

Principi programskih jezikov

2021-02-18

Anatomija programskega jezika

Programski jezik je zasnovan kot sistem, ki ima naslednje komponente:

- **sintaksa:** pravila, kako se piše kodo (na primer "vsak oklepaj mora imeti svoj zaklepaj")
- **statična semantika:** preverjanje, ali je program smiselen (na primer "spremenljivka i ni nikjer deklarirana")
- **dinamična semantika:** kako se program izvede
- **denotacijska semantika:** matematični pomen programa

Aritmetični izrazi

- cela števila
- + in ×
- spremenljivke

Primer : $37 + x * (8 + y)$

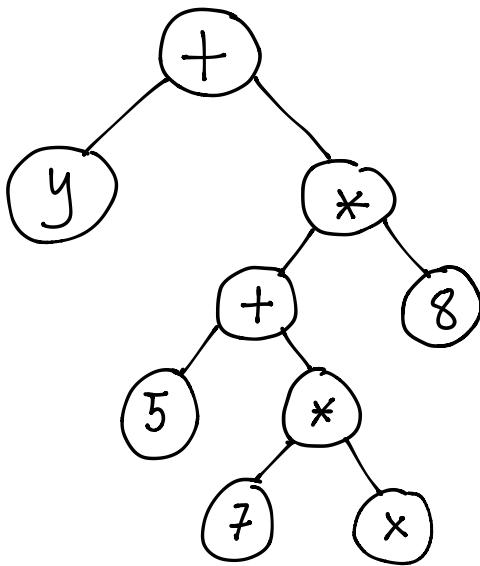
Sintaksa

↳ konkretna ← za ljudi
↳ abstraktna ← za matematike & računalnike

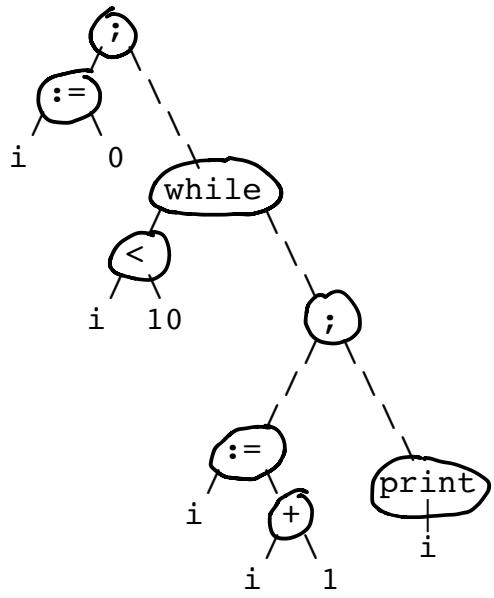
Konkretna :

"y + (5 + 7 * x) * 8"

Abstraktna:



Primer:



$i = 0$;
while ($i < 10$) {
 $i = i + 1$;
 print(i);
}

$i = 0$
while $i < 10$:
 $i = i + 1$
print(i)

Slovnica (gramatika) & slovníčna pravila

BNF pravila: náčin za podajanie slovnice

.(izraz) ::= (aditivni-izraz) EOF
 (aditivni-izraz) ::= (multiplikativni-izraz) | (aditivni-izraz) + (multiplikativni-izraz)
 (multiplikativni-izraz) ::= (osnovni-izraz) | (multiplikativni-izraz) * (osnovni-izraz)
 (osnovni-izraz) ::= (spremenljivka) | (številka) | ((aditivni-izraz))
 (spremenljivka) ::= [a-zA-Z]+ regularni izraz
 (številka) ::= -? [0-9]+

007
 <enota> ::= 1.moznost | 2.moznost | ...
 ali
 ali
 [znaki...]
 + 1 ali več ponavitev
 * 0 ali več ponavitev
 ? 0 ali 1 ponavitev

" $x * (5 + 8)$ "

Leksična analiza: niz znakov razbijemo na gradnike (lexeme, token)

Razčlenjevanje (angl. parsing):

niz gradnikov predelamo v drevo z uporabo slavnice

Za aritmetične izraze so osnovni gradniki:

- EOF poseben gradnik, ki pomeni "konec"
- PLUS znak za seštevanje
- KRAT znak za množenje
- SPREMENLJIVKA(x) spremenljivka
- ŠTEVILKA(n) številka
- OKLEPAJ in ZAKLEPAJ

Predstavke in komentarje spustimo

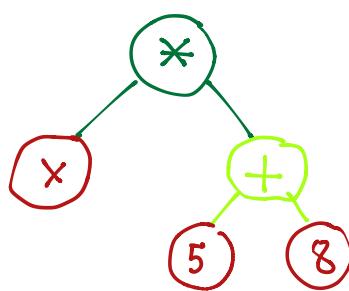
"8 53 + xy (" → leksična analiza
 (lexer, tokenizer)

OKLEPAJ ŠTEVILKA(8) ŠTEVILKA(53) PLUS SPREMENLJIVKA("xy") OKLEPAJ EOF

```

<izraz> ::= <aditivni-izraz> EOF
<aditivni-izraz> ::= <množevalni-izraz> | <aditivni-izraz> + <množevalni-izraz>
<množevalni-izraz> ::= <osnovni-izraz> | <množevalni-izraz> * <osnovni-izraz>
<osnovni-izraz> ::= <spremenljivka> | <številka> | (<aditivni-izraz>)
<spremenljivka> ::= [a-zA-Z]*
<številka> ::= -? [0-9]*

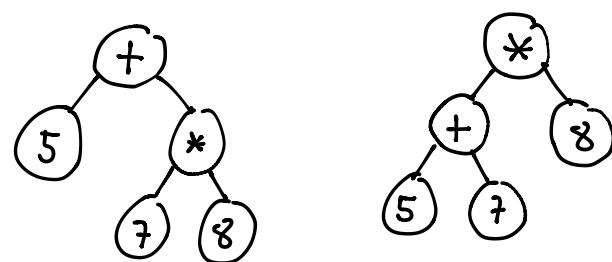
```



$\langle \text{izraz} \rangle ::= \langle \text{spremenljivka} \rangle$
 $| \langle \text{številka} \rangle$
 $| \langle \text{izrat} \rangle + \langle \text{izrat} \rangle$
 $| \langle \text{izrat} \rangle * \langle \text{izrat} \rangle$
 $| (\langle \text{izrat} \rangle)$

Dvostruka slovnica!

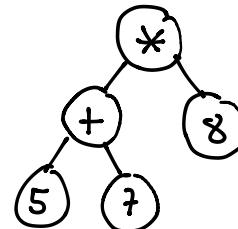
"5 + 7 * 8"



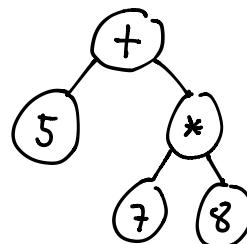
Dvojnost odpravimo s prioritetami in asociativnostjo operatorjev.

Prioriteta pove, kateri operator ima prednost, veže močnejše

- + ima višjo prioriteto kot *



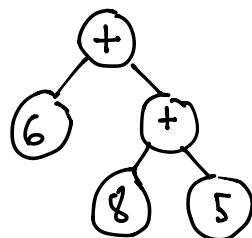
- * ima višjo prioriteto kot +



Asociativnost:

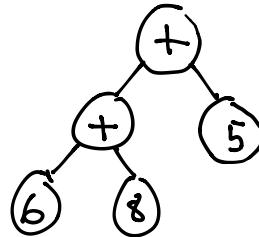
" $6 + 8 + 5$ "

NI VSEENO!!!



$6 + (8 + 5)$

desno



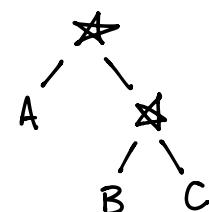
$(6 + 8) + 5$

levo

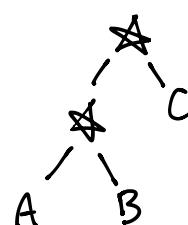
asociativnost

Operator \star :

- desno asociativen: $A \star B \star C$



- levo asociativen: $A \star B \star C$



Ljudske itročilo:

Priorite: * /
+ -

Asociativnost:

+	$x+y+z = (x+y)+z$	levo
-	$x-y-z = (x-y)-z$	levo
*		levo
/		levo

Operacijska semantika

Okolje (runtime environment):

preslikava iz imen spremenljivih v vrednosti (cela števila)

$$\eta = [x \mapsto 5, y \mapsto 3, z \mapsto 10] \quad \eta \text{ "eta"} \\ \left\{ \begin{array}{ll} \xi & \xi \\ \text{zeta} & \text{ksi} \\ \{ & \} \end{array} \right.$$

1. Semantika velikih korakov

$$\eta \models e \hookrightarrow n$$

"v okolju η se izraz e evalvira/izračuna v število n "

$$[x \mapsto 10, y \mapsto 5] \models 7 + (x * 3) \hookrightarrow 37$$

Pravila sklepanja:

$P_1 P_2 \dots P_n$ predpostavke

S

sklep

" Če smo dokazali /izpoljili /ugotovili P_1, \dots, P_n potem sklepamo S . "

— aksiom "S velja"

Primeri:

$$\frac{}{x = x}$$

refleksivnost

$$\frac{x = y}{y = x}$$

simetričnost

$$\frac{x = y \quad y = z}{x = z}$$

transitivnost

$$\eta(x) = n$$

$$\eta \mid x \hookrightarrow n$$

$$\eta \mid n \hookrightarrow n$$

↑
Source code \mathbb{Z}

$$\eta \mid e_1 \hookrightarrow n_1$$

$$\eta \mid e_2 \hookrightarrow n_2$$

matematično
množenje

$$n_1 \cdot n_2 = n$$

$$\eta \mid e_1 * e_2 \hookrightarrow n$$

↑
znak "*"

$$\eta \mid e_1 \hookrightarrow n_1$$

$$\eta \mid e_2 \hookrightarrow n_2$$

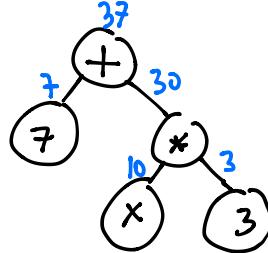
matematično seštevanje

$$n_1 + n_2 = n$$

$$\eta \mid e_1 + e_2 \hookrightarrow n$$

↑
znak v source code

$$\eta = [x \mapsto 10, y \mapsto 7]$$



2. Semantika matih konstruktor

$$7 + x * 3 \mapsto 7 + 10 * 3 \mapsto 7 + 30 \mapsto 37$$

$$\eta \mid e \mapsto e'$$

V okolju η se izraz e v enem koraku izračuna v e'

$$\frac{\eta(x) = n}{\eta \mid x \mapsto n}$$

NI PRAVILA $\frac{}{\eta \mid n \mapsto n}$ (se začila)

x spremenljivka
 e izraz
 n število

$$1) \frac{\eta \mid e_1 \mapsto e_1'}{\eta \mid e_1 + e_2 \mapsto e_1' + e_2}$$

Najprej delaj korake v e_1

$$2) \frac{\eta \mid e_2 \mapsto e_2'}{\eta \mid n_1 + e_2 \mapsto n_1 + e_2'}$$

Nato v e_2

$$3) \frac{n_1 + n_2 = n}{\eta \mid n_1 + n_2 \mapsto n}$$

Oba argumenta sta števili

$$\frac{\eta \mid e_1 \mapsto e_1'}{\eta \mid e_1 * e_2 \mapsto e_1' * e_2}$$

$$3 + (5+8) \xrightarrow{(2)} 3+11 \mapsto \dots$$

$$\frac{\eta \mid e_2 \mapsto e_2'}{\eta \mid n_1 * e_2 \mapsto n_1 * e_2'}$$

$$\underline{3} + (5+8) \xrightarrow{(1)} \text{NE GRE}$$

$$\frac{n_1 * n_2 = n}{\eta \mid n_1 * n_2 \mapsto n}$$

$$\eta = [x \mapsto 5]$$

$$\frac{\eta \mid y \mapsto}{\eta \mid y+2 \mapsto} \text{STOP}$$

$$\eta \mid (3+4) + (y+2) \xrightarrow{(1)} ?$$

SE ZATAKNE