Sumário

[INTRODUÇÃO 3](#_Toc232408334)

[SINTAXE, SEMÂNTICA E DESCRIÇÃO 3](#_Toc232408335)

[1. Programas 3](#_Toc232408336)

[1.1 Programa 3](#_Toc232408337)

[2. Comandos 3](#_Toc232408338)

[2.1 Comando 3](#_Toc232408339)

[2.2 Comando Declaração 4](#_Toc232408340)

[2.3 Atribuição 4](#_Toc232408341)

[2.4 While 4](#_Toc232408342)

[2.5 For 4](#_Toc232408343)

[2.6 If Then Else 4](#_Toc232408344)

[2.7 Repeat Until 5](#_Toc232408345)

[2.8 IO 5](#_Toc232408346)

[2.9 Skip 5](#_Toc232408347)

[2.10 Chamada Procedimento 5](#_Toc232408348)

[2.11 Comando Declaração 5](#_Toc232408349)

[3. Expressões 6](#_Toc232408350)

[3.1 Expressão 6](#_Toc232408351)

[3.2 Valor 6](#_Toc232408352)

[3.3 Expressão Unária 6](#_Toc232408353)

[3.4 Expressão Binária 6](#_Toc232408354)

[3.5 Lista Expressão 7](#_Toc232408355)

[4. Declarações 7](#_Toc232408356)

[4.1 Declaração 7](#_Toc232408357)

[4.2 Declaração Variável 7](#_Toc232408358)

[4.3 Declaração Procedimento 7](#_Toc232408359)

[4.4 Lista Declaração Parâmetro 8](#_Toc232408360)

[5. Tipos 8](#_Toc232408361)

[5.1 Tipo 8](#_Toc232408362)

[5.2 Valor Inteiro 8](#_Toc232408363)

[5.3 Valor Booleano 8](#_Toc232408364)

[5.4 Valor String 8](#_Toc232408365)

[5.5 Id 9](#_Toc232408366)

[PALAVRAS RESERVADAS 9](#_Toc232408367)

[PROGRAMAS EXEMPLIFICATIVOS 10](#_Toc232408368)

[1. Programa 1 10](#_Toc232408369)

[2. Programa 2 10](#_Toc232408370)

[3. Programa 3 10](#_Toc232408371)

[CONCLUSÃO 10](#_Toc232408372)

[BIBLIOGRAFIA 10](#_Toc232408373)

INTRODUÇÃO

Esse documento contém detalhes da especificação e desenvolvimento por trás do projeto de desenvolvimento de um interpretador em Haskell. Foram descritas neste documento as características da linguagem que deve ser interpretada pelo interpretador, bem como exemplos de programas implementados nesta linguagem. Estes programas tentam englobar todas as funcionalidades da linguagem.

O objetivo deste documento é que, se um programador tiver acesso à ele, conseguirá não só implementar programas usando a linguagem, como entender o que cada palavra reservada, comando ou expressão oferecem à linguagem.

SINTAXE, SEMÂNTICA E DESCRIÇÃO

1. Programas
   1. Programa
   2. Sintaxe

Programa ::= Comando

* 1. Semântica
  2. Descrição
  3. Exemplo

1. Comandos
2. Comando
   1. Sintaxe

Comando ::= ComandoDeclaracao

| Atribuicao

| While

| For

| IfThenElse

| RepeatUntil

| IO

| Skip

| ChamadaProcedimento

| Comando ";" Comando

* 1. Semântica
  2. Descrição
  3. Exemplo

1. Comando Declaração
   1. Sintaxe

ComandoDeclaracao ::= "{" Declaracao ";" Comando "}"

* 1. Semântica
  2. Descrição
  3. Exemplo

1. Atribuição
   1. Sintaxe

Atribuicao ::= Id ":=" Expressao

* 1. Semântica
  2. Descrição
  3. Exemplo

1. While
   1. Sintaxe

While ::= "while" Expressao "do" Comando

* 1. Semântica
  2. Descrição
  3. Exemplo

1. For
   1. Sintaxe

For ::= "for" Atribuicao "to" ValorInteiro "do" Comando

* 1. Semântica
  2. Descrição
  3. Exemplo

1. If Then Else
   1. Sintaxe

IfThenElse ::= "if" Expressao "then" Comando "else" Comando

* 1. Semântica
  2. Descrição
  3. Exemplo

1. Repeat Until
   1. Sintaxe

RepeatUntil ::= “repeat” Comando “until” Expressao

* 1. Semântica
  2. Descrição
  3. Exemplo

1. IO
   1. Sintaxe

IO ::= "write" "(" Expressao ")"

| "read" "(" Id ")"

* 1. Semântica
  2. Descrição
  3. Exemplo

1. Skip
   1. Sintaxe

Skip ::= "skip"

* 1. Semântica
  2. Descrição
  3. Exemplo

1. Chamada Procedimento
   1. Sintaxe

ChamadaProcedimento ::= "call" Id "(" ListaExpressao ")"

* 1. Semântica
  2. Descrição
  3. Exemplo

1. Comando Declaração
   1. Sintaxe

ComandoDeclaracao ::= "{" Declaracao ";" Comando "}"

* 1. Semântica
  2. Descrição
  3. Exemplo

1. Expressões
   1. Expressão
2. Sintaxe

Expressao ::= Valor

| ExpUnaria

| ExpBinaria

| Id

| "&"Id

| "\*"Id

1. Semântica
2. Descrição
3. Exemplo
   1. Valor
4. Sintaxe

Valor ::= ValorInteiro

| ValorBooleano

| ValorString

1. Semântica
2. Descrição
3. Exemplo
   1. Expressão Unária
4. Sintaxe

ExpUnaria ::= "-" Expressao

| "!" Expressao

1. Semântica
2. Descrição
3. Exemplo
   1. Expressão Binária
4. Sintaxe

ExpBinaria ::= Expressao "+" Expressao

| Expressao "-" Expressao

| Expressao "&&" Expressao

| Expressao "||" Expressao

| Expressao "==" Expressao

| Expressao "++" Expressao

1. Semântica
2. Descrição
3. Exemplo
   1. Lista Expressão
4. Sintaxe

ListaExpressao ::= Expressao

| Expressao "," ListaExpressao

1. Semântica
2. Descrição
3. Exemplo
4. Declarações
5. Declaração
   1. Sintaxe

Declaracao ::= DeclaracaoVariavel

| DeclaracaoProcedimento

| Declaracao "," Declaracao

* 1. Semântica
  2. Descrição
  3. Exemplo

1. Declaração Variável
   1. Sintaxe

DeclaracaoVariavel ::= "var" Id "=" Expressao

| "pointer" Id "=" "^"Tipo

* 1. Semântica
  2. Descrição
  3. Exemplo

1. Declaração Procedimento
   1. Sintaxe

DeclaracaoProcedimento ::= "proc" Id "(" [ ListaDeclaracaoParametro ] ")" "{" Comando "}"

* 1. Semântica
  2. Descrição
  3. Exemplo

1. Lista Declaração Parâmetro
   1. Sintaxe

ListaDeclaracaoParametro ::= Id ["^"]Tipo

| ListaDeclaracaoParametro "," ListaDeclaracaoParametro

* 1. Semântica
  2. Descrição
  3. Exemplo

1. Tipos
2. Tipo
3. Sintaxe

Tipo ::= "string"

| "int"

| "boolean"

1. Semântica
2. Descrição
3. Exemplo
4. Valor Inteiro
5. Sintaxe
6. Semântica
7. Descrição
8. Exemplo
9. Valor Booleano
10. Sintaxe
11. Semântica
12. Descrição
13. Exemplo
14. Valor String
15. Sintaxe
16. Semântica
17. Descrição
18. Exemplo
19. Id
20. Sintaxe
21. Semântica
22. Descrição
23. Exemplo

PALAVRAS RESERVADAS

1. “Skip”

2. “,”

3. “:=”

4. “&”

5. “\*”

6. “-”

7. “!”

8. “+”

9. “&&”

10 “||”

11. “==”

12. “++”

13. “,”

14. “var”

15. “=”

16. “pointer”

17. “^”

18. “proc”

19. “[]”

20. “{}”

21. “string”

22. “int”

23. “boolean”

24. “while”

25. “do”

26. “for”

27. “to”

28. “if”

29. “then”

30. “else”

31. “repeat”

32. “until”

33. “write”

34. “read”

35. “call”

PROGRAMAS EXEMPLIFICATIVOS

1. Programa 1
2. Programa 2
3. Programa 3

CONCLUSÃO

BIBLIOGRAFIA