

Programação de computadores

APRESENTAÇÃO DO PLANO DE ENSINO

Coordenação

Coordenador Cursos Tecnólogos: Profa. Dra. Cristiane Camilo Hernandez
(cristiane.camilo@cruzeirosul.edu.br)

Coordenador Cursos Bacharelados: Prof. Dr. Ismar Frango Silveira
(ismar.silveira@cruzeirosul.edu.br)

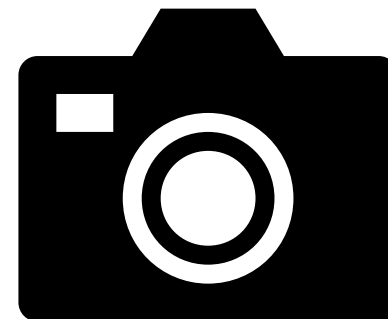
Adjuntos:

Prof. Dr. Alcides Teixeira Barboza Junior
(alcides.junior@cruzeirosul.edu.br)

Prof. Dr. Cleber Silva Ferreira da Luz
(cleber.luz@cruzeirosul.edu.br)

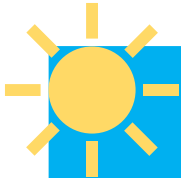
Assistente Administrativo:

Pedro Henrique (phpereira@cruzeirosul.edu.br)



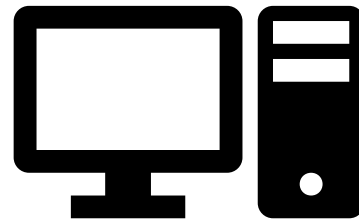
Coordenação - Avisos

- ❑ Acompanhem os **avisos no ambiente Blackboard** constantemente, principalmente das disciplinas online, atenção aos prazos de entregas das atividades destas disciplinas.

 60h 1h30 por módulo	1º Módulo	Intervalo (10 min)	2º Módulo
	19h10 às 20h25	20h25 às 20h35	20h35 às 21h50

Ementa

Estudo dos aspectos (conceitos, técnicas e ferramentas)
relacionados ao desenvolvimento de programas para
computador



Conteúdo

Introdução:

- ✓ Apresentação e discussão do Plano de Ensino, focando objetivos, conteúdos, estratégias, avaliação e bibliografia.
- ✓ Contextualização da programação de computadores no semestre, curso, área e mercado de trabalho.

Conceitos básicos:

- ✓ Edição, compilação, execução, depuração.
- ✓ Programas e Linguagens de Programação.
- ✓ Linguagem *Python* para programação.



Conteúdo

Atribuição e E/S (Entrada e Saída):

- ✓ Comando de atribuição;
- ✓ Leitura de dados;
- ✓ Exibição de dados;
- ✓ Exemplos de algoritmos e implementação em *Python*.

Operações Numéricas:

- ✓ Tipos e operadores numéricos;
- ✓ Precedência de operadores;
- ✓ Conversão de tipos;
- ✓ Exemplos de algoritmos e implementação em *Python*.



Conteúdo

Operações lógicas:

- ✓ Tipo booleano ou lógico;
- ✓ Operações lógicas (and or not);
- ✓ Exemplos de algoritmos e implementação em *Python*.

Tomadas de decisão (condicional):

- ✓ Comando Se...então;
- ✓ Comando Se...então...Senão;
- ✓ Composição de condições (and or not);
- ✓ Condicionais aninhados;
- ✓ Exemplos de algoritmos e implementação em *Python*.



Conteúdo

Repetição

- ✓ Comando Enquanto...;
- ✓ Comando Faça... Enquanto;
- ✓ Comando Para... até....;
- ✓ Composição de comandos de repetição e condicionais;
- ✓ Comandos de repetição aninhados;
- ✓ Exemplos de algoritmos e implementação em *Python*.

Processamento de textos (operações com cadeias de caracteres):

- ✓ Conceito de string;
- ✓ Operações com strings;
- ✓ Exemplos de algoritmos e implementação em *Python*.



Conteúdo

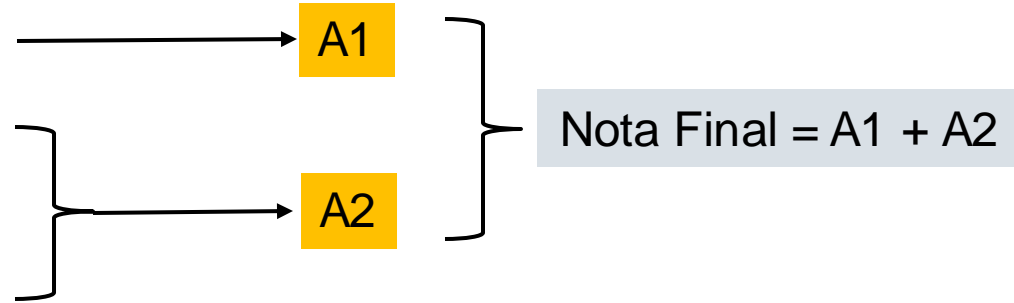
Vetores - estruturas lineares:

- ✓ Conceito de vetor;
- ✓ Operações básicas em vetores (criação inicialização inserção remoção troca e busca);
- ✓ Exemplos de algoritmos e implementação em *Python*.

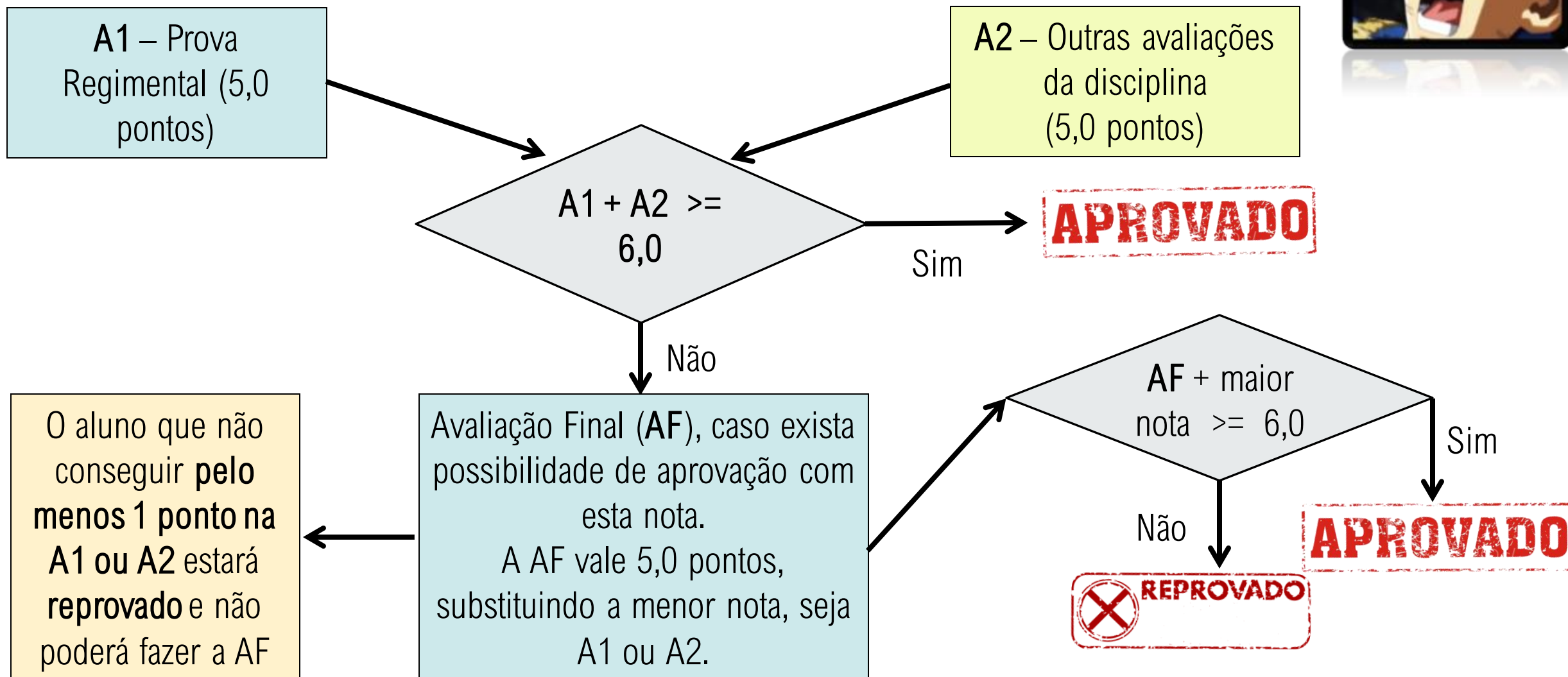
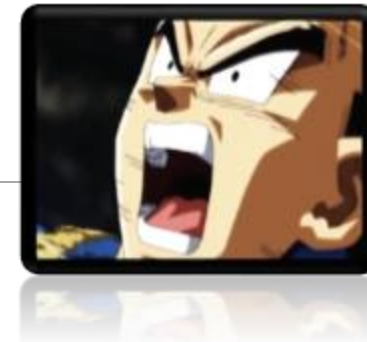


Avaliação

- ✓ Prova Regimental - 5,0
- ✓ Avaliação Parcial - 3,0
- ✓ Exercícios/Projetos - 2,0



Sistema de avaliação

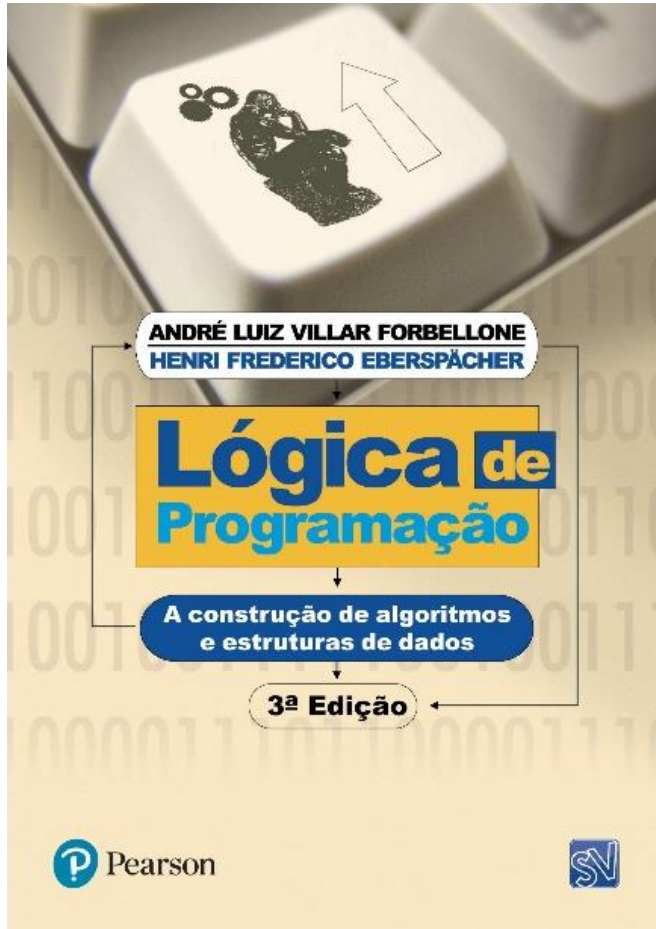


Bibliografia Básica



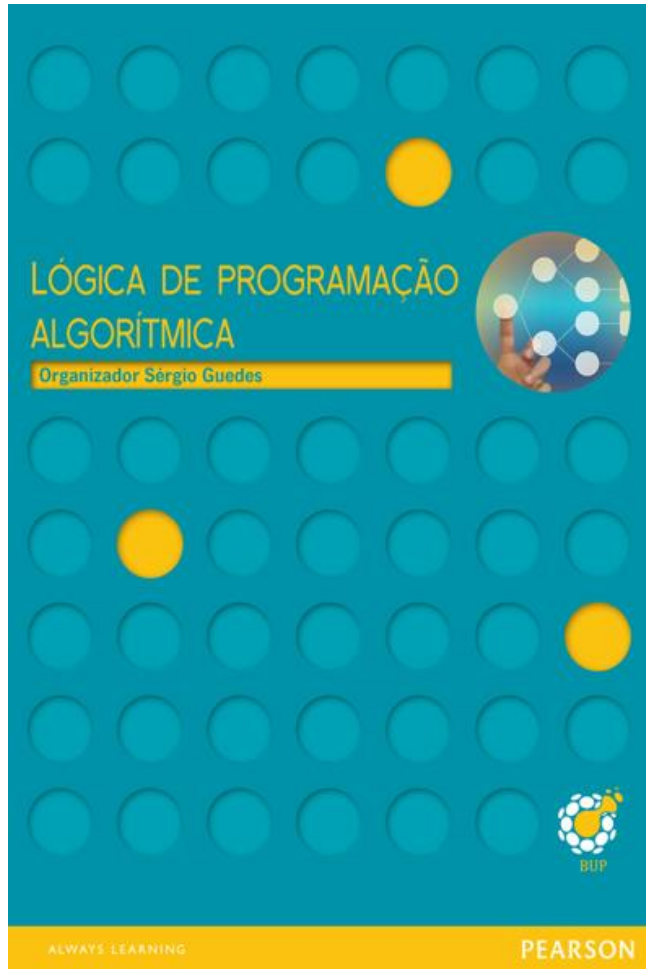
- 1) ARAUJO, S. **Lógica de programação e algoritmos**. Curitiba: Contentus, 2020. (e-book).

Bibliografia Básica



2) FORBELLONE, A. L. V.; EBERSPACHER, H. F.
Logica de Programação: A Construção de Algoritmos e Estrutura de Dados. 3. ed. São Paulo: Pearson Prentice Hall, 2008.

Bibliografia Básica



3) GUEDES, S. **Lógica de Programação Algorítmica**. São Paulo: Pearson Education do Brasil, 2014 (e-book)

Bibliografia Complementar

ASCENCIO, Ana F. G.; CAMPOS, Edilene A. V. Fundamentos da Programação de Computadores - 3ª edição São Paulo: Pearson, 2012. (e-book).

DASGUPTA, SANJOY; PAPADIMITRIOU, CHRISTOS; VAZIRANI, UMESH. Algoritmos. Porto Alegre: Grupo A, 2011. (e-book).

PYTHON SOFTWARE FOUNDATION. A Biblioteca Padrão do Python. Disponível em: <https://docs.python.org/pt-br/3/library/index.html>. Acesso em: 30 jan. 2023.

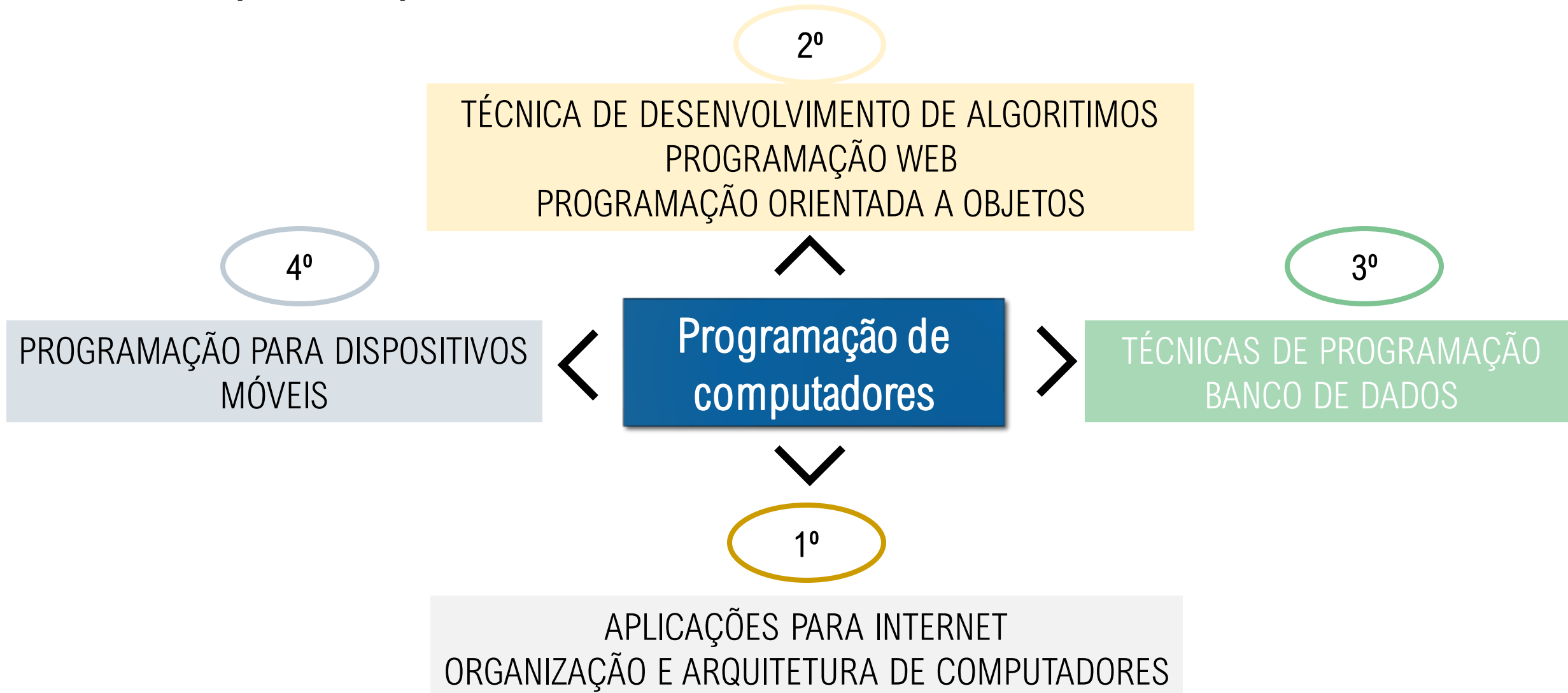
PYTHON SOFTWARE FOUNDATION. A Referência da Linguagem Python. Disponível em: <https://docs.python.org/pt-br/3/reference/index.html>. Acesso em: 30 jan. 2023.

PYTHON SOFTWARE FOUNDATION. O tutorial de Python. Disponível em: <https://docs.python.org/pt-br/3/tutorial/index.html>. Acesso em: 30 jan. 2023.

+ Materiais de aula (resumo).

+ Materiais disponíveis na Internet e indicados pelos professores quando for pertinente.

Relação (diretas) com disciplinas do curso



Linguagem de programação Python

Nesta disciplina iremos utilizar a linguagem de programação Python.

- ✔ A linguagem Python foi concebida no final de 1989 no Instituto de Pesquisa Nacional para Matemática e Ciência da Computação em Amsterdã.



Guido van Rossum em
2006**



** https://pt.wikipedia.org/wiki/Guido_van_Rossum

Ferramentas

Nesta disciplina iremos utilizar as seguintes ferramentas:

- ✓ Caderno/Lápis/Borracha
- ✓ Python



Ferramentas – Python

Acesse o site do Python para baixar e instalar a versão 3 ou superior:

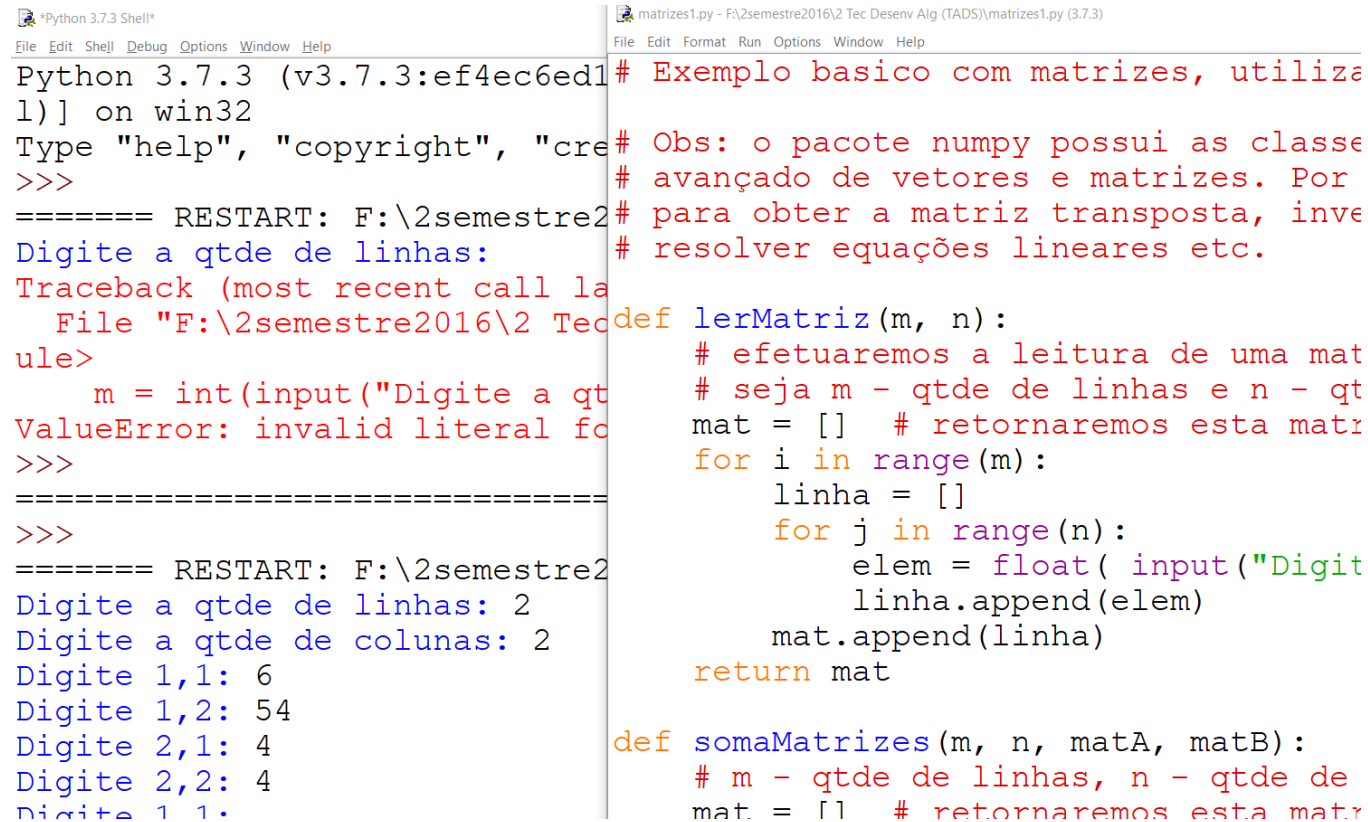


Detecta automaticamente o SO

<https://www.python.org/downloads/>

Ferramentas – Python

Após instalar o Python, teremos a opção de usar a linha de comando ou o IDLE*



The image shows two side-by-side screenshots of a Python 3.7.3 environment. The left window is a command prompt titled '*Python 3.7.3 Shell*' showing the execution of a script. The right window is a text editor titled 'matrizes1.py - F:\2semestre2016\2 Tec Desenv Alg (TADS)\matrizes1.py (3.7.3)' showing the source code of the script.

```
Python 3.7.3 (v3.7.3:ef4ec6ed1) on win32
Type "help", "copyright", "credits()" or "quit()" for more
>>>
===== RESTART: F:\2semestre2016\2 Tec Desenv Alg (TADS)\matrizes1.py (3.7.3)
Digite a qtde de linhas: 2
Traceback (most recent call last):
  File "F:\2semestre2016\2 Tec Desenv Alg (TADS)\matrizes1.py", line 1, in <module>
    m = int(input("Digite a qtde de linhas: "))
ValueError: invalid literal for int() with base 10: '2'
>>>
===== RESTART: F:\2semestre2016\2 Tec Desenv Alg (TADS)\matrizes1.py (3.7.3)
Digite a qtde de linhas: 2
Digite a qtde de colunas: 2
Digite 1,1: 6
Digite 1,2: 54
Digite 2,1: 4
Digite 2,2: 4
Digite 1,1: 1
```

```
# Exemplo basico com matrizes, utiliza o pacote numpy
# Obs: o pacote numpy possui as classe avançado de vetores e matrizes. Por para obter a matriz transposta, inverse resolver equações lineares etc.

def lerMatriz(m, n):
    # efetuaremos a leitura de uma matriz
    # seja m - qtde de linhas e n - qtde de colunas
    mat = [] # retornaremos esta matriz
    for i in range(m):
        linha = []
        for j in range(n):
            elem = float(input("Digite o elemento %d,%d: " % (i+1, j+1)))
            linha.append(elem)
        mat.append(linha)
    return mat

def somaMatrizes(m, n, matA, matB):
    # m - qtde de linhas, n - qtde de colunas
    mat = [] # retornaremos esta matriz
```

* IDLE é um ambiente de desenvolvimento integrado para Python

Python IDLE / Editor de código

1. PyCharm

2. Spyder

3. Sublime

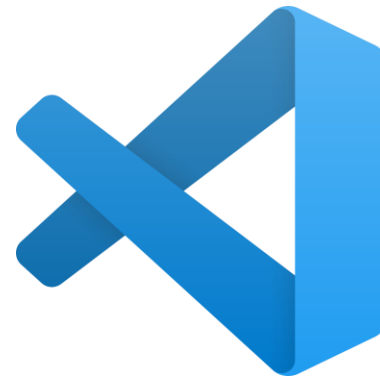
4. Jupyter

5. Atom

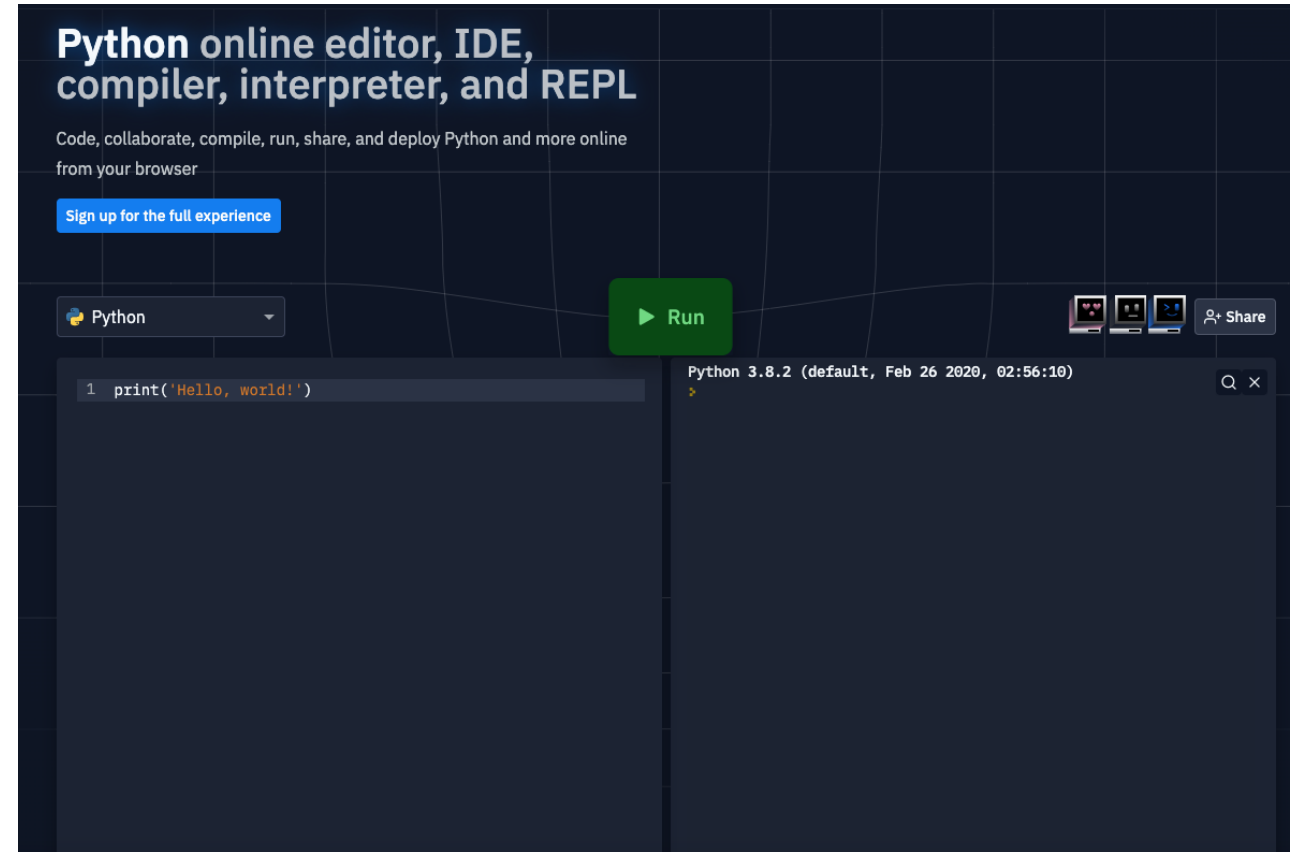
6. Python IDLE

7. VSCode

8. Colaboratory



Opções On-line



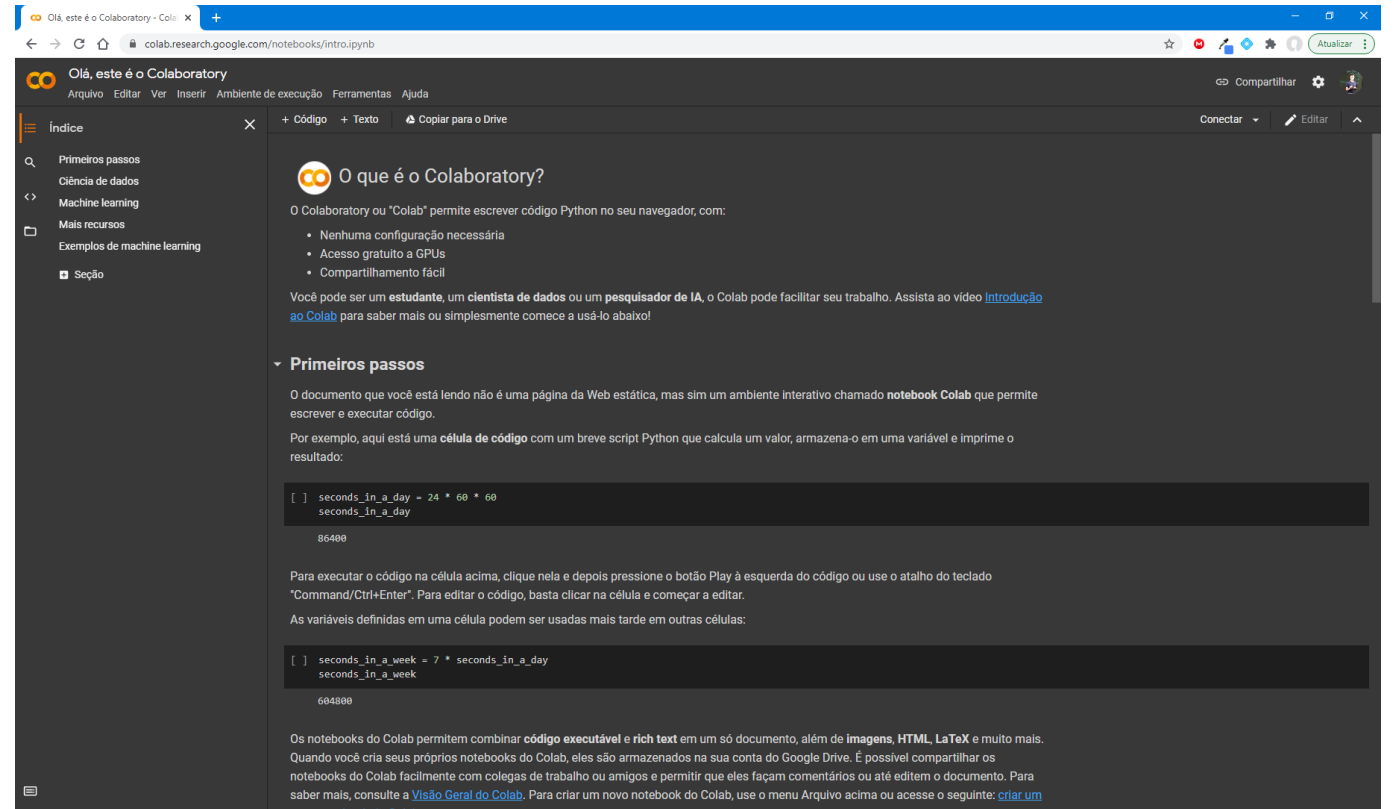
<https://repl.it/languages/python3>

Opções On-line



O Colaboratory ou "Colab" permite escrever código Python no seu navegador, com:

- ✓ Nenhuma configuração necessária
- ✓ Acesso gratuito a GPUs
- ✓ Compartilhamento fácil



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

Outras opções



<https://www.programiz.com/python-programming/online-compiler/>



<https://www.w3schools.com/python/default.asp>



<https://docs.python.org/3/reference/index.html>

Por que todos deveriam aprender a programar?

<https://www.youtube.com/watch?v=mHW1Hsqlp6A>

<http://bit.ly/TDA-aula1>



"Todas as pessoas nesse país deveriam aprender a programar um computador

Por que todos deveriam aprender a programar?



Linguagens de programação?!



Linguagens



<https://spectrum.ieee.org/top-programming-languages-2022>

Feb 2023	Feb 2022	Change	Programming Language	
1	1			Python
2	2			C
3	4	▲		C++
4	3	▼		Java
5	5			C#
6	6			Visual Basic
7	7			JavaScript
8	10	▲		SQL

<https://www.tiobe.com/tiobe-index/>

Desafios? Quer aprender mais?



<https://www.beecrowd.com.br/judge/pt/login>

Resolva os problemas disponíveis utilizando as 11 linguagens de programação, competindo com os outros usuários. Como desafio, melhore seu ranking, resolvendo o máximo de problemas e aperfeiçoando seu código fonte.

Reflexão (motivação)



Qual é a mensagem da imagem ao lado?

Se você acredita que você não pode, você está certo.

Se você acredita que você pode, você está certo.

“As oportunidades se multiplicam quando você as aproveita e morrem quando são ignoradas.

Aquele que não luta pelo futuro que quer, deve aceitar o futuro que vier.
É fazendo que se aprende, aquilo que se deve aprender a fazer.”

(Aristóteles)

Alguma dúvida????

Quando tiver
alguma dúvida, não
duvide, pergunte!

 PENSADOR

Ivani Pereira

Vamos começar???

