

Construção de Compiladores 1 - 2015.1 - Prof. Daniel Lucrédio

Aula 08 - Tabela de Símbolos - roteiro

Demonstração 1 – Construção da tabela de símbolos “na mão”

1. Criar um novo arquivo, no Desktop, com um programa de exemplo

```
let x=2+1, y=3+4 in x+y
```

2. Abrir o NetBeans, e criar um novo projeto Java “AnalizadorExpressoesAritmeticas”
3. Criar classe analisadorexpressoesaritmeticas.EntradaTabelaDeSimbolos

```
public class EntradaTabelaDeSimbolos {  
    public String nome;  
    public int valor;  
}
```

4. Criar classe analisadorexpressoesaritmeticas.TabelaDeSimbolos

```
public class TabelaDeSimbolos {  
  
    private HashMap<String, EntradaTabelaDeSimbolos> tabelaDeSimbolos;  
  
    public TabelaDeSimbolos() {  
        tabelaDeSimbolos = new HashMap<String, EntradaTabelaDeSimbolos>();  
    }  
  
    public void inserir(String nome, int valor) {  
        EntradaTabelaDeSimbolos etds = new EntradaTabelaDeSimbolos();  
        etds.nome = nome;  
        etds.valor = valor;  
        tabelaDeSimbolos.put(nome, etds);  
    }  
  
    public EntradaTabelaDeSimbolos verificar(String nome) {  
        if(!tabelaDeSimbolos.containsKey(nome))  
            return null;  
        else return tabelaDeSimbolos.get(nome);  
    }  
}
```

5. Criar classe analisadorexpressoesaritmeticas.Escopos

```
public final class Escopos {  
    private LinkedList<TabelaDeSimbolos> pilhaDeTabelas;  
  
    public Escopos() {  
        pilhaDeTabelas = new LinkedList<TabelaDeSimbolos>();  
        criarNovoEscopo();  
    }  
}
```

```

public void criarNovoEscopo() {
    pilhaDeTabelas.push(new TabelaDeSimbolos());
}

public TabelaDeSimbolos pegarEscopoAtual() {
    return pilhaDeTabelas.peek();
}

public List<TabelaDeSimbolos> percorrerEscoposAninhados() {
    return pilhaDeTabelas;
}

public void abandonarEscopo() {
    pilhaDeTabelas.pop();
}
}

```

6. Criar o seguinte código na classe principal

```

ANTLRInputStream input = new ANTLRInputStream(new
FileInputStream("/Users/daniellucredio/NetBeansProjects/projetosCompiladores/
teste.txt"));
ExpressoesAritmeticasLexer lexer = new
ExpressoesAritmeticasLexer(input);
CommonTokenStream tokens = new CommonTokenStream(lexer);
ExpressoesAritmeticasParser parser = new
ExpressoesAritmeticasParser(tokens);
int val = parser.programa().val;
System.out.println("Valor=" + val);

```

7. Abrir o ANTLRWorks e criar a seguinte gramática

```

grammar ExpressoesAritmeticas;

@members {
Escopos escoposAninhados = new Escopos();
}

programa returns [int val]
:      exp { $programa.val = $exp.val; } ;
exp returns [int val]
:      { int termo1 = 0, termo2 = 0; }
      ( '(' expPar=exp ')'
        { termo1 = $expPar.val;
        }
      | variavel=ID
        { List<TabelaDeSimbolos> escopos =

```

```

escoposAninhados.percorrerEscoposAninhados();
boolean encontrou = false;

```

```

        for(TabelaDeSimbolos ts:escopos) {
            EntradaTabelaDeSimbolos etds =
ts.verificar($variavel.getText());
            if(etds != null) {
                termol = etds.valor;
                encontrou = true;
                break;
            }
        }
        if(!encontrou) {
            throw new RuntimeException("Erro semântico: "
                +$variavel.getText()+" não foi declarada antes
do uso");
        }
    }
    | constante=INT { termol =
Integer.parseInt($constante.getText()); }
    | 'let' { escoposAninhados.criarNovoEscopo(); }
        listaDecl 'in' subexp=exp
        { termol = $subexp.val;
          escoposAninhados.abandonarEscopo();
        }
    )
    ('+' termo=exp
        { termo2 += $termo.val; }
    )*
    { $exp.val = termol + termo2; }
;
listaDecl returns [int val]
:      decl (',' decl)*;
decl returns [int val]
:      nome=ID '=' exp
        { TabelaDeSimbolos escopoAtual =
escoposAninhados.pegarEscopoAtual();
          if(escopoAtual.verificar($nome.getText()) != null) {
              throw new RuntimeException("Erro semântico:
"+$nome.getText()+
                                     " declarada duas vezes num
mesmo escopo");
          }
          else escopoAtual.inserir($nome.getText(), $exp.val);
        }
;

ID  : ('a'..'z'|'A'..'Z'|'_') ('a'..'z'|'A'..'Z'|'0'..'9'|'_')*;
INT : '0'..'9'+;
WS  : ( ' '
      | '\t'
      | '\r'
      | '\n'
      ) {skip();};

```

8. Testar com vários programas

```
let x=2+1, y=3+4 in x+y
```

```
let x=2, y=3 in  
  (let x=x+1, y=(let z=3, x=4 in x+y+z)  
   in (x+y)  
  )
```

```
let x=2,y=x+1 in (let x=x+y,y=x+y in y)
```

```
let x=2,x=3 in x+1
```

```
let x=2 in x+y
```

```
let x=2 in (let x=3 in x)
```

```
let x=2+1, y=(let z=3 in z+x) in let z=4 in x+y+z
```