

# Metody kompilacji

## Laboratorium 2

Piotr Błaszyński

6 marca 2017

Zadania (wyjaśnienie w dalszej części dokumentu):

- przygotować liste leksemów w analizatorze leksykalnym (plik zX.1)
- przygotować listę tokenów w części początkowej analizatora składniowego (plik def.y),
- w analizatorze leksykalnym dopisać zwracanie tokenów - dla pojedynczych symboli jest to kod ASCII symbolu, dla słów kluczowych, liczb, identyfikatorów i symboli wieloznakowych jest kod tokenu.

Plik z opisem analizatora leksykalnego składa się z 3 części oddzielonych od siebie podwójnym znakiem '%'. Pierwsza część to część nagłówkowa - pozwala ona na dołączenie plików nagłówkowych, przygotowanie prototypów funkcji wywoływanych w pozostałych częściach oraz zapisanie definicji dla flexa. Druga część zawiera reguły przetwarzania tekstu składające się z wzorca i akcji, która jest wykonywana, jeżeli wzorzec zostanie dopasowany. W tej części (akcja) możemy również zwrócić kod tokenu do analizatora składniowego. Trzecia część to kod w języku C kopiowany bezpośrednio do pliku wynikowego w C.

Podstawne elementy reguł: Znaki '[' i ']' (nawiasy kwadratowe) pozwalają zapisać regułę dotyczącą jednego elementu wyrażenia regularnego, jeżeli po takiej regule następuje znak '+' to reguła musi wystąpić w dopasowanym tekście na odpowiednim miejscu przynajmniej raz (może się powtarzać wielokrotnie, 1-), jeżeli następnym znakiem jest '\*' to reguła może wystąpić wielokrotnie ale nie musi (0-). Na końcu listy reguł powinna się znaleźć kropka ('.'), która w tym kontekście oznacza brak wcześniejszego dopasowania - co powoduje błąd (nieznany leksem).

Plik z opisem analizatora składniowego jest zbudowany również z 3 sekcji: nagłówkowa, reguły i akcje (tutaj nazywane semantycznymi - na tym etapie nie będziemy jeszcze z nich korzystać), kod w C/C++. Na obecnym etapie potrzebujemy

jedynie zdefiniować tokeny (przykłady poniżej) dla wszystkich słów kluczowych, liczb, identyfikatorów i symboli wieloznakowych (dla pojedynczych znaków kodem tokenu jest jego kod ASCII).

Po zdefiniowaniu tokenów i przetworzeniu pliku def.y przy pomocy bisona możemy (bison wygeneruje nam m.in. plik nagłówkowy z liczbowymi kodami tokenów) zwracać kody tokenów z poziomu analizatora leksykalnego. Przykłady do znalezienia w katalogach z4, z5 i z6.

Definicja tokenu bez określania typu:

```
%token LEQ
```

Definicja kilku tokenów bez określania typu:

```
%token LEQ GEQ EQ
%token FOR INT DOUBLE
```

Definicja tokenu z określeniem typu:

```
%token <text> ID
```

Zwrócenie tokenu (całe reguły z akcjami w wersji podstawowej) z analizatora leksykalnego:

```
\=                {return '=';}
\<\=             {return LEQ;}
"int"            {return INT;}
[A-Za-z_][A-Za-z0-9_]* {return ID;}

```