $\Theta(1) + \Theta(2) + \Theta(2)$ not really have tion.) $\cdots + \Theta(n-1) + \Theta(n)$ does e a reasonable interpreta-
5 A	symptotic Notation	22. Feb. 2020
r, Harald Räcke		11/18

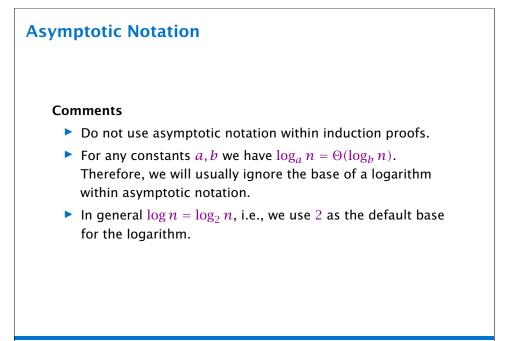




Asymptotic Notation in Equations

Then an asymptotic equation can be interpreted as containement btw. two sets:





Asymptotic Notation

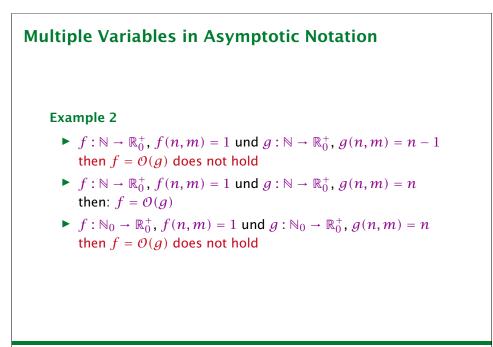
In general asymptotic classification of running times is a good measure for comparing algorithms:

- If the running time analysis is tight and actually occurs in practise (i.e., the asymptotic bound is not a purely theoretical worst-case bound), then the algorithm that has better asymptotic running time will always outperform a weaker algorithm for large enough values of n.
- However, suppose that I have two algorithms:
 - Algorithm A. Running time $f(n) = 1000 \log n = O(\log n)$.
 - Algorithm B. Running time $g(n) = \log^2 n$.

Clearly f = o(g). However, as long as $\log n \le 1000$ Algorithm B will be more efficient.

החוחל	5 Asymptotic
🛛 🕒 🛛 🖉 Ernst Mayr, Harald Räcke	

```
5 Asymptotic Notation
```



Multiple Variables in Asymptotic Notation

Sometimes the input for an algorithm consists of several parameters (e.g., nodes and edges of a graph (n and m)).

If we want to make asympotic statements for $n \to \infty$ and $m \to \infty$ we have to extend the definition to multiple variables.

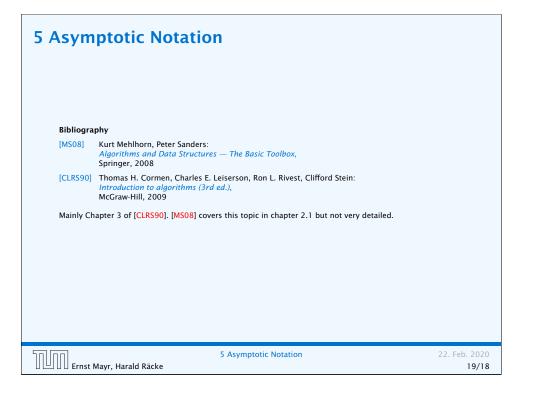
Formal Definition

Let f, g denote functions from \mathbb{N}^d to \mathbb{R}_0^+ .

• $\mathcal{O}(f) = \{g \mid \exists c > 0 \exists N \in \mathbb{N}_0 \forall \vec{n} \text{ with } n_i \ge N \text{ for some } i : [g(\vec{n}) \le c \cdot f(\vec{n})]\}$

(set of functions that asymptotically grow not faster than $f\)$

החוחר	5 Asymptotic Notation	22. Feb. 2020
🛛 🕒 🛛 Ernst Mayr, Harald Räcke		17/18



22. Feb. 2020 18/18

16/18