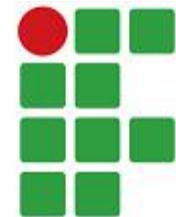


PROGRAMAÇÃO I

Curso Técnico Subsequente em Informática

Lucas Sampaio Leite



**INSTITUTO
FEDERAL**

Baiano

Professor

Lucas Sampaio Leite

lucas.leite@ifbaiano.edu.br



Objetivo

- Aplicar os conceitos das estruturas de dados estáticas e dinâmicas na construção de softwares para resolução de problemas reais.

Ementa

- Técnicas de modularização, passagem de parâmetros e recursividade;
- Ambientes e técnicas de desenvolvimento de aplicações.

Conteúdo Programático

- Utilizando uma linguagem de programação
 - Apresentação de uma IDE
 - Tipos de dados, Constantes e Variáveis
 - Atribuição, Operadores Lógicos e Aritméticos
 - Comandos de Entrada e Saída
- Recapitulando Estruturas
 - Estruturas Condicionais
 - Estrutura de Repetição
 - Vetores e Matrizes

Conteúdo Programático

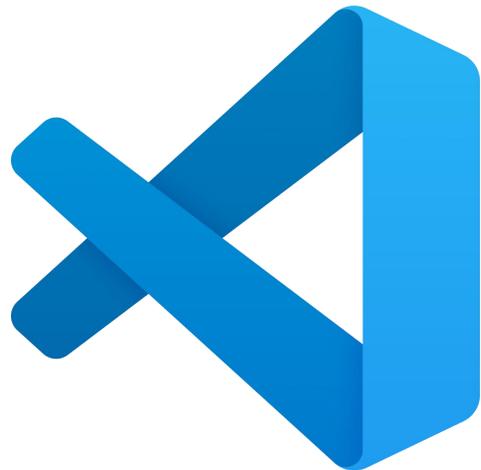
- Modularização de algoritmos
 - Utilizando procedimentos e funções
 - Passagem de parâmetros (valor x referência)
 - Escopo de variáveis
 - Recursividade
- Acesso a arquivos
 - O que é um arquivo?
 - Arquivo-texto
 - Operação de manipulação de arquivos

Conteúdo Programático

- Introdução a Estrutura de Dados
 - Listas
 - Filas
 - Pilhas
 - Busca e ordenação de listas

Ferramentas e ambientes de desenvolvimento

- IDEs: VS Code, PyCharm, etc
- Notebook Jupyter com Google Colab
- Árbitro Virtual Beecrowd (listas de exercícios)



Metodologia

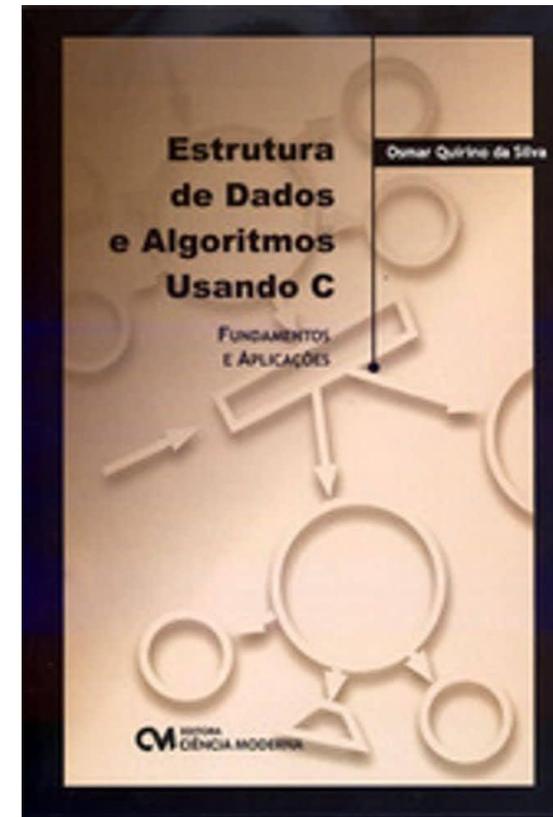
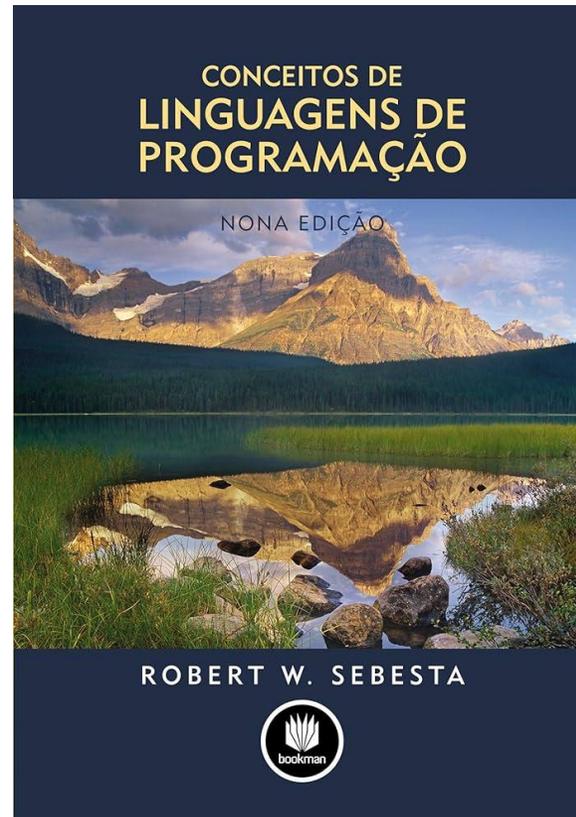
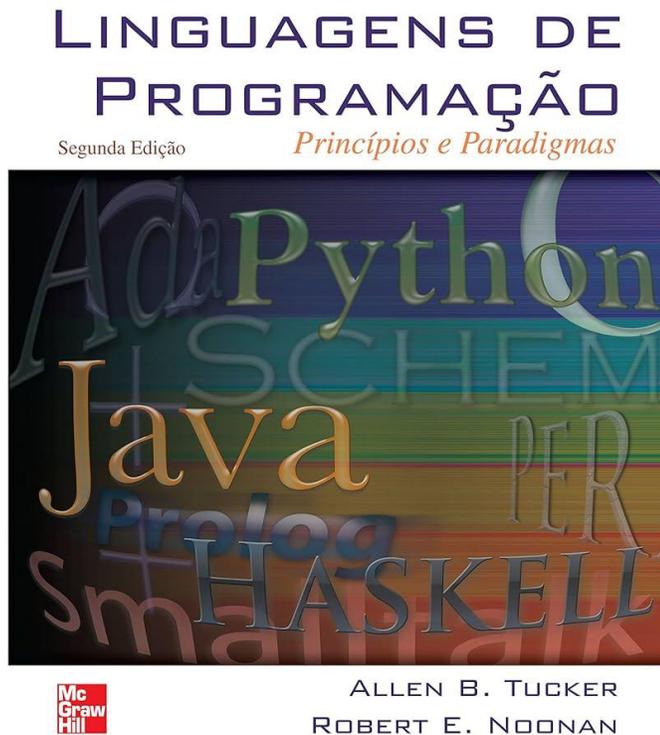
- Aulas síncronas e presenciais
 - Aulas expositivas e dialogadas
 - Abordagem prática
 - Aulas de resolução de exercícios
- **Com chamada**

Avaliação

- Avaliação 1: Listas de exercício 30% + Prova 70%
- Avaliação 2: Listas de exercício 30% + Projeto 70%
- Recuperação paralela (substitui a menor nota)
- Prova final



Bibliografia Básica



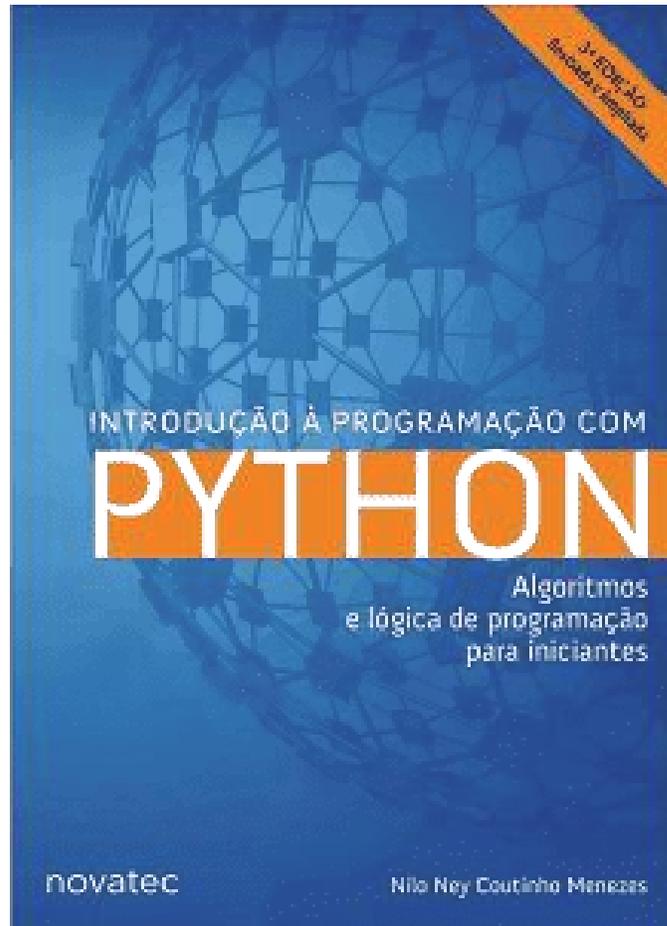
Bibliografia Básica

- TUCKER, Allen B. Linguagens de programação: princípios e paradigmas. 2. ed. São Paulo (SP): McGraw-Hill, 2008. 594 p. IL. ISBN 9788577260447.
- SEBESTA, Robert W. Conceitos de Linguagens de Programação. 9. ed. Porto Alegre: Bookman, 2011. 792 p. il. ISBN 9788577807918.
- SILVA, Osmar Quirino da. Estrutura de dados e algoritmos usando C: fundamentos e aplicações. Ciência Moderna, 2007. ISBN 9788573936117.

Bibliografia Complementar

- BENEDUZZI, Humberto Martins; METZ, Joao Ariberto. Lógica e linguagem de programação: introdução ao desenvolvimento de software. Curitiba: Editora do Livro Técnico, 2010. 144 p. ISBN 9788563687111.
- BORATTI, Isaias Camilo; OLIVEIRA, Álvaro Borges de. Introdução à programação: algoritmos. 3. Visual Books, 2007. ISBN 978857502215.
- CORMEN, Thomas H et al. Algoritmos: teoria e prática. Rio de Janeiro: Elsevier, 2012. 926 p. ISBN 9788535236996.

Outras Referências



UNIVERSIDADE FEDERAL DE SANTA MARIA
CAMPUS CACHOEIRA DO SUL

Python 101

Autores:
Prof^ª. Dr^ª Ana Luisa Soubhia
Elias Teixeira da Costa
Flavio Luan Müller Freitas
Laís Brum Menezes
Marcos Alves dos Santos
Prof. Dr. Vinícius Maran

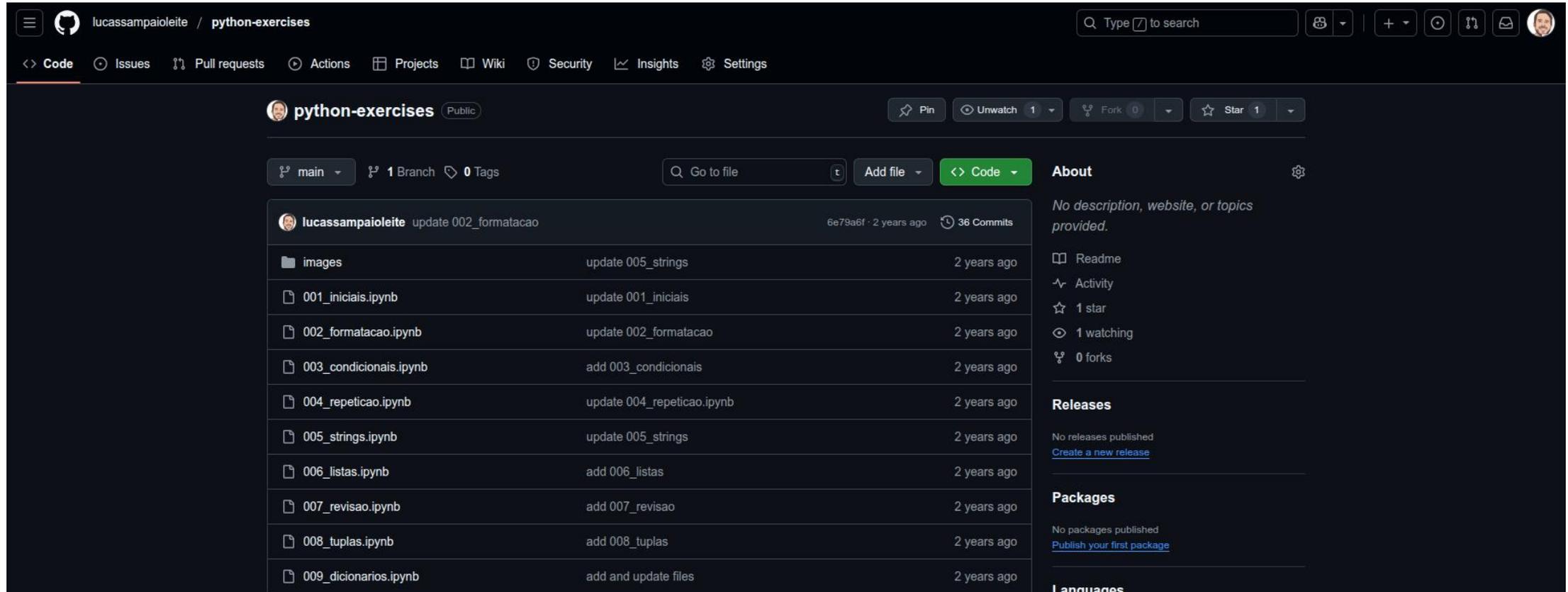
Link:

https://www.ufsm.br/app/uploads/sites/6/79/2019/08/Apostila_Python_v_1.pdf

Outras Referências

- Documentação do Python: <https://docs.python.org/pt-br/3/>
- <https://www.w3schools.com/python/default.asp>
- <https://ricardoduarte.github.io/python-para-desenvolvedores>

Referência de exercícios Python

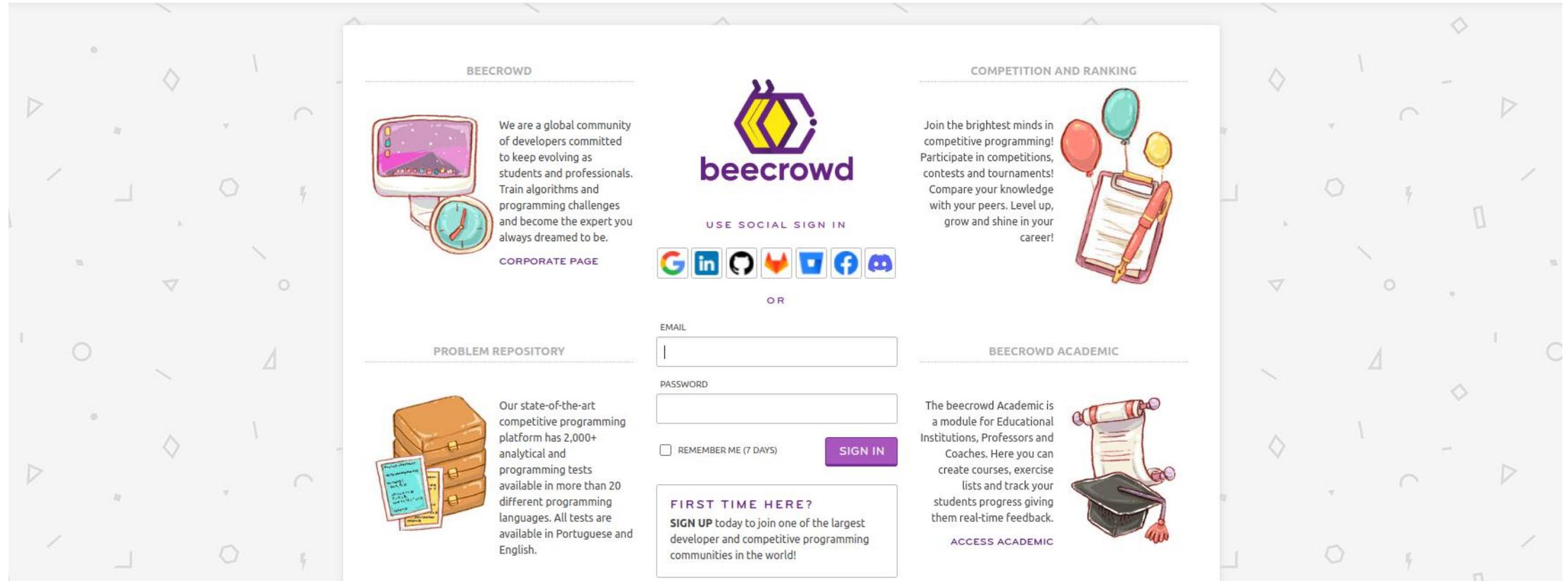


The screenshot shows a GitHub repository page for 'python-exercises' by user 'lucassampaioleite'. The repository is public and has 1 star and 0 forks. The main branch is 'main'. The repository contains a folder 'images' and several Python notebooks (ipynb files) numbered 001 to 009. The files are: '001_iniciais.ipynb', '002_formatacao.ipynb', '003_condicionais.ipynb', '004_repeticao.ipynb', '005_strings.ipynb', '006_listas.ipynb', '007_revisao.ipynb', '008_tuplas.ipynb', and '009_dicionarios.ipynb'. The repository also has a 'Code' button and a search bar.

File	Commit	Time
images	update 005_strings	2 years ago
001_iniciais.ipynb	update 001_iniciais	2 years ago
002_formatacao.ipynb	update 002_formatacao	2 years ago
003_condicionais.ipynb	add 003_condicionais	2 years ago
004_repeticao.ipynb	update 004_repeticao.ipynb	2 years ago
005_strings.ipynb	update 005_strings	2 years ago
006_listas.ipynb	add 006_listas	2 years ago
007_revisao.ipynb	add 007_revisao	2 years ago
008_tuplas.ipynb	add 008_tuplas	2 years ago
009_dicionarios.ipynb	add and update files	2 years ago

<https://wiki.python.org.br/ListaDeExercicios>

Referência de exercícios Python (com árbitro digital)



BEECROWD

We are a global community of developers committed to keep evolving as students and professionals. Train algorithms and programming challenges and become the expert you always dreamed to be.

[CORPORATE PAGE](#)

PROBLEM REPOSITORY

Our state-of-the-art competitive programming platform has 2,000+ analytical and programming tests available in more than 20 different programming languages. All tests are available in Portuguese and English.

COMPETITION AND RANKING

Join the brightest minds in competitive programming! Participate in competitions, contests and tournaments! Compare your knowledge with your peers. Level up, grow and shine in your career!

BEECROWD ACADEMIC

The beecrowd Academic is a module for Educational Institutions, Professors and Coaches. Here you can create courses, exercise lists and track your students progress giving them real-time feedback.

[ACCESS ACADEMIC](#)

USE SOCIAL SIGN IN

[G](#) [in](#) [@](#) [f](#) [d](#)

OR

EMAIL

PASSWORD

REMEMBER ME (7 DAYS) [SIGN IN](#)

FIRST TIME HERE?

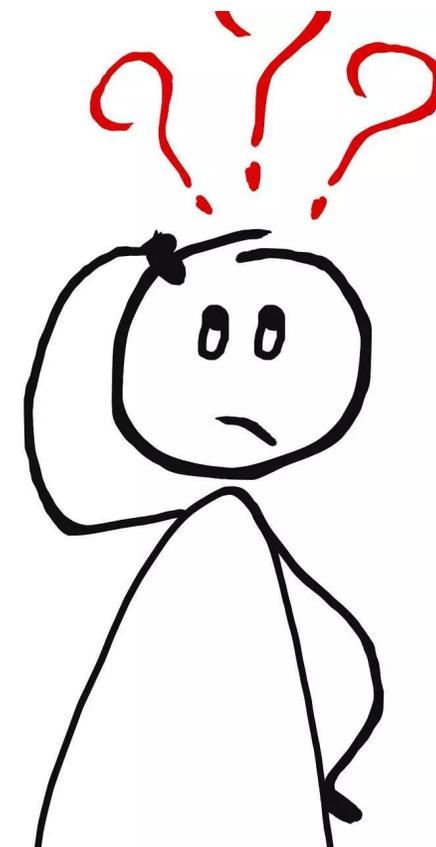
[SIGN UP](#) today to join one of the largest developer and competitive programming communities in the world!

<https://judge.beecrowd.com/>

Boas práticas

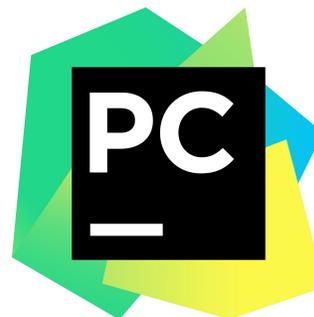
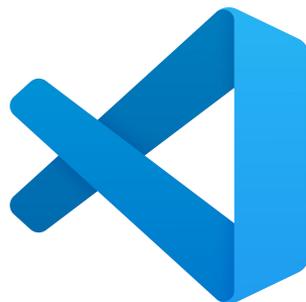
- Para a dinâmica de aprendizagem da disciplina funcionar, é importante realizar as práticas e exercícios passados em sala (não deixem acumular atividades).
- Quem deixa acumular as atividades, tende a ter um desempenho inferior.
- Não deixem as listas de exercício para última hora.
- Organizem seu tempo entre as disciplinas.

O que é um IDE?



IDE

- Uma IDE (do inglês Integrated Development Environment, ou Ambiente de Desenvolvimento Integrado) é um software que reúne, em um só lugar, diversas ferramentas necessárias para desenvolver programas de computador.
- Ou seja, em vez de usar um editor de texto, depois abrir o terminal para compilar, e depois usar outra ferramenta para depurar, a IDE junta tudo em uma interface única e integrada, facilitando a vida do programador.



Google Colab

- O Google Colab (ou Google Colaboratory) é uma plataforma gratuita oferecida pelo Google que permite escrever e executar códigos Python diretamente no navegador, sem precisar instalar nada no computador.
- Ele é muito usado em ciência de dados, aprendizado de máquina, inteligência artificial e ensino de programação, porque já vem integrado com várias bibliotecas prontas para uso.



Google Colab

- Principais características:
 - Baseado em Jupyter Notebook;
 - Execução na nuvem;
 - Gratuito;
 - Integração com Google Drive;
 - Bibliotecas pré-instaladas.

Google Colab



Google Colaboratory

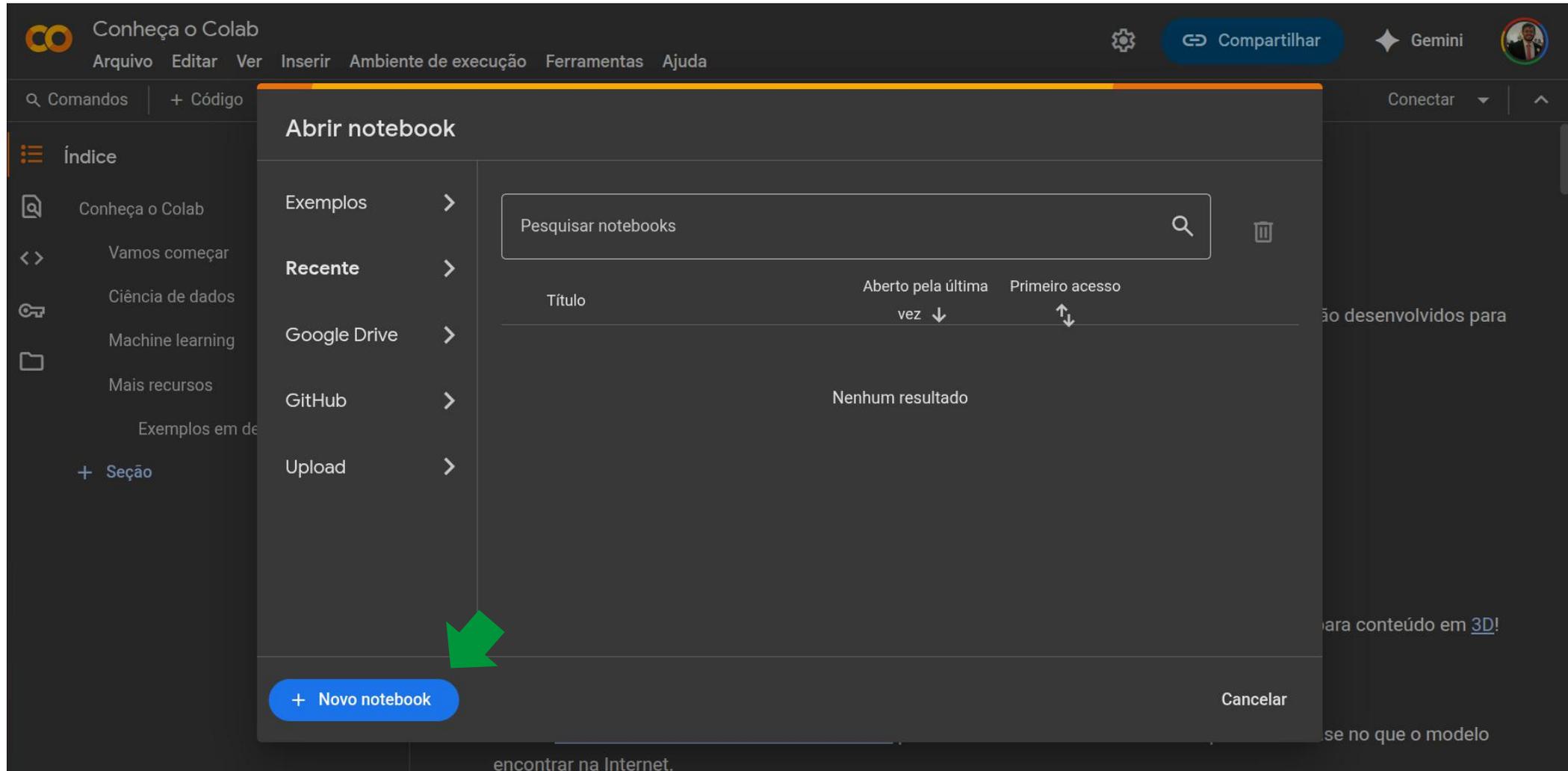
Colab is a hosted Jupyter Notebook service that requires no setup to use and provides free access to computing resources, including GPUs and TPUs. Colab is especially well suited to machine learning, data science, and education.



Open Colab

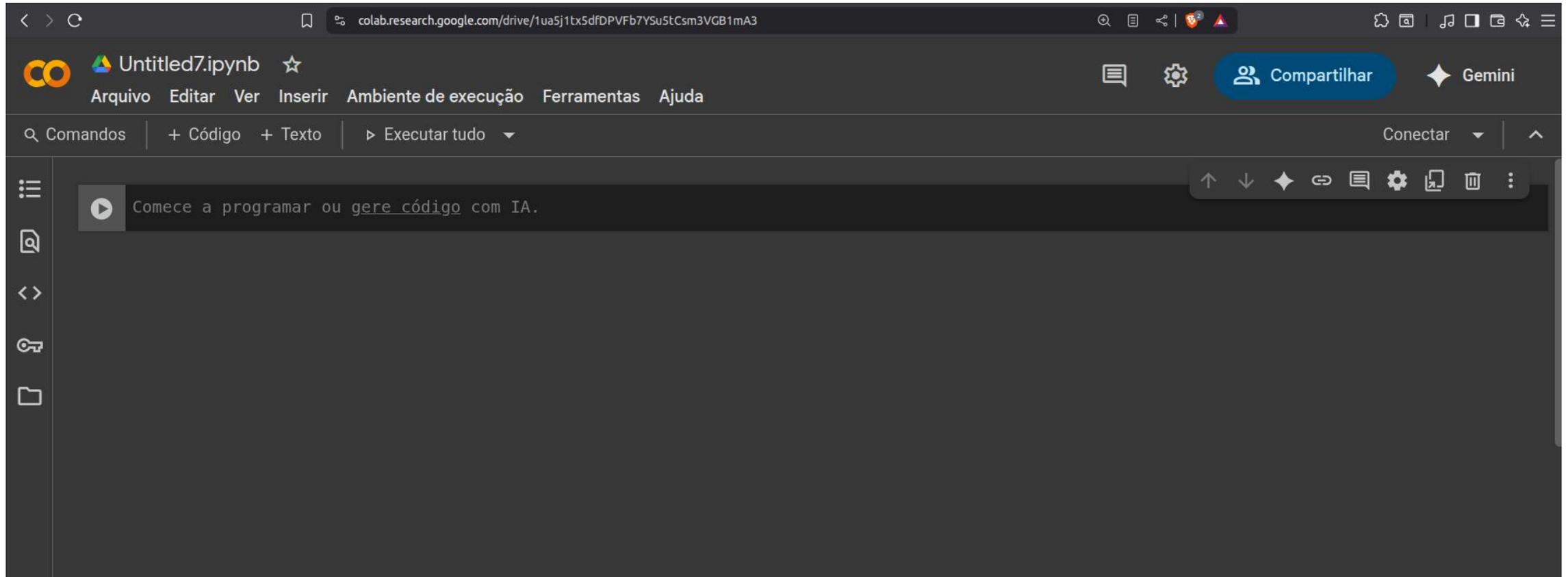
New Notebook

Google Colab

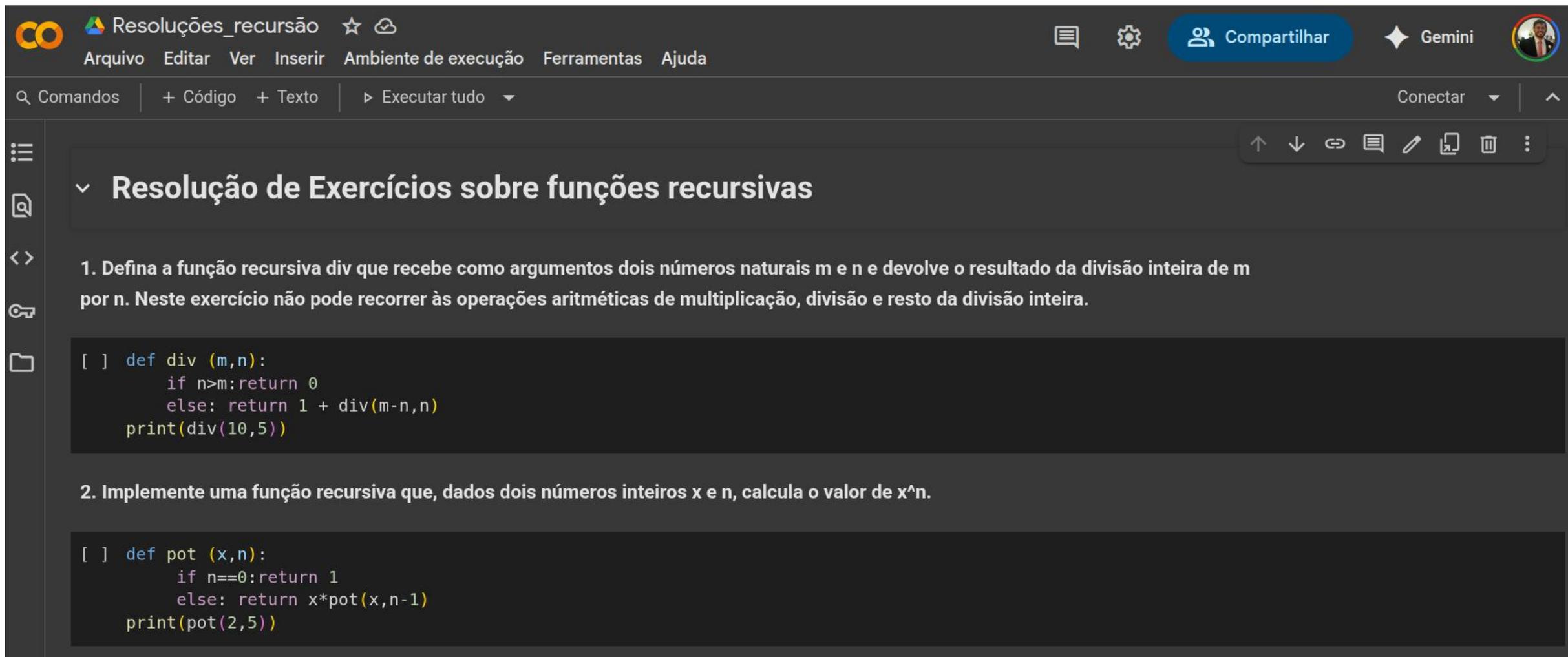


The screenshot shows the Google Colab interface with the 'Abrir notebook' dialog box open. The dialog box has a search bar at the top with the placeholder text 'Pesquisar notebooks'. Below the search bar, there are columns for 'Título', 'Aberto pela última vez', and 'Primeiro acesso'. The main area of the dialog box is empty, displaying 'Nenhum resultado'. At the bottom left of the dialog box, there is a blue button labeled '+ Novo notebook', which is highlighted by a green arrow. At the bottom right, there is a 'Cancelar' button. The background interface shows the 'Conheça o Colab' header, a menu bar with 'Arquivo', 'Editar', 'Ver', 'Inserir', 'Ambiente de execução', 'Ferramentas', and 'Ajuda', and a sidebar with an 'Índice' section containing items like 'Conheça o Colab', 'Vamos começar', 'Ciência de dados', 'Machine learning', and 'Mais recursos'.

Google Colab



<https://colab.research.google.com/#>



Resoluções_recurso

Arquivo Editar Ver Inserir Ambiente de execução Ferramentas Ajuda

Comandos + Código + Texto ▶ Executar tudo

Resolução de Exercícios sobre funções recursivas

- Defina a função recursiva `div` que recebe como argumentos dois números naturais `m` e `n` e devolve o resultado da divisão inteira de `m` por `n`. Neste exercício não pode recorrer às operações aritméticas de multiplicação, divisão e resto da divisão inteira.

```
[ ] def div (m,n):
    if n>m:return 0
    else: return 1 + div(m-n,n)
print(div(10,5))
```
- Implemente uma função recursiva que, dados dois números inteiros `x` e `n`, calcula o valor de x^n .

```
[ ] def pot (x,n):
    if n==0:return 1
    else: return x*pot(x,n-1)
print(pot(2,5))
```

Jupyter

- O Jupyter Notebook é uma ferramenta interativa que permite escrever e executar código, visualizar resultados, adicionar textos explicativos, imagens e até fórmulas matemáticas em um mesmo documento.
- O nome Jupyter vem de Julia, Python e R, três linguagens para as quais ele foi inicialmente pensado, mas hoje suporta várias outras.
- O formato padrão dos notebooks criados no Jupyter Notebook é o `.ipynb`



Jupyter

- Principais características:
 - Células de código: você escreve trechos de código (geralmente em Python) e pode executar de forma independente.
 - Células de texto (Markdown): permitem escrever explicações, colocar títulos, imagens e fórmulas matemáticas (em LaTeX).
 - Execução interativa: os resultados aparecem logo abaixo do código (tabelas, gráficos, imagens).
 - Documentação integrada: é possível misturar teoria + prática no mesmo arquivo, ótimo para ensino e pesquisa.
 - Extensível: suporta bibliotecas como NumPy, Pandas, Matplotlib, TensorFlow etc.

Jupyter no VsCode



Jupyter
Microsoft  microsoft.com |  94.423.581 |  (336)

Jupyter notebook support, interactive programming and computing that supports Intellisense, debu...

[Desabilitar](#) | [Desinstalar](#) | [Alternar para a Versão de Pré-Lançamento](#) Atualização Automática 

Jupyter no VsCode



```
strings.ipynb > M Exercícios: Strings > string1 = input("Escreva a primeira string: ")
Gerar + Código + Markdown | ▶ Executar Tudo ↺ Reiniciar ≡ Limpar Todas as Saídas | Variáveis do Jupyter ... Python 3.8.10
```

Exercícios: Strings

1. Faça um programa que leia 2 strings e informe o conteúdo delas seguido do seu comprimento. Informe também se as duas strings possuem o mesmo comprimento e são iguais ou diferentes no conteúdo.

```
string1 = input("Escreva a primeira string: ")
string2 = input("Escreva a segunda string: ")

print(f"A primeira string possui conteúdo {string1} e tamanho {len(string1)}.")
print(f"A segunda string possui conteúdo {string2} e tamanho {len(string2)}.")

if(len(string1) == len(string2)):
    print("As strings possuem o mesmo comprimento e ", end="")
else:
    print("As strings possuem diferentes comprimentos e ", end="")

if(string1 == string2):
    print('o mesmo conteúdo.')
else:
```

[1]

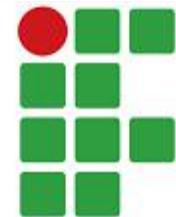
Exercícios

1. Crie um notebook no Colab.
2. Insira uma célula de texto (Markdown) com o título: "Minha primeira experiência no Colab".
3. Em uma célula de código, imprima: "Olá Colab"
4. Em uma célula de código, escreva um programa que calcule a soma, subtração, multiplicação e divisão de dois números fornecidos pelo usuário (use input).
5. Em uma célula de código, escreva um programa que exiba todos os números pares de 1 a 50.
6. Exporte o notebook criado no Colab no formato .ipynb e abra no VSCode. Utilize a extensão Jupyter para executar os códigos criados.

PROGRAMAÇÃO I

Curso Técnico Subsequente em Informática

Lucas Sampaio Leite



**INSTITUTO
FEDERAL**

Baiano