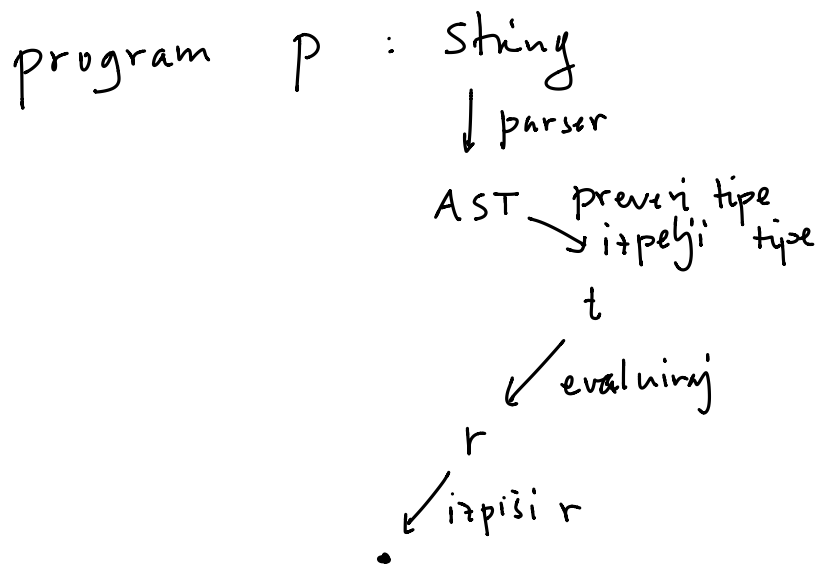


# Haskell & razredi tipov type classes



Haskell

[a]

AVLtree a

OCaml

'a list

'a avltree

Funkcije :

$x \mapsto e$

$\lambda x. e$

fun x → e

\ x → e

matematika

$\lambda$ -raun

OCaml

Haskell

# Razredi tipov (type classes)

	<u>Specifikacija</u>	<u>Implementacija</u>
Ocaml	module type T = sig .... end	module M : T = struct .... end
Java	interface I { ..... } ↑ Kaj pomeni funkcionalnost I	class C implements I { ..... razred C ima funkcionalnost I }
Python?	# Bla bla bla	hack hack
Haskell	class C a where ..... } Kaj pomeni, da ima a funkcionalnost C Opis funkcionalnosti C za tip a	instance C t where ..... Implementiramo funkcionalnost C za tip t

## Monoid:

množica  $M$   
 element  $e \in M$  enota  
 operacija  $* : M \rightarrow M \rightarrow M$

$$x * (y * z) = (x * y) * z \quad \text{asociativna}$$

$$e * x = x$$

$$x * e = x$$

Primeri:

$(\mathbb{N}, 0, +)$

$(\mathbb{Z}, 0, +)$

$(\mathbb{N}, 1, \cdot)$

$(\mathbb{R}^{n \times n}, \begin{bmatrix} 1 & & \\ & \ddots & \\ & & 1 \end{bmatrix}, \text{ množenje matric})$

$(\text{bool}, \text{false}, \text{or})$

$(\text{bool}, \text{true}, \text{and})$

$([a], [], ++)$   $\rightarrow$  stĺpcové seznamy

$$s ++ (t ++ u) = (s ++ t) ++ u$$

$$s ++ [] = s$$