

Lua

Uma linguagem de Programação simples,
rápida e bastante poderosa.

Por Thiago Almeida e Werton Guimarães



Agenda

- Motivação
- O que é Lua?
- História
- Características
 - Vantagens
 - Desvantagens
- Python vs Lua
- Exemplos
- Jogos e TV Digital
- Conclusões
- Referências



Motivação

- Lua é usada em muitas **aplicações industriais** (e.x.: Adobe Photoshop Lightroom), com ênfase em **sistemas embutidos** (e.x.: o middleware Ginga para TV digital) e **jogos** (e.x.: World of Warcraft).
- Lua é, atualmente, a linguagem de script mais usada em jogos.

O que é Lua?

- Lua é uma linguagem de script **imperativa**, leve, projetada para expandir aplicações em geral.
- Semelhança
 - Python
 - Ruby
 - Entre outras.

História

- Inteiramente projetada, implementada e desenvolvida no Brasil.
- PUC-Rio + Petrobrás = Parceria que deu certo.
 - Lua foi projetada pelo de time de desenvolvedores da Tecgraf da PUC-Rio.
 - Criada, a princípio, para ser usada em um projeto da Petrobrás.
 - Superação das expectativas.
 - Eficiência
 - Clareza
 - Facilidade no aprendizado

História

- Evolução
 - 1993 – Lua 1.0;
 - 1994 – Primeira versão para o público;
 - 1997 – Aplicações em jogos;
 - 2003 – Lua se destaca como a linguagem mais popular para criação de scripts de jogos. É lançado o livro “Programming in Lua”. Liberada a versão 5.0.1;
 - Adode e Social Media Press juntam-se à Lua no programa de patrocínio corporativo. Liberada versão 5.2(alfa).

Características

- Lua foi desenvolvida, inicialmente, para ser embutida em outra linguagem, para poder estendê-la.
 - C/C++,Java, Fortran.
- Linguagem modular.
- Ocupa pouco espaço no HD.
- Pequeno número de estruturas:
 - Booleanos, números e strings.
 - Estruturas de dados comuns:
 - Matrizes, conjuntos, tabelas, listas e registros.

Vantagens

- **Rápida** - Lua tem um ótimo desempenho comparada a outras linguagens de scripts interpretadas.
- **Portátil** - O mesmo trecho de código pode ser compilado em várias plataformas sem precisar de alterações.
- **Embutível** - Lua é rápida e pequena podendo ser embutida em diversas aplicações.
- **Pequena** - Lua 5.1.4 + documentação + exemplos = 860K

Desvantagens

- Mais demorado que C.
- Vulnerabilidade:
 - Injeção de comandos, condição de competição, corrupção de arquivos por código malicioso, referência insegura a tabelas, etc.



Python e Lua

- Em jogos
 - Python: 7%
 - Lua: 20%
 - World WarCraft – Blizzard
 - The Sims 2 e Sim City 4 - EA



	Lua	Python
Binary size	about 200KB (Lua version 5.1)	About 2MB (Python version 2.6)
Memory usage	Lua is more compact	
Speed	Lua is faster ^{*1}	
Multi-threading	Thread-safe	Poor-quality
Standard Library	Small & compact	Rich & powerful
Glue ability	Easy	Normal

Exemplos

- Básico
 - Hello World
 - Criar Função
 - Recursividade

```
print ( "Hello World")
```

```
function fact(n)  
  if n == 0 then  
    return 1  
  else  
    return n * fact(n - 1)  
  end  
end
```

Exemplos

- Básico
 - Hello World
 - Criar Função
 - Recursividade
 - Números Perfeitos

```
function perfeitos(n)
  cont=0
  x=0
  print('Os numeros perfeitos sao ')
  repeat
    x=x+1
    soma=0
    for i=1,(x-1) do
      if math.mod(x,i)==0 then
        soma=soma+i;
      end
    end
    if soma == x then
      print(x)
      cont = cont+1
    end
  until cont==n
  print('Pressione qualquer tecla para finalizar...')
end
```

Exemplos

- Básico
 - Hello World
 - Criar Função
 - Recursividade
 - Números Perfeitos
 - Vetor de Objetos

```
function Point(x, y)    -- "Point" object constructor
    return { x = x, y = y } -- Creates and returns a new object (table)
end
array = { Point(10, 20), Point(30, 40), Point(50, 60) } -- Creates array of points
print(array[2].y)
```

Exemplos

- Escopo Léxico

```
print("foo")
```

```
do
```

```
    local oldprint = print -- Grava a variável "print" em "oldprint"
```

```
    print = function(s) -- Redefine a função "print"
```

```
        if s == "foo" then
```

```
            oldprint("bar")
```

```
        else
```

```
            oldprint(s)
```

```
        end
```

```
    end
```

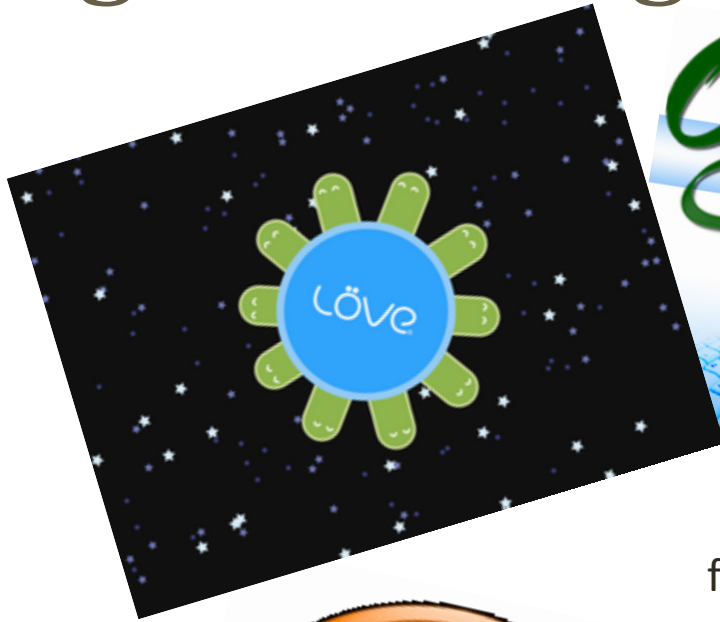
```
    oldprint("foo")
```

```
end
```

```
print("foo")
```

```
oldprint("foo")
```

Jogos e TV Digital

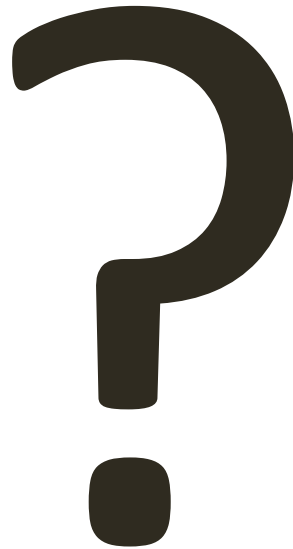


```
fisica = require("physics")  
physics.start()
```

```
retangulo = display.newRect(200, 200, 30, 70)  
retangulo:setFillColor(0, 0, 255)  
fisica.addBody(retangulo, "dynamic",  
{density=0, friction=1, bounce=0.4})
```

```
chao = display.newRect(0, 500, 460, 50)  
chao:setFillColor(255, 0, 0)  
fisica.addBody(chao, "static", {friction=1})
```

Dúvidas?



Referências

- <http://www.slideshare.net/cybrshin/lua-vs-python>
- www.lua.org
- [http://pt.wikipedia.org/wiki/Lua_\(linguagem_de_programaço](http://pt.wikipedia.org/wiki/Lua_(linguagem_de_programa%C3%A7%C3%A3o))