

Floor function

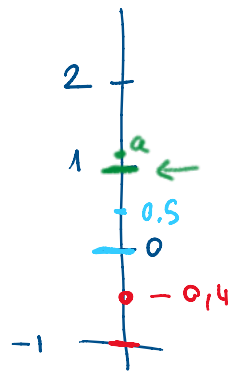
$$\lfloor \pi \rfloor = \lfloor 3.14159 \rfloor = \lfloor 3.1 \rfloor = 3$$

$$\lfloor 2 \rfloor = 2$$

$$\lfloor 1.99 \rfloor = 1$$

forall $x \in \mathbb{R}$

$$\lfloor x \rfloor := \{ n \in \mathbb{Z} \mid \text{biggest } n \text{ such that } n \leq x \}$$

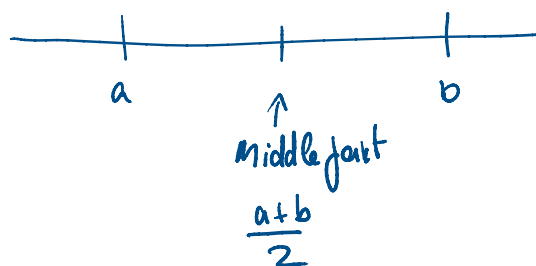


$$\lfloor a \rfloor = 1$$

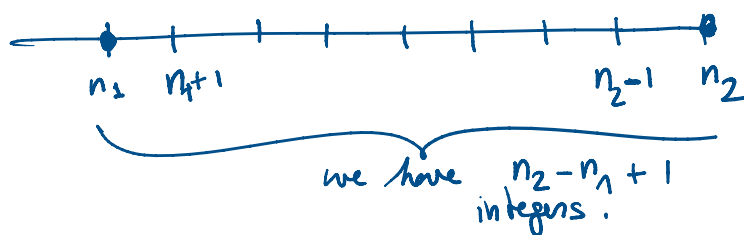
$$\lfloor 0.5 \rfloor = 0$$

$$\lfloor -0.4 \rfloor = -1$$

such that $n \leq x$
n as big as possible.

Middle point $a, b \in \mathbb{R}$ 

Number of integers in an interval
with end points



$$\sqrt{n} = n^{\frac{1}{2}} \in O(n^{1-\varepsilon}) \quad \text{for } \varepsilon > 0$$

example $\varepsilon = \frac{1}{2}$

$$2. \quad T(n) = \Theta(n^{\log_b a} \ln n) \text{ if } f(n) = \Theta(n^{\log_b a}).$$

ex. for $b = a = 2 \quad | \quad \log_2 2 = 1$

thus means

$$T(n) = \Theta(n \ln n) \quad \text{if } f(n) = \Theta(n)$$

Exercise 6 Work out the computational complexity of the following piece of code:

Algorithm

```

for i = n to 1 do
  for j = 1 to n do
    for k = 0 to n do
      ... // constant number of operations
      k = k + 2
    end for
    j = j * 2
  end for
  i = i / 2
end for

```

Handwritten annotations:

- Left of the code: $\log_2 n$ (bracketed next to the outer loop)
- Inside the inner loop: $\log_2 n$ (bracketed next to the inner loop)
- Right of the code: $\begin{bmatrix} c_1 \\ c_2 \end{bmatrix} \} \frac{n}{2} \text{-times}$
- Far right: $(c_1 + c_2) \cdot \frac{n}{2}$

Handwritten sequence: $1 \quad 2 \quad 4 \quad 8 \quad 16 \quad \dots \quad n$

Underneath the sequence: $\cong \log_2 n$

Handwritten sequence: $1 \quad n \quad \frac{n}{2} \quad \frac{n}{4} \quad \dots \quad 1$

Underneath the sequence: $\log_2 n$

$$L_2 = L_3 \cdot \log_2 n$$

$$L_1 = L_2 \cdot \log_2 n = L_3 \cdot \log_2 n \cdot \log_2 n$$

(