

Linguagens de Programação 2

Trabalho Prático N°3

João Costa Seco

Outubro de 2005

- 1 Introdução
- 2 Parte 1 - Análise Sintática
- 3 Parte 2 - Análise Semântica
- 4 Parte 3 - Geração de Código .NET

Objectivos

O que se pretende. . .

Uma linguagem imperativa com comandos e expressões. Também vamos acrescentar comandos de controlo de fluxo (while,if).
Comandos de output.

Exemplos de expressões

- var(0)

Objectivos

O que se pretende. . .

Uma linguagem imperativa com comandos e expressões. Também vamos acrescentar comandos de controlo de fluxo (while,if).
Comandos de output.

Exemplos de expressões

- `var(0)`
- `var(x+y*4)`

Objectivos

O que se pretende. . .

Uma linguagem imperativa com comandos e expressões. Também vamos acrescentar comandos de controlo de fluxo (while,if).
Comandos de output.

Exemplos de expressões

- `var(0)`
- `var(x+y*4)`
- `!var(x+y*4)`

Objectivos

O que se pretende. . .

Uma linguagem imperativa com comandos e expressões. Também vamos acrescentar comandos de controlo de fluxo (while,if).
Comandos de output.

Exemplos de expressões

- `var(0)`
- `var(x+y*4)`
- `!var(x+y*4)`
- `!x*3`

Objectivos

O que se pretende. . .

Uma linguagem imperativa com comandos e expressões. Também vamos acrescentar comandos de controlo de fluxo (while,if).
Comandos de output.

Exemplos de expressões

- `var(0)`
- `var(x+y*4)`
- `!var(x+y*4)`
- `!x*3`
- `not (3 > x) & !y`

Objectivos

O que se pretende. . .

Uma linguagem imperativa com comandos e expressões. Também vamos acrescentar comandos de controlo de fluxo (while,if).
Comandos de output.

Exemplos de expressões

- `var(0)`
- `var(x+y*4)`
- `!var(x+y*4)`
- `!x*3`
- `not (3 > x) & !y`
- `3 = !y`

Objectivos

O que se pretende. . .

Uma linguagem imperativa com comandos e expressões. Também vamos acrescentar comandos de controlo de fluxo (while,if).
Comandos de output.

Exemplos de expressões

Exemplos de comandos

- `print (!(var(0)))`

Objectivos

O que se pretende. . .

Uma linguagem imperativa com comandos e expressões. Também vamos acrescentar comandos de controlo de fluxo (while,if).
Comandos de output.

Exemplos de expressões

Exemplos de comandos

- `print (!(var (0)))`
- `print (0); print (1)`

Objectivos

O que se pretende. . .

Uma linguagem imperativa com comandos e expressões. Também vamos acrescentar comandos de controlo de fluxo (while,if).
Comandos de output.

Exemplos de expressões

Exemplos de comandos

- `print (!(var(0)))`
- `print (0); print (1)`
- `decl x=var(0) in if !x=2 then print(2) else print (!x)`

Objectivos

O que se pretende. . .

Uma linguagem imperativa com comandos e expressões. Também vamos acrescentar comandos de controlo de fluxo (while,if).
Comandos de output.

Exemplos de expressões

Exemplos de comandos

- `print (!(var(0)))`
- `print (0); print (1)`
- `decl x=var(0) in if !x=2 then print(2) else print (!x)`
- `decl x=var(0) in x:=1; print (!x) end`

Objectivos

O que se pretende. . .

Uma linguagem imperativa com comandos e expressões. Também vamos acrescentar comandos de controlo de fluxo (while,if).
Comandos de output.

Exemplos de expressões

Exemplos de comandos

- `print (!(var(0)))`
- `print (0); print (1)`
- `decl x=var(0) in if !x=2 then print(2) else print (!x)`
- `decl x=var(0) in x:=1; print (!x) end`
- `decl x=var(0) in while 10>!x do print (!x);x:=!x+1 end end`

Lexer - Reconhecimento de tokens

Tokens

```
| < SEMICOLON: ";" >  
| < BANG: "!" >  
| ...  
| < VAR: "var" >  
| < PRINT: "print" >  
| ...
```

Gramática

```
S ::=
  E := E
  while E do S end
  if E then S else S end
  S ; S
  decl id1 = E1 id2 = E2 ... idn = En in S end
  print(E)

E ::= E = E | E > E | E & E | not E | var(E) | !E |
```

Avaliação de referências

Referências

Utilização de um novo tipo de valor, uma célula de memória, é um objecto utilizado para conter outro valor passível de ser modificado.

Modificações ao trabalho anterior

- interface `IVal`.
- `IVal evaluate(IEnvironment env) ...`
- classe `IntVal` contendo um field `int`, um constructor e um “getter”.
- classe `CellVal` para implementar as células de memória.
página 160 (acetatos das aulas teóricas).

Geração de código .Net

Slides das aulas teóricas 168 e seguintes

CheckList

- Tokens:
 - “:=”,
 - “!”,
 - “var”
- Gramática: ...
- AST: Ref, Deref, Assign, ...
- Semântica:
 - IVal,
 - IntVal,
 - CellVal,...
- Compilação: ...