

Por que você está aqui?



Por que você está aqui?

Projetar e implementar soluções para resolver problemas!

Impacto gigante na sociedade, em todas as áreas!

Apreciar a beleza do raciocínio lógico, matemático e de idéias da computação!

Computação não é sobre computadores ou programação

Computação é sobre como resolver problemas importantes para nós.

Esta disciplina é uma visão abrangente (não muito profunda) de diversas áreas da computação, tendo como foco:

- 1) Fundamentos da computação
- 2) Fundamentos da programação
- 3) Visão geral atual
- 4) Visão futura

Tópicos abordados:

- Pensamento computacional
- Abstração, algoritmos, recursão
- Estruturas de dados, funções como dados
- Representação de dados
- Bancos de dados
- Programação C
- Programação Python
- Programação Web
- Complexidade de algoritmos
- ...

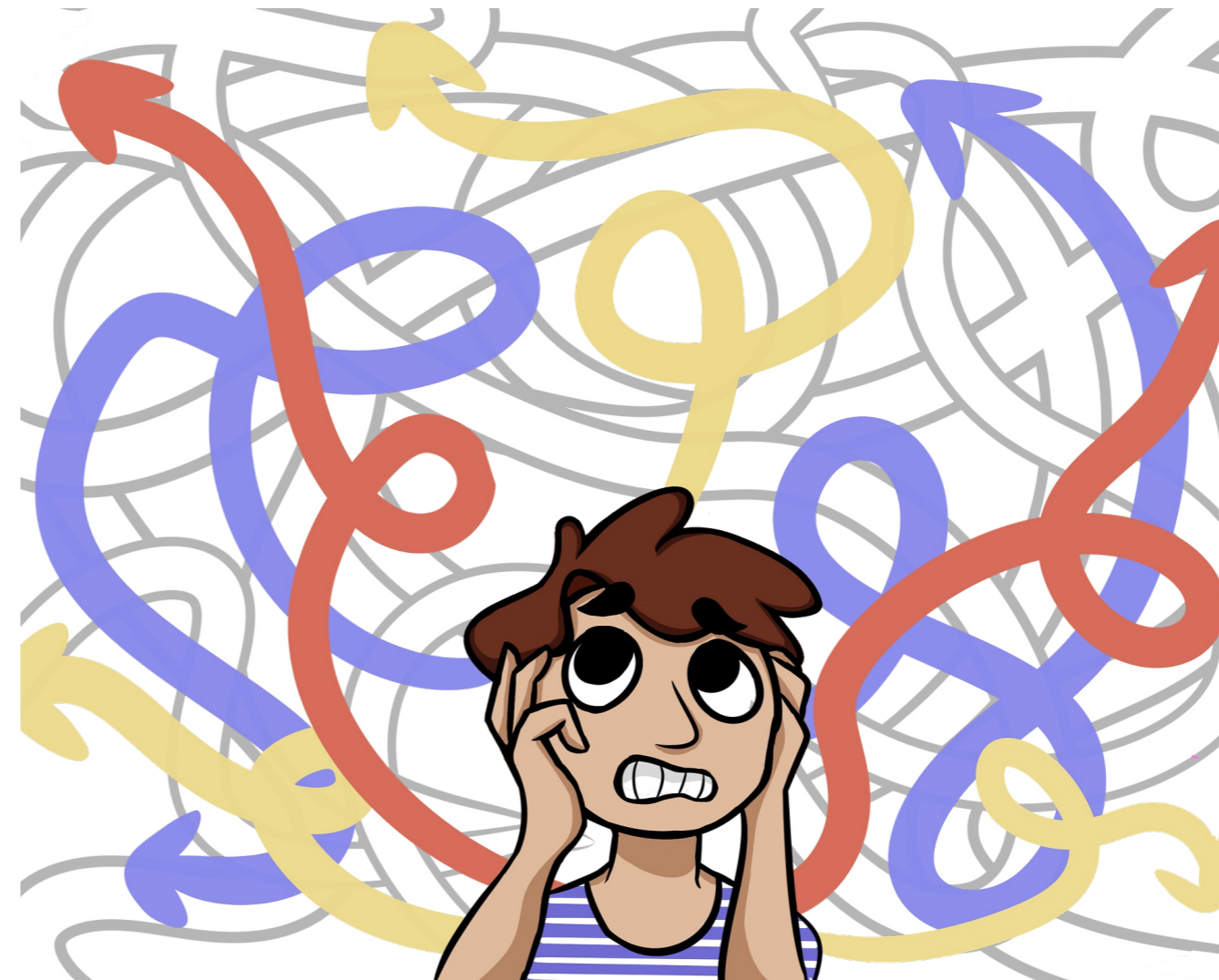


Imagem: MMillustrates, no Pixabay (<https://pixabay.com/illustrations/unordered-chaos-3192273/>)

Sim, é muita coisa...



complexidade de algoritmos

compilação

arrays

SQL

ponteiros

algoritmos

fundamentos da programação

estruturas de dados

internet

web

linux

pensamento computacional

memória

C

fundamentos da computação

... é como tentar beber de um hidrante. Mas você conseguirá!



"Getting an education from MIT is like taking a drink from a Fire Hose."

Jerome Weisner
Former MIT President
(1971-1980)

Estudantes do MIT, em 1991, transformaram um hidrante em um bebedor.

O que é necessário?

- Coragem
- Resiliência
- Estudo (muito!)
- Sair da zona de conforto



Imagem: waldryano, no Pixabay
(<https://pixabay.com/illustrations/resilience-victory-force-1697546/>)

Em média, 2/3 ou mais dos alunos NUNCA FIZERAM UM CURSO DE COMPUTAÇÃO ANTES.
E, mesmo os que já estudaram alguma coisa de computação, irão aprender muito mais do que imaginam! Ao final, todos estarão "na mesma página".



Imagem: geralt, no Pixabay (<https://pixabay.com/illustrations/banner-header-butterfly-development-940636/>)

Estou indo bem na disciplina?

Ao final do curso não importa tanto onde você está em relação aos seus colegas, mas onde você chegou em relação a onde estava no começo!



Visão geral da disciplina



The Beauty and Joy of Computing
<https://bjc.berkeley.edu> | <https://bjc.edc.org/>

Berkeley CS10

<https://cs10.org>

MIT 6.001 MIT 6.0001

<https://ocw.mit.edu>



<https://www.computacaoraiz.com.br>
<https://www.youtube.com/computacaoraiz>
<https://www.abrantes.pro.br>

This is CS50
Harvard University's
Introduction to Computer
Science

<https://cs50.harvard.edu/x/>

Por que 6.100b? Fazer junto com CS50?

Dinâmica do curso

Aulas

(presenciais/gravadas)

<https://cursos.computacaoraiz.com.br>

Monitorias

(presenciais/gravadas)

Diários de Aprendizagem

(presenciais/online)

Autograders:

- **Computação Raiz: Autolab**

<https://autolab.computacaoraiz.com.br>

(acesso restrito, só alunos presenciais)

- **Harvard CS50: submit50/check50**

<https://cs50.harvard.edu/x/>

(acesso público)

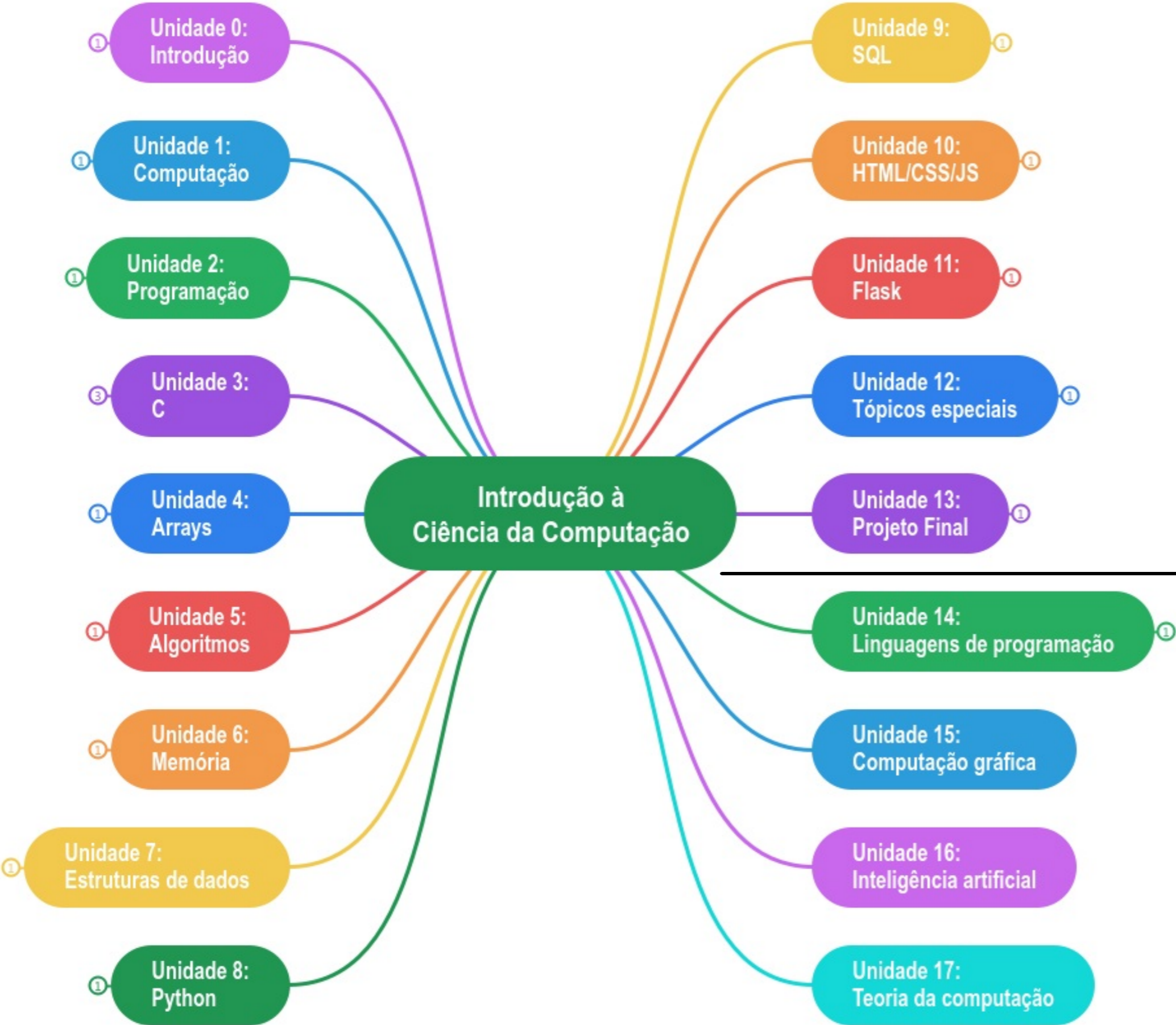
Provas:

- **Computação Raiz: Autolab**

<https://cursos.computacaoraiz.com.br>

(acesso restrito, só alunos presenciais)

Conteúdo do curso: mapa geral



Unidade 1: fundamentos da computação



Unidade 2: fundamentos da programação

The image shows the Scratch programming environment. At the top, there is a purple navigation bar with the Scratch logo, 'Settings', 'File', 'Edit', and 'Tutorials' menus, and 'Join Scratch' and 'Sign In' buttons. Below this is a light blue header with 'Code', 'Costumes', and 'Sounds' tabs. The main workspace is a large grid. On the left, the 'Motion' category is selected in the block palette, showing various movement blocks like 'move 10 steps', 'turn 15 degrees', 'go to random position', 'glide 1 secs to random position', 'point in direction 90', 'change x by 10', 'set x to 0', 'change y by 10', 'set y to 0', 'if on edge, bounce', and 'set rotation style left-right'. On the right, the 'Sprite1' control panel is visible, showing the sprite's current position (x: 0, y: 0), size (100), and direction (90). The stage area is currently empty, with the Scratch cat sprite visible in the top right corner of the workspace.

Unidade 3: fundamentos da programação (agora na linguagem C)

```
1 #include <stdio.h>
2
3 int main(void)
4 {
5     printf("%s\n", "Olá, mundo!");
6     return 0;
7 }
```

Unidade 4: compilação, debug, memória, arrays, strings...



Imagem: CS50 Lecture 0
(<https://cdn.cs50.net/2022/fall/lectures/0/lecture0.pdf>)

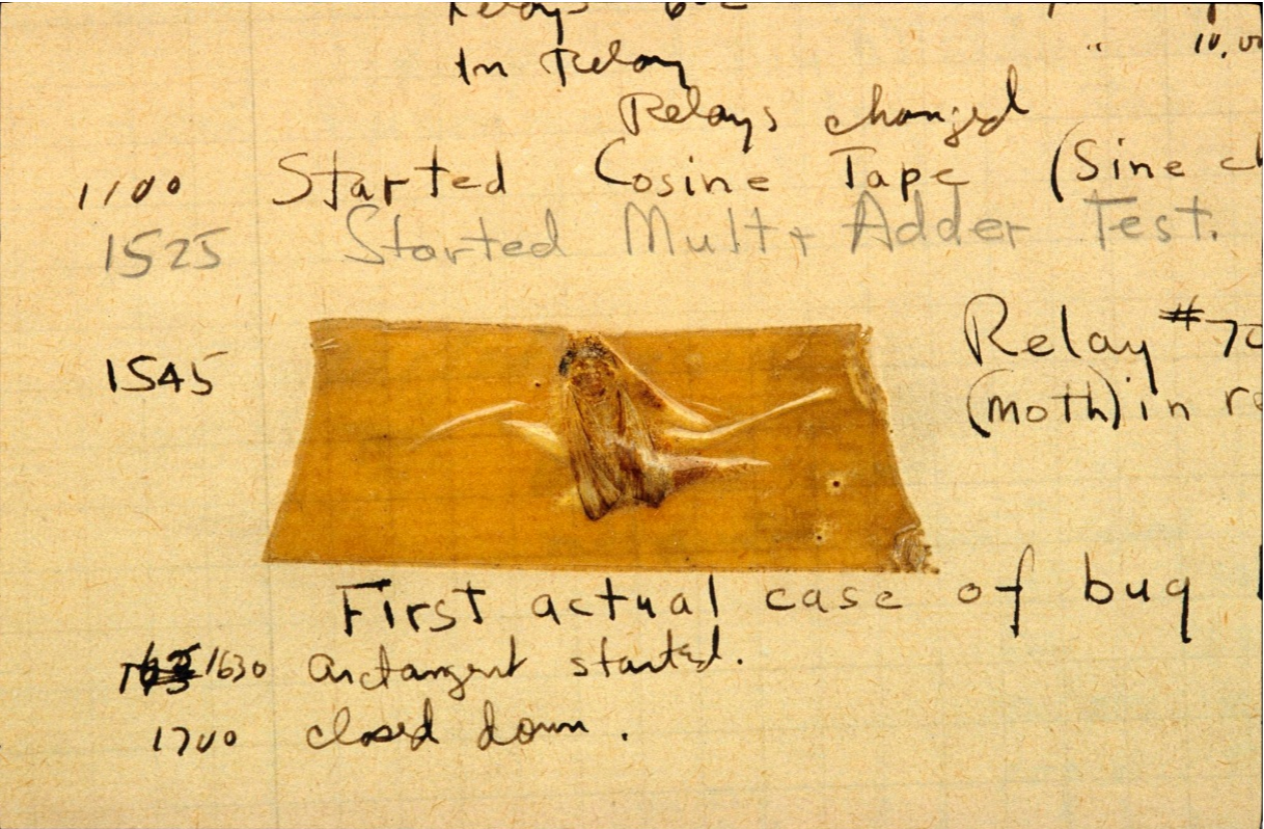
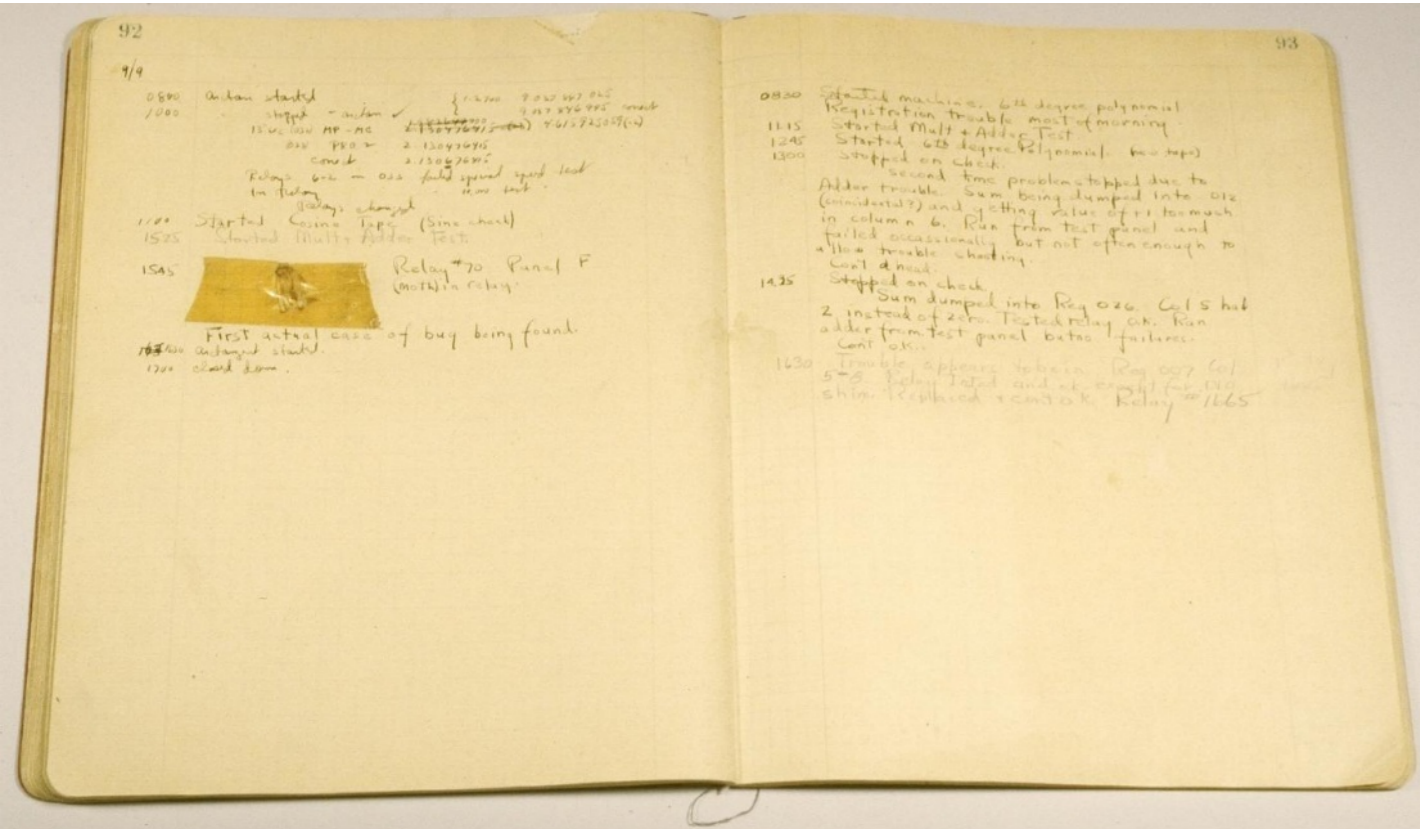


Imagem: Smithsonian National Museum os American History (https://americanhistory.si.edu/collections/search/object/nmah_334663)

Unidade 5: algoritmos, complexidade, recursão

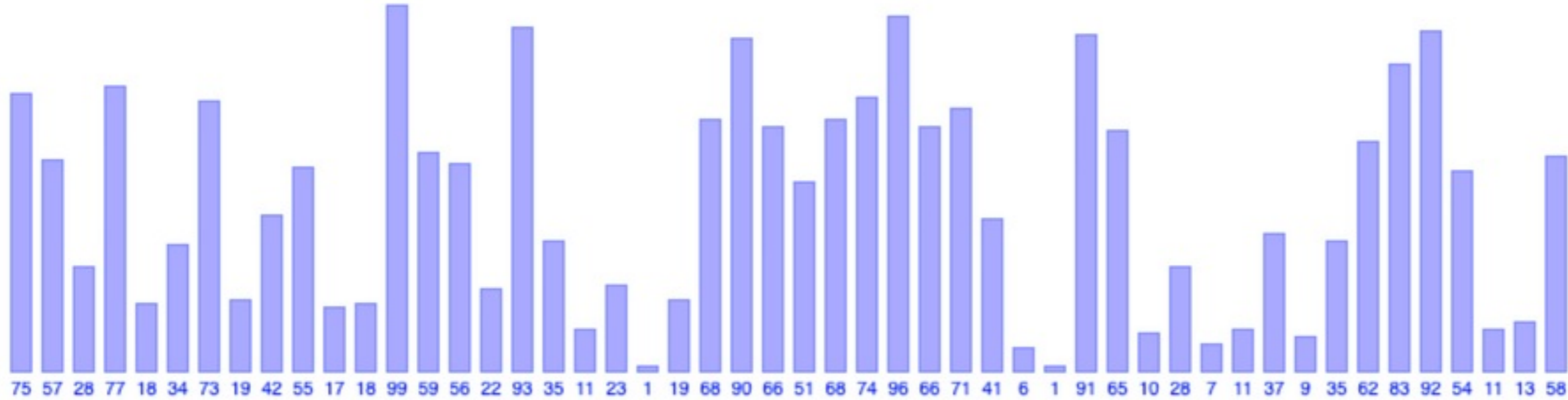


Imagem: CS50 Lecture 0 (<https://cdn.cs50.net/2022/fall/lectures/0/lecture0.pdf>)

Unidade 6: memória, ponteiros, alocação dinâmica, stack, heap...



Imagem: CS50 Lecture 0 (<https://cdn.cs50.net/2022/fall/lectures/0/lecture0.pdf>)

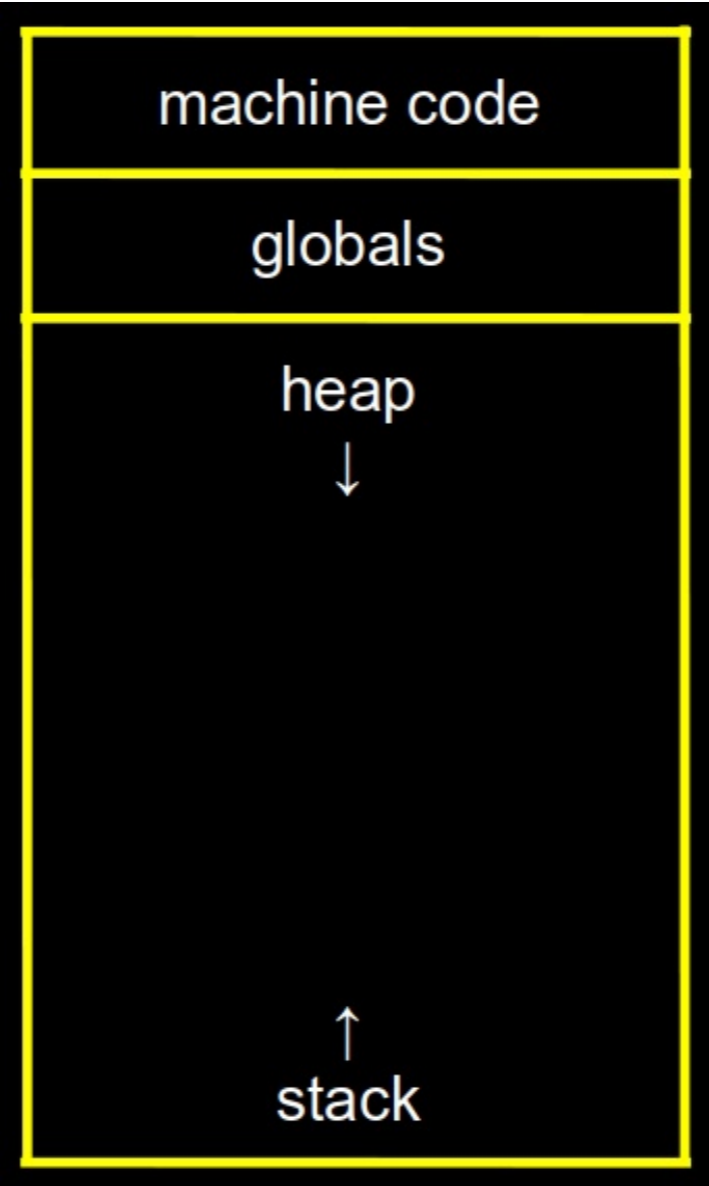


Imagem: CS50 Lecture 0 (<https://cdn.cs50.net/2022/fall/lectures/0/lecture0.pdf>)

Unidade 7: estruturas de dados, tipos abstratos de dados, pilhas...

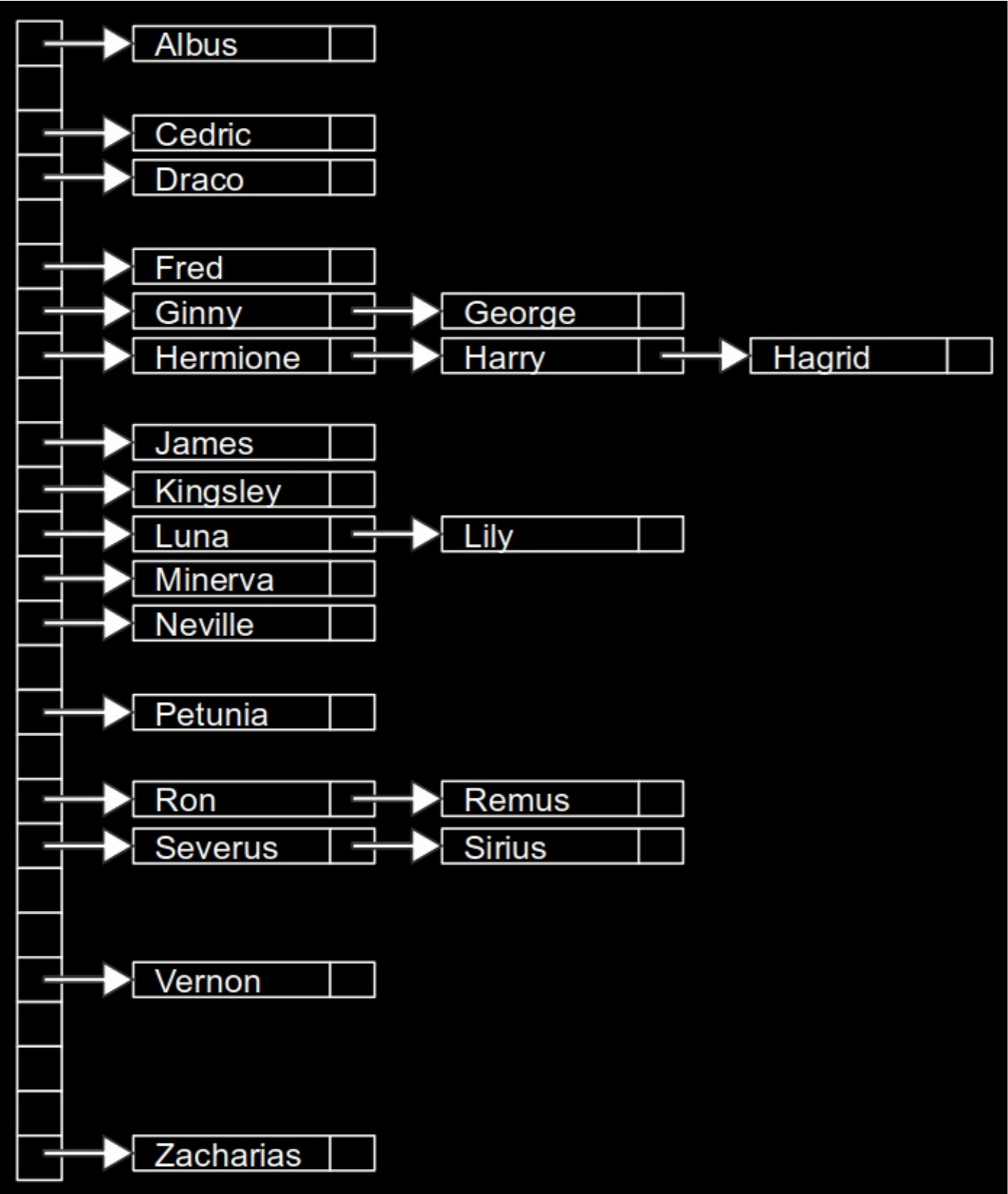
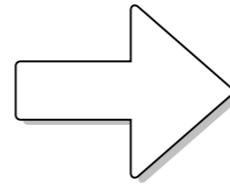


Imagem: CS50 Lecture 0 (<https://cdn.cs50.net/2022/fall/lectures/0/lecture0.pdf>)

Unidade 8: Python (fundamentos da programação, em linguagem moderna)

```
1 #include <stdio.h>
2
3 int main(void)
4 {
5     printf("%s\n", "Olá, mundo!");
6     return 0;
7 }
```



```
1 print("Olá, mundo!")
```


Unidade 9: bancos de dados, sql, transações...

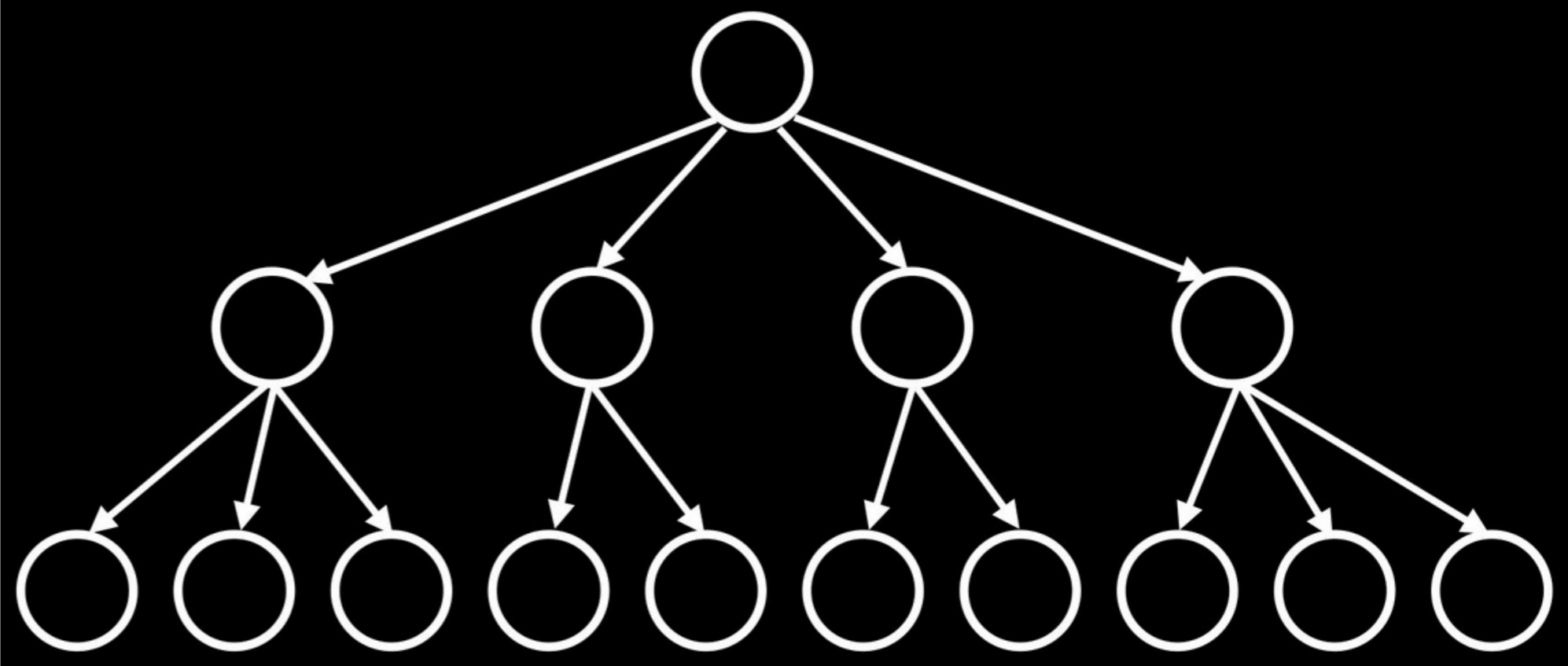
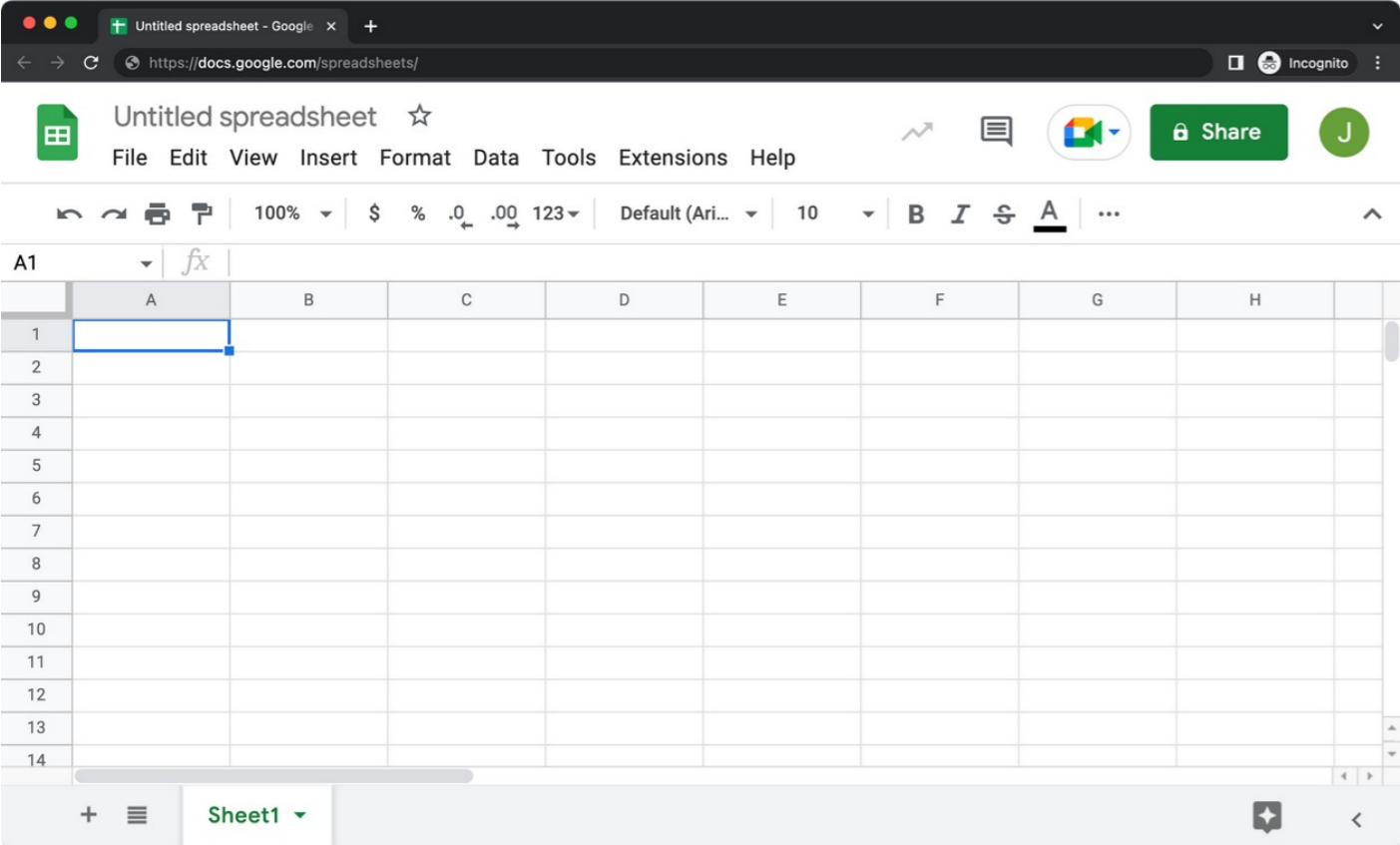


Imagem: CS50 Lecture 0 (<https://cdn.cs50.net/2022/fall/lectures/0/lecture0.pdf>)

Unidade 10: Internet, HTML, CSS, JavaScript

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <title>
      hello, title
    </title>
  </head>
  <body>
    hello, body
  </body>
</html>
```

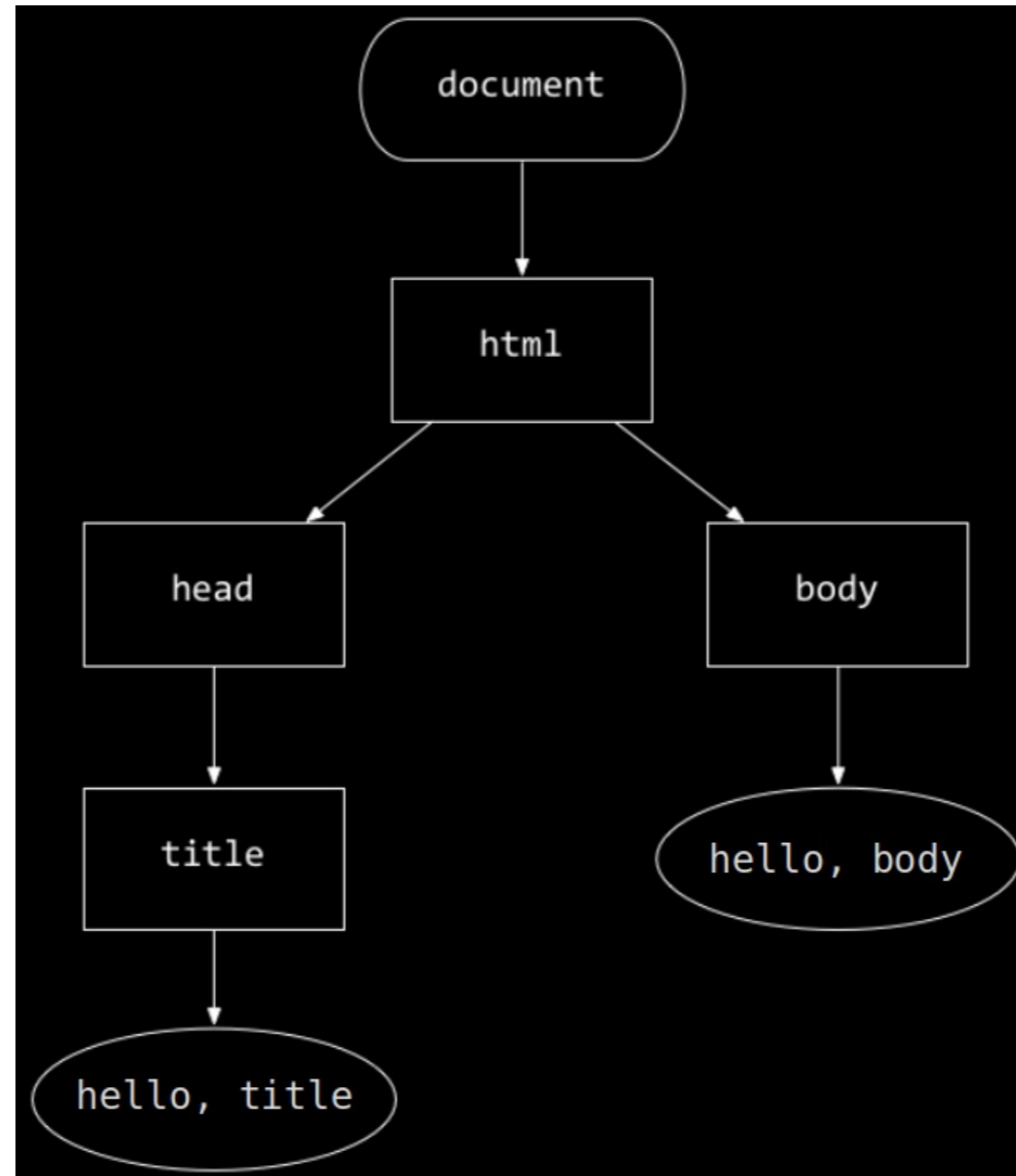


Imagem: CS50 Lecture 0 (<https://cdn.cs50.net/2022/fall/lectures/0/lecture0.pdf>)

Unidade 11: Flask, frameworks, libraries, web...



- Calendar
- Champions
- FAQ's
- Guide to Frosh IMs
- Headlines
- How to get involved
- Photos
- Point tallies
- Records
- Register
- Registrants
- Rules
- Schedules and results
- Whom to contact

Headlines

Past headlines
For headlines posted prior to the past seven days, click [here](#).

Unidade 12: tópicos especiais para o projeto final



Imagem: <https://github.com/logos>

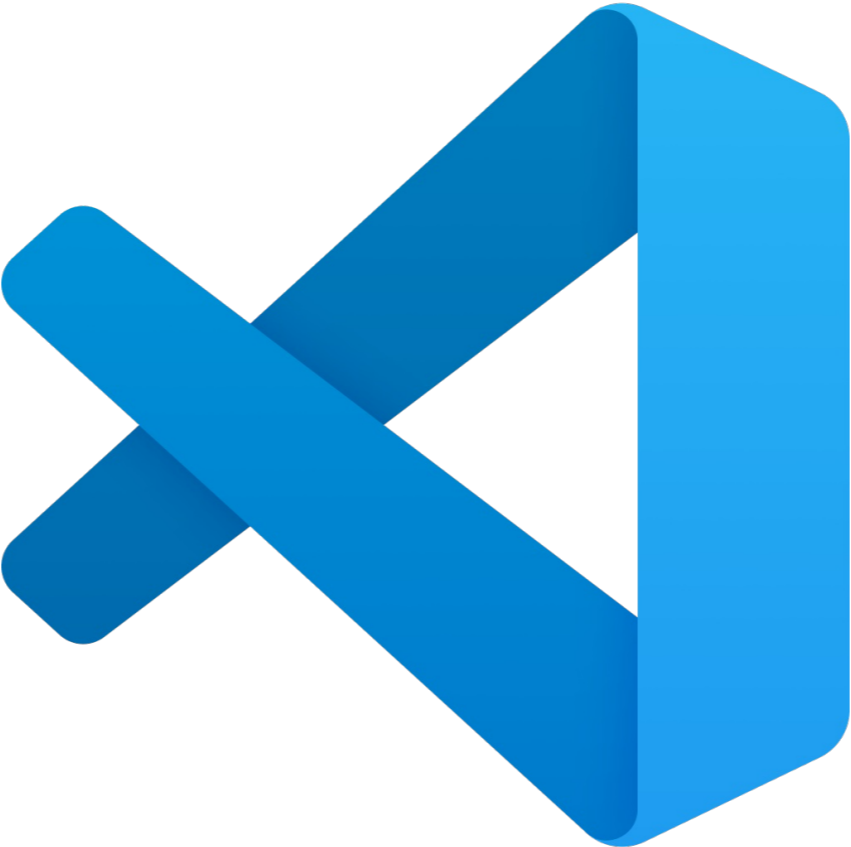


Imagem: <https://code.visualstudio.com/brand>



Imagem: <https://www.lua.org/images/>

Unidade 13: projeto final



Imagem: geralt, no Pixabay (<https://pixabay.com/photos/question-mark-a-notice-duplicate-3585355/>)

Unidade 14: linguagens de programação

Lisp **Perl** **C** **C++** **Forth** **Lua**

AWK **Racket**

ADA **Scheme** **Go** **Scratch**

R **PHP**

Algol **Java** **OCaml** **Assembly** **Haskell** **Snap!**

Kotlin

Rust **Pascal** **Shell** **Logo** **COBOL**

Python

SML

Ruby **Fortran** **Prolog** **Erlang** **JavaScript**

Unidade 15: computação gráfica

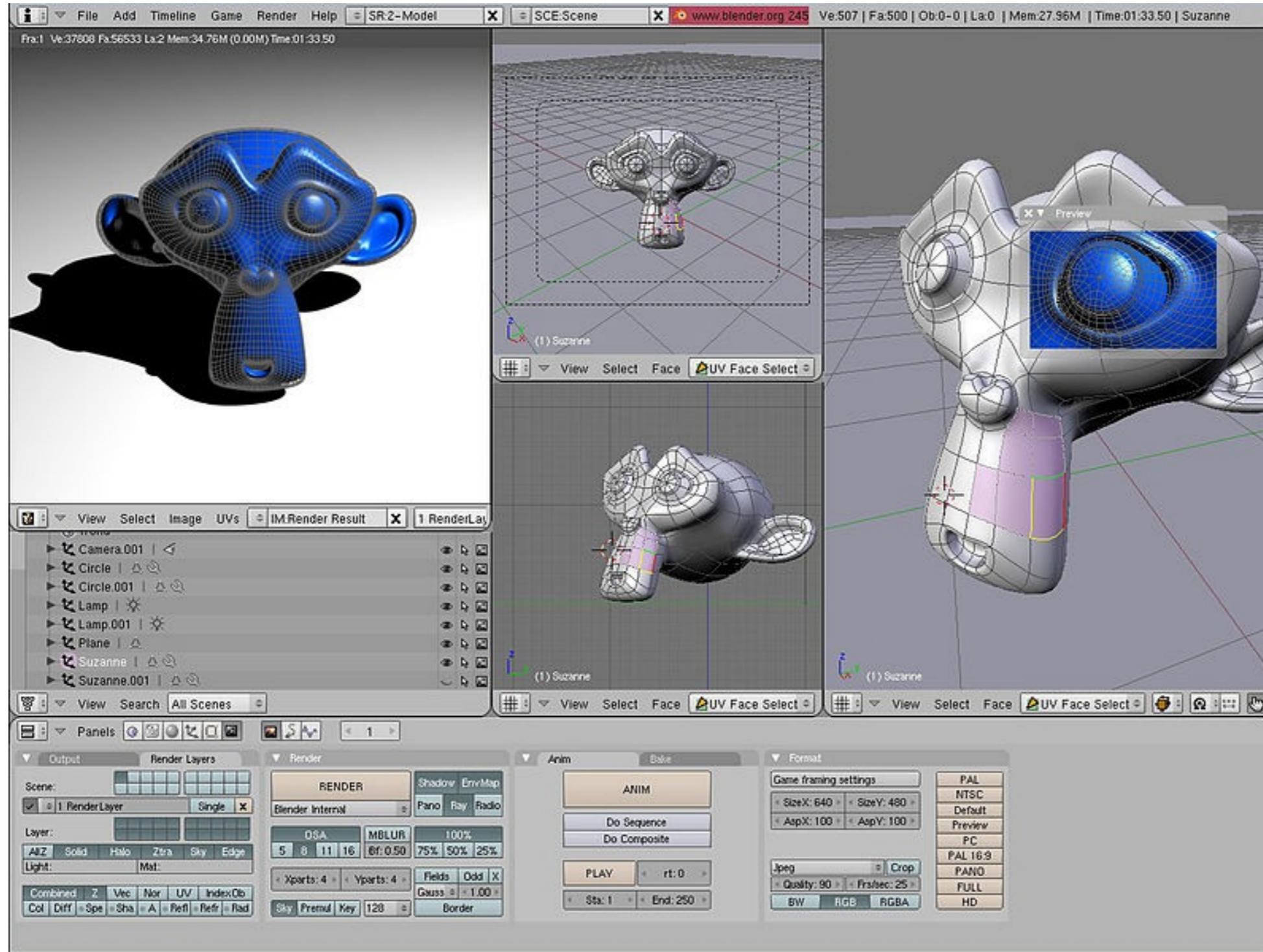


Imagem: Auisis, no Wikimedia Commons (https://commons.wikimedia.org/wiki/File:Blender_2.45_screenshot.jpg)

Unidade 16: inteligência artificial

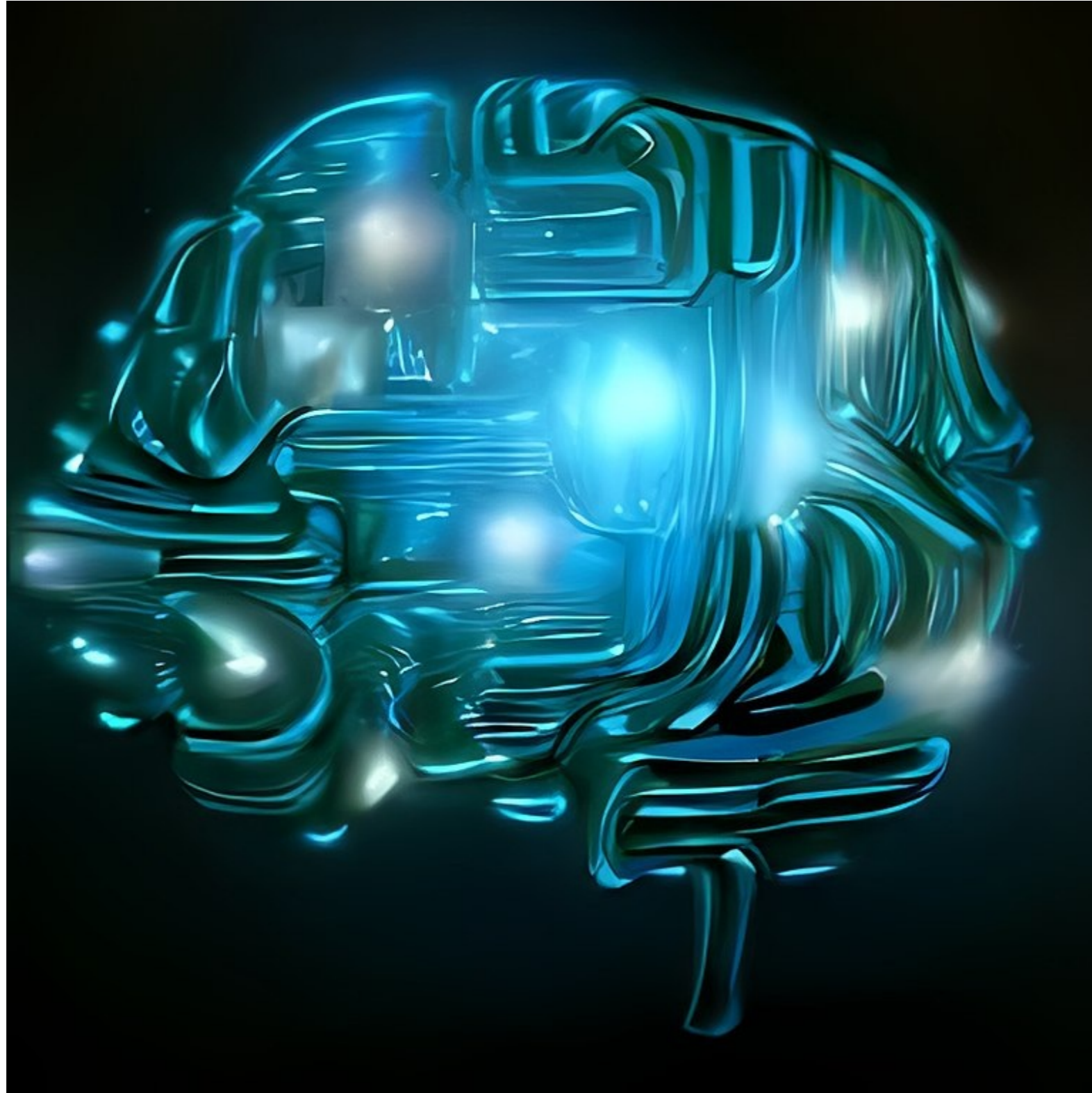


Imagem: JPxGuis, no Wikimedia Commons

(https://commons.wikimedia.org/wiki/File:Artificial_intelligence_prompt_completion_by_dalle_mini.jpg)

Unidade 17: teoria da computação



Imagem: Autor desconhecido, no Wikimedia Commons
(https://commons.wikimedia.org/wiki/File:Alan_Turing_az_1930-as_%C3%A9vekben.jpg)

Unidade ∞ : sobre o futuro

Há todo um mundo na ciência da computação que nem foi citado aqui, durante o estudo introdutório, na CR6.100B. Por exemplo:

- Sistemas operacionais
- Redes
- Arquitetura de computadores
- Compiladores
- Engenharia de software
- Análise de dados
- ...

O que você pode ter certeza é o seguinte: você já **sabe muito mais do que sabia** no começo, e está **totalmente preparado** para esses tópicos mais avançados. Inclusive para **estudar e aprender por conta própria!**