

Paradigmas de Programação da linguagem LUA

Projeto Integrador VI - Paradigmas de linguagens de
programação

Mário Sergio e Pedro Martins

22 de setembro de 2013

Introdução

Motivações
Objetivos

História

Análise Léxica e Sintática

Construções léxicas
Sintaxe

Semântica das variáveis

Variáveis
Vinculação
Verificação de Tipos
Escopo

- ▶ Este projeto acadêmico se refere ao desenvolvimento de um estudo e pesquisa, relativo aos paradigmas e conceitos da linguagem de programação Lua.



Figura 1: Logo - Lua Org

Introdução

Motivações
Objetivos

História

Análise Léxica e Sintática

Construções léxicas
Sintaxe

Semântica das variáveis

Variáveis
Vinculação
Verificação de Tipos
Escopo

Motivações

⇒ Linguagem dinâmica, similar à python, ou seja, de fácil entendimento;

Introdução

Motivações
Objetivos

História

Análise Léxica e Sintática

Construções léxicas
Sintaxe

Semântica das variáveis

Variáveis
Vinculação
Verificação de Tipos
Escopo

Motivações

- ⇒ Linguagem dinâmica, similar à python, ou seja, de fácil entendimento;
- ⇒ Única linguagem criada fora do eixo de países desenvolvidos com relevância internacional;

Introdução

Motivações
Objetivos

História

Análise Léxica e
Sintática
Construções léxicas
Sintaxe

Semântica das variáveis

Variáveis
Vinculação
Verificação de Tipos
Escopo

Motivações

- ⇒ Linguagem dinâmica, similar à python, ou seja, de fácil entendimento;
- ⇒ Única linguagem criada fora do eixo de países desenvolvidos com relevância internacional;
- ⇒ O nicho de aplicação de Lua é muito vasto;

Introdução

Motivações
Objetivos

História

Análise Léxica e
Sintática
Construções léxicas
Sintaxe

Semântica das variáveis

Variáveis
Vinculação
Verificação de Tipos
Escopo

Motivações

- ⇒ Linguagem dinâmica, similar à python, ou seja, de fácil entendimento;
- ⇒ Única linguagem criada fora do eixo de países desenvolvidos com relevância internacional;
- ⇒ O nicho de aplicação de Lua é muito vasto;
- ⇒ Leve, com apenas 20.000 linhas de código C que podem ser construídos em um intérprete executável 182K em um Linux;

Introdução

Motivações
Objetivos

História

Análise Léxica e
Sintática
Construções léxicas
Sintaxe

Semântica das variáveis

Variáveis
Vinculação
Verificação de Tipos
Escopo

Motivações

⇒ Portável, é utilizada em qualquer plataforma com um compilador C ANSI. Lua Pode ser usada em:

Introdução

Motivações
Objetivos

História

Análise Léxica e Sintática

Construções léxicas
Sintaxe

Semântica das variáveis

Variáveis
Vinculação
Verificação de Tipos
Escopo

Motivações

- ⇒ Portável, é utilizada em qualquer plataforma com um compilador C ANSI. Lua Pode ser usada em:
- ▶ Microcontroladores;
 - ▶ Plataformas móveis; Consoles de jogos;
 - ▶ Navegadores (traduzido para JavaScript);
 - ▶ Aplicações de TV digital;
 - ▶ Programas de manipulação de imagens.

Introdução

Motivações
Objetivos

História

Análise Léxica e
Sintática
Construções léxicas
Sintaxe

Semântica das variáveis

Variáveis
Vinculação
Verificação de Tipos
Escopo

Exemplos de aplicações Lua



Figura 2: teste



Figura 3: Subfiguras

Introdução

Motivações
Objetivos

História

Análise Léxica e Sintática

Construções léxicas
Sintaxe

Semântica das variáveis

Variáveis
Vinculação
Verificação de Tipos
Escopo

Objetivos

- ▶ O objetivo do principal do projeto é aplicar os conhecimentos obtidos na disciplina de paradigmas de linguagem de programação à linguagem LUA.

Introdução

Motivações
Objetivos

História

Análise Léxica e Sintática

Construções léxicas
Sintaxe

Semântica das variáveis

Variáveis
Vinculação
Verificação de Tipos
Escopo

Objetivos

- ▶ O objetivo do principal do projeto é aplicar os conhecimentos obtidos na disciplina de paradigmas de linguagem de programação à linguagem LUA.
 - ⇒ Levantar os paradigmas de programação da linguagem;

Introdução

Motivações
Objetivos

História

Análise Léxica e Sintática

Construções léxicas
Syntaxe

Semântica das variáveis

Variáveis
Vinculação
Verificação de Tipos
Escopo

Objetivos

- ▶ O objetivo do principal do projeto é aplicar os conhecimentos obtidos na disciplina de paradigmas de linguagem de programação à linguagem LUA.
 - ⇒ Levantar os paradigmas de programação da linguagem;
 - ⇒ Analisar Sintaxe e Semântica;

Introdução

Motivações
Objetivos

História

Análise Léxica e Sintática

Construções léxicas
Sintaxe

Semântica das variáveis

Variáveis
Vinculação
Verificação de Tipos
Escopo

Objetivos

- ▶ O objetivo do principal do projeto é aplicar os conhecimentos obtidos na disciplina de paradigmas de linguagem de programação à linguagem LUA.
 - ⇒ Levantar os paradigmas de programação da linguagem;
 - ⇒ Analisar Sintaxe e Semântica;
 - ⇒ Explicar e exemplificar o funcionamento de variáveis. Tipos, sua vinculação, verificação de tipo e escopo;

Introdução

Motivações
Objetivos

História

Análise Léxica e Sintática

Construções léxicas
Sintaxe

Semântica das variáveis

Variáveis
Vinculação
Verificação de Tipos
Escopo

Objetivos

- ▶ O objetivo do principal do projeto é aplicar os conhecimentos obtidos na disciplina de paradigmas de linguagem de programação à linguagem LUA.
 - ⇒ Levantar os paradigmas de programação da linguagem;
 - ⇒ Analisar Sintaxe e Semântica;
 - ⇒ Explicar e exemplificar o funcionamento de variáveis. Tipos, sua vinculação, verificação de tipo e escopo;
 - ⇒ Entender as vantagens, desvantagens e as áreas a qual LUA melhor se aplica;

Introdução

Motivações
Objetivos

História

Análise Léxica e Sintática

Construções léxicas
Sintaxe

Semântica das variáveis

Variáveis
Vinculação
Verificação de Tipos
Escopo

Objetivos

- ▶ O objetivo do principal do projeto é aplicar os conhecimentos obtidos na disciplina de paradigmas de linguagem de programação à linguagem LUA.
 - ⇒ Levantar os paradigmas de programação da linguagem;
 - ⇒ Analisar Sintaxe e Semântica;
 - ⇒ Explicar e exemplificar o funcionamento de variáveis. Tipos, sua vinculação, verificação de tipo e escopo;
 - ⇒ Entender as vantagens, desvantagens e as áreas a qual LUA melhor se aplica;
 - ⇒ Criação códigos para exemplificar os conceitos apresentados.

Introdução

Motivações
Objetivos

História

Análise Léxica e Sintática

Construções léxicas
Sintaxe

Semântica das variáveis

Variáveis
Vinculação
Verificação de Tipos
Escopo

Projeto Inicial

- ⇒ A construção da linguagem veio de um projeto entre a PETROBRAS e a PUC-RIO, a fim de produzir um programa de interfaces gráficas para várias aplicações;

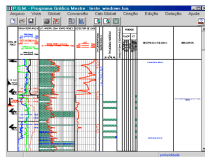


Figura 4: Logo - Lua Org

Introdução

Motivações
Objetivos

História

Análise Léxica e Sintática

Construções léxicas
Sintaxe

Semântica das variáveis

Variáveis
Vinculação
Verificação de Tipos
Escopo

Projeto Inicial

- ⇒ A construção da linguagem veio de um projeto entre a PETROBRAS e a PUC-RIO, a fim de produzir um programa de interfaces gráficas para várias aplicações;

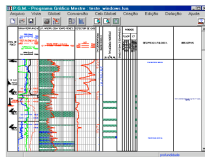


Figura 4: Logo - Lua Org

- ⇒ Logo surgiu o DEL - Linguagem para Especificação de Diálogos;

Introdução

Motivações
Objetivos

História

Análise Léxica e Sintática

Construções léxicas
Sintaxe

Semântica das variáveis

Variáveis
Vinculação
Verificação de Tipos
Escopo

Projeto Inicial

- ⇒ A construção da linguagem veio de um projeto entre a PETROBRAS e a PUC-RIO, a fim de produzir um programa de interfaces gráficas para várias aplicações;

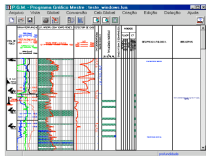


Figura 4: Logo - Lua Org

- ⇒ Logo surgiu o DEL - Linguagem para Especificação de Diálogos;
- ⇒ 'SOL' - Simple Object Language, uma linguagem para descrição de objetos, inspirada no bibTex.

```
type @track {x:number, y:number=23, z}
type @line {t:@track=@track(x=0), z:number*}

-- create an object 't1', of type 'track'
t1 = @track{y=9, x=10, s="hi!"}

l = @line{t=@track{x=t1.y, y=t1.x, s={2,3,4}}}
```

Introdução

Motivações
Objetivos

História

Ânálise Léxica e Sintática

Construções léxicas
Sintaxe

Semântica das variáveis

Variáveis
Vinculação
Verificação de Tipos
Escopo

Esforço

- ⇒ No entanto, DEL e SOL tinha várias limitações como, pouco recurso para construção de diálogos e pouca abstração de dados, se comparadas à linguagens contemporâneas;

Introdução

Motivações
Objetivos

História

Análise Léxica e Sintática

Construções léxicas
Sintaxe

Semântica das variáveis

Variáveis
Vinculação
Verificação de Tipos
Escopo

Esforço

- ⇒ No entanto, DEL e SOL tinha várias limitações como, pouco recurso para construção de diálogos e pouca abstração de dados, se comparadas à linguagens contemporâneas;
- ⇒ As propostas de solução era formular uma nova linguagem de configuração genérica com as seguintes características:

Introdução

Motivações
Objetivos

História

Análise Léxica e Sintática

Construções Léxicas
Sintaxe

Semântica das variáveis

Variáveis
Vinculação
Verificação de Tipos
Escopo

Esforço

- ⇒ No entanto, DEL e SOL tinha várias limitações como, pouco recurso para construção de diálogos e pouca abstração de dados, se comparadas à linguagens contemporâneas;
- ⇒ As propostas de solução era formular uma nova linguagem de configuração genérica com as seguintes características:
 - ▶ Facilmente acoplável;

Introdução

Motivações
Objetivos

História

Análise Léxica e Sintática

Construções léxicas
Sintaxe

Semântica das variáveis

Variáveis
Vinculação
Verificação de Tipos
Escopo

Esforço

- ⇒ No entanto, DEL e SOL tinha várias limitações como, pouco recurso para construção de diálogos e pouca abstração de dados, se comparadas à linguagens contemporâneas;
- ⇒ As propostas de solução era formular uma nova linguagem de configuração genérica com as seguintes características:
 - ▶ Facilmente acoplável;
 - ▶ Portável

Introdução

Motivações
Objetivos

História

Análise Léxica e
Sintática

Construções Léxicas
Sintaxe

Semântica das
variáveis

Variáveis
Vinculação
Verificação de Tipos
Escopo

Esforço

- ⇒ No entanto, DEL e SOL tinha várias limitações como, pouco recurso para construção de diálogos e pouca abstração de dados, se comparadas à linguagens contemporâneas;
- ⇒ As propostas de solução era formular uma nova linguagem de configuração genérica com as seguintes características:
 - ▶ Facilmente acoplável;
 - ▶ Portável
 - ▶ Simples e de sintaxe fácil

Introdução

Motivações
Objetivos

História

Análise Léxica e Sintática

Construções léxicas
Sintaxe

Semântica das variáveis

Variáveis
Vinculação
Verificação de Tipos
Escopo

⇒ O resultado desse projeto foi dado o nome LUA, como um contráste da antiga SOL.



Figura 6: Logo - Lua Org

Introdução

Motivações
Objetivos

História

Ânálise Léxica e Sintática

Construções léxicas
Sintaxe

Semântica das variáveis

Variáveis
Vinculação
Verificação de Tipos
Escopo

- ▶ Em Lua, os nomes podem ser qualquer cadeia de letras, dígitos, e sublinhados que não começam com um dígito;
- ▶ Os identificadores são usados para nomear variáveis e campos de tabelas;
- ▶ Lua é uma linguagem que diferencia letras minúsculas de maiúsculas;
- ▶ As seguintes cadeias denotam outros itens léxicos: `+ - * == = ~= < > <= >= <> [] ; : ,`
- ▶ Existem outras construções lexicas, porém estão mais ligadas à convenções da comunidade Lua.

- Aqui está a sintaxe completa de Lua na notação BNF estendida. (Ela não descreve as precedências dos operadores).

```
trecho ::= {comando [';']} [ultimocomando [';']]

bloco ::= trecho

comando ::= listavar '=' listaexp |
           chamadadefuncao |
           do bloco end |
           while exp do bloco end |
           repeat bloco until exp |
           if exp then bloco {elseif exp then bloco} [else bloco] end |
           for Nome '=' exp [, exp] do bloco end |
           for listadenomes in listaexp do bloco end |
           function nomedafuncao corpodafuncao |
           local function Nome corpodafuncao |
           local listadenomes ['=' listaexp]

ultimocomando ::= return [listaexp] | break

nomedafuncao ::= Nome { '.' Nome } [ ':' Nome ]

listavar ::= var { ',' var }

var ::= Nome | expprefixo '[' exp `']' | expprefixo '.' Nome

listadenomes ::= Nome { ',' Nome }

listaexp ::= {exp `','} exp

exp ::= nil | false | true | Numero | Cadeia | `...' | funcao |
       expprefixo | construtortabela | exp opbin exp | opunaria exp

exprefixo ::= var | chamadadefuncao | `(' exp `')'

chamadadefuncao ::= expprefixo args | expprefixo ':' Nome args
```

`args ::= '(' [listaexp] ')' | construtortabela | Cadeia`

`funcao ::= function corpodafuncao`

`corpodafuncao ::= '(' [listapar] ')' bloco end`

`listapar ::= listadenomes [' , ' ... '] | ' ... '`

`construtortabela ::= '{' [listadecampos] '}'`

`listadecampos ::= campo {separadordecampos campo} [separadordecampos]`

`campo ::= '[' exp ']' | '=' exp | Nome '=' exp | exp`

`separadordecampos ::= ' , ' | ' ; '`

`opbin ::= '+' | '-' | '*' | '/' | '^' | '%' | '..' |
< | <= | > | >= | == | ~= |
and | or`

`opunaria ::= '-' | not | '#'`

Introdução

Motivações
Objetivos

História

Análise Léxica e Sintática

Construções Léxicas
Sintaxe

Semântica das variáveis

Variáveis
Vinculação
Verificação de Tipos
Escopo

⇒ Em Lua existem três tipos de variáveis, sendo elas as seguintes.

- ▶ Variáveis locais;
- ▶ Variáveis globais;
- ▶ Variáveis de tabelas.

```
local X = 1 -- VARIÁVEL LOCAL
Y = 10 -- VARIÁVEL GLOBAL

Tabela = {} -- DECLARAÇÃO DE TABELA GLOBAL
Tabela["primeiro_indice"] = 100 -- VARIÁVEL DE TABELA
```

Introdução

Motivações
Objetivos

História

Análise Léxica e Sintática

Construções léxicas
Sintaxe

Semântica das variáveis

Variáveis
Vinculação
Verificação de Tipos
Escopo

⇒ Em Lua existem três tipos de variáveis, sendo elas as seguintes.

- ▶ Variáveis locais;
- ▶ Variáveis globais;
- ▶ Variáveis de tabelas.

⇒ A diferença entre variáveis locais e globais é o uso da palavra reservada 'local', antes do nome da variável.

```
local X = 1 -- VARIÁVEL LOCAL
Y = 10 -- VARIÁVEL GLOBAL

Tabela = {} -- DECLARAÇÃO DE TABELA GLOBAL
Tabela["primeiro_indice"] = 100 -- VARIÁVEL DE TABELA
```

Introdução

Motivações
Objetivos

História

Análise Léxica e Sintática

Construções léxicas
Sintaxe

Semântica das variáveis

Variáveis
Vinculação
Verificação de Tipos
Escopo

- ▶ Em Lua, os nomes podem ser qualquer cadeia de letras, dígitos, e sublinhados que não começam com um dígito;
- ▶ Os identificadores são usados para nomear variáveis e campos de tabelas;
- ▶ Lua é uma linguagem que diferencia letras minúsculas de maiúsculas;
- ▶ As seguintes cadeias denotam outros itens léxicos: + - * == = ~= < > <= >= () [] ; : ,
- ▶ Existem outras construções lexicais, porém estão mais ligadas à convenções da comunidade Lua.

- ▶ Em Lua, os nomes podem ser qualquer cadeia de letras, dígitos, e sublinhados que não começam com um dígito;
- ▶ Os identificadores são usados para nomear variáveis e campos de tabelas;
- ▶ Lua é uma linguagem que diferencia letras minúsculas de maiúsculas;
- ▶ As seguintes cadeias denotam outros itens léxicos: `+ - * == = ~= < > <= >= <> [] ; : ,`
- ▶ Existem outras construções lexicais, porém estão mais ligadas à convenções da comunidade Lua.

- ▶ Em Lua, os nomes podem ser qualquer cadeia de letras, dígitos, e sublinhados que não começam com um dígito;
- ▶ Os identificadores são usados para nomear variáveis e campos de tabelas;
- ▶ Lua é uma linguagem que diferencia letras minúsculas de maiúsculas;
- ▶ As seguintes cadeias denotam outros itens léxicos: + - * == = ~= < > <= >= () [] ; : ,
- ▶ Existem outras construções lexicas, porém estão mais ligadas à convenções da comunidade Lua.