

Estruturas de Linguagem

Francisco Sant'Anna

francisco@ime.uerj.br

<http://github.com/fsantanna-uerj/EDL>

Forma de Avaliação

- Tarefas e Apresentações
 - são considerados “continuamente”
- Prova(s)
- Presença $\geq 75\%$

Online

- GitHub

- <https://github.com/fsantanna-uerj/EDL/>

- Lista de discussão

- <https://groups.google.com/d/forum/edl-uerj/>

Tarefa-01: Lista & Github

(até domingo 16/04)

- Cadastrar-se na lista da turma
 - <https://groups.google.com/d/forum/edl-uerj/>
- Dar um *Fork* no repositório da disciplina
 - <https://github.com/fsantanna-uerj/EDL/>
 - Habilitar a aba the *Issues*
 - <http://softwareengineering.stackexchange.com/questions/179468/forking-a-repo-on-github-but-allowing-new-issues-on-the-fork/179470#179470>
- Adicionar um arquivo tarefa-01/RESPOSTA.md
 - texto “pessoal” qualquer formatado em *Markdown*
 - <https://help.github.com/articles/basic-writing-and-formatting-syntax/>
- Mandar um e-mail para a lista com o link do seu projeto

Tarefa 02 - Artigo

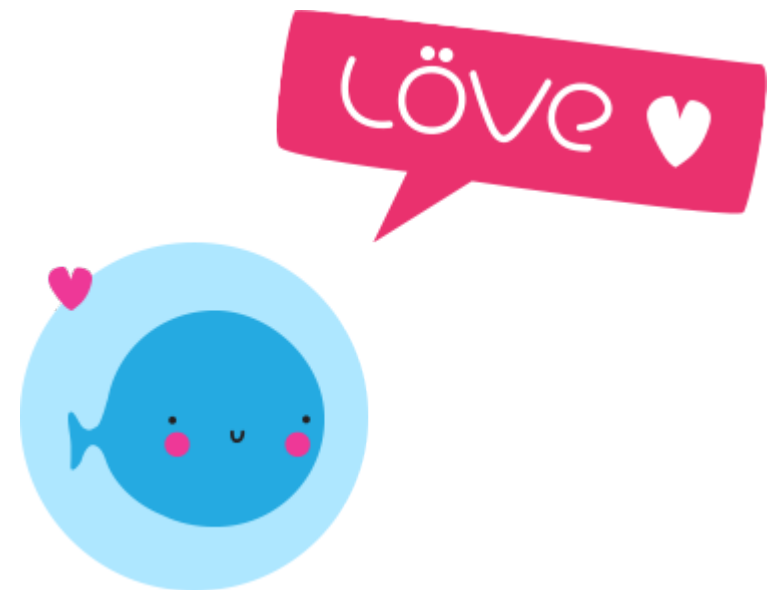
(até domingo 07/05)

- Escolher uma linguagem com a qual você **não está familiarizado**.
 - evitar duplicatas com outros colegas
 - instalar e escrever pequenos programas com a linguagem
 - usar pelo menos uma funcionalidade de alta expressividade
- Escrever um pequeno artigo (estilo *Wikipedia*):
 - [1.0] origens e influências (linha do tempo)
 - [1.0] classificação (imp/func/log/oo, est/din, usos)
 - [4.0] avaliação em comparação com linguagens que você conhece (read/write, expressividade)
 - [4.0] exemplos de código representativos (vs outra linguagem)
 - `trabalho-01/ARTIGO.md`
- Slides de apresentação (5-10 slides)
 - `trabalho-01/slides.pdf`

Tarefa 03 - Lua/Löve

(até domingo 30/04)

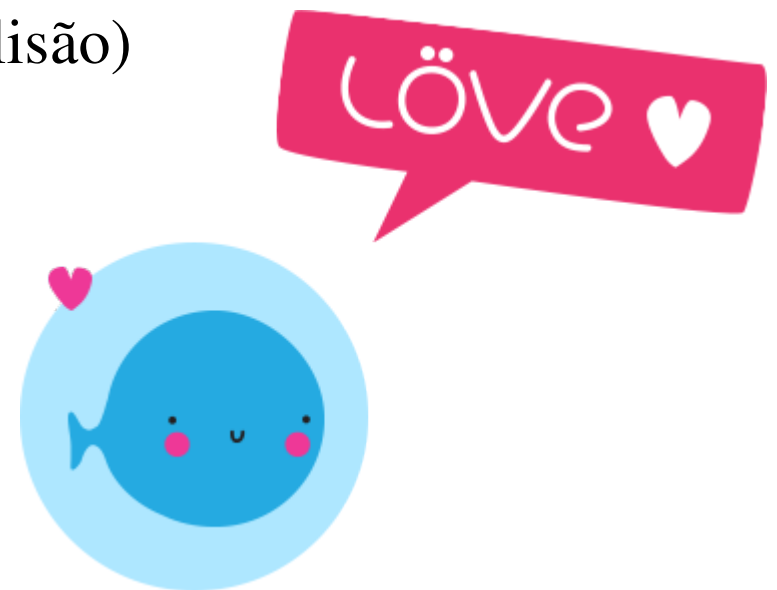
- <https://love2d.org/>
- Mostrar instalação/execução do ambiente:
 - Código no GitHub
 - Screenshot no GitHub



Trabalho 04 – Joguinho

(até domingo 14/05)

- Individual
- Fazer um jogo qualquer
 - Teclado e/ou Mouse
 - Imagens e/ou Retângulos
 - Animações (i.e., tempo como input)
 - Interação entre objetos (e.g., colisão)



Trabalho 05 – Bindings

(até domingo 21/05)

- Para cada “binding time” de Lua (*design, compile, run*), identificar no joguinho dois exemplos (com características diferentes).
- Adicionar comentários no próprio código fonte do jogo com as identificações e explicações.
- Exemplo:

```
function inc (v)
    return v + 1
end
-- Nome: variável “v”
-- Propriedade: endereço
-- Binding time: execução
-- Explicação: dado que “v” é uma variável
-- local de uma função, seu endereço só pode
-- ser determinado em tempo de execução.
```


Trabalho 06 - Tipos de Dados

(até domingo 04/06)

- Individual
- Identificar no jogo valores de tipos de dados não primitivos (pelo menos 4 diferentes)
 - enumeração, registro, tupla, array, dicionário, união
- Caso não existam 4 diferentes, altere o jogo para que tenham
- Usar comentários para facilitar a identificação
 - - - trabalho - 06

Trabalho 07 - Arrays

(até domingo 11/06)

- Individual
- Adicionar ao jogo uma coleção dinâmica de objetos
 - criar novos objetos periodicamente
 - timer ou evento (e.g., tecla pressionada)
 - remover objetos
 - timer ou evento (e.g., colisão)
 - objetos devem interagir entre si
 - e.g., colisão
- Descrever o ciclo de vida da coleção e de um objeto
 - escopo, tempo de vida, alocação/desalocação

Tarefas

- Tarefa-01: **Dom, 16/04**: Lista & GitHub
- Tarefa-02: **Dom, 07/05**: Artigo
- Tarefa-03: **Dom, 30/04**: Instalação Löve
- Tarefa-04: **Dom, 14/05**: Jogoinho
- Tarefa-05: **Dom, 21/05**: Bindings
- Tarefa-06: **Dom, 04/06**: Tipos de Dados
- Tarefa-07: **Dom, 11/06**: Arrays

Provas

Bibliografia

- Concepts of Programming Languages
 - 11th edição
 - Robert W. Sebesta
- `https://www.pearsonhighered.com/program/Sebesta-Concepts-of-Programming-Languages-11th-Edition/PGM270801.html`

Calendário

- 10/04, 12/04

- Introdução

- 17/04, 19/04

- Sintaxe, Semântica, Avaliação, Lua

- 24/04

- Lua, Löve

- 26/04

- Lab

- 03/05

- Bindings

- 08/05

- Ciclo de Vida

- 10/05

- Escopo

- 15/05

- Dados

