

Construção de Compiladores

Aula 6 - Declaração de Vars

Bruno Müller Junior

Departamento de Informática
UFPR

30 de Agosto de 2014

1 Declaração de Variáveis e Comando Composto

2 Regras da declaração de Variáveis

- Ações do compilador
- Regras do Comando Composto
- Regra da Atribuição



Declaração de Variáveis e Comando Composto

- Esta aula trata de declaração de variáveis e do comando composto (só `BEGIN ...END`).

Regras da declaração de Variáveis

- Regras para declaração de variáveis: 2,8,9,10
2. `<bloco> ::= [<parte declara rotulos>]
[<parte define tipos>] (*)
[<parte declara variáveis>]
[<parte declara subrotinas>]
<Comando Composto>`
 8. `<parte declara variáveis> ::= var <declara variáveis>`
`{ ; <declara variáveis> }`
 9. `<declara variáveis> ::= <lista de idents> : <tipo>`
 10. `<lista de idents> ::= <ident> { , <ident> }`

Ações do compilador

- As regras tem de ser convertidas para a notação LR.
- Ao encontrar a regra 8 (parte declara variáveis), o compilador deve:
 - inserir os símbolos na Tabela de Símbolos;
 - gerar o código AMEM k , onde k é o número de variáveis.
 - problemas:
 - Ao encontrar a var. simples, sabe-se o nível léxico e deslocamento. Já o tipo das variáveis, só se sabe ao fim da declaração. Como atualizar o tipo?
 - Como tratar o caso abaixo (vários AMEM)?

```
program declaraVars (input, output);  
  var a, b: integer  
      c, d, e: integer  
      f: integer  
  ...  
end.
```

```
INPP  
AMEM 2  
AMEM 3  
AMEM 1  
...  
PARA
```

Regras do Comando Composto

- O comando composto é quem tem as regras de tradução dos comandos pascal, e será alvo de várias aulas.
- Hoje, só atribuição.

```
16. <comando composto> ::= begin <comando>
                                { ; <comando> } end
17. <comando> ::= [numero :] <comando sem rótulo>
18. <comando sem rótulo> ::= <atribuição> |
                                <chamada de procedimento> |
                                <desvio> |
                                <comando composto> |
                                <comando condicional> |
                                <comando repetitivo>
```



Regra da Atribuição

19. `<atribuição> ::= <variavel> := <expressão>`

- Esta regra contém dois problemas:
 - Calcular o tipo resultante da expressão, como por exemplo em:
`a:=a+b` ou `a:=a>b` ou `a:=1`. Este será o tópico da próxima aula.
 - Como armazenar o elemento da esquerda (com quem o tipo da expressão deve ser comparado).
 - Sugestão: variável global `l_elem`.