

Statycznie typowany funkcyjny język programowania o ewaluacji leniwej

Jakub Grobelny

Kurs języka Prolog (Q1) – projekt końcowy

1 Struktura leksykalna i składnia

Klasyfikacja znaków:

- *biały znak* — znaki ASCII: HT, LF, CR, LF, SPACE
- *cyfra* — znaki ASCII: 0-9
- *mała litera* — znaki Unicode będące małymi literami ¹
- *wielka litera* — znaki Unicode będące wielkimi literami ²

1.1 Struktura leksykalna

Komentarz to ciąg znaków rozpoczynających się znakiem #, zakończony znakiem nowej linii.

Token to najdłuższy ciąg znaków niezawierający *komentarzy* ani *białych znaków*, taki, że opisuje go poniższa gramatyka:

$$\begin{aligned} \langle \text{literał-całkowitoliczbowy} \rangle &::= \text{cyfra} \\ &| \text{cyfra } \langle \text{literał-całkowitoliczbowy} \rangle \end{aligned}$$

¹Wszystkie takie znaki **X**, że spełniony jest predykat `char_type(X,lower)` w SWI Prologu.

²Analogicznie jak wyżej, tylko `char_type(X, upper)`.

$\langle \text{litera\l-znakowy} \rangle ::= ' \langle \text{znak} \rangle '$

$\langle \text{litera\l-napisowy} \rangle ::= " \langle \text{ciag-znakow} \rangle "$

$\langle \text{litera\l-zmiennopozycyjny} \rangle ::= \langle \text{u\lamek} \rangle$
 $\quad | \quad \langle \text{u\lamek} \rangle \langle \text{wyk\ladnik} \rangle$

$\langle \text{s\lowo-kluczowe} \rangle ::= \text{let} \mid \text{and} \mid \text{in} \mid \text{if} \mid \text{then}$
 $\quad | \quad \text{match} \mid \text{else} \mid \text{with} \mid \text{type} \mid \text{of} \mid \text{where}$
 $\quad | \quad \text{true} \mid \text{false} \mid \text{import} \mid \text{fun}$

$\langle \text{operator} \rangle ::= \langle \text{znak-specjalny} \rangle \mid \langle \text{operator} \rangle \langle \text{znak-specjalny} \rangle$

$\langle \text{identyfikator} \rangle ::= \text{ma\l a litera} \langle \text{alfanum} \rangle$, taki, \u017ce nie jest $\langle \text{s\lowo-kluczowe} \rangle$

$\langle \text{nazwa-typu} \rangle ::= \text{wielka litera} \langle \text{alfanum} \rangle$

gdzie

$\langle \text{znak-specjalny} \rangle ::= + \mid - \mid * \mid / \mid = \mid : \mid >$
 $\quad | \quad < \mid \& \mid ! \mid | \mid \$ \mid @ \mid \% \mid ^ \mid \sim$

$\langle \text{znak} \rangle ::= \text{dowolny znak Unicode oprócz } \backslash, ", '$
 $\quad | \quad \backslash \mathbf{n} \mid \backslash \mathbf{b} \mid \backslash \mathbf{f} \mid \backslash \mathbf{a} \mid \backslash \mathbf{r} \mid \backslash \mathbf{t} \mid \backslash \mathbf{v} \mid \backslash \backslash \mid \backslash ' \mid \backslash "$

$\langle \text{ciag-znakow} \rangle ::= \langle \text{znak} \rangle$
 $\quad | \quad \langle \text{ciag-znakow} \rangle \langle \text{znak} \rangle$

$\langle \text{litera} \rangle ::= \text{ma\l a litera}$
 $\quad | \quad \text{wielka litera}$

$\langle \text{alfanum} \rangle ::= \langle \text{litera} \rangle$
 $\quad | \quad \langle \text{alfanum} \rangle \langle \text{litera} \rangle$
 $\quad | \quad \langle \text{alfanum} \rangle \langle \text{cyfra} \rangle$

$\langle \text{u\lamek} \rangle ::= \langle \text{litera\l-ca\lkowitoliczbowy} \rangle . \langle \text{litera\l-ca\lkowitoliczbowy} \rangle$

$\langle \text{wyk\ladnik} \rangle ::= \mathbf{e} - \langle \text{litera\l-ca\lkowitoliczbowy} \rangle$
 $\quad | \quad \mathbf{e} \langle \text{litera\l-ca\lkowitoliczbowy} \rangle$

1.2 Składnia

$$\begin{aligned}\langle \textit{program} \rangle &::= \langle \textit{wyrażenie} \rangle \\ &\quad | \quad \langle \textit{definicja-typu} \rangle \langle \textit{program} \rangle \\ &\quad | \quad \langle \textit{import} \rangle \langle \textit{program} \rangle \\ \\ \langle \textit{wyrażenie} \rangle &::= \langle \textit{wyrażenie} \rangle ; \langle \textit{wyrażenie} \rangle \\ &\quad | \quad \langle \textit{definicja} \rangle \textbf{in} \langle \textit{wyrażenie} \rangle \\ &\quad | \quad (\langle \textit{wyrażenie} \rangle) \\ &\quad | \quad \langle \textit{wyrażenie-arytmetyczne} \rangle \\ &\quad | \quad \langle \textit{literał-napisowy} \rangle \\ &\quad | \quad \langle \textit{literał-znakowy} \rangle\end{aligned}$$

2 Semantyka