

löve

semana 9

LÖVE é uma framework de
desenvolvimento de jogos 2d em lua.

Apenas um único arquivo é
necessário para que LÖVE funcione:

`main.lua`

para rodar nosso jogo da linha de
comando podemos utilizar o
comando:

```
love caminho/até/o/diretório
```

ou, caso já estivermos no diretório:

```
love .
```

LÖVE funciona através de uma abordagem modular. Diversos módulos diferentes nos proporcionam diferentes funcionalidades.

love.graphics, por exemplo, é responsável por desenhar formas, imagens e pelo gerenciamento da geometria em nossa tela.

love.audio providencia uma interface de áudio através da qual podemos inicializar e tocar arquivos de som.

love.window providencia uma
interface com a janela do nosso jogo.

LÖVE possui muitos outros módulos
que nos permitem realizar muitas
outras coisas:

love.filesystem

love.event

love.data

love.joystick

love.mouse

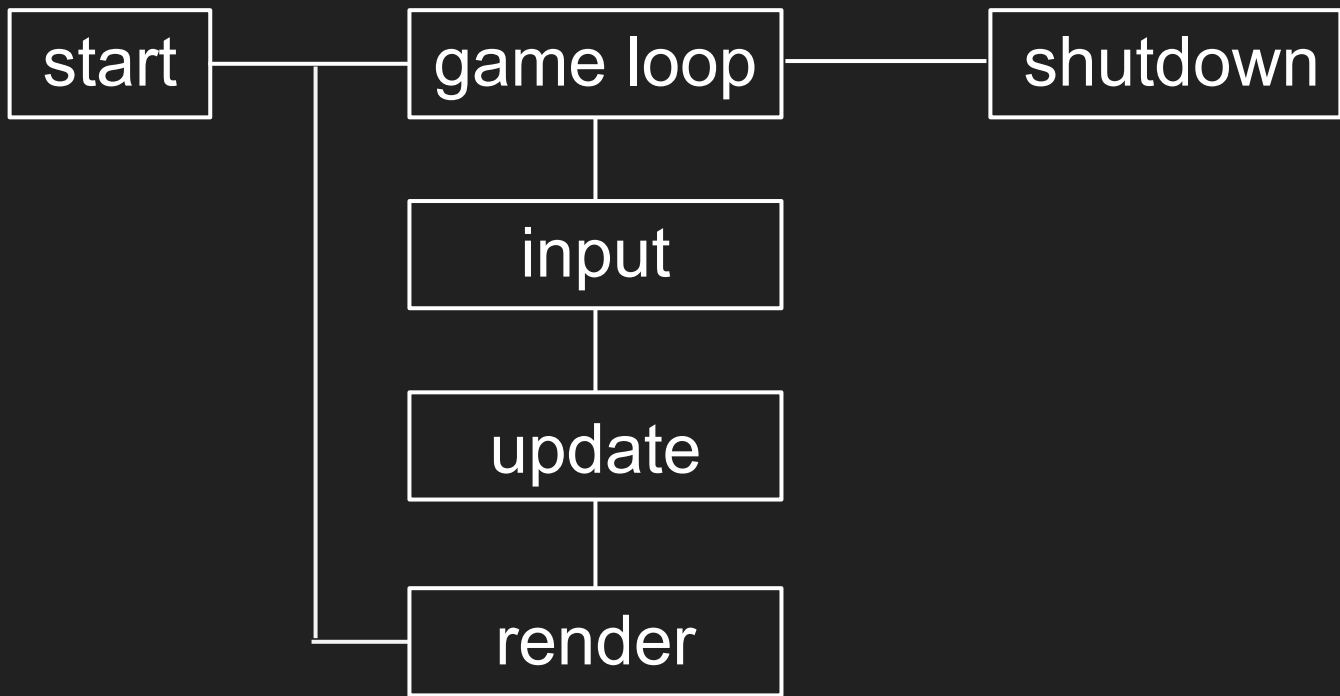
love.keyboard

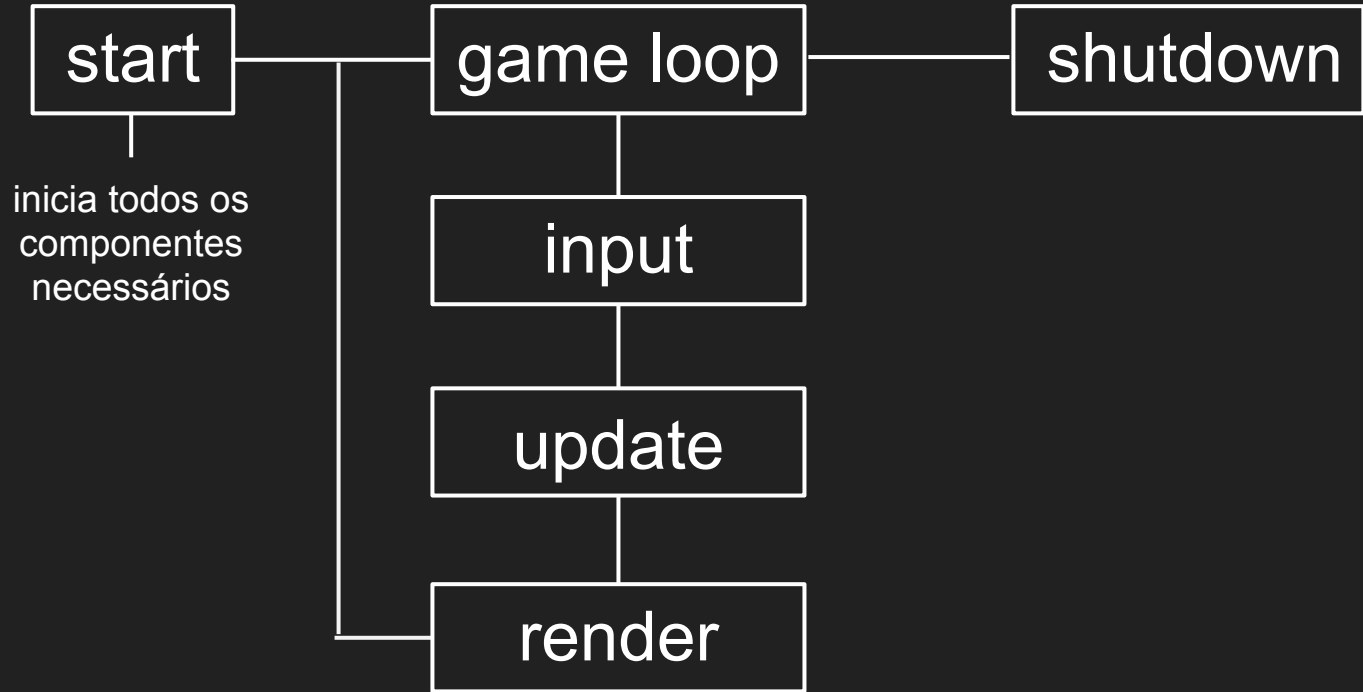
...

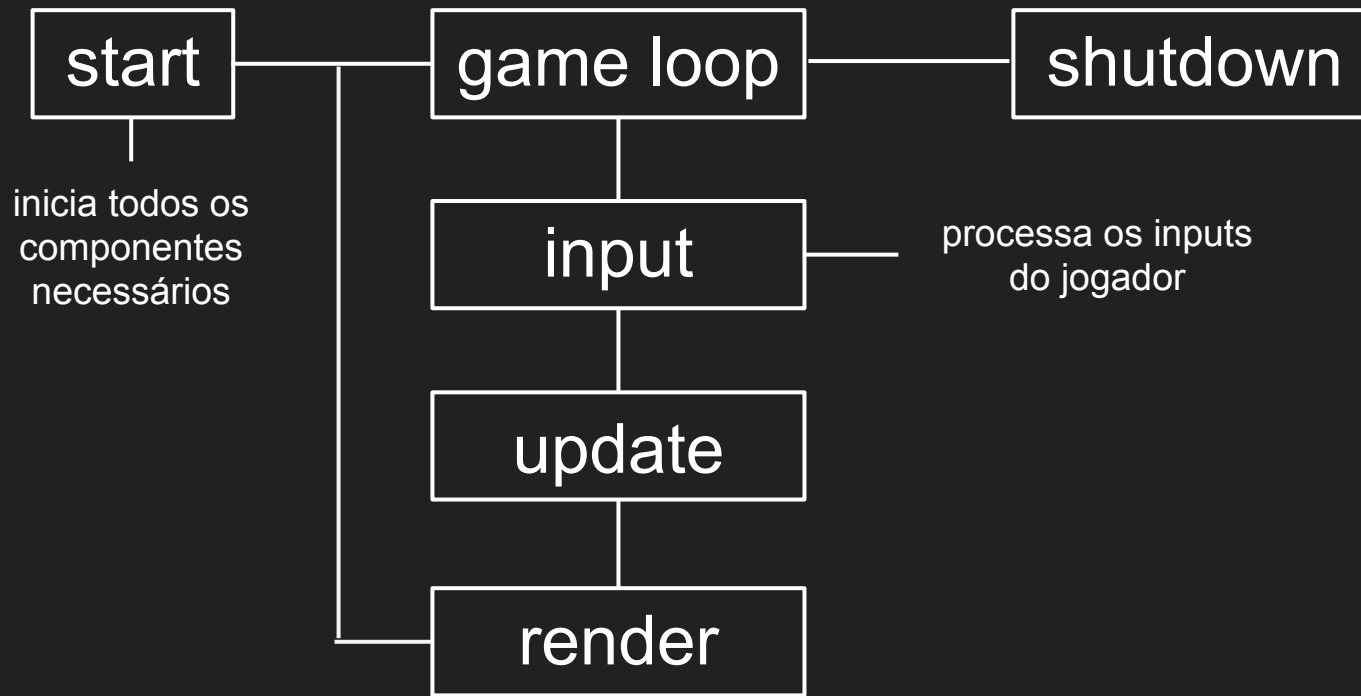
Outra parte importante de LÖVE são
as callback functions.

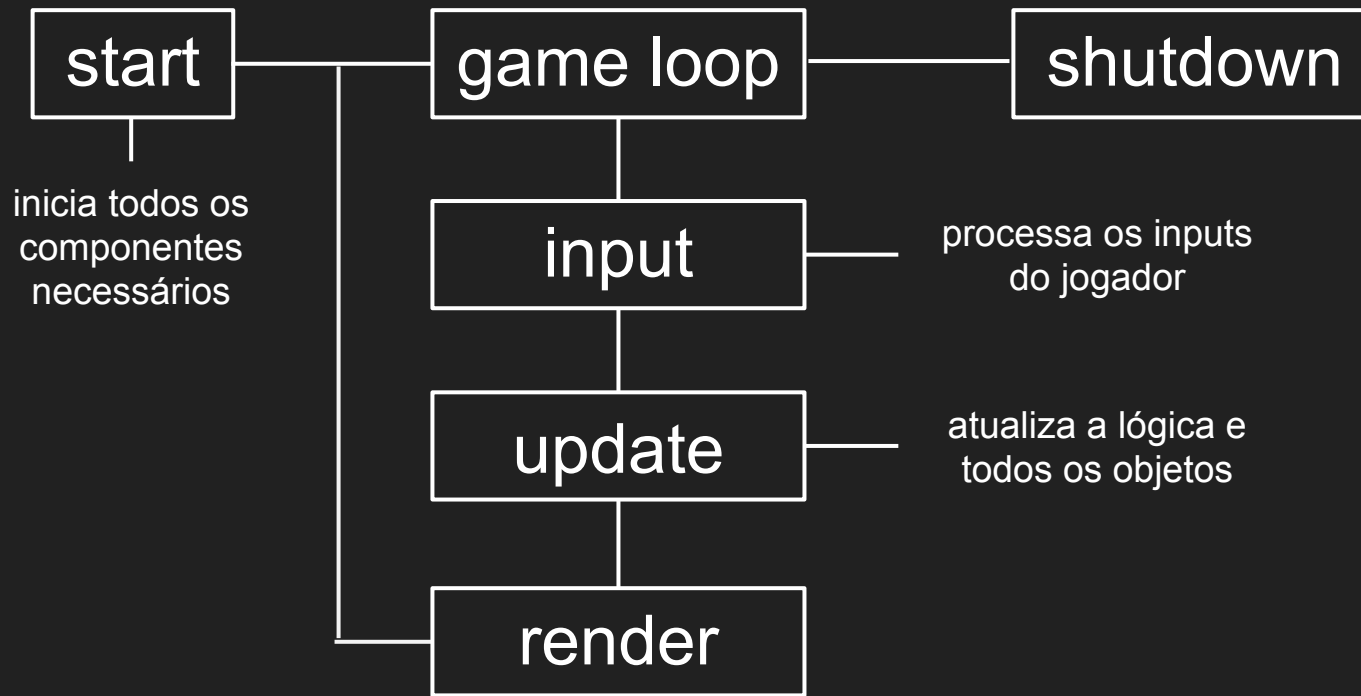
Diferente de funções convencionais,
que são chamadas por nós, as
callback functions são chamadas por
LÖVE em momentos específicos.

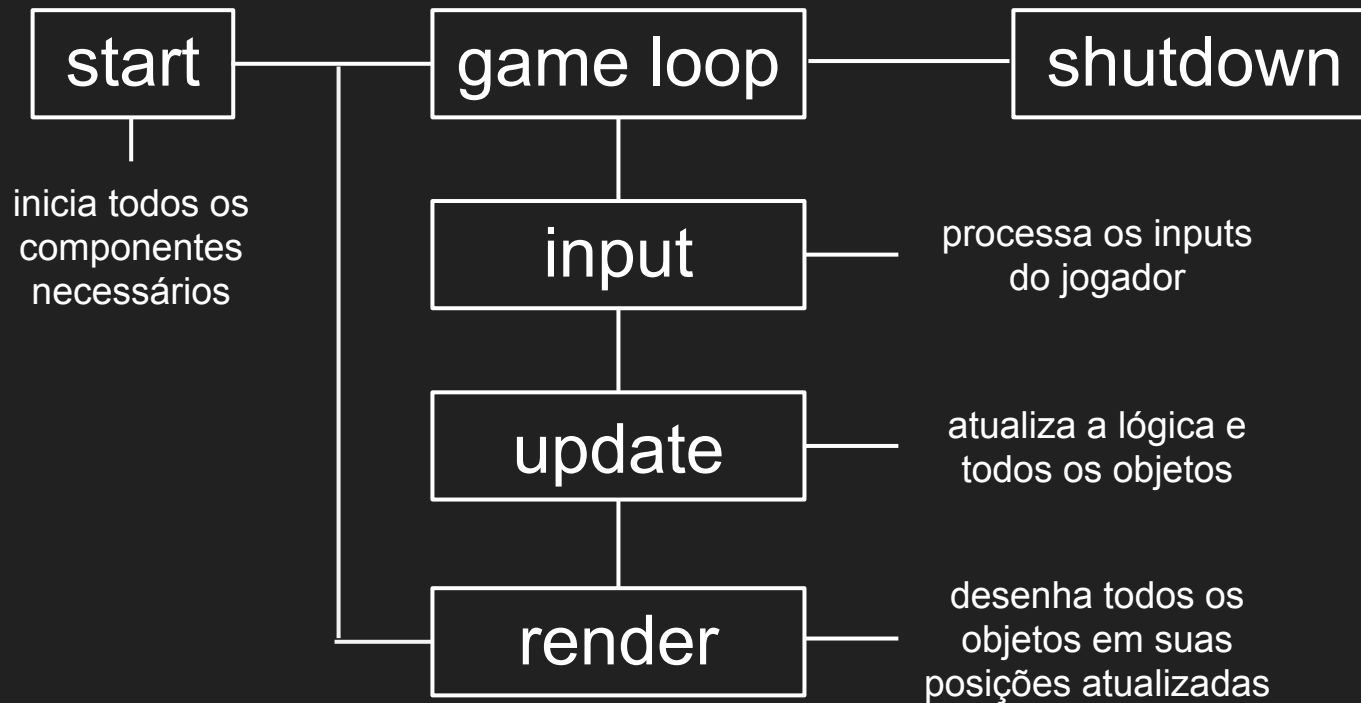
Algumas *callback functions* são chamadas a cada frame do nosso jogo, enquanto outras são chamadas apenas uma vez, ou em momentos específicos (quando uma tecla é apertada, por exemplo).

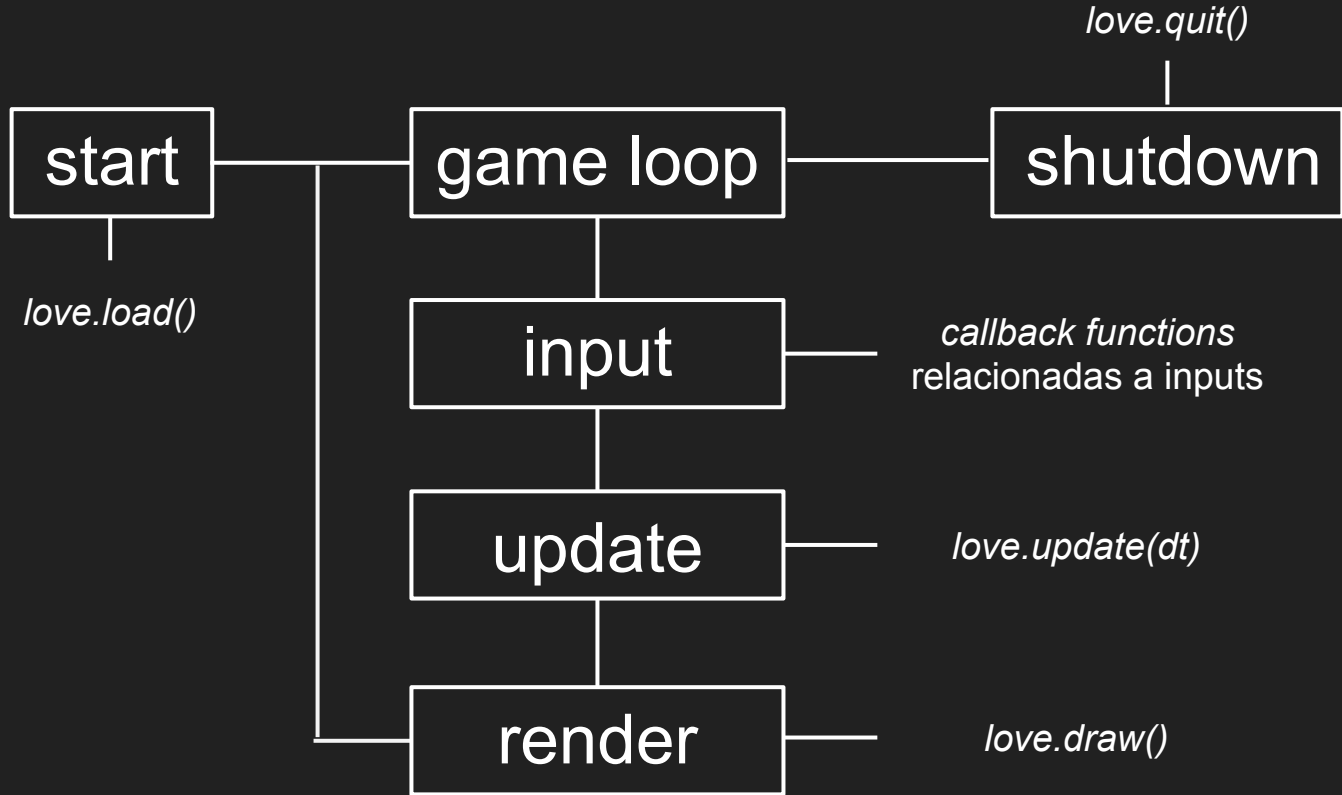












love.load()

Chamada uma única vez quando iniciamos o jogo. Usamos esta função para inicializar os componentes do nosso jogo ou para rodar qualquer código que deve ser executado apenas uma vez.

love.update(dt)

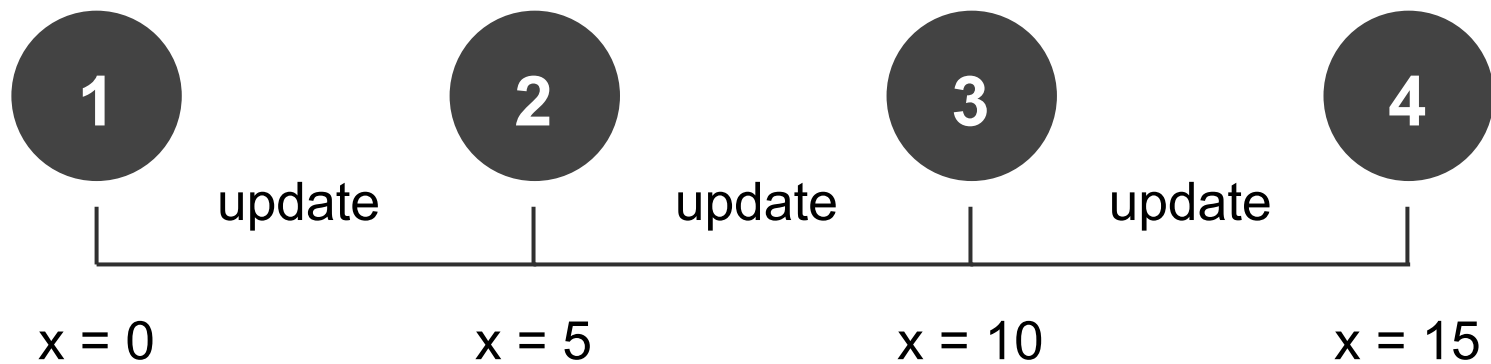
Chamada em toda iteração do game loop. Nesta função reside toda a lógica do jogo (movimentação, controle de estado, pontuação, vida, etc).

dt = delta time

Delta Time é o tempo de processamento entre a frame atual e a última frame. Utilizar delta time permite que nosso jogo seja executado com velocidade constante, independente da performance da máquina utilizada.

$$dt = 1$$

$$x = x + 5 * dt$$



$$dt = 3$$

$$x = x + 5 * dt$$



$$x = 0$$

update



$$x = 15$$

love.draw()

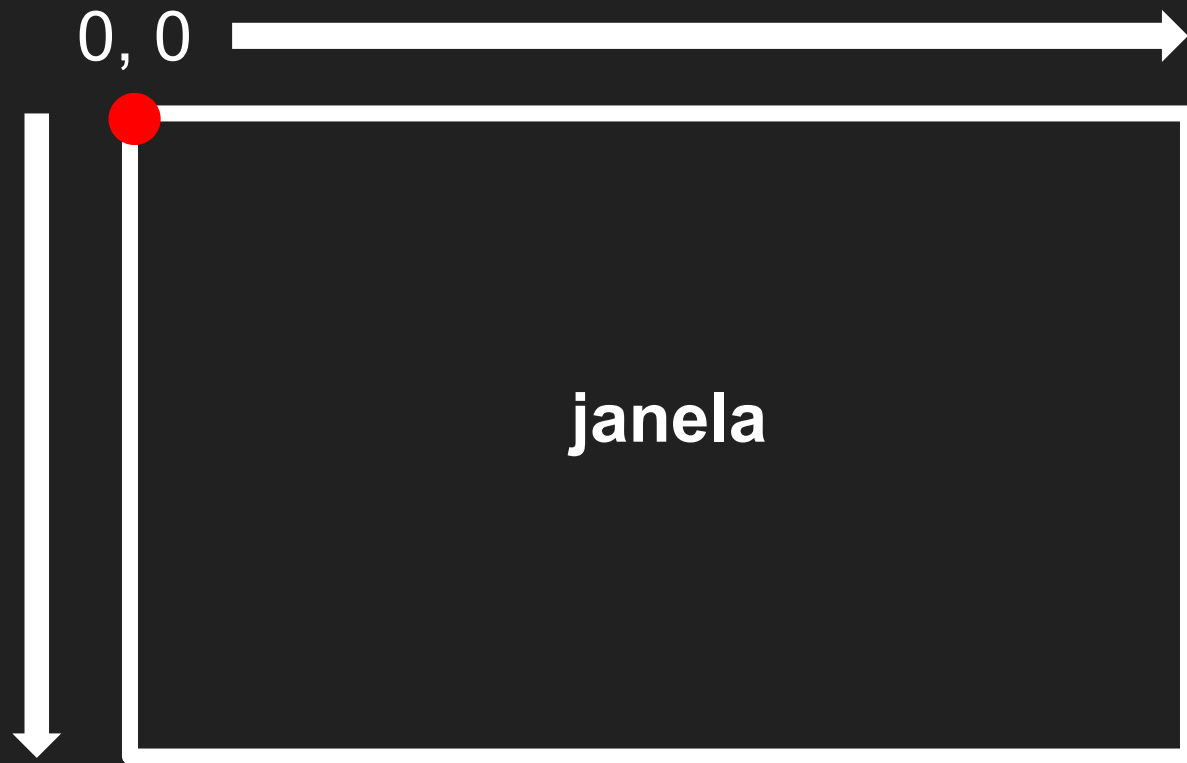
Chamada após love.update(). Esta função encapsula todo o código responsável por desenhar o jogo na tela.

love.keypressed(*key*)

Chamada toda vez que uma tecla no teclado for pressionada. *key* é a variável que identifica a tecla.

Hello, LÖVE!

```
love.graphics.print( text, x, y, r, sx, sy, ox, oy, kx, ky )
```



Drawing shapes

```
love.window.setMode( width, height, flags )
```

```
love.graphics.rectangle( mode, x, y, width, height )
```

```
love.graphics.circle( mode, x, y, radius )
```



```
love.graphics.polygon( mode, vertices )
```

```
love.graphics.setColor( red, green, blue, alpha )
```

valores entre **0** e **1**

Drawing sprites

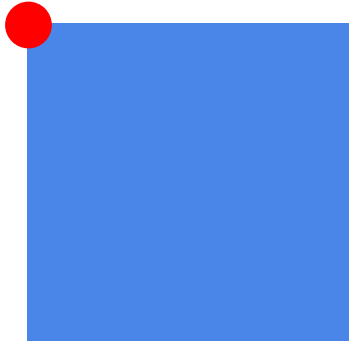
```
love.graphics.newImage( filename )
```

```
love.graphics.setBackgroundColor( red, green, blue, alpha )
```

```
love.graphics.draw( drawable, x, y, r, sx, sy, ox, oy, kx, ky )
```

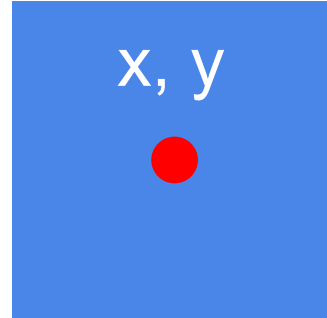
sem offset

x, y



com offset

x, y



```
image.getWidth()  
image.getHeight()
```


Keyboard inputs

```
math.random(...)  
math.randomseed(...)  
os.time()
```

```
love.graphics.printf( text, x, y, limit, align, r, sx, sy, ox, oy, kx, ky )
```

Movement

`math.max(...)`
`math.min(...)`

```
love.keyboard.isDown( key )
```

`love.keyboard.isDown(key) ~= love.keypressed(key)`

Sounds


```
love.audio.newSource( filename, type )
```

source:play()
source:pause()
source:stop()
source:setLooping(bool)
...

Fonts

```
love.graphics.newFont( filename, size, ... )
```

Próxima aula

11/11 - Pong!

Atividade: Desenhar a tela principal do seu Pong.

- Pás
- Bola
- Fonte