Rekurzija

```java
public static int f(int n) {
    if (n == 0) {
        return 1 ;
    } else {
        return n * f(n - 1)
    }
}
```

→ rekurzivni klic

```python
def f(n):
  if n == 0:
      return 1
  else:
      return n * f(n -1)
```

```ocaml
let rec f n =
    if n = 0 then 1 else n * f (n - 1)
```

} razstavimo na telo in rekurzijo

Funkcije višjega reda
(take, ki sprejmejo funkcijo kot argument)

<u>Primeri:</u>

$\int_0^1$ sprejme funkcijo $f:[0,1] \longrightarrow \mathbb{R}$

vrne: število

$$\underline{I}(\underline{f}) = \int_0^1 f(x)\,dx$$

$$\text{map } f \ [x_0; x_1, \dots; x_n] = [f\,x_0; f\,x_1; \dots; f\,x_n]$$

Operacija "kompozitum"

$$\overbrace{g \circ f}$$

↑ višjega reda

$$g : \mathbb{N} \to \mathbb{R}$$
$$f : \mathbb{Z} \to \mathbb{N}$$
$$g \circ f : \mathbb{Z} \longrightarrow \mathbb{R}$$

$$\begin{tikzcd} \mathbb{Z} \arrow[r, "f"] \arrow[rr, bend right, "g \circ f"'] & \mathbb{N} \arrow[r, "g"] & \mathbb{R} \end{tikzcd}$$

$$\underbrace{(\text{fun } y \mapsto y+3)\ x}_{x+3}$$

① $f\ x = \cdots\cdots\ f(\cdots)\ \cdots\cdots$ rekurzivna definicija f

② $f = \text{fun } x \to \cdots\cdots f(\cdots)\ \cdots\cdots$

③ $f = \underbrace{(\text{fun } g \to \text{fun } x \mapsto \cdots g(\cdots)\cdots)}_{\text{telo}}\ f$

④ $\boxed{f = \text{telo } f}$ kjer je $\text{telo} = \text{fun } g \mapsto \text{fun } x \mapsto \cdots g(\cdots)\cdots$

RAZPRLI SMO REKURZIJO

$f = \text{telo } f$

$\underbrace{f = \text{rek telo}}$ kjer je $\text{rek } \underset{\uparrow \atop \text{telo}}{t} = t\ (\text{rek } t)$

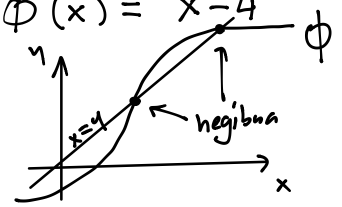
Torej: let rec rek $t = t\,(\text{rek } t)$ EDINA REKURZIJA

let faktoriela $= \text{rek }(\text{fun } g \mapsto \text{fun } x \mapsto \text{if } x=0 \text{ then } 1 \text{ else } x * g(x-1))$
$= \text{fun } x \mapsto \text{if } x=0 \text{ then } 1 \text{ else } x * (\cdots\cdots)(x-1)$

rek (fun g ↦ fun x ↦ ⋯⋯)

<u>Negibna točka</u> funkcije $\phi: A \to A$ je tak $a \in A$, da velja $a = \phi(a)$

Primer: $\phi: \mathbb{R} \to \mathbb{R}$
$\phi(x) = x^2 - 4$



$a = \phi(a)$
$a = a^2 - 4$
$a^2 - a - 4 = 0$
$a = \frac{1}{2}(1 \pm \sqrt{1+16})$

Rekurzivna definicija

$$f = \underbrace{\phi}_{\text{telo rekurzivne definicije}}(f)$$

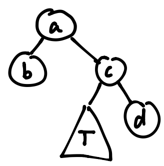
$$\boxed{\text{REKURZIJA} = \text{NEGIBNA TOČKA}}$$

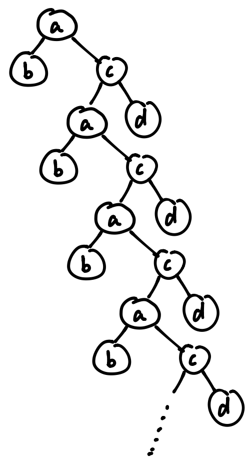
Rekurzivno definiran seznam $\longrightarrow$ $l = \phi(l)$

neskončen $\quad \phi(h) = 1 :: 2 :: h$

$$l = 1 :: (2 :: l)$$

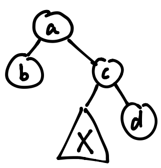
$$l = [1; 2; 1; 2; 1; 2; \ldots]$$

$T =$ 



neskončno

$$T = \phi(T)$$

$\phi :$  $\mapsto$ 

Rekurzivne podatkovne strukture

Seznam je

- → • prazen Nil
- → • sestavljen iz glave in repa Cons(glava, rep)

prvi element (celo število) preostanek seznama (seznam)

[1; 2; 3] Cons(1, Cons(2, Cons(3, Nil)))

$$\text{Seznam} = \underbrace{\{\text{Nil}\}}_{L_1(\dots)} + \underbrace{\mathbb{Z} \times \text{Seznam}}_{L_2(\dots)}$$

$A + B$

$L_1(x)$ $x \in A$

$L_2(y)$ $y \in B$

Rekurzivno definirana množica seznamov:

$L_1(\text{Nil})$	[]
$L_2(42, L_1(\text{Nil}))$	[42] 42 :: []
$L_2(23, L_2(42, L_1(\text{Nil})))$	[23; 42]

```
type N =
  | Zero
  | Succ of N
```

0 Zero

1 Succ Zero

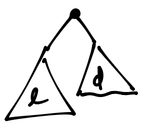
2 Succ (Succ Zero)

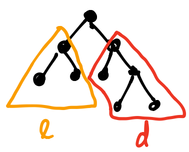
⋮

∞? Succ(Succ(Succ(.......)..) ?

Drevo:

- prazno Empty
- sestavljeno Tree(l, d)





Tree(l, d)

```
shl0 (shl1 (shl0 (shl1 0)))
```

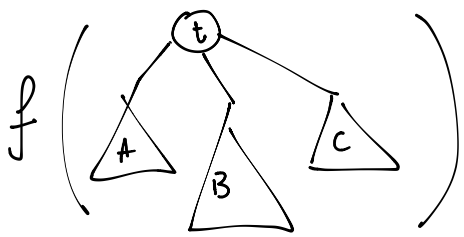
01

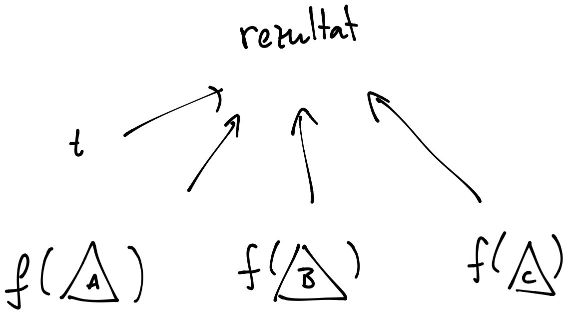
010

0101

$01010_2 = 8+2 = 10$

Strukturna rekurzija





pogoj

$$f(x) = \begin{cases} x^2+3 & \text{če } x > 0 \\ x-7 & \text{če } x = 0 \\ x+13 & \text{če } x < 0 \end{cases}$$

function
| $p_1 \to c_1$
| $p_2 \to c_2$
⋮

DO 10:25

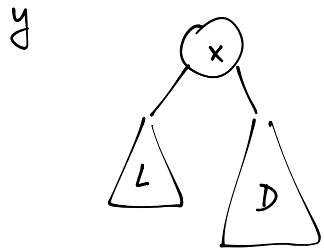
SNEMAJ!

"boolean blindness"

primerjava x in y → TRIJE odgovori

1. $x < y$
2. $x = y$
3. $x > y$

$x \leq y$ → bool DVA odgovora



$y = x$ ✓

$y < x$ išči v L

$y > x$ išči v D

if x=y then ....

elseif x<y then ....

else ....