

Construção de Compiladores

Período Especial

Aula 7: Expressões

Bruno Müller Junior

Departamento de Informática
UFPR

2020

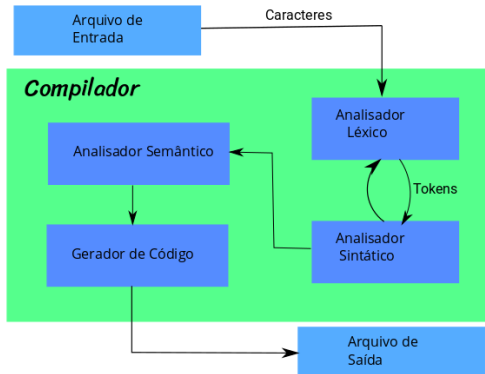
- 1 Objetivos
- 2 Análise semântica
 - Tradução Dirigida pela Sintaxe
 - Atributos Sintetizados
 - Atributos Sintetizados
 - Atributos Sintetizados
 - Atributos Sintetizados
 - Atributos Herdados
- 3 Expressões Booleanas
 - Implementação
- 4 Projeto

Objetivos

- Após a aula de hoje, o compilador desenvolvido pelos alunos deverá ser capaz de receber o código da esquerda como entrada e detectar erros semânticos.

```
program varsGlobais (input, output);  
var a, b: integer;  
    k1, temp : integer;  
begin  
    a:=a>0;  
end.
```

Análise semântica



Analizador sintático e semântico

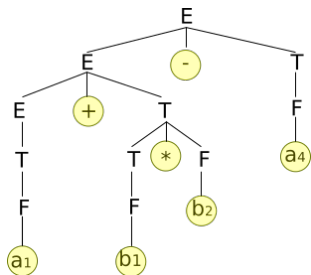
- Como já foi discutido, a análise semântica ocorre na execução dos "vértices" ou "nós" semânticos;
- Assim, os nós semânticos são executados durante a construção da árvore sintática;
- Didaticamente, é mais fácil explicar considerando que o analisador sintático constrói toda a árvore (incluindo os vértices semânticos) e passa a árvore toda para analisador semântico;
- Este, por sua vez caminha in order pela árvore e executa os vértices semânticos quando aparecerem.

Atributos Sintetizados

```
G4 = { E ::= E+T | T
      T ::= T*F | F
      F ::= a | b
      }
```

- como usar para verificar tipos?
- lembre do caminhamento inorder!
- onde “descobre” o tipo do ident?
- onde “compara” os tipos dos idents?

$a + b * b - a$



Atributos Sintetizados

- TDSs é uma abstração (algoritmo).
- Como utilizar atributos sintetizados no nosso compilador?
- Várias possibilidades.
- Destaco duas:
 - atributos posicionais do bison;
 - pilhas;

Atributos Sintetizados: bison

- O bison contém mecanismo para referenciar os elementos de uma produção usando os símbolos $\$ \$$, $\$ 1$, $\$ 2$, \dots , conhecidos como pseudo-variáveis.
- O número indica a posição. Na produção $E ::= E + T$, temos:

$\$ \$ == E$ (o da esquerda, antes do $::=$)

$\$ 1 == E$ (o da direita, após do $::=$)

$\$ 2 == +$

$\$ 3 == T$

- Exemplo: $E ::= E + T \{ \$ \$ = \$ 1 + \$ 2 \}$
- Qual o tipo de $\$?$

Atributos Sintetizados: pilha

- Uma alternativa que deixa o código mais legível é utilizar uma ou mais pilhas.

```
E ::= E + T { t1 = desempilha(E);  
              t2 = desempilha(T);  
              se (t1 == t2)  
                então empilha(E,t1);  
                senão erro("...");  
            }
```


Atributos Sintetizados: pilha

- Como utilizar atributos herdados no nosso compilador?
- Variáveis globais;

Expressões Booleanas

- Instruções que geram ou lidam com booleanos.
- $M[s-1] := (M[s-1] \text{ op } M[s]); s := s-1; i := i+1$
- $true == 1$ e $false == 0$

Instrução	Ação		Exemplo de Tradução	
			Expressão	Código MEPA equivalente
CMIG	$M[s-1] := (M[s-1] == M[s])$ $s := s-1; i := i+1$	Compara Igual	$a > b \text{ and } b = c$	CRVL a
CMMA	$M[s-1] := (M[s-1] > M[s])$ $s := s-1; i := i+1$	Compara Maior		CRVL b
CMME				CMMA
CMDG				CRVL b
...				CRVL c
CONJ	$M[s-1] := (M[s-1] \text{ and } M[s])$ $s := s-1; i := i+1$	AND		CMIG
...				CONJ

Implementação

- A regra que aplica as construções relacionais é:
25. `<expressão> ::= <expressão simples> [<relação> <expressão simples>]`
26. `<relação> ::= = | <> | < | <= | > | >=`

Projeto

- Implemente as regras de declaração de declaração de variáveis utilizando atributos herdados para saber o "k" de AMEM k.
- Implemente as regras relacionadas com expressões (regras 25 até 30) utilizando atributos sintetizados para verificar tipos.
- Dica: a regra 30 é a única que acessa a tabela de símbolos. Nas demais, usa-se as construções (como pilha ou \$\$).

- Página para anotações

Licença

- Slides desenvolvidos somente com software livre:
 - \LaTeX usando beamer;
 - Inkscape.
- Licença:
 - Creative Commons Atribuição-Uso Não-Comercial-Vedada a Criação de Obras Derivadas 2.5 Brasil License. <http://creativecommons.org/licenses/by-nc-nd/2.5/br/>