

# Paradigmas de Programação da linguagem LUA

## Projeto Integrador VI

Mário Sergio e Pedro Martins

7 de outubro de 2013

### Roteiro

#### Introdução

- Objetivos
- Motivações

#### História

#### Análise Léxica e Sintática

- Construções léxicas
- Sintaxe

#### Semântica das variáveis

- Variáveis
- Vinculação
- Verificação de Tipos
- Escopo

#### Conclusão

- 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

## Roteiro

### Introdução

Objetivos  
Motivações

### História

### Análise Léxica e Sintática

Construções léxicas  
Sintaxe

### Semântica das variáveis

Variáveis  
Vinculação  
Verificação de Tipos  
Escopo

### Conclusão

## Roteiro

### Introdução

Objetivos  
Motivações

### História

### Análise Léxica e Sintática

Construções léxicas  
Sintaxe

### Semântica das variáveis

Variáveis  
Vinculação  
Verificação de Tipos  
Escopo

### Conclusão

- Objetivos
- Motivações
- História da Linguagem
- Análise Léxica e Sintática
- Semântica das variáveis
- 
- 
- 
- Considerações Finais

## Objetivos

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

### Roteiro

#### Introdução

Objetivos  
Motivações

#### História

#### Análise Léxica e Sintática

Construções léxicas  
Sintaxe

#### Semântica das variáveis

Variáveis  
Vinculação  
Verificação de Tipos  
Escopo

#### Conclusão

## 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;

### Roteiro

#### Introdução

Objetivos  
Motivações

#### História

#### Análise Léxica e Sintática

Construções léxicas  
Sintaxe

#### Semântica das variáveis

Variáveis  
Vinculação  
Verificação de Tipos  
Escopo

#### Conclusão

## 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;

### Roteiro

#### Introdução

Objetivos  
Motivações

#### História

#### Análise Léxica e Sintática

Construções léxicas  
Sintaxe

#### Semântica das variáveis

Variáveis  
Vinculação  
Verificação de Tipos  
Escopo

#### Conclusão

## 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;

### Roteiro

#### Introdução

Objetivos  
Motivações

#### História

#### Análise Léxica e Sintática

Construções léxicas  
Sintaxe

#### Semântica das variáveis

Variáveis  
Vinculação  
Verificação de Tipos  
Escopo

#### Conclusão

## 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;

### Roteiro

#### Introdução

Objetivos  
Motivações

#### História

#### Análise Léxica e Sintática

Construções léxicas  
Sintaxe

#### Semântica das variáveis

Variáveis  
Vinculação  
Verificação de Tipos  
Escopo

#### Conclusão



## 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;
  - \* Criar códigos para exemplificar os conceitos apresentados.

### Roteiro

#### Introdução

Objetivos  
Motivações

#### História

#### Análise Léxica e Sintática

Construções léxicas  
Sintaxe

#### Semântica das variáveis

Variáveis  
Vinculação  
Verificação de Tipos  
Escopo

#### Conclusão

## Motivações

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

Roteiro

Introdução

Objetivos  
Motivações

História

Análise Léxica e  
Sintática

Construções léxicas  
Sintaxe

Semântica das  
variáveis

Variáveis  
Vinculação  
Verificação de Tipos  
Escopo

Conclusão

## Motivações

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

Roteiro

Introdução

Objetivos  
Motivações

História

Análise Léxica e  
Sintática

Construções léxicas  
Sintaxe

Semântica das  
variáveis

Variáveis  
Vinculação  
Verificação de Tipos  
Escopo

Conclusão

## 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;

Roteiro

Introdução

Objetivos  
Motivações

História

Análise Léxica e  
Sintática

Construções léxicas  
Sintaxe

Semântica das  
variáveis

Variáveis  
Vinculação  
Verificação de Tipos  
Escopo

Conclusão

## 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;

Roteiro

Introdução

Objetivos  
Motivações

História

Análise Léxica e  
Sintática

Construções léxicas  
Sintaxe

Semântica das  
variáveis

Variáveis  
Vinculação  
Verificação de Tipos  
Escopo

Conclusão

## Motivações

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

Roteiro

Introdução

Objetivos  
Motivações

História

Análise Léxica e  
Sintática

Construções léxicas  
Sintaxe

Semântica das  
variáveis

Variáveis  
Vinculação  
Verificação de Tipos  
Escopo

Conclusão

## 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.

## Exemplos de aplicações Lua



Figura 2: Jogos feitos em Lua



Figura 3: Aplicações Lua

Roteiro

Introdução

Objetivos

Motivações

História

Análise Léxica e  
Sintática

Construções léxicas  
Sintaxe

Semântica das  
variáveis

Variáveis

Vinculação

Verificação de Tipos  
Escopo

Conclusão



## 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: Programa Gráfico Mestre

### Roteiro

#### Introdução

Objetivos  
Motivações

#### História

#### Análise Léxica e Sintática

Construções léxicas  
Sintaxe

#### Semântica das variáveis

Variáveis  
Vinculação  
Verificação de Tipos  
Escopo

#### Conclusão

## 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: Programa Gráfico Mestre

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

### Roteiro

#### Introdução

Objetivos  
Motivações

#### História

#### Análise Léxica e Sintática

Construções léxicas  
Sintaxe

#### Semântica das variáveis

Variáveis  
Vinculação  
Verificação de Tipos  
Escopo

#### Conclusão

## 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;

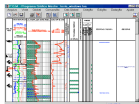


Figura 4: Programa Gráfico Mestre

- 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=8), z:number*)
-- create an object 't1', of type 'track'
t1 = @track(y=9, x=10, z="hi!")
l = @line(t=@track(x=t1.y, y=t1.x), z={2,3,4})
```

Figura 5: SOL

### Roteiro

### Introdução

Objetivos  
Motivações

### História

### Análise Léxica e Sintática

Construções léxicas  
Sintaxe

### Semântica das variáveis

Variáveis  
Vinculação  
Verificação de Tipos  
Escopo

### Conclusão

## Esforço

- No entanto, DEL e SOL tinha várias limitações;

### Roteiro

#### Introdução

Objetivos  
Motivações

#### História

#### Análise Léxica e Sintática

Construções léxicas  
Sintaxe

#### Semântica das variáveis

Variáveis  
Vinculação  
Verificação de Tipos  
Escopo

#### Conclusão

## Esforço

- No entanto, DEL e SOL tinha várias limitações;
- As propostas de solução era formular uma nova linguagem de configuração genérica com as seguintes características:

### Roteiro

#### Introdução

Objetivos  
Motivações

#### História

#### Análise Léxica e Sintática

Construções léxicas  
Sintaxe

#### Semântica das variáveis

Variáveis  
Vinculação  
Verificação de Tipos  
Escopo

#### Conclusão

## Esforço

- No entanto, DEL e SOL tinha várias limitações;
- As propostas de solução era formular uma nova linguagem de configuração genérica com as seguintes características:
  - \* Facilmente acoplável;

### Roteiro

#### Introdução

Objetivos  
Motivações

#### História

#### Análise Léxica e Sintática

Construções léxicas  
Sintaxe

#### Semântica das variáveis

Variáveis  
Vinculação  
Verificação de Tipos  
Escopo

#### Conclusão

## Esforço

- No entanto, DEL e SOL tinha várias limitações;
- As propostas de solução era formular uma nova linguagem de configuração genérica com as seguintes características:
  - \* Facilmente acoplável;
  - \* Portável

### Roteiro

#### Introdução

Objetivos  
Motivações

#### História

#### Análise Léxica e Sintática

Construções léxicas  
Sintaxe

#### Semântica das variáveis

Variáveis  
Vinculação  
Verificação de Tipos  
Escopo

#### Conclusão

## Esforço

- No entanto, DEL e SOL tinha várias limitações;
- As propostas de solução era formular uma nova linguagem de configuração genérica com as seguintes características:
  - \* Facilmente acoplável;
  - \* Portável
  - \* Simples e de sintaxe fácil

### Roteiro

#### Introdução

Objetivos  
Motivações

#### História

#### Análise Léxica e Sintática

Construções léxicas  
Sintaxe

#### Semântica das variáveis

Variáveis  
Vinculação  
Verificação de Tipos  
Escopo

#### Conclusão



## Esforço

- No entanto, DEL e SOL tinha várias limitações;
- As propostas de solução era formular uma nova linguagem de configuração genérica com as seguintes características:
  - \* Facilmente acoplável;
  - \* Portável
  - \* Simples e de sintaxe fácil
- Envolvidos: Roberto Ierusalimsky, Luiz Henrique de Figueiredo e Waldemar Celes;

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



## Roteiro

### Introdução

Objetivos  
Motivações

### História

### Análise Léxica e Sintática

Construções léxicas  
Sintaxe

### Semântica das variáveis

Variáveis  
Vinculação  
Verificação de Tipos  
Escopo

### Conclusão

## Roteiro

### Introdução

Objetivos  
Motivações

### História

### Análise Léxica e Sintática

Construções léxicas  
Sintaxe

### Semântica das variáveis

Variáveis  
Vinculação  
Verificação de Tipos  
Escopo

### Conclusão

- Em Lua, os nomes podem ser qualquer cadeia de letras, dígitos, e sublinhados que não começam com um dígito;

## Roteiro

### Introdução

Objetivos  
Motivações

### História

### Análise Léxica e Sintática

Construções léxicas  
Sintaxe

### Semântica das variáveis

Variáveis  
Vinculação  
Verificação de Tipos  
Escopo

### Conclusão

- 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;

## Roteiro

### Introdução

Objetivos  
Motivações

### História

### Análise Léxica e Sintática

Construções léxicas  
Sintaxe

### Semântica das variáveis

Variáveis  
Vinculação  
Verificação de Tipos  
Escopo

### Conclusão

- 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;

- 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 | '#'
```

## Roteiro

## Introdução

Objetivos  
Motivações

## História

## Análise Léxica e Sintática

Construções léxicas  
Sintaxe

## Semântica das variáveis

Variáveis  
Vinculação  
Verificação de Tipos  
Escopo

## Conclusão

- Em Lua existem três tipos de variáveis, sendo elas as seguintes:

## Roteiro

### Introdução

Objetivos  
Motivações

### História

### Análise Léxica e Sintática

Construções léxicas  
Sintaxe

### Semântica das variáveis

Variáveis  
Vinculação  
Verificação de Tipos  
Escopo

### Conclusão



## Roteiro

### Introdução

Objetivos  
Motivações

### História

### Análise Léxica e Sintática

Construções léxicas  
Sintaxe

### Semântica das variáveis

Variáveis  
Vinculação  
Verificação de Tipos  
Escopo

### Conclusão

- Em Lua existem três tipos de variáveis, sendo elas as seguintes:
  - \* Variáveis locais;

## Roteiro

### Introdução

Objetivos  
Motivações

### História

### Análise Léxica e Sintática

Construções léxicas  
Sintaxe

### Semântica das variáveis

Variáveis  
Vinculação  
Verificação de Tipos  
Escopo

### Conclusão

- Em Lua existem três tipos de variáveis, sendo elas as seguintes:
  - \* Variáveis locais;
  - \* Variáveis globais;

## Roteiro

### Introdução

Objetivos  
Motivações

### História

### Análise Léxica e Sintática

Construções léxicas  
Sintaxe

### Semântica das variáveis

Variáveis  
Vinculação  
Verificação de Tipos  
Escopo

### Conclusão

- Em Lua existem três tipos de variáveis, sendo elas as seguintes:
  - \* Variáveis locais;
  - \* Variáveis globais;
  - \* Variáveis de tabelas.

## Roteiro

### Introdução

Objetivos  
Motivações

### História

### Análise Léxica e Sintática

Construções léxicas  
Sintaxe

### Semântica das variáveis

Variáveis  
Vinculação  
Verificação de Tipos  
Escopo

### Conclusão

- 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
```

- Lua é uma linguagem dinamicamente tipada;

## Roteiro

### Introdução

Objetivos  
Motivações

### História

### Análise Léxica e Sintática

Construções léxicas  
Sintaxe

### Semântica das variáveis

Variáveis  
Vinculação  
Verificação de Tipos  
Escopo

### Conclusão

## Roteiro

### Introdução

Objetivos  
Motivações

### História

### Análise Léxica e Sintática

Construções léxicas  
Sintaxe

### Semântica das variáveis

Variáveis  
Vinculação  
Verificação de Tipos  
Escopo

### Conclusão

- Lua é uma linguagem dinamicamente tipada;
- A linguagem trabalha com vinculação dinâmica de tipos;

## Roteiro

### Introdução

Objetivos  
Motivações

### História

### Análise Léxica e Sintática

Construções léxicas  
Sintaxe

### Semântica das variáveis

Variáveis  
Vinculação  
Verificação de Tipos  
Escopo

### Conclusão

- Lua é uma linguagem dinamicamente tipada;
- A linguagem trabalha com vinculação dinâmica de tipos;
- Existem oito tipos de dados básicos em Lua:

- Lua é uma linguagem dinamicamente tipada;
- A linguagem trabalha com vinculação dinâmica de tipos;
- Existem oito tipos de dados básicos em Lua:
  - \* nil - boolean - number - string- thread;



- Lua é uma linguagem dinamicamente tipada;
- A linguagem trabalha com vinculação dinâmica de tipos;
- Existem oito tipos de dados básicos em Lua:
  - \* nil - boolean - number - string- thread;
  - \* **function - userdata - table.**

- Lua é uma linguagem dinamicamente tipada;
- A linguagem trabalha com vinculação dinâmica de tipos;
- Existem oito tipos de dados básicos em Lua:
  - \* nil - boolean - number - string- thread;
  - \* **function - userdata - table.**
- O tempo de vida das variáveis é definido pelo fato de ela se global ou local;

- A verificação de tipos em Lua é feita em tempo de execução pelo interpretador Lua;

```
local valor1 = 10
local valor2 = "vinte"

if valor1 < valor2 then
    print "valor1 menor"
end
```

Figura 6: Trecho de código Lua

```
/home/mario/Desktop/bs.lua:30: attempt to compare number with string
traceback:
home/mario/Desktop/bs.lua:30: in main chunk
```

Figura 7: Log de execução Lua

## Roteiro

### Introdução

Objetivos  
Motivações

### História

### Análise Léxica e Sintática

Construções léxicas  
Sintaxe

### Semântica das variáveis

Variáveis  
Vinculação  
Verificação de Tipos  
Escopo

### Conclusão

- Lua é uma linguagem com escopo léxico;

## Roteiro

### Introdução

Objetivos  
Motivações

### História

### Análise Léxica e Sintática

Construções léxicas  
Sintaxe

### Semântica das variáveis

Variáveis  
Vinculação  
Verificação de Tipos  
**Escopo**

### Conclusão

- Lua é uma linguagem com escopo léxico;
- Baseia-se na sequência de chamadas de subprogramas;

## Roteiro

### Introdução

Objetivos  
Motivações

### História

### Ánálise Léxica e Sintática

Construções léxicas  
Sintaxe

### Semântica das variáveis

Variáveis  
Vinculação  
Verificação de Tipos  
Escopo

### Conclusão

- Lua é uma linguagem com escopo léxico;
- Baseia-se na sequência de chamadas de subprogramas;
- O escopo pode ser determinado em tempo de execução;

## Roteiro

### Introdução

Objetivos  
Motivações

### História

### Análise Léxica e Sintática

Construções léxicas  
Sintaxe

### Semântica das variáveis

Variáveis  
Vinculação  
Verificação de Tipos  
Escopo

### Conclusão

- Lua é uma linguagem com escopo léxico;
- Baseia-se na sequência de chamadas de subprogramas;
- O escopo pode ser determinado em tempo de execução;

## Roteiro

### Introdução

Objetivos  
Motivações

### História

### Análise Léxica e Sintática

Construções léxicas  
Sintaxe

### Semântica das variáveis

Variáveis  
Vinculação  
Verificação de Tipos  
Escopo

### Conclusão

- Lua é uma linguagem com escopo léxico;
- Baseia-se na sequência de chamadas de subprogramas;
- O escopo pode ser determinado em tempo de execução;
- Variáveis locais podem ser livremente acessadas por funções definidas dentro do seu escopo ou bloco;

```
x = 10          -- variável global
do             -- bloco novo
  local x = x   -- novo 'x', com valor 10
  print(x)      --> 10
  x = x+1
do             -- outro bloco
  local x = x+1 -- outro 'x'
  print(x)      --> 12
end
print(x)        --> 11
end
print(x)        --> 10 (o x global)
```

Figura 8: Trecho de código Lua

## Roteiro

### Introdução

Objetivos

Motivações

### História

### Análise Léxica e Sintática

Construções léxicas

Sintaxe

### Semântica das variáveis

Variáveis

Vinculação

Verificação de Tipos

Escopo

### Conclusão



- A iniciação do projeto foi a maior motivação para continuação do trabalho;

## Roteiro

### Introdução

Objetivos  
Motivações

### História

### Análise Léxica e Sintática

Construções léxicas  
Sintaxe

### Semântica das variáveis

Variáveis  
Vinculação  
Verificação de Tipos  
Escopo

### Conclusão

- A iniciação do projeto foi a maior motivação para continuação do trabalho;
- O grupo compreendeu a complexidade de se aplicar os conceitos em um projeto de linguagem;

## Roteiro

### Introdução

Objetivos  
Motivações

### História

### Análise Léxica e Sintática

Construções léxicas  
Sintaxe

### Semântica das variáveis

Variáveis  
Vinculação  
Verificação de Tipos  
Escopo

### Conclusão

- A iniciação do projeto foi a maior motivação para continuação do trabalho;
- O grupo compreendeu a complexidade de se aplicar os conceitos em um projeto de linguagem;

## • Próximos Passos

### Roteiro

### Introdução

Objetivos  
Motivações

### História

### Análise Léxica e Sintática

Construções léxicas  
Sintaxe

### Semântica das variáveis

Variáveis  
Vinculação  
Verificação de Tipos  
Escopo

### Conclusão

- A iniciação do projeto foi a maior motivação para continuação do trabalho;
- O grupo compreendeu a complexidade de se aplicar os conceitos em um projeto de linguagem;

## • Próximos Passos

- \* Levantar os aspectos dos tipos de dados da linguagem Lua;

### Roteiro

#### Introdução

Objetivos  
Motivações

#### História

#### Análise Léxica e Sintática

Construções léxicas  
Sintaxe

#### Semântica das variáveis

Variáveis  
Vinculação  
Verificação de Tipos  
Escopo

#### Conclusão

- A iniciação do projeto foi a maior motivação para continuação do trabalho;
- O grupo compreendeu a complexidade de se aplicar os conceitos em um projeto de linguagem;

## • Próximos Passos

- \* Levantar os aspectos dos tipos de dados da linguagem Lua;
- \* Verificar a implementação e o comportamento dos subprogramas;

Roteiro

Introdução

Objetivos  
Motivações

História

Análise Léxica e  
Sintática

Construções léxicas  
Sintaxe

Semântica das  
variáveis

Variáveis  
Vinculação  
Verificação de Tipos  
Escopo

Conclusão

- A iniciação do projeto foi a maior motivação para continuação do trabalho;
- O grupo compreendeu a complexidade de se aplicar os conceitos em um projeto de linguagem;

## • Próximos Passos

- \* Levantar os aspectos dos tipos de dados da linguagem Lua;
- \* Verificar a implementação e o comportamento dos subprogramas;
- \* Aprofundar no paradigma de 'Orientação à tabelas' de Lua.

Roteiro

Introdução

Objetivos  
Motivações

História

Análise Léxica e  
Sintática

Construções léxicas  
Sintaxe

Semântica das  
variáveis

Variáveis  
Vinculação  
Verificação de Tipos  
Escopo

Conclusão