0. Uvod

0.1. Algoritem

1. Ali je algoritem pravilen?
2. Kako učinkovit je?
   - časovna zahtevnost
   - prostorska zahtevnost
   - paralelizabilnost
   - sprotnost

     online / off-line algoritem
3. Ali znamo še bolje?

Al Horizmi
* ~780 Hiva
† ≥846 Bagdad

0.2. Fibonaccijeva števila

$$F_0 = 0 \qquad F_1 = 1 \qquad F_n = F_{n-1} + F_{n-2}$$

vhod $n \in \mathbb{N}$

izhod F_n

```
def fib_1(n):
    če n = 0:
        vrni 0
    če n = 1:
        vrni 1
    sicer:
        vrni fib_1(n-1) + fib_1(n-2)
```

1. ✓
2. $O(2^n)$

$$T_1(n) = O(1) + T_1(n-1) + T_1(n-2)$$
$$T_1(0) = O(1)$$
$$T_1(1) = O(1)$$
$$T_1(n) = O(F_n)$$ z indukcijo

③ Da, znamo boljše

vhod $n \in \mathbb{N}$
izhod F_n

def $fib_2(n)$:
    pripravi tabelo t velikosti $n+1$
    $t[0] = 0$
    $t[1] = 1$
    za $i = 2 \ldots n$:
        $t[i] = t[i-1] + t[i-2]$
    vrni $t[n]$

① pokažemo z invariantami

② $O(n)$ (v resnici $O(n^2)$, če upoštevamo da je seštevanje linearno v dolžini predstavitve)

③ Ja, poglej učbenik

0.3. Notacija velikega $\mathcal{O}$

$$f = \mathcal{O}(g) \iff \exists c > 0 .\ \boxed{\exists n_0 \in \mathbb{N} ,\ \forall n > n_0} .\ f(n) < c \cdot g(n)$$

ni v učbeniku

$f, g : \mathbb{N} \to [0, \infty)$

$$f = \Omega(g) \iff g = \mathcal{O}(f)$$

$$f = \Theta(g) \iff f = \mathcal{O}(g) \wedge f = \Omega(g)$$