

Implementace imperativního jazyka IFJ14

Tým 113

Fakulta informačních technologií
Vysokého učení technického v Brně

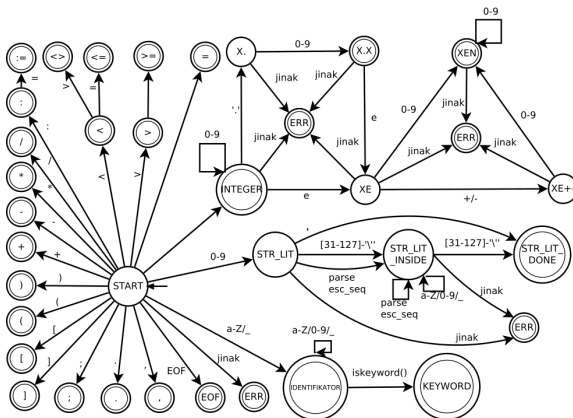
15. prosince 2014

Vedoucí týmu: Antonín Marko

- **Lexikální analyzátor:** Tomáš Pružina a Martin Juřík
- **Syntaktický a sémantický analyzátor:** Antonín Marko
- **Interpret:** Tomáš Pružina
- **Algoritmy předmětu IAL:** Petr David a Martin Juřík
- **LL-gramatika a testování:** Martin Kubíček

Lexikální analyzátor

Schéma konečného automatu

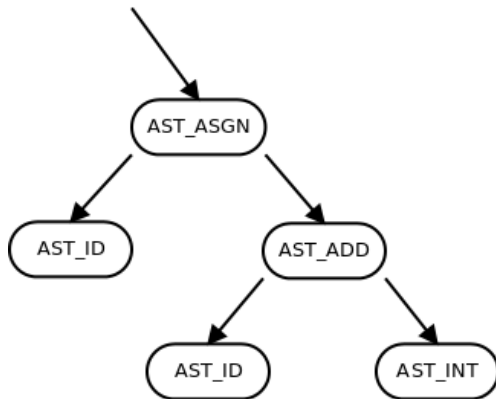


Syntaktický a semantický analyzátor

- Analýza shora dolů
- Kontrola datových typů
- Načítání podle LL-tabulky
- Vnitřní kód řešen Abstraktním syntaktickým stromem
- Výrazy řešeny Shunting-yard algoritmem

Syntaktický a semantický analyzátor

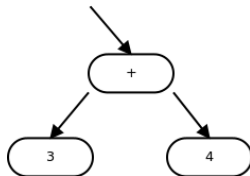
Příklad AST



Shunting-yard algoritmus

Přímé generování AST

- Metoda zpracování výrazů
- Převod infixového zápisu do postfixového
- **Modifikace:** Přímé generování AST
 - Pouze jediný průchod
- Př.: $3 + 4 \Rightarrow 3 \ 4 \ +$



Interpret

- Interpretuje AST
- Pracuje s tabulkou symbolů
- Kontrolování platných operací
 - Neinicializované proměnné
 - Dělení nulou

Algoritmy do předmětu IAL

Funkce sort

QuickSort

- Složitost: $n * \log(n)$
- Nestabilní algoritmus
- Problém volby pivotu

Algoritmy do předmětu IAL

Funkce find

Knuth-Morris-Prattův algoritmus

- Slouží na vyhledávání podřetězců v řetězcích
- Konečný automat využívající hrany ANO/NE
- Pole FAIL uchovávající hodnoty posunů zpět
- Nevýhodou KMP je, že z každého uzlu vychází tolik hran, kolik je znaků abecedy

Implementovaná rozšíření

ARRAY, FOR, REPEAT, BOOLOP, ELSEIF

ARRAY

- Implementované v tabulce symbolů

FOR cyklus

- Doplnění pravidel do LL-tabulky

REPEAT cyklus

- Podobný postup analýzy jako u cyklu while

BOOLOP

- Doplněné operátory do Shunting-yard algoritmu

ELSEIF

- Podpora vynechání klíčového slova else

Jednoduchý příklad

```
AST_STT
AST_CMD
  AST_BOL [ TRUE ]
  AST_ASG
    AST_ID [ b ]
  AST_CMD
    AST_INT [ 10 ]
    AST_FOR_TO
      AST_INT [ 0 ]
      AST_ASG
        AST_ID [ id ]
    AST_FOR
      AST_CMD
        AST_IF
          AST_ID [ b ]
          AST_CMD
            AST_NOT
              AST_ID [ b ]
            AST_ASG
              AST_ID [ b ]
          AST_CMD
            AST_CMD
              AST_NON
                AST_WRITE
                  AST_CMD
                    AST_NOT
                      AST_ID [ b ]
                    AST_ASG
                      AST_ID [ b ]
                    AST_CMD
                      AST_CMD
AST_CMD

var
  b: boolean;
  id: integer;

begin
  b := true;
  for id := 0 to 10 do
  begin
    if b then
    begin
      b := NOT b
    end;
    write(b);
    b := NOT b
  end;
end.
```