



Construção de Compiladores

Alterado por: prof. Rodrigo Pereira

Alterado por: prof. Marlon Oliveira

Elaborado por: Prof^a Christine Vieira

ANÁLISE SEMÂNTICA

há um nível de correção que vai além da gramática

- Jogo dos 6 erros:

```
foo(a,b,c,d) {  
    int a, b, c, d;  
    ...  
}  
  
bar() {  
    int f[3],g[0], h, i, j, k;  
    char *p;  
    foo(h,i,"ab",j, k);  
    k = f * i + j;  
    h = g[17];  
    printf("<%s,%s>.\n",p,q);  
    p = 10;  
}
```

- Jogo dos 6 erros:

```
foo(a,b,c,d) {  
    int a, b, c, d;  
    ...  
}  
bar() {  
    int f[3],g[0], h, i, j, k;  
    char *p;  
    foo(h,i,"ab",j, k);  
    k = f * i + j;  
    h = g[17];  
    printf("<%s,%s>.\n",p,q);  
    p = 10;  
}
```

O que há de errado?

(vamos contar...)

- número de argumentos de foo()
- declarou g[0], usou g[17]
- "ab" não é int
- dimensão errada no uso de f
- variável não declarada q
- 10 não é uma string

Tudo isso está além da sintaxe

Antes de gerar código, o compilador pode responder as questões do tipo:

- “x” é escalar, array, ou função? “x” foi declarado?
- Há nomes não declarados? Há nomes Declarados mas não usados?
- Qual declaração de “x” é usada por uma referência a “x”?
- A expressão “x * y + z” é corretamente tipada?
- Em “a[i,j,k]”, a foi declarado com três dimensões?
- Onde uma variável “z” pode ficar armazenada? (registrador, local, global, heap, estática)
- Em “f ← 15”, como representar a constante 15?
- Quantos argumentos uma função “foo()” recebe? E “printf()” ?
- “x” é definido antes de ser usado?

ANÁLISE SEMÂNTICA

O próximo passo no processo de construção do nosso compilador refere-se a análise semântica, normalmente composta por um conjunto de ações, as **ações semânticas**, responsáveis pela determinação do significado de cada instrução do programa fonte.

A avaliação das ações semânticas pode **criar e manter a tabela de símbolos**, gerar código-objeto e em mensagens de erro.

A análise semântica constitui-se, em linhas gerais, um verificador de tipos. É de sua responsabilidade verificar se um operador contém operandos do mesmo tipo, se uma variável foi declarada, entre outras atribuições.

Se alguma dessas condições não for satisfeita, o sistema reportará um erro semântico.

Por exemplo, se tivéssemos declarado uma variável como `integer` uma variável **y** como `string` e quiséssemos fazer uma atribuição do tipo **$x := x + y$** , certamente seria gerado um erro semântico, porque não é possível adicionarmos um valor `string` a um inteiro e atribuí-lo a outro inteiro.

Nos compiladores usuais, compiladores dirigidos pela sintaxe, o analisador sintático ativa a execução das ações semânticas sempre que forem atingidos determinados estados do reconhecimento ou sempre que determinadas transições ocorrerem durante a análise do programa fonte.

Deve-se então incluir na gramática símbolos que indiquem as ações semânticas a serem efetuadas no processo de compilação.

Neste caso **o algoritmo da análise sintática deve ser alterado** para que, ao detectar um símbolo com uma ação semântica no topo da pilha sintática, retire-o do topo da pilha e execute a ação semântica correspondente.

A análise semântica, trabalha com uma tabela chamada o “**tabela de símbolos**” e com o decorrer das ações semânticas encontradas na gramática esta tabela é preenchida ou não.

A tabela de símbolos armazena informações relativas identificadores encontrados no programa fonte durante processo de compilação. Sua organização depende da estrutura da linguagem que está sendo implementada.

As linhas da tabela de símbolos devem conter as seguintes informações:

Nome	Categoria	Tipo	Nível
------	-----------	------	-------

Nome: Nome do identificador

Categoria:

- variável
- procedure
- parâmetro

Tipo:

- Real
- Inteiro

Nível: Nível onde foi declarado o identificador

As operações para a manipulação da tabela de símbolos são:

Busca: verifica a existência de um identificador na tabela de símbolos. Caso afirmativo efetua as ações pertinentes. Caso negativo indica a condição de erro: “identificador não declarado”.

Inserção: verifica a existência de um identificador na tabela de símbolos. Caso afirmativo, indica condição de erro: “identificador já declarado”, caso negativo insere na tabela.

Remoção: retira os nomes da tabela de símbolos. Deve-se lembrar de verificar o nível que está retirando.

Atividade

- Mesmo grupo do trabalho final.
- Preparar um material (slide ou relatório) para apresentar a turma sobre:
 - Gramática de Atributos semânticos (herdados e sintetizados), ou
 - Algoritmos de Semântica Dirigida por Sintaxe (S-atributo e L-atributo)
- Serão sorteados 2 grupos para apresentar, valendo 1 ponto extra na prova.
- Deverá conter teoria e exemplos.