

# Desenhandando com Python no Navegador



**Bernardo Fontes**

Ribeirão Preto / SP

25 de Outubro de 2019

Olar!

**berinhard.github.io/sketches**

**twitter - @bbfontes**

**berinfontes.com/talks**



## Programação Criativa

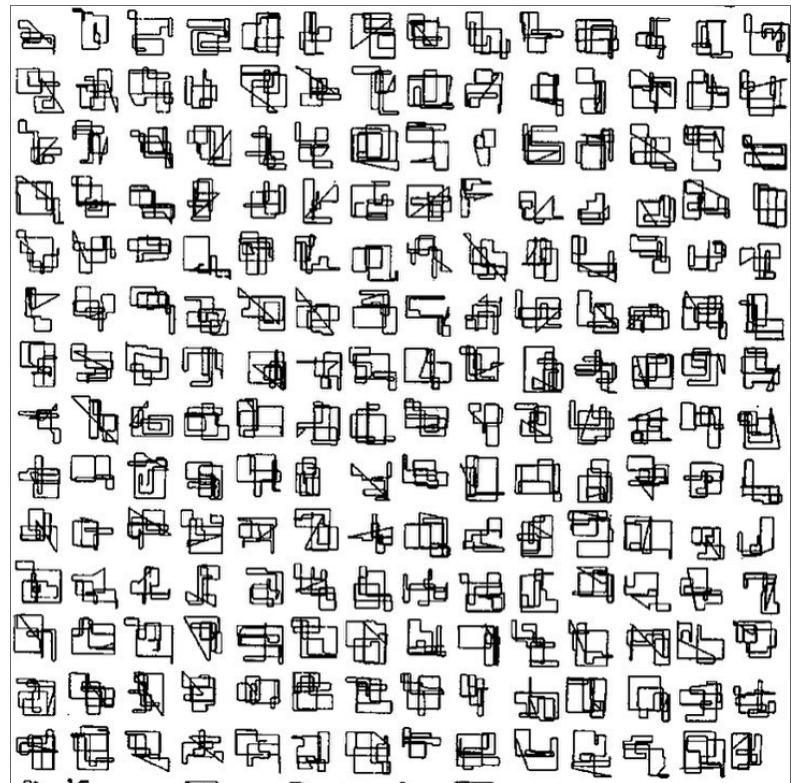
- Programar para a **expressão** ao invés de resolução de problemas
- A **experimentação** precedendo a funcionalidade ou assertividade
- Tem com música, vídeo, desenho, fotografia...

## Generative Art

- Objetos **criados com código**
- O **algoritmo como sistema autônomo** determina o comportamento da composição
- A pessoa controlando a máquina cria e programa os algoritmos e **controla sua randomicidade**
- Programação + **técnicas**

# Georg Ness - Generative Computergrafik

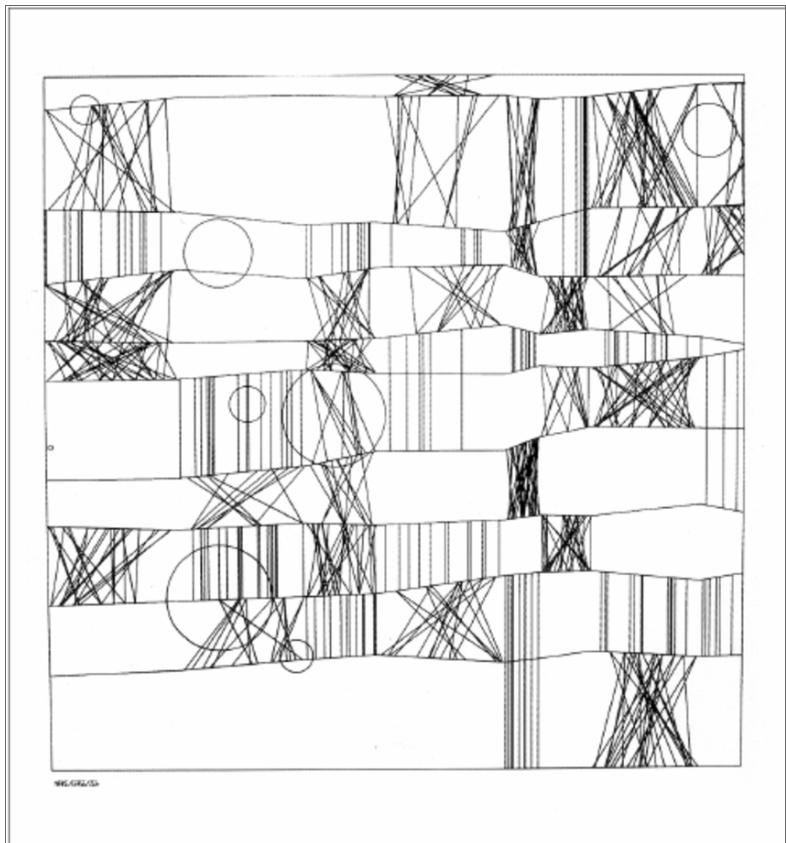
1965



23-Ecke (Polygon of 23 vertices)

Frieder Nake

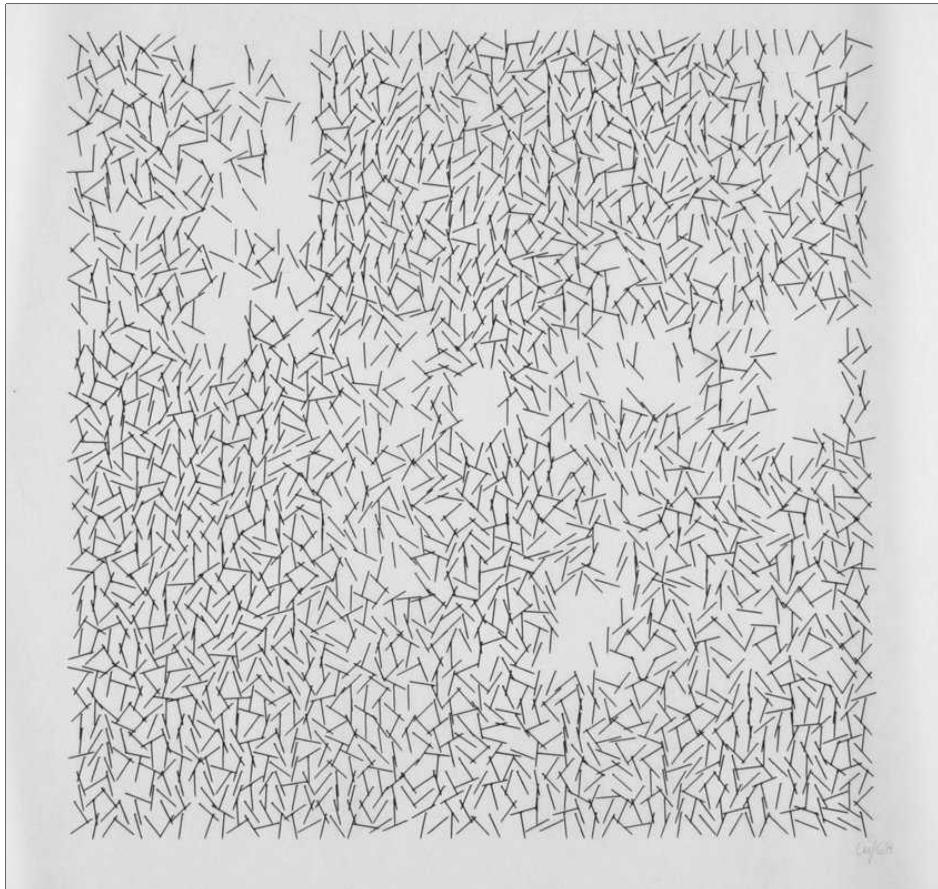
1965



**Hommage à Paul Klee**

Vera Molnár

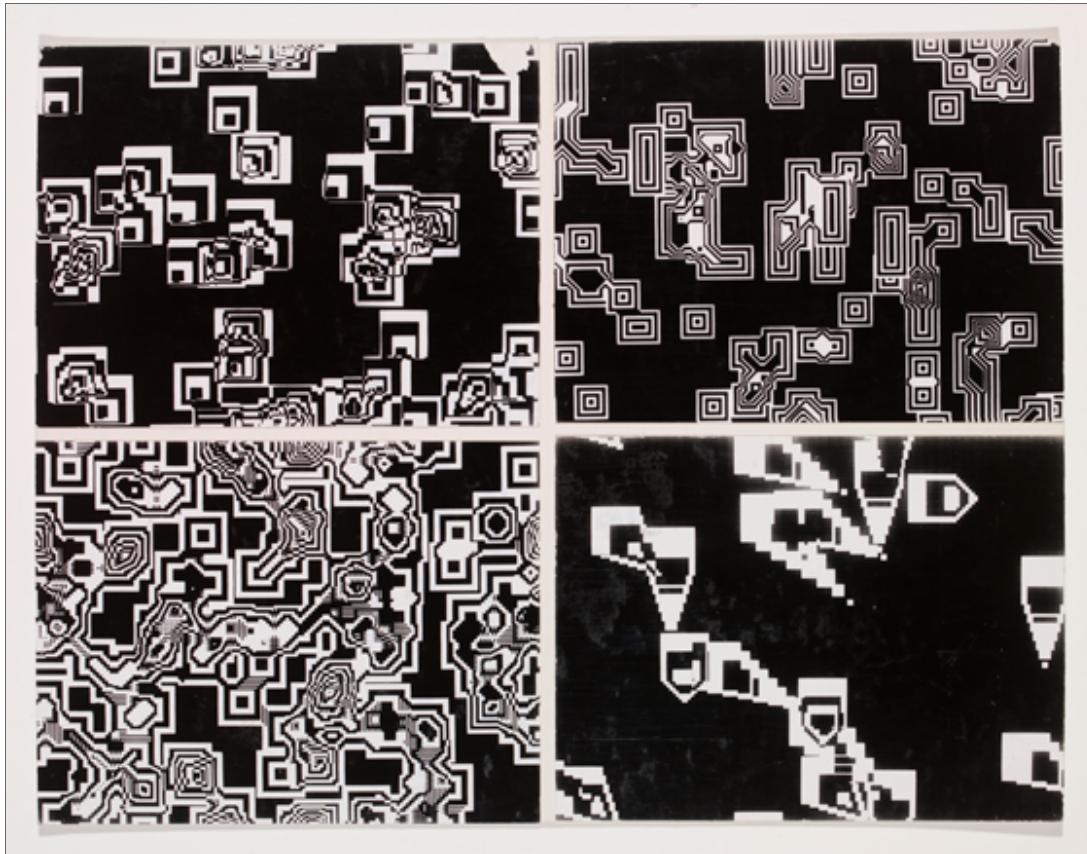
1968 / 1969



**Interruptions**

Lillian Schwartz

1970



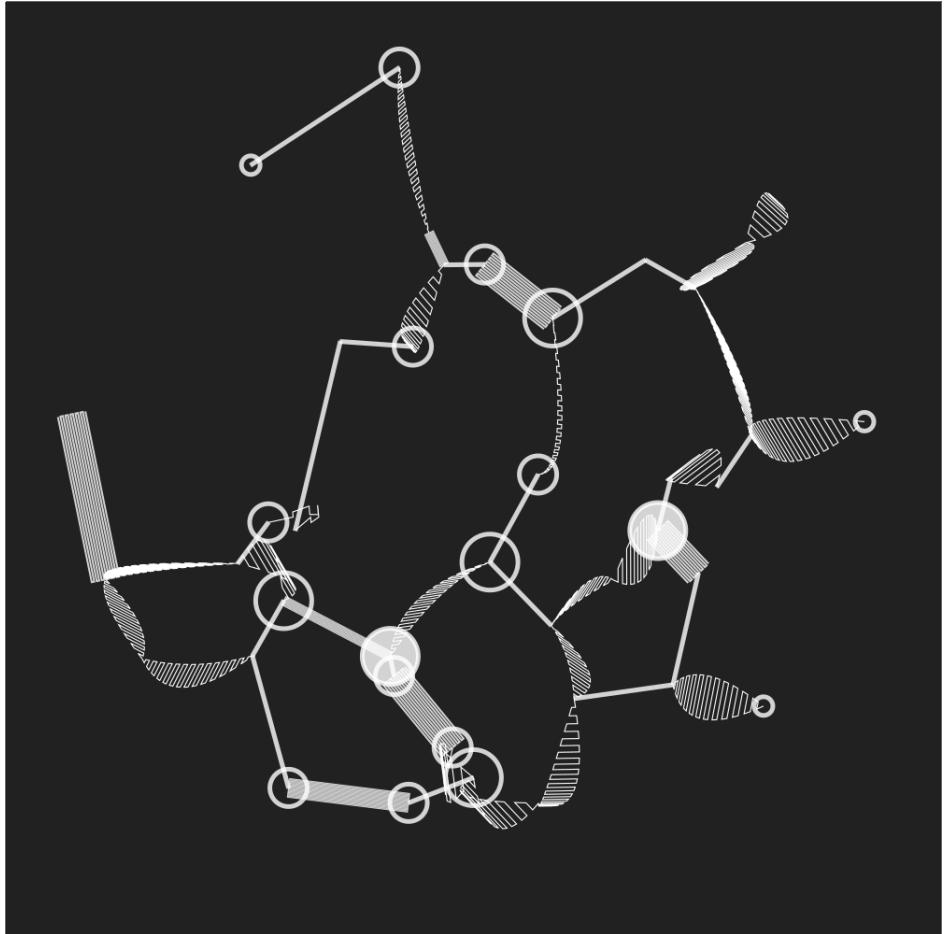
**Pixillation, photographic film stills**

Hoje em dia...

@manoloidee

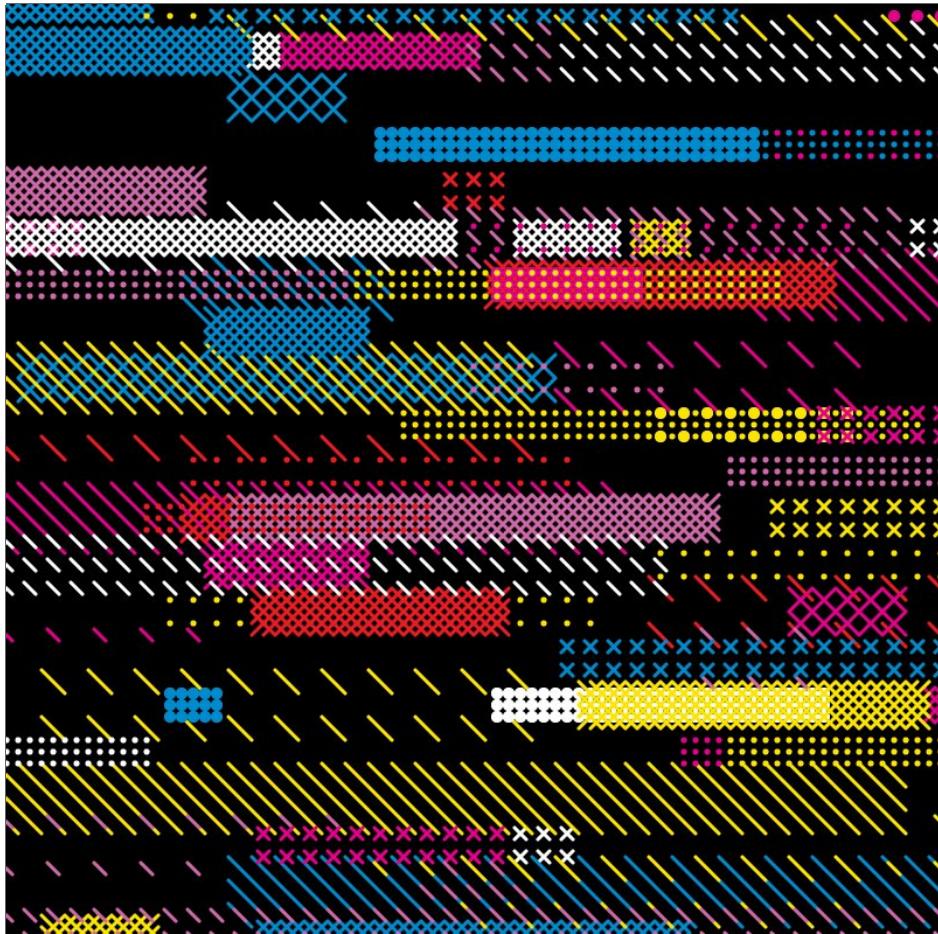


@inconvergent



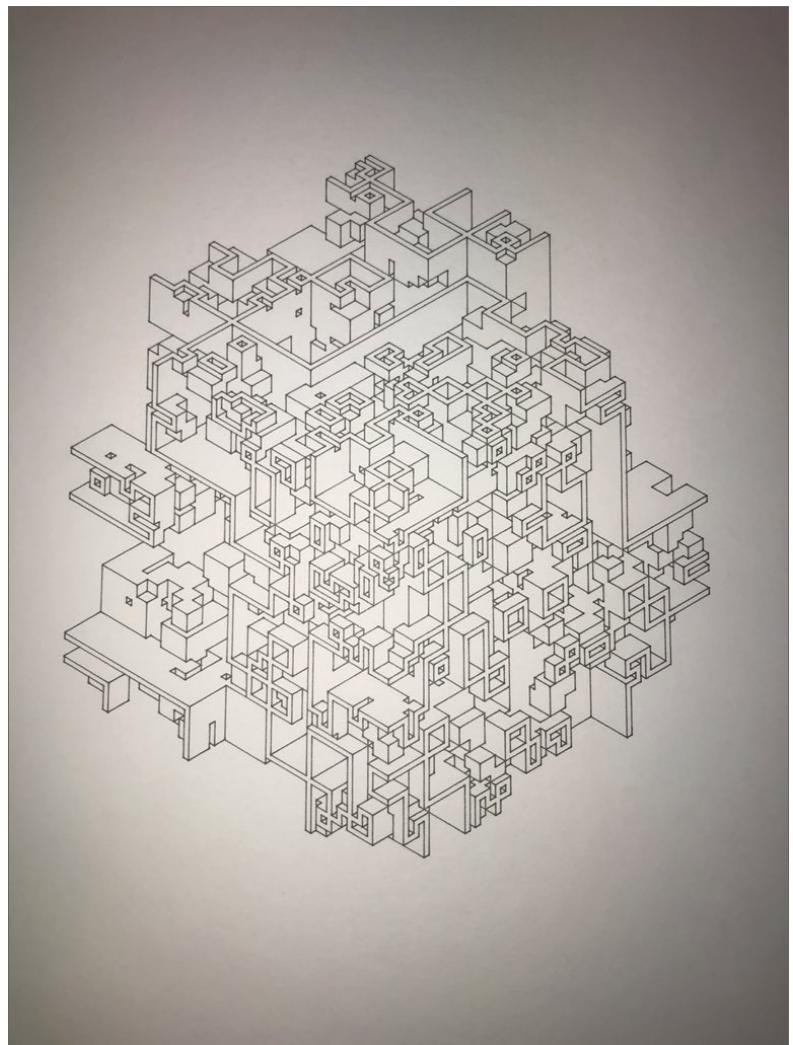
Saskia Freeke

@sasj\_nl



Frederik Vanhoutte

@wblut



## O que tem em comum?

- Todos usam **Processing**

## Processing

- Uma IDE que usa Java para servir como um **sketchbook**
- Ideal para estudantes e artistas visuais por conta de sua **API simples e versátil**
- Possui um **modo Python (Jython** por baixo dos panos)

## Um exemplo

```
# Author: Berin
# Sketches repo: https://github.com/berinhard/sketches
from random import choice

WHITE = color(228, 228, 228, 200)
RED = color(228, 12, 10, 200)
BLACK = color(17, 17, 17)

def setup():
    size(900, 900)
    noFill()
    background(BLACK)

def draw():
    radius = random(20, 300)
    x_positions = range(0, width, 20)
    y_positions = range(0, height, 20)
    x, y = choice(x_positions), choice(y_positions)

    if random(1) < 0.5:
        c = WHITE
    else:
        c = RED

    stroke(c)
    while radius > 20:
        ellipse(x, y, radius, radius)
        radius -= 20
```

Incríve né!?

- Mas.....

## Problemas do Modo Python

- É necessário **ativá-lo manualmente** pela IDE do Processing
- Praticamente **preso à IDE** do Processing
- Não tem acesso as **libs Python instaladas no sistema**
- Ainda está no **Python 2.7**
- O Jython não possui **nenhum** roadmap para migrar pra Python 3

## Python 2.7 "acaba" em 2 meses



Será que temos alternativas?

## 2 principais alternativas com limitações

- **Pesquisa** do Alexandre Villares
  - Python 3 puro
  - Reescrita ainda incompleta
  - API bastante diferente da API do Processing
- **BrythonIDE** - Prof. Esperança - COPPE/UFRJ
  - Roda no navegador
  - Utiliza **Brython**
  - Tem questões de performance

Quem poderá nos ajudar?



# Python + p5.js

**pyp5js**

## pyp5js: Python to P5.js Transcriptor

 PASSED

Processing ideas and Python 3 together with [P5.js](#) in the browser, using [Transcrypt](#).

This project started from a proof of concept based in [Axel Tanner's "Transcrypt & p5js" blogpost](#).

The project's main goal was to use Tanner's approach combined with decorator and global variables control to enable P5.js API from being called "directly" from the Python code as clean as possible.

`pyp5js` covers all the methods, variables and event handlers listed in [the p5.js API documentation](#). Here's an example of a valid Python code using p5.js API:

# p5.js

Processing fun times JavaScript quirkiness

[Home](#)

## Reference

Can't find what you're looking for? You may want to check out  
[p5.sound](#).

You can also download an offline version of the reference.

[Download](#)

[Get Started](#)

[Reference](#)

[Libraries](#)

[Learn](#)

[Examples](#)

[Books](#)

[Community](#)

[Forum](#)

## Color

[Creating & Reading](#)    [Setting](#)

<a href="#">alpha()</a>	<a href="#">background()</a>
<a href="#">blue()</a>	<a href="#">clear()</a>
<a href="#">brightness()</a>	<a href="#">colorMode()</a>
<a href="#">color()</a>	<a href="#">fill()</a>

[GitHub](#)

[Twitter](#)

## Search the reference

```
$ pip install pyp5js  
$ pyp5js serve
```

## API Básica

```
from py5js import *
from random import choice

colors = [
    (235, 235, 0),
    (235, 0, 235),
    (0, 235, 235),
]

def setup():
    createCanvas(900, 900)
    background(27)
    rectMode(CENTER)

def draw():
    size = 20

    for x in range(0, width + size, size): # itera por colunas
        for y in range(0, height + size, size): # itera por linhas

            fill(choice(colors)) # escolhe uma cor randomicamente
            if random(1) > 0.5: # condição para desenhar um retângulo ou círculo
                ellipse(x, y, size, size)
            else:
                rect(x, y, size, size)

noLoop()
```

## O que acontece embaixo dos panos?

- A interface web é uma abstração para a CLI
- A criação do sketch via form equivale ao comando `$ pyp5js new sketch_name`
- O diretório `~/sketchbook-pyp5js/sketch_name/` é criado com todo o código e arquivos necessários
- Ao acessar `/sketch/pybr_001/` a lib "compila" o código Python para código JS
- A compilação é através do comando `$ pyp5js transcrypt pybr_001`
- Dentro do diretório do sketch, os JS finais são colocados dentro do diretório `target`

## API de Eventos

```
from pyp5js import *
from random import choice

shape_type = 'RECT'
shape_size = 30
live_shapes = []

def add_shape():
    x, y = mouseX, mouseY
    colors = [
        (235, 235, 0),
        (235, 0, 235),
        (0, 235, 235),
    ]
    live_shapes.append((
        shape_type,
        x,
        y,
        shape_size,
        choice(colors)
    ))

def keyPressed():
    global shape_type, shape_size

    if key == 'c':
```

# Manipulando o DOM

```
from py5js import *

add_library("p5.dom.js")

rect_base_size = 30
positions = []
w_size, h_size = None, None

def setup():
    global w_size, h_size

    createP("Hi! This is an example of how to use p5.dom.js with py5js")

    # creates a container div
    slider_div = createDiv()
    slider_div.style("display", "block")

    # creates the sliders
    w_size = createSlider(0, 600, 100)
    w_size.style('width', '50%')

    h_size = createSlider(0, 600, 100)
    h_size.style('width', '50%')

    # adds the slider to the container div
    slider_div.child(h_size)
    slider_div.child(w_size)
```

Mas nem tudo são flores...



- No fundo, estamos usando JS e isso traz algumas questões
- As vezes o uso de memória não é otimizado
- Não ter toda a standard lib do Python
- Algumas **outras diferenças** do CPython

## Já vivemos no futuro



- **Pyodide**: The Python scientific stack, compiled to **WebAssembly**
- Já existe uma **issue aberta** para migrar o `pyp5js`

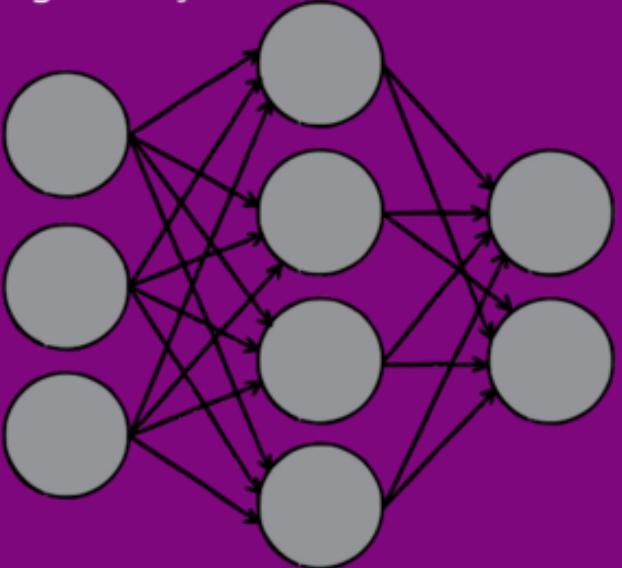
## Call to Action!

- Usem, quebrem, desmontem o `pyp5js`
- Ajudem a melhorar a documentação
- Criem novos exemplos e abram PRs com eles
- Estarei presente nos Sprints

No Garoa (São Paulo) toda última terça-feira do mês

# NOITE DE PROCESSING

arte e programação



terça-feira 24 de setembro de 2019 - 19h30

**Inteligência Artificial e Aprendizado de Máquina  
com Processing – Luciana Marques**

Rua Costa Carvalho, 567 fundos - São Paulo  
Garoa Hacker Clube - [garoa.net.br/wiki/Noite\\_de\\_Processing](http://garoa.net.br/wiki/Noite_de_Processing)



Em Recife toda última quarta-feira do mês

# NOITE DE PROCESSING - RECIFE

Arte e Programação



## PROCESSAMENTO DE IMAGENS

---

25 de Setembro de 2019  
Quarta-feira, 19h

---

[arteprog.space/noite-processing-recife/](http://arteprog.space/noite-processing-recife/)

Labcodes Software Studio  
Praça do Derby, 149  
11º Andar - Sala 1109  
Derby - Recife - PE

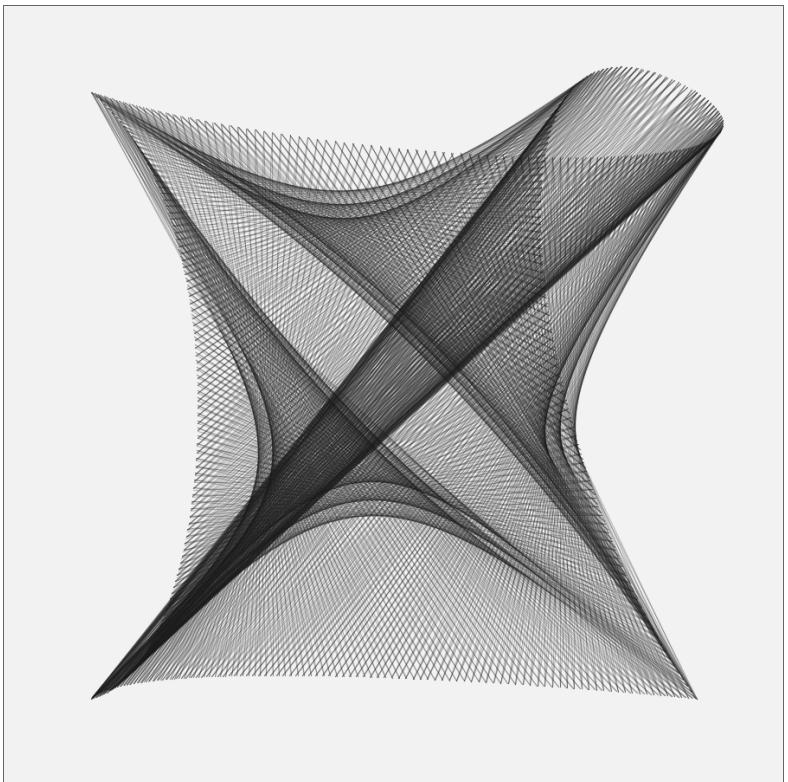


## processing-brasil@googlegroups.com

The screenshot shows a Google Groups forum interface for the 'processing-brasil' group. The top navigation bar includes buttons for 'NOVO TÓPICO', search, 'Marcar tudo como lido', 'Ações', 'Filtros', and settings. The main content area displays a list of 15 topics, each with a small profile icon, the topic title, the poster's name, the number of posts, and the date. The topics listed are:

- Apresente-se! (1) - Por John F - 13 postagens - 83 visualizações (2 de abr)
- Referências (3) - Por Mariana Leal - 3 postagens - 2 visualizações (22 de abr)
- Nóide de Processing - Recife - 24/04 (1) - Por eu - 1 postagem - 0 visualização (22 de abr)
- Saiu o livro da Patricia Oakim! - Arte feita em código (3) - Por Alexandre B A Villares - 3 postagens - 4 visualizações (4 de abr)
- Links sobre GIF loops (1) - Por eu - 2 postagens - 4 visualizações (28 de mar)
- 2ª Noite Processing - Recife (1) - Por eu - 1 postagem - 3 visualizações (25 de mar)
- Terça 26/3 tem Noite de Processing em São Paulo (1) - Por Alexandre B A Villares - 1 postagem - 4 visualizações (23 de mar)

Obrigado!



**berinfontes.com**

**berinhard.github.io/sketches**

**twitter - @bbfontes**