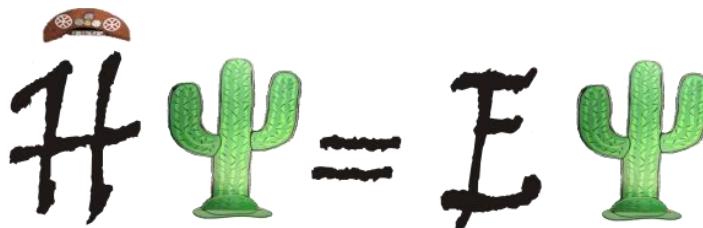




# Universidade Federal da Paraíba

Departamento de Química

Laboratório de Química Quântica Computacional



Grupo de Química Computacional mais Oriental das Américas  
L.Q.Q.C. - UFPB - João Pessoa - PB

## Minicurso: Noções de Python para Químico(a)s

Gerd Bruno Rocha

[gbr@academico.ufpb.br](mailto:gbr@academico.ufpb.br)

[www.quantum-chem.pro.br](http://www.quantum-chem.pro.br)



# Ementa do Minicurso

## 1. Introdução

- Motivação: Por que aprender a programar?
- Por onde começar?
- Noção de algoritmos
- Pseudo código
- Linguagens de Programação
- Python: aspectos gerais e informações
- IDEs e onde programar em Python
- Google Colab

# Ementa do Minicurso

## 2. Parte Prática

(Dia 10/11): Noções de Python

1. Histórico do Python e instalação
2. Bibliotecas e módulos
3. Tipos de dados, constantes e variáveis; vetores e matrizes; listas, tuplas e dicionários;
4. Input e output com arquivos externos
5. Controlando o fluxo de dados do seu programa
6. Repetindo partes do programa
7. Funções;
8. Bibliotecas básicas do Python

# Ementa do Minicurso

(Dia 11/11): Aplicações para Químico(a)s

Bibliotecas de Python para químicos

1. RDKit
2. ASE
3. PySCF
4. AmberTools, Pytraj e NGLView

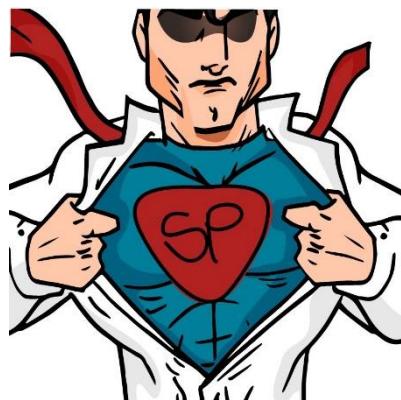
Preencher o formulário:

**<https://forms.gle/xUmuL44D7otM7Kt17>**

# Motivações

**Por que é importante aprender a programar?**

**"É A COISA MAIS PRÓXIMA QUE EXISTE DE TER SUPERPODERES"**



**Drew Houston**



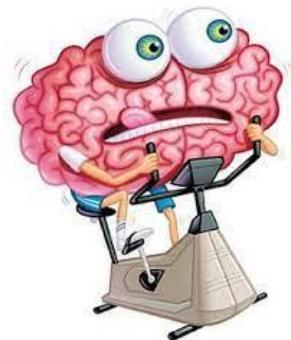
<https://www.youtube.com/watch?v=t4dfngUxXVA>

<https://www.youtube.com/watch?v=nEEUjIjK-bE>

# Motivações

## Por que é importante aprender a programar?

1. Exercício mental
2. Procedimento fixador de conhecimento
3. Automatizar procedimentos repetitivos
4. Economia de tempo
5. Diversão
6. Resolução de procedimentos complexos



Vídeo:

- <https://www.youtube.com/watch?v=t4dfngUxXVA>

# Motivações

[Classroom](#)[Staffroom](#)[Science](#)[Collections](#) ▾[Post-lockdown teaching support](#)[Get the print issue](#)[RSC Education >>](#)[Opinion](#)

## Why should chemistry students learn to code?

By Cate Cropper | 26 October 2017



### How the University of Liverpool is training students to meet the demands of industry

As a chemistry lecturer, my job is to take cohorts of students who have recently passed their A-levels and turn them into employable graduates. In three years.

On a generic chemistry degree, in addition to the core content, students learn how to process data, present, write reports, operate spectrometers, model molecules and navigate their virtual learning environment. This is enough to get a degree, but is it enough for employers? Not these days.



Source: best backgrounds / shutterstock.com

[Latest](#)[Popular](#)

Give context to your teaching of biofuels



3 key skills your post-16 chemists need for university

<https://edu.rsc.org/opinion/why-should-chemistry-students-learn-to-code/3008177.article#:~:text=For%20chemists%20in%20particular%2C%20coding,the%20discovery%20of%20new%20materials.>

# Motivações



SCROLL TO CONTINUE WITH CONTENT

ACS Publications

C&EN

CAS

Become a Member

Login



TOPICS ▾

MAGAZINE ▾

COLLECTIONS ▾

VIDEOS

Jobs



EDUCATION

## Teaching chemistry students to use the labs of the future

Educators want to prepare chemists for a time when autonomous machines will do the work

by Sam Lemonick

December 2, 2018 | A version of this story appeared in **Volume 96, Issue 48**

<https://cen.acs.org/education/Teaching-chemistry-students-use-labs/96/i48>

# Motivações

The screenshot shows the homepage of C&EN (Chemical & Engineering News). At the top, there's a navigation bar with links for ACS, Publications, C&EN (highlighted in red), and CAS. To the right are buttons for 'Become a Member' and 'Login'. Below the navigation is the C&EN logo and the text 'CHEMICAL & ENGINEERING NEWS'. A red banner across the top has links for TOPICS, MAGAZINE, COLLECTIONS, VIDEOS, JOBS, and a search icon. On the far right of the banner are social media icons for Facebook, Instagram, and Twitter. Below the banner is a dark blue section featuring an advertisement for 'THE JOURNAL OF PHYSICAL CHEMISTRY A' and the 'International Symposium on Molecular Spectroscopy'. The text 'ADVERTISEMENT' is above the ad, and 'VIRTUAL SPECIAL ISSUE' is to its right. In the middle of the page, there's a red box containing the word 'EDUCATION'. Below this, a large, bold, black header reads 'Perspectives: Teaching chemists to code'. Underneath the header is a subtext: 'Providing undergraduate chemistry majors with computer programming skills can make them more efficient and effective scientists'. It includes byline information: 'by Charles J. Weiss' and 'September 4, 2017 | A version of this story appeared in Volume 95, Issue 35'. The main article text begins with a large, bold, black 'S' followed by the first paragraph.

**S**preadsheets are a standard tool in chemistry for simple tasks such as data analysis and graphing. Chemistry students are often introduced to spreadsheets their first year of college, if not earlier, and those who continue on to do research will likely use them as a means of handling and visualizing data.



<https://cen.acs.org/articles/95/i35/Perspectives-spreadsheets-programming.html>

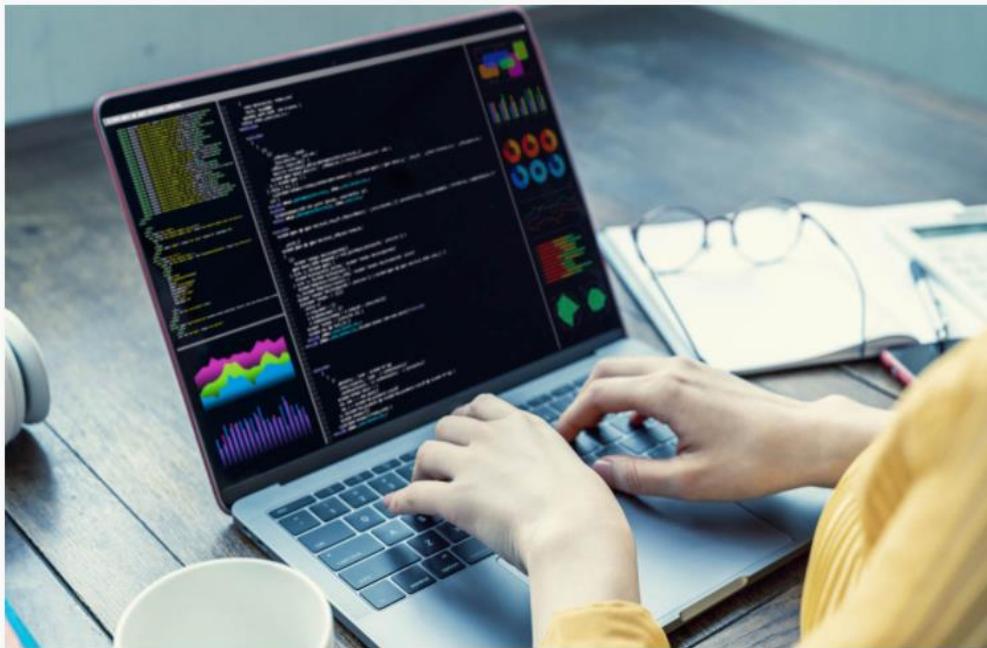
# Motivações

GEN// **Exatas**

HOME ENGENHARIAS TECNOLOGIA E INOVAÇÃO MEIO AMBIENTE EDUCAÇÃO ARTIGOS CARREIR

HOME > ENGENHARIAS > COMPUTAÇÃO > POR QUE APRENDER PROGRAMAÇÃO SERÁ TÃO NECESSÁRIO QUANTO SABER LER?

Por que aprender programação será tão necessário quanto saber ler?



INSCREVA-SE EM  
NOSSA NEWSLETTER

---

Nome

E-mail

Perfil

Não sou um robô reCAPTCHA  
Privacidade + Termos

Se o não sou um robô não for marcado,  
não receberemos o seu contato

MAIS LIDOS



<https://genexatas.com.br/por-que-aprender-programacao-sera-tao-necessario-quanto-saber-ler/>

# Motivações



home do blog

categorias ▾

ebook gratuito

assine a newsletter

buscar artigos



PROGRAMAÇÃO

## 7 bons motivos para aprender a programar



Por Escola Conquer

Publicado em 20.09.2019 | Atualizado em 12.03.2020

COMPARTILHE



*“Todos deveriam aprender a programar um computador, porque isso ensina a pensar.”*

Esta frase é de Steve Jobs, criador de uma das maiores empresas de tecnologia do mundo.

<https://escolaconquer.com.br/blog/7-motivos-para-aprender-a-programar/>

# Por onde começar?

 [Website](#) ago 17, 2021 Andrei L. 21min de leitura

## 25 Sites Para Aprender Como Programar de Graça



 HOSTINGER TUTORIAIS

Domine o WordPress em 7 Dias com um Curso Grátis

Insira seu nome aqui...

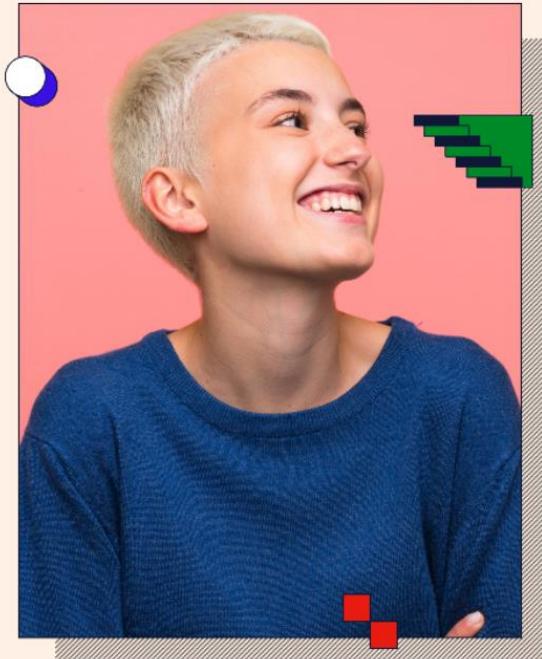
Insira seu email aqui...

Ao inserir seu email, você vai se inscrever na newsletter da Hostinger.  
 Você pode cancelar essa inscrição a qualquer momento. Veja nossa política de privacidade

**Inscreve-se**

<https://www.hostinger.com.br/tutoriais/11-sites-aprender-como-programar-de-graca>

# Por onde começar?



Join the millions learning to code  
with Codecademy for free

Email

Password

 ⓘ

**Sign Up**

By signing up for Codecademy, you agree to  
Codecademy's [Terms of Service](#) & [Privacy Policy](#).

Or sign up using:



<https://www.codecademy.com/>

# Por onde começar?



Catálogo do Curso    Projetos    Sobre

Criar ▾    Entrar    ?

Aprenda Ciência da Computação.  
Mude o mundo.

Comece a aprender

Assista ao vídeo

Take a Code Break! Your weekly dose of inspiration, community, and computer science.

Saiba mais

<https://code.org/>

# Por onde começar?

Caiena,

Home

Tecnologia

Design

Projetos

Newsletter

Site

...



## Mulheres programadoras: 8 iniciativas que mudam o cenário

Tecnologia • Set 10, 2019

<https://blog.caiena.net/mulheres-programadoras-8-iniciativas/>

# Algoritmo

Para resolver um problema no computador é necessário:

1. Saber descrever o problema de forma clara e precisa.
  - Ex: quero calcular o pH de uma solução ácida!
2. Encontrar uma seqüência de passos que permitam que o problema seja resolvido, ou seja, encontrar o **algoritmo**.
3. Utilizar uma linguagem de programação para fazer a interface homem-máquina.
4. Executar o programa e conferir o resultado (depuração).

A noção de algoritmo é central para toda a computação.

# **Algoritmo**

**Um algoritmo é um conjunto finito de regras que fornece uma sequência de operações para resolver um problema específico.**

**Algoritmo = receita, procedimento, rotina ....**

Um algoritmo opera sobre um conjunto de entradas de modo a gerar uma saída que seja útil (ou agradável) para o usuário.

Ex: Preparação de um bolo

Entradas: ingredientes

Saída esperada: bolo pronto

# Algoritmo

Um algoritmo tem cinco características importantes:

- 1. Finitude:** Um algoritmo deve sempre terminar após um número finito de passos.
- 2. Definição:** Cada passo de um algoritmo deve ser precisamente definido. As ações devem ser definidas rigorosamente e sem ambiguidades.
- 3. Entradas:** Um algoritmo deve ter zero ou mais entradas, isto é informações que são lhe fornecidas antes do algoritmo iniciar.
- 4. Saídas:** Um algoritmo deve ter uma ou mais saídas, isto é quantidades que tem uma relação específica com as entradas.
- 5. Efetividade:** Um algoritmo deve ser efetivo. Isto significa que todas as operações devem ser suficientemente básicas de modo que possam ser em princípio executadas com precisão em um tempo finito por um humano usando papel e lápis.

# **Algoritmo: um exemplo**

Ir trabalhar pela manhã:

1. Acordar
2. Levantar da cama
3. Tomar banho
4. Escovar os dentes
5. Se vestir
6. Sair de casa
7. Ir até o trabalho
8. Começar a trabalhar

**Executamos algoritmos todo instante**

# Algoritmo: observando alguns detalhes

Problema: Ir trabalhar pela manhã.

Para executarmos o algoritmo para esse problema precisamos:

- Definir as entradas:
  - Que horas acordar?
  - Qual o meio de transporte utilizar?
  - Que caminho percorrer?
  - Etc..
- Prever situações possíveis e tomar decisões:
  - Se estiver chovendo levo um guarda-chuvas
  - Se estiver engarrafado vou por outro caminho
  - Etc...

**Existem uma grande variedade de  
algoritmos para resolver esse problema**

# Representação de algoritmos

**Linguagem Natural:** Os algoritmos são expressos diretamente em linguagem natural.

**Fluxograma Convencional:** Representação gráfica que emprega formas geométricas padronizadas para indicar as diversas ações que devem ser executadas e decisões que devem ser tomadas para resolver o problema.

**Pseudo-linguagem:** Emprega uma linguagem intermediária entre a linguagem natural e uma linguagem de programação para descrever os algoritmos.

# Fluxogramas

Esta forma de representação de algoritmos emprega várias formas geométricas para descrever cada uma das possíveis ações durante a execução do algoritmos. Cada uma destas formas se aplica a uma determinada ação.

## Fluxogramas



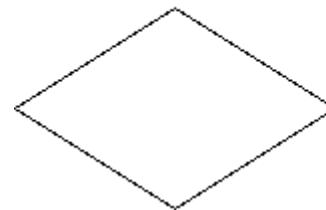
Início e final de fluxograma



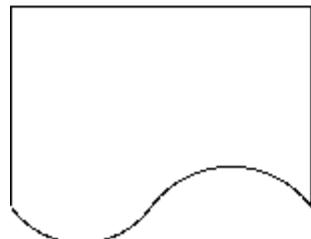
Operação de atribuição de valores e cálculos de expressões



Operação de entrada de dados



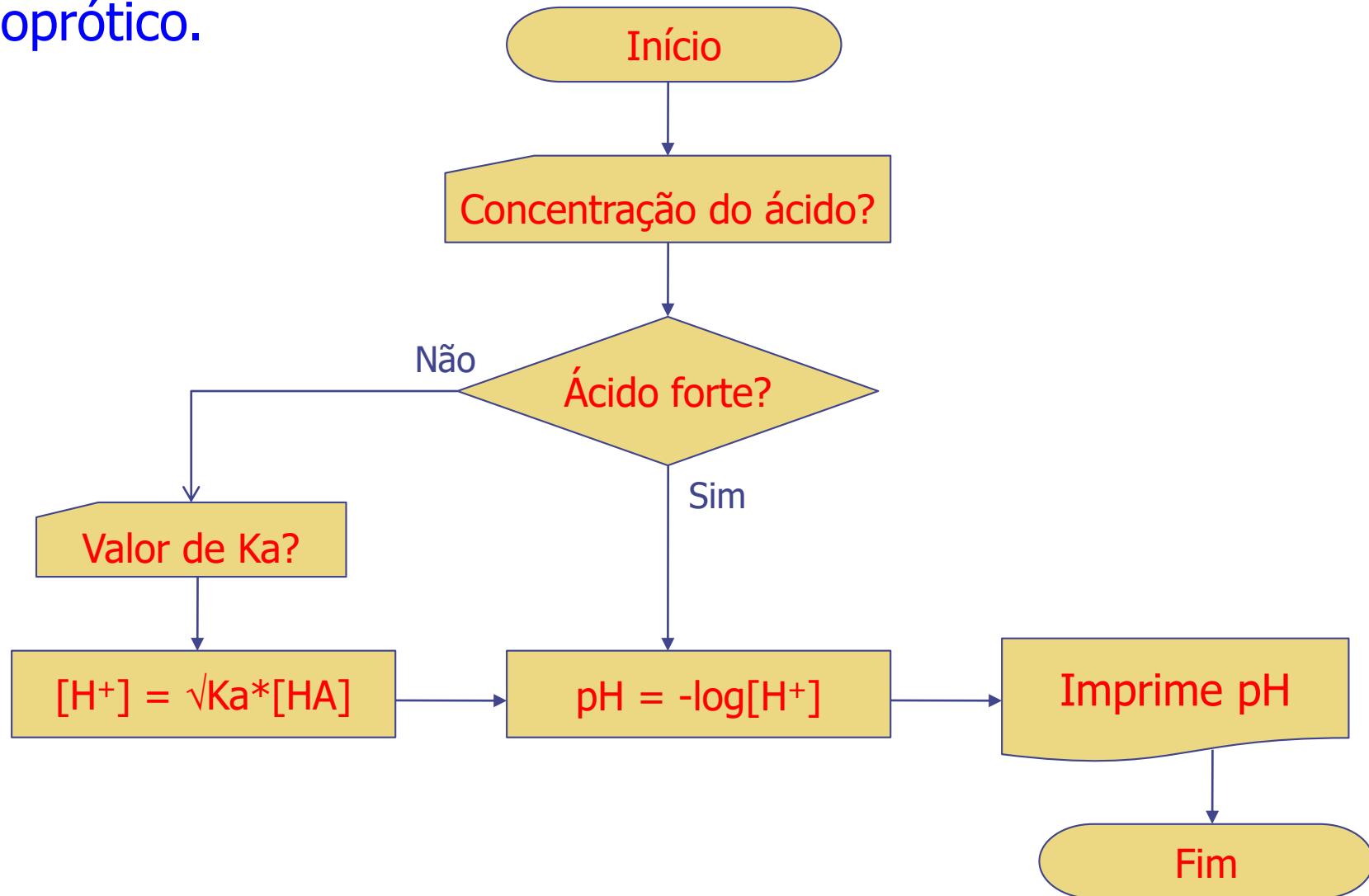
Decisões



Operação de saída de dados

# Exemplo de fluxograma

Problema: calcular o pH de uma solução de um ácido monoprótico.



# Pseudo Código

Este modo de representar algoritmos procura empregar uma linguagem que esteja o mais próximo possível de uma linguagem de programação de computadores de alto nível mas evitando de definir regras de construção gramatical muito rígidas. A ideia é usar as vantagens do emprego da linguagem natural, mas restringindo o escopo da linguagem.

Normalmente estas linguagens são versões ultra reduzidas de linguagens de alto nível do tipo Fortran, Pascal ou C.

<http://lite.acad.univali.br/portugol/>

<http://www.c5.cl/ieinvestiga/actas/ribie98/118.html>



# Exemplo do uso de pseudo código

Problema: calcular o pH de uma solução de um ácido monoprótico.

Início do algoritmo de cálculo do pH de uma solução de um ácido monoprótico

Leia a concentração do ácido (HA)

O ácido é forte?

Se (ácido for forte) então

pH  $\leftarrow$  -log(HA)

Senão

Leia o valor de Ka (Ka)

x  $\leftarrow$  raiz quadrada (Ka\*HA)

pH  $\leftarrow$  -log(x)

Fim do teste

Imprima pH

Fim do algoritmo

# Exemplo do uso de pseudo código

Problema: calcular as raízes de uma equação do segundo grau.

Algoritmo para cálculo de uma equação do segundo grau.

Leia os coeficientes a, b e c

Se ( $a = 0.0$ ) então

    imprima “Essa equação não é do segundo grau”

    Encerre o algoritmo

Senão

$\text{delta} \leftarrow b^{**2} - 4.0 * a * c$

    Se ( $\text{delta} < 0.0$ ) então

        imprima “Essa equação não tem raízes reais”

        Encerre o algoritmo

    Senão

$\text{raiz} \leftarrow \text{delta}^{**0.5}$

$x1 \leftarrow (-b + \text{raiz}) / (2 * a)$

$x2 \leftarrow (-b - \text{raiz}) / (2 * a)$

        imprima “As raízes da equação são x1 e x2”

    Fim do teste

    Fim do teste

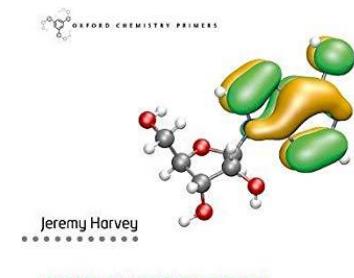
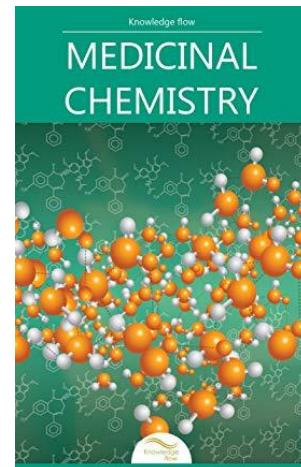
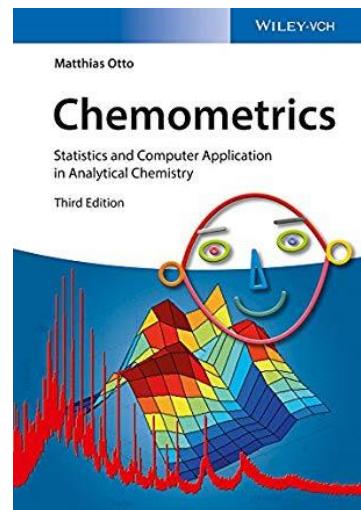
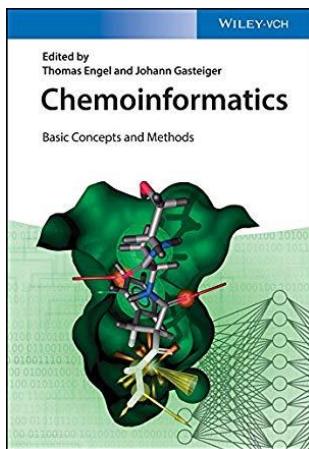
Fim do algoritmo para cálculo de uma equação do segundo grau.

## **Exercícios:**

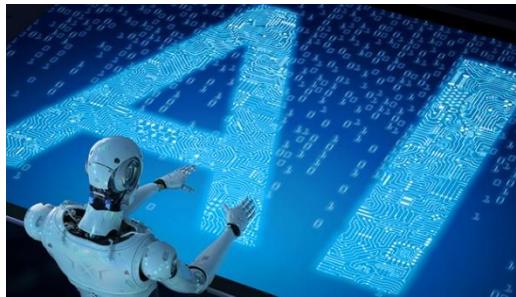
## **Escreva os pseudo-algoritmos para:**

1. Calcular o números de isômeros óticos a partir do número de carbonos assimétricos existentes numa molécula.
2. Verificar se o comprimento de onda de um fóton está na região do infravermelho, visível ou ultravioleta.
3. Calcule a pressão em bar para 2,5 mol de um gás que está confinado num recipiente de 100mL e numa temperatura de 30°C.

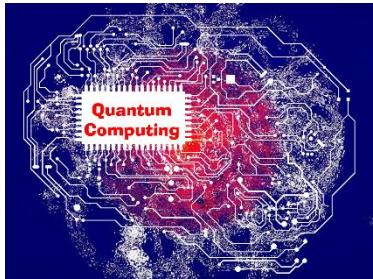
# Subáreas da química que usam a computação mais intensamente



# **Hot topics no século 21**



HPC



# Novos algoritmos

Será que esses grandes temas são importantes para química?

# Linguagens de Programação

1. Interpretadas: precisam de um **interpretador** para executar o código
  - Python
  - Matlab, Octave, R
  - Mathematica, Maple, Maxima, Mathcad
2. Compiladas: precisam de um **compilador** para executar o código
  - Fortran
  - C/C++
  - Basic
  - Pascal

# Linguagens de Programação

## Melhores Linguagens de Programação Para Aprender em 2021

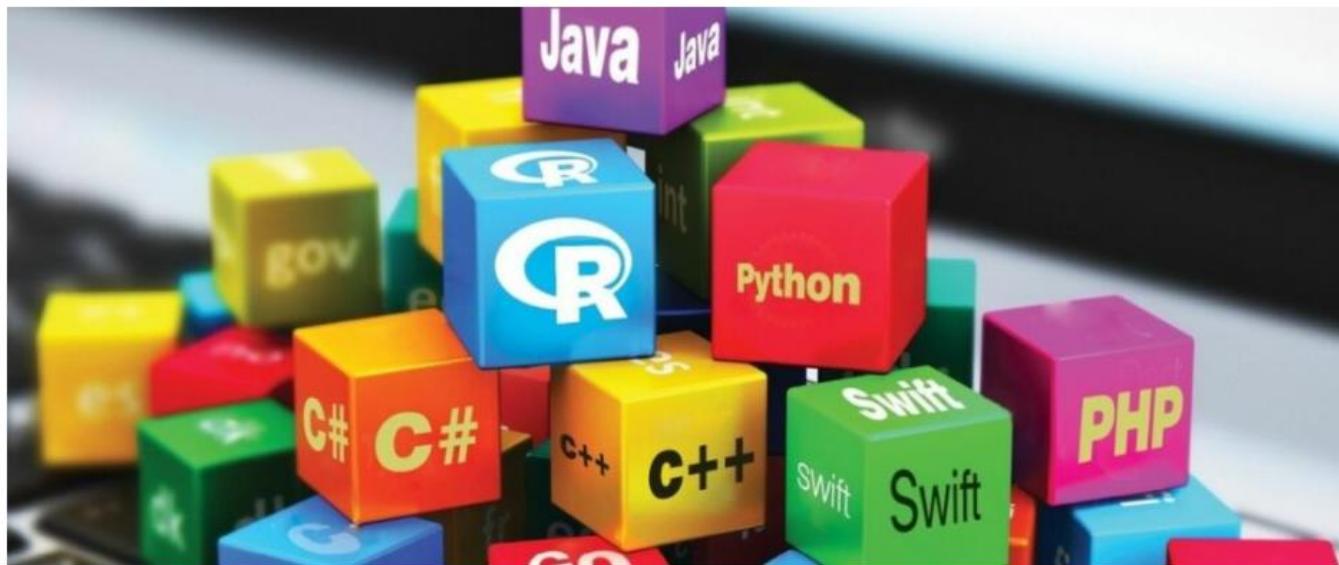


Robson dos Santos

22 de fevereiro de 2021

7 Comentários

Leitura: 7 min



<https://www.brasilcode.com.br/melhores-linguagens-de-programacao-para-aprender-em-2021/>

# **Quais as linguagens mais populares para os químicos?**

## **1. Fortran e C/C++**

- Para modelagem molecular, química computacional e simulações moleculares
- Para computação de alta performance

## **2. Python, R**

- Para aplicações de propósito geral, incluindo applets
- Bancos de dados

## **3. Tcl/tk, Perl e Bash Script**

- Linguagens para montar scripts
- Automatização

## **4. HTML e Java**

- Aplicações web e applets

# Livros



## **"Teaching Programming Across the Chemistry Curriculum:" An ACS Symposium Series Book**

SEPTEMBER 25, 2020    2799 VIEWS

Prof. Ashley Ringer McDonald, Associate Professor in the Dept. of Chemistry and Biochemistry at Cal Poly and a MolSSI Associate, and Dr. Jessica Nash, MolSSI's Education Lead, are accepting contributions for an [ACS Symposium Series](#) book entitled "Teaching Programming Across the Chemistry Curriculum," to be published in 2021.

[https://molssi.org/2020/09/25/acs\\_symposium\\_series\\_book/](https://molssi.org/2020/09/25/acs_symposium_series_book/)  
<https://pubs.acs.org/isbn/9780841298194>

# **Assuntos para dominar usando linguagens de programação?**

1. Operações algébricas e de álgebra linear com Vetores e matrizes
2. Definição de Funções matemáticas
3. Derivadas, Limites, e Integrais de funções
4. Gráficos de funções matemáticas e de dados
5. Análise de regressão linear
6. Leitura e escrita de dados externos
7. Análises estatísticas
8. Otimização
9. Banco de dados

# Python

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

The screenshot shows the official Python website at <https://www.python.org/>. The page features a dark blue header with the Python logo and navigation links for Python, PSF, Docs, PyPI, Jobs, and Community. Below the header is a search bar with a magnifying glass icon and a "GO" button. A "Donate" button is also visible. The main content area includes a code snippet demonstrating a for loop:

```
# For loop on a list
>>> numbers = [2, 4, 6, 8]
>>> product = 1
>>> for number in numbers:
...     product = product * number
...
>>> print('The product is:', product)
The product is: 384
```

On the right, there's a section titled "All the Flow You'd Expect" which discusses Python's control flow statements. It includes a link to "More control flow tools in Python 3". At the bottom of the main content area are five numbered buttons (1, 2, 3, 4, 5). The footer contains a brief description of Python and a "Learn More" link.

Python is a programming language that lets you work quickly  
and integrate systems more effectively. [» Learn More](#)

**Nós vamos usar Python 3**

# Python

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

Python Brasil   [Impressionese](#) ▾   [Inicie-se](#) ▾   [Aprenda mais](#) ▾   [Participe](#) ▾



**A comunidade Python Brasil reune grupos de usuários em todo o Brasil interessados em difundir e divulgar a linguagem de programação.**

**Impressionese »**  
Descubra como Python está presente em seu dia-a-dia.

**Inicie-se »**  
Veja como é fácil começar a usar a linguagem.

**Aprenda mais »**  
Conheça mais sobre a linguagem e torne-se um verdadeiro pythonista.

**Participe »**  
Encontre e participe da comunidade e compartilhe suas dúvidas e idéias.

# Python – IDEs e ambientes de programação

<https://www.spyder-ide.org/>

The screenshot shows the Spyder IDE interface. The top menu bar includes Arquivo, Editar, Pesquisar Código, Executar, Depurar, Consoles, Projetos, Ferramentas, Ver, and Ajuda. The toolbar contains icons for file operations like Open, Save, and Run. The main window has tabs for 'temp.py' and 'pressao.py\*'. The code editor displays a script named 'temp.py' with the following content:

```
# -*- coding: utf-8 -*-
"""
Created on Wed Sep 23 11:12:33 2020

Esse programa calcula a pressão de um gás usando a lei dos gases ideais, dados:
    - quantidade de matéria,
    - temperatura e
    - volume.

@author: RochaGB
"""

n = float(input("Insira a quantidade de matéria: "))
print("Quantidade de matéria é igual a:", n)

V = float(input("Insira o volume do recipiente em litros: "))
print("Volume do recipiente em litros é igual a:", V)

T = float(input("Insira a temperatura da amostra em Kelvin: "))
print("Temperatura em Kelvin do gás é igual a:", T)

R = 0.082
p = n*R*T/V

print()

print("A pressão calculada, em atm, é igual a:", p)
```

A yellow warning icon is present at the top left of the code editor area. A tooltip box titled 'Uso' provides information about using the help feature in Spyder. The bottom right corner shows the Spyder status bar with 'Console 1/A >', 'Explorador de variáveis', 'Ajuda' (selected), 'Análise de código', 'Figuras', 'Arquivos', and system metrics like CPU usage.

The console output window shows the execution of the script:

```
Quantidade de matéria é igual a: 2.0
Insira o volume do recipiente em litros: 1.0
Volume do recipiente em litros é igual a: 1.0
Insira a temperatura da amostra em Kelvin: 300.0
Temperatura em Kelvin do gás é igual a: 300.0
A pressão calculada, em atm, é igual a: 49.2
```

The status bar at the bottom indicates 'LSP Python: pronto', 'conda: base (Python 3.7.1)', 'Line 25, Col 1', 'UTF-8', 'CRLF', 'RW', 'Mem 56%', 'CPU 20%', and the time '11:20'.

# Python – IDEs e ambientes de programação

<https://jupyter.org/>



# Python – IDEs e ambientes de programação

<https://notepad-plus-plus.org/>

The screenshot shows the Notepad++ interface with a Python file named 'pressao.py' open. The code calculates pressure using the ideal gas law. The Notepad++ window includes a toolbar, menu bar, status bar at the bottom, and a Task List window on the right.

```
1 # -*- coding: utf-8 -*-
"""
Created on Wed Sep 23 11:12:33 2020

Esse programa calcula a pressão de um gás usando a lei dos gases ideais, dados:
    quantidade de matéria, temperatura e volume.
@author: RochaGB
"""

n = float(input("Insira a quantidade de matéria: "))
print("Quantidade de matéria é igual a:", n)
V = float(input("Insira o volume do recipiente em litros: "))
print("Volume do recipiente em litros é igual a:", V)
T = float(input("Insira a temperatura da amostra em Kelvin: "))
print("Temperatura em Kelvin do gás é igual a:", T)
R = 0.082
p = n*R*T/V
print()
print(" A pressão calculada, em atm, é igual a:", p)
```

Python file length:677 lines:29 Ln:29 Col:1 Sel:0|0 Windows (CR LF) UTF-8 INS

# Python – IDEs e ambientes de programação

<https://colab.research.google.com/notebooks/intro.ipynb>

The screenshot shows the Google Colaboratory interface. At the top, there's a browser-like header with tabs and icons. Below it, the main workspace has a sidebar on the left containing a navigation tree with sections like 'Índice', 'Primeiros passos', 'Ciência de dados', 'Machine learning', 'Mais recursos', 'Exemplos de machine learning', and '+ Seção'. The main area displays a notebook cell with the title 'O que é o Colaboratory?' and a list of features. It also includes a message about being a student, scientist, or AI researcher, and a code cell example.

Olá, este é o Colaboratory

Arquivo Editar Ver Inserir Ambiente de execução Ferramentas Ajuda

Compartilhar

Índice X + Código + Texto Copiar para o Drive Conectar | Editar

O que é o Colaboratory?

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

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

Você pode ser um **estudante**, um **cientista de dados** ou um **pesquisador de IA**, o Colab pode facilitar seu trabalho. Assista ao vídeo [Introdução ao Colab](#) para saber mais ou simplesmente comece a usá-lo abaixo!

▼ Primeiros passos

O documento que você está lendo não é uma página da Web estática, mas sim um ambiente interativo chamado **notebook Colab** que permite escrever e executar código.

Por exemplo, aqui está uma **célula de código** com um breve script Python que calcula um valor, armazena-o em uma variável e imprime o resultado:

```
[ ] seconds_in_a_day = 24 * 60 * 60
```

# Python – IDEs e ambientes de programação

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

The screenshot shows the Google Colab interface. At the top, there's a toolbar with various icons for file operations, search, and help. Below the toolbar, the title bar shows the URL and the name of the notebook: "Untitled1.ipynb". The menu bar includes "Arquivo", "Editar", "Ver", "Inserir", "Ambiente de execução", "Ferramentas", "Ajuda", and a message stating "Todas as alterações foram salvas". On the left, there's a sidebar with icons for code, text, and other notebook components. The main area contains a code editor with the following Python script:

```
print("Volume do recipiente em litros é igual a:", V)
T = float(input("Insira a temperatura da amostra em Kelvin: "))

print("Temperatura em Kelvin do gás é igual a:", T)

R = 0.082

p = n*R*T/V

print()

print(" A pressão calculada, em atm, é igual a:", p)
```

Below the code editor, the output window displays the results of running the script:

```
Insira a quantidade de matéria: 1.0
Quantidade de matéria é igual a: 1.0
Insira o volume do recipiente em litros: 2.0
Volume do recipiente em litros é igual a: 2.0
Insira a temperatura da amostra em Kelvin: 350
Temperatura em Kelvin do gás é igual a: 350.0

A pressão calculada, em atm, é igual a: 14.350000000000001
```

A vantagem desse é que você não precisa instalar o Python no seu computador, basta ter apenas uma conexão de internet.

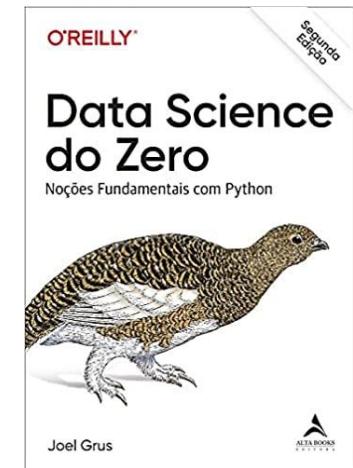
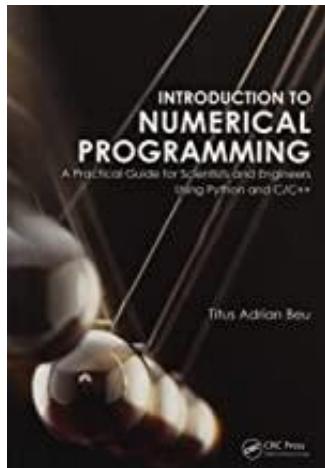
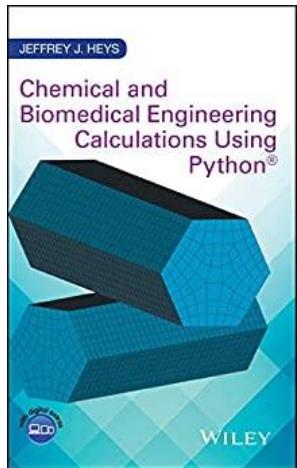
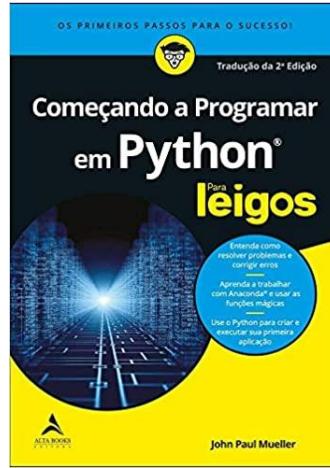
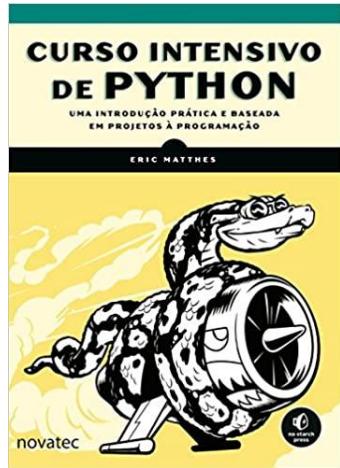
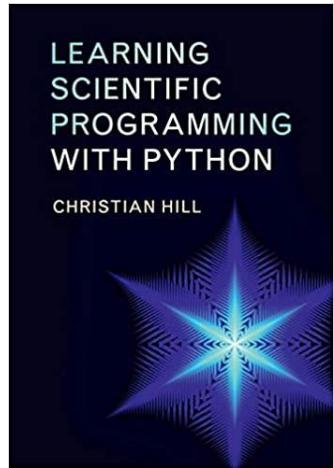
# Python – Tutoriais, Livros, Videos e links úteis

- <https://docs.python.org/pt-br/3/tutorial/>
- <https://wiki.python.org.br/AprendaProgramar>
- <https://www.youtube.com/watch?v=S9uPNppGsGo>
- <https://www.youtube.com/playlist?list=PLlrxD0HtieHhS8VzuMCfQD4uJ9yne1mE6>
- [https://python-guide-pt-br.readthedocs.io/pt\\_BR/latest/intro/learning.html](https://python-guide-pt-br.readthedocs.io/pt_BR/latest/intro/learning.html)
- <https://codigosimples.net/2016/02/12/1000-livros-gratuitos-sobre-programacao/>
- <https://panda.ime.usp.br/panda/default/index>
- <https://computerworld.com.br/2019/07/09/programacao-conheca-5-cursos-online-gratuitos-sobre-python/>
- <https://pythoninchemistry.org/>
- <https://wiki.python.org.br/PythonBrasil>

# Repositórios de bibliotecas de Python

1. <https://pypi.org/>
2. <https://github.com/lmmentel/awesome-python-chemistry>
3. <http://lukaszmentel.com/blog/awesome-python-chemistry/index.html>
4. <https://vegibit.com/24-popular-python-repositories/>
5. <https://towardsdatascience.com/top-11-github-repositories-to-learn-python-e75e8676757a>
6. <https://github.com/vinta/awesome-python>
7. <https://github.com/pythonbrasil>

# Livros



# Livros



Link: [l1nq.com/4Gcxx](https://l1nq.com/4Gcxx)

<https://github.com/pythonchembook>

# **Usando Python**

**Agora vamos para a parte prática  
do minicurso**

O material do curso está no link público:

[https://github.com/RochaGerd/Chemistry\\_with\\_Python](https://github.com/RochaGerd/Chemistry_with_Python)

# Matlab

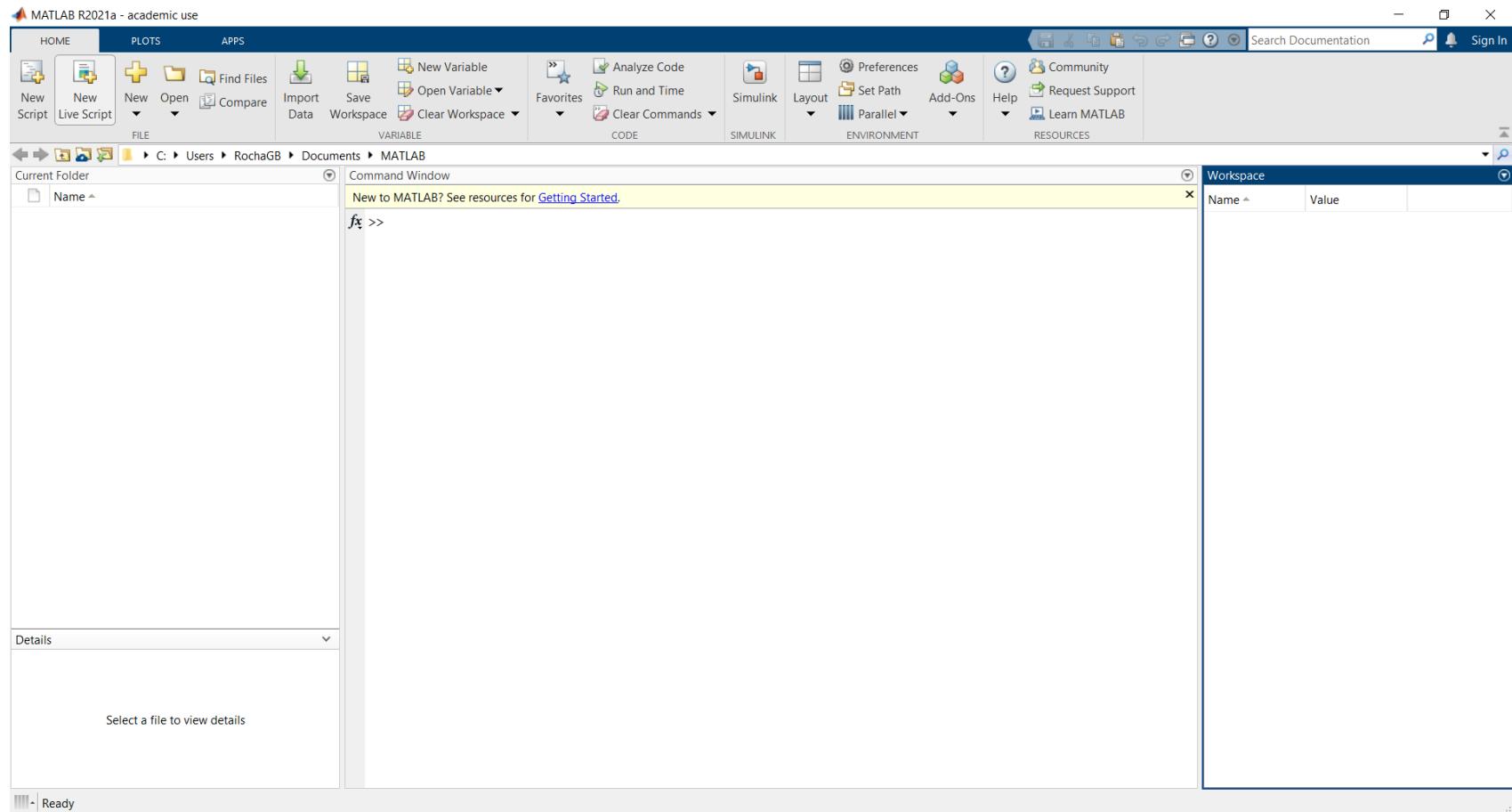
<https://www.mathworks.com/products/matlab.html>

The screenshot shows the official MATLAB website. At the top, there's a navigation bar with the MathWorks logo, a search bar, and links for Products, Solutions, Academia, Support, Community, and Events. Below the header, a large banner features a scientist in profile and the text "Math. Graphics. Programming." followed by a description of MATLAB as a platform for engineers and scientists. A prominent orange button says "Download a free trial". To the right, a video player window is open, showing a MATLAB script for image classification, a confusion matrix plot, and a sample image of a cell. The video player has a timestamp of "1:37".

## A comunidade da UFPB possui a licença do Matlab

- <https://www.ufpb.br/ufpb/contents/noticias/ufpb-disponibiliza-a-alunos-e-professores-software-de-simulacao-e-calculo-numerico>
- <https://www.mathworks.com/academia/tah-portal/ufpb-universidade-federal-da-paraiba-31545846.html>

# Matlab



<https://www.mathworks.com/products/matlab-online.html#license-types>

# Projeto R

<https://cran.r-project.org/>

The screenshot shows a Microsoft Edge browser window with the URL [cran.r-project.org](https://cran.r-project.org/) in the address bar. The page content is the "The Comprehensive R Archive Network" (CRAN) homepage.

**Left sidebar:**

- CRAN**
- [Mirrors](#)
- [What's new?](#)
- [Task Views](#)
- [Search](#)
- About R**
- [R Homepage](#)
- [The R Journal](#)
- Software**
- [R Sources](#)
- [R Binaries](#)
- [Packages](#)
- [Other](#)
- Documentation**
- [Manuals](#)
- [FAQs](#)
- [Contributed](#)

**Main Content Area:**

## The Comprehensive R Archive Network

### Download and Install R

Precompiled binary distributions of the base system and contributed packages. **Windows and Mac** users most likely want one of these versions of R:

- [Download R for Linux](#)
- [Download R for \(Mac\) OS X](#)
- [Download R for Windows](#)

R is part of many Linux distributions, you should check with your Linux package management system in addition to the link above.

### Source Code for all Platforms

Windows and Mac users most likely want to download the precompiled binaries listed in the upper box, not the source code. The sources have to be compiled before you can use them. If you do not know what this means, you probably do not want to do it!

- The latest release (2020-06-06, See Things Now) [R-4.0.1.tar.gz](#), read [what's new](#) in the latest version.
- Sources of [R alpha and beta releases](#) (daily snapshots, created only in time periods before a planned release).
- Daily snapshots of current patched and development versions are [available here](#). Please read about [new features and bug fixes](#) before filing corresponding feature requests or bug reports.
- Source code of older versions of R is [available here](#).
- Contributed extension [packages](#)

### Questions About R

- If you have questions about R like how to download and install the software, or what the license terms are, please read our [answers to frequently asked questions](#) before you send an email.

### What are R and CRAN?

R is 'GNU S', a freely available language and environment for statistical computing and graphics which provides a wide variety of statistical and graphical techniques: linear and

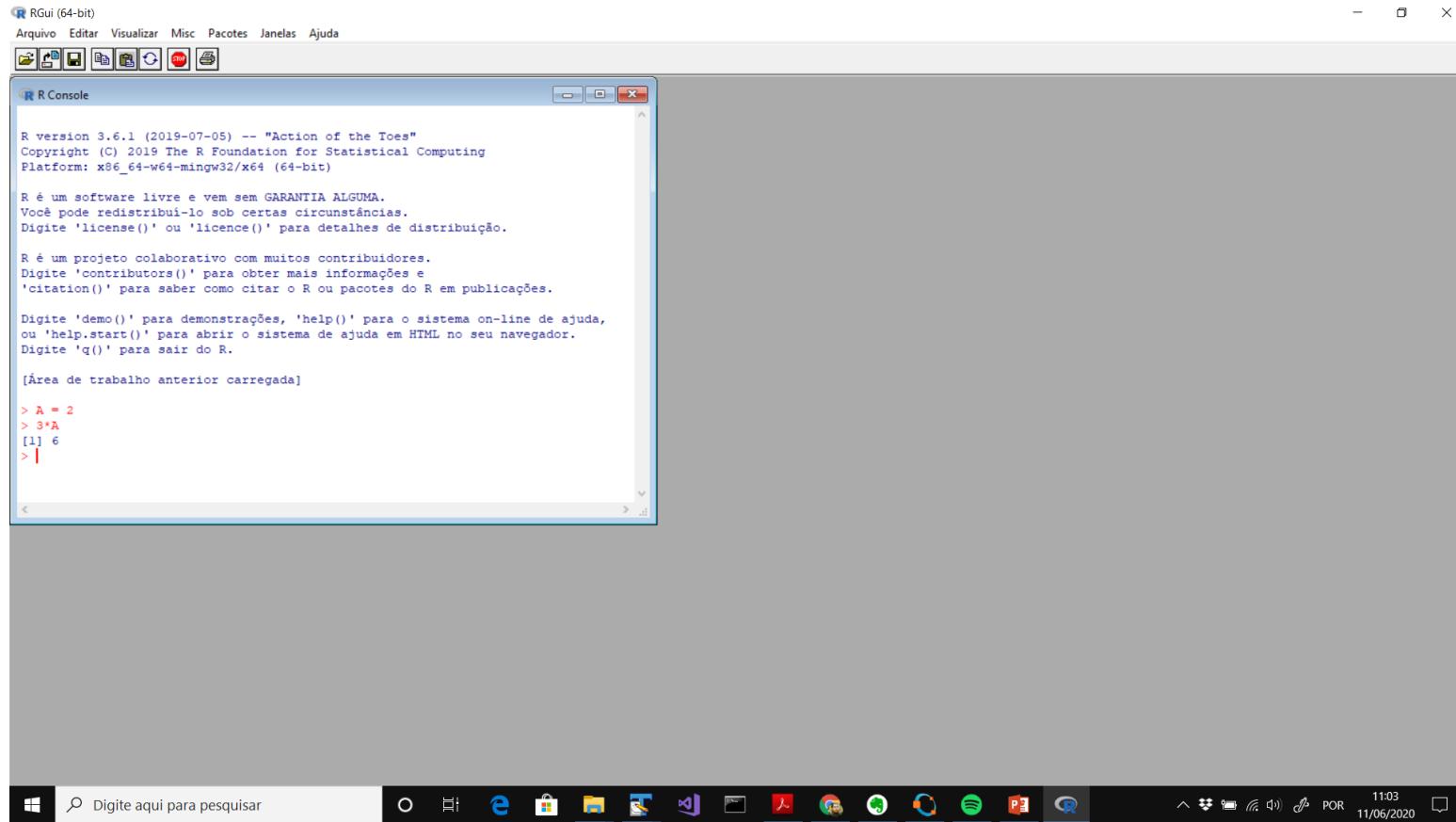
general-2.1.1.tar.gz

Digitate aqui para pesquisar

11:05  
11/06/2020

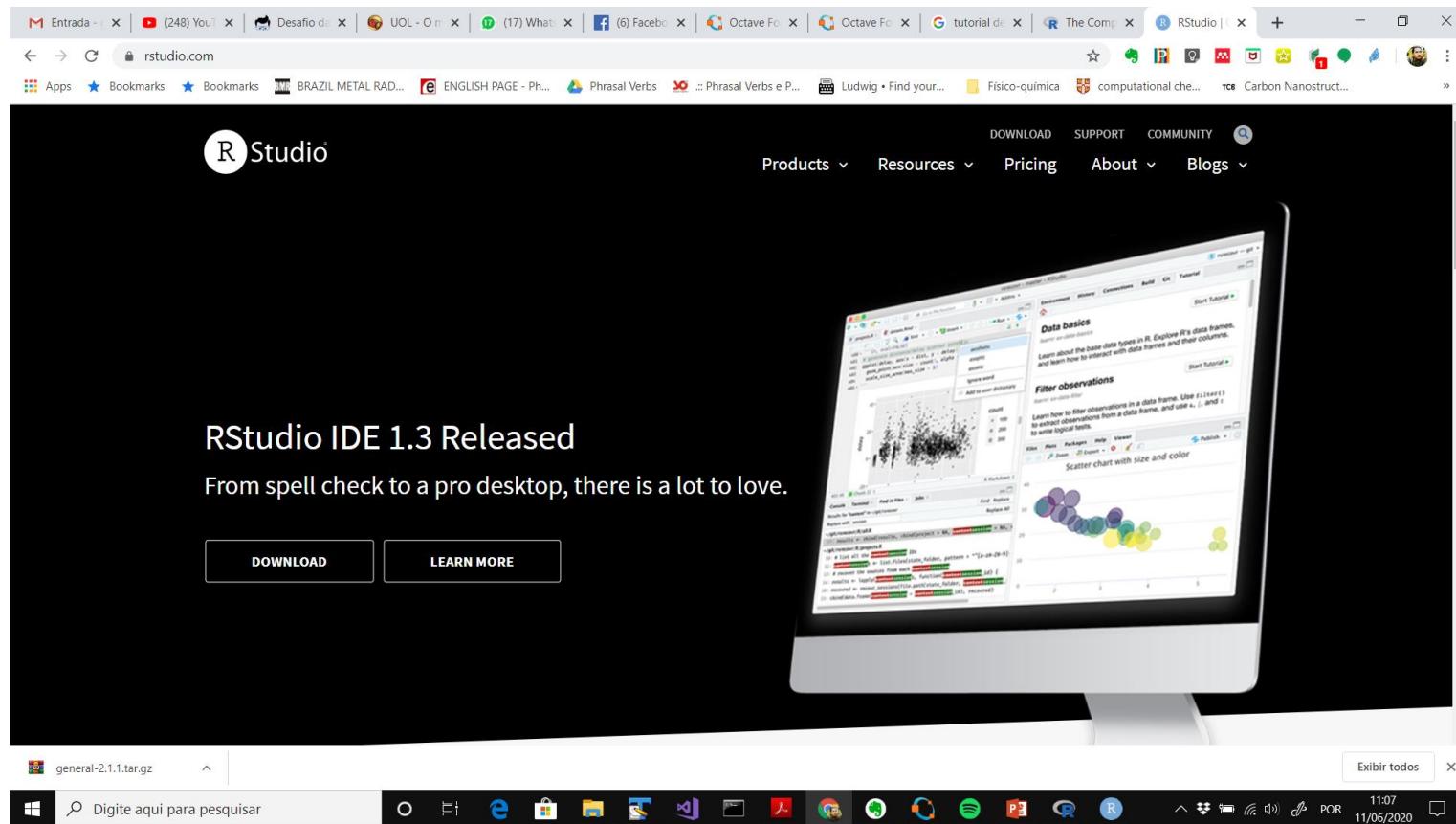
# Projeto R

<https://cran.r-project.org/>



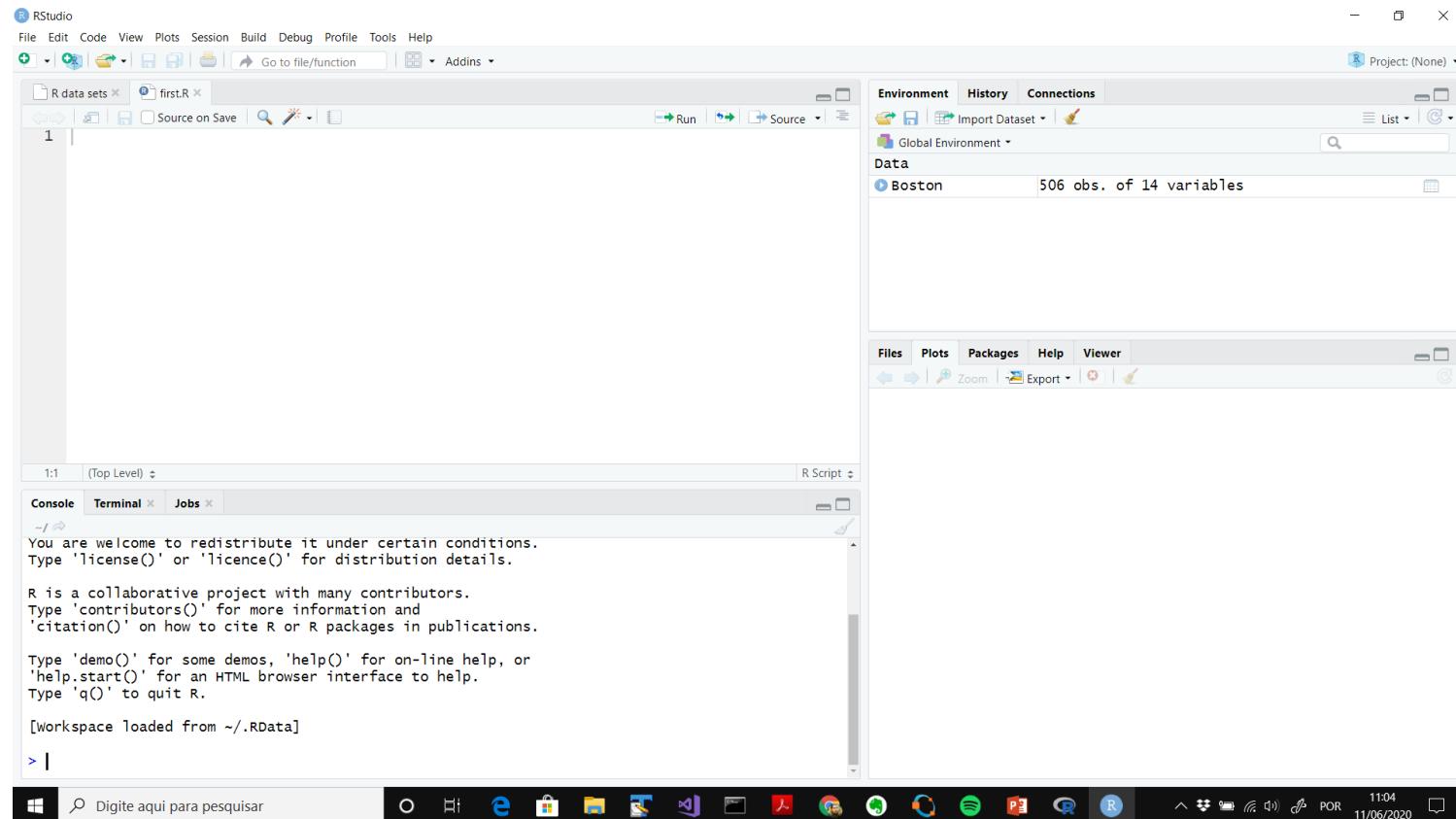
# RStudio

<https://rstudio.com/>



# RStudio

<https://rstudio.com/>



# Projeto R – Tutoriais, Livros, Videos e links úteis

- <http://ecologia.ib.usp.br/bie5782/doku.php>
- <https://www.listendata.com/p/r-programming-tutorials.html>
- <https://producaoanimalcomr.wordpress.com/2015/12/15/livros-r-gratuitos-para-downloads/>
- <https://www.youtube.com/watch?v=tfPsmDzS74c&list=PLyqOvdQmGdTQ5dE6hSD7ZGBu8bud70wYf>
- <https://www.youtube.com/watch?v=C6tslbHNHr0>
- [https://www.youtube.com/watch?v=8HQHf5XCS7g&list=PLuQWWXrHQLiejECp\\_IdRPEjuVnJrG7fVS](https://www.youtube.com/watch?v=8HQHf5XCS7g&list=PLuQWWXrHQLiejECp_IdRPEjuVnJrG7fVS)
- <https://rdrr.io/snippets/> e <https://replit.com/languages/rlang>

# Projeto R – Tutoriais, Livros, Videos e links úteis

**Using R**

**Índice**

**O Curso**

- Introdução
- Curso IBUSP
- Curso UFABC
- Atividades Preparatórias
- Os 10 Mandamentos do R
- Mini Curso
- Tópicos Especiais

**Material de Apoio**

- Apostila on line
- Slide das Aulas
- Códigos de Aula
- Tutoriais
- Exercícios
- Arquivos de Dados
- Outros

**Área dos Alunos**

- Wiki Alunos
- Trabalho Final
- Postar Exercícios
- Avaliação
- Prazos de Postagem

**Visitantes**

**Apresentação**

“Uma das coisas mais importantes que você pode fazer é dedicar um tempo para aprender uma linguagem de programação de verdade. Aprender a programar é como aprender outro idioma: exige tempo e treinamento, e não há resultados práticos imediatos. Mas se você supera essa primeira subida íngreme da curva de aprendizado, os ganhos como cientista são enormes. Programar não vai apenas livrar você da camisa de força dos pacotes estatísticos, mas também irá aguçar suas habilidades analíticas e ampliar os horizontes de modelagem ecológica e estatística.”

*Tradução um tanto livre de Gotelli & Ellison, 2004. A Primer of Ecological Statistics. Sunderland, Sinauer.*

“A chave para entender o R é que trata-se de uma linguagem. Uma linguagem para manipular objetos.”

Bill Venables

“A pergunta certa sobre uma análise em R não é se é possível fazê-la, e sim como fazê-la.”

Paulo Justiniano Ribeiro

“Tudo que ensinas a uma criança, impede-a de descobrir.”

Jean Piaget

**Tabela de conteúdos**

- Apresentação
- BEM VINDO(A)
- Objetivos
  - Objetivos Específicos
- Responsáveis
- Conteúdo
- Arquivos de Dados
  - Matrícula

Entrar

Alterações recentes Índice

start

Digitando aqui para pesquisar

# Projeto R – Tutoriais, Livros, Videos e links úteis



Using R

Entrar



Alterações recentes Índice

Índice



O Curso

- Introdução
- Curso IBUSP
- Curso UFABC
- Atividades Preparatórias

Área Restrita

Área Restrita

Cursos Anteriores

- Alunos 2018
- Alunos 2017
- Alunos 2016
- Alunos 2015
- Alunos 2014
- Alunos 2013
- Alunos 2012
- Alunos 2011
- Alunos 2010
- Alunos 2009
- Curso Manaus 2012
- Curso Natal 2013

Linques



Laboratório de Ecologia de Florestas Tropicais



CENTRO DE MÉTODOS QUANTITATIVOS - CMQ



## Apostila on line

### Apresentação

Esta apostila foi desenvolvida para apoio do curso [Uso da Linguagem R em Ecologia](#). Depois de muitos ciclos de aperfeiçoamento na internet como um material para estudo individual.

Ela está em constante aperfeiçoamento, e ficaremos felizes com suas críticas e sugestões. Entre em contato com os editores. Veja também a nossa seção de [tutoriais](#).

### Conteúdo

- [1a. Introdução](#): histórico e filosofia de trabalho do R, breve histórico da linguagem S e do R, exemplo de uma sessão de trabalho no R.
- [1b. Primeiros passos](#): interação com programa, sintaxe e comandos, como obter ajuda.
- [2. Funções no R e sua Aplicação](#): funções matemáticas, funções estatísticas e operações lógicas.
- [3. Leitura e Manipulação de Dados](#): tipos de variáveis e de estrutura dos dados, entrada e importação de dados, transformações de dados, operações vetoriais e matriciais, indexação.
- [4. Análise Exploratória de Dados](#): estatísticas descritivas, gráficos exploratórios.
- [5a. Criando e Editando Gráficos](#): criação e modificação de gráficos finais para apresentação e publicação.
- [5b. Gráficos Avançado](#): procedimentos simples para a construção de gráficos mais elaborados.
- [6. Teste de significância - em construção!](#)
- [7. Modelos Lineares](#): lógica geral em R, regressão linear, análise de variância, verificação das pressuposições dos modelos lineares.
- [8. Reamostragem e Simulação](#): funções ligadas às distribuições de probabilidade, simulação de distribuições, reamostragem de dados.
- [9. Noções de Programação](#): fundamentos de programação orientada a objetos; lógica e controle de fluxo em linguagem S; procedimentos vetoriais de programação.

### Citação

Batista, J.L.F., Prado, P.I. e Oliveira, A. A. (Eds.) 2009. Introdução ao R - Uma Apostila on-line. URL: <http://ecologia.ib.usp.br/bie5782>.

bie5782:03\_apostila:start

Tabela de conteúdos

- Apostila on line
- Apresentação
- Autores
  - Editores
  - Colaboradores
- Conteúdo
- Citação



TeX



TeX

# Linguagem R

## Operações algébricas e de álgebra linear com Vetores e matrizes

```
#  
# passos iniciais no R  
#  
a <- 1 # Armazena o valor 1 na variável a  
b <- a # Atribui o valor da variável a para a variável b  
print(a) # Imprime a variável a  
[1] 1  
print(b) # Imprime a variável b  
[1] 1  
a + 4 # Soma o valor 4 a variável a  
[1] 5  
a*5 # Multiplica a variável a por 5  
[1] 5
```

# Linguagem R

## Operações algébricas e de álgebra linear com Vetores e matrizes

```
#  
# manipulação de vetores e matrizes  
#  
A1 <- c(1,2,3) # Cria o vetor A1  
print(A1) # Imprime o vetor A1  
[1] 1 2 3  
A2 <- c(10,20,30) # Cria o vetor A2  
print(A2) # Imprime o vetor A2  
[1] 10 20 30  
b <- c(A1,A2) # Cria um vetor b com a junção dos vetores A1 e  
A2  
print(b) # Imprime o vetor b  
[1] 1 2 3 10 20 30  
p = c(1, 10, 3.4, pi, pi/4, exp(-1), log( 2.23 ), sin(pi/7) ) #  
Cria o vetor p  
print(p) # Imprime o vetor p  
[1] 1.0000000 10.0000000 3.4000000 3.1415927 0.7853982  
0.3678794 0.8020016 0.4338837  
p[2:4] # Mostra os elementos nas posições de 2 até 4  
[1] 10.000000 3.400000 3.141593
```

# Linguagem R

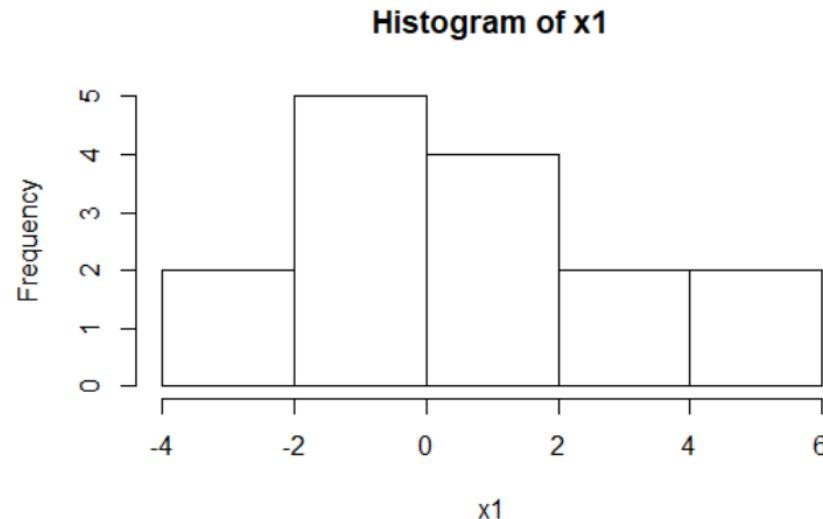
## Operações algébricas e de álgebra linear com Vetores e matrizes

```
b1 = 1:8 # Cria um vetor b1 de uma sequência iniciando em 1 e terminando em 8
print (b1) # Imprime o vetor b1
[1] 1 2 3 4 5 6 7 8
b2 = seq(10, 30, by=2) # Cria um vetor b2 de uma sequência iniciando em 10 e terminando em 30 em passos 2
print (b2) # Imprime o vetor b2
[1] 10 12 14 16 18 20 22 24 26 28 30
minha.matriz <- matrix(data=1:12,nrow=3,ncol=4) # Cria uma matriz 3 x 4
print(minha.matriz) # Imprime a matriz minha.matriz
[,1] [,2] [,3] [,4]
[1,] 1 4 7 10
[2,] 2 5 8 11
[3,] 3 6 9 12
elementos <- matrix(c(1,2,3,4,5,6,7,8,9),ncol=3,nrow=3) # Com os elementos do vetor, cria a matriz elementos
print(elementos) # Imprime os dados na matriz elementos
[,1] [,2] [,3]
[1,] 1 4 7
[2,] 2 5 8
[3,] 3 6 9
```

# Linguagem R

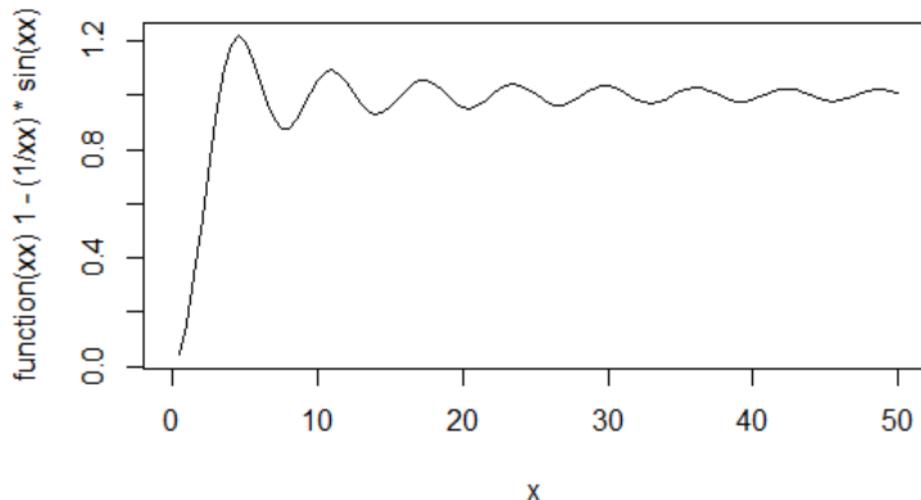
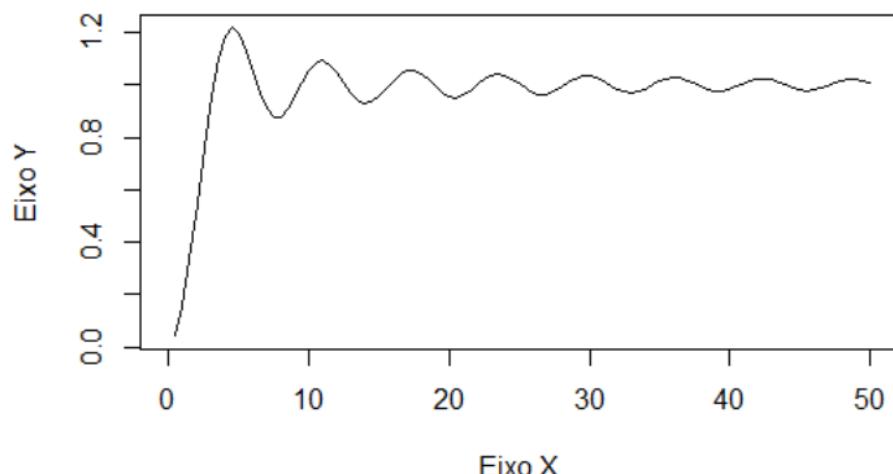
## Definição de Funções matemáticas, gráficos e integração numérica

```
#  
# Gráfico, definição de função e integrais numéricas  
# http://leg.ufpr.br/~paulojus/CE223/ce223/ce223se5.html  
#  
x1 <- rnorm(n=15, mean=1, sd=3) # Cria um vetor x1 de números aleatórios com  
média 1 e desvio padrão 3  
print(x1) # Imprime o vetor x1  
[1] -0.2732125 -1.7278447 4.4125847 1.9617774 -2.5610144 1.2279162 -2.7630620 -  
1.1746744 [9] -1.3545851 2.7376578 5.5775200 3.5844448 -1.8119393 0.8081351  
1.3512801  
hist(x1) # Plota o gráfico de Histograma com os valores de x1
```



# Linguagem R

```
xx1 <- seq(0, 50, 1 = 101) # Gera um vetor xx1
print(xx1)
[1] 0.0 0.5 1.0 1.5 2.0 2.5 3.0 3.5 4.0 4.5 5.0 5.5 6.0 6.5 7.0 7.5 8.0 [18] 8.5
9.0 9.5 10.0 10.5 11.0 11.5 12.0 12.5 13.0 13.5 14.0 14.5 15.0 15.5 16.0 16.5
[35] 17.0 17.5 18.0 18.5 19.0 19.5 20.0 20.5 21.0 21.5 22.0 22.5 23.0 23.5 24.0
24.5 25.0 [52] 25.5 26.0 26.5 27.0 27.5 28.0 28.5 29.0 29.5 30.0 30.5 31.0 31.5
32.0 32.5 33.0 33.5 [69] 34.0 34.5 35.0 35.5 36.0 36.5 37.0 37.5 38.0 38.5 39.0
39.5 40.0 40.5 41.0 41.5 42.0 [86] 42.5 43.0 43.5 44.0 44.5 45.0 45.5 46.0 46.5
47.0 47.5 48.0 48.5 49.0 49.5 50.0
yy1 <- 1 - (1/xx1) * sin(xx1) # Gera um vertor com a fórmula matemática
plot(xx1, yy1, xlab="Eixo X", ylab="Eixo Y", type = "l") # Faz o gráfico da
função matemática
plot(function(xx) 1 - (1/xx) * sin(xx), 0, 50) # Forma alternativa de fazer
gráfico de funções matemáticas
```



# Linguagem R

## Definição de Funções matemáticas, gráficos e integração numérica

```
fx <- function(x) x^2 # Define uma função de x, fx
integrate(fx, -3, 3) # Calcula a integral da função fx
18 with absolute error < 2e-13
fx <- function(x) { +
+ (1/(9 * sqrt(2 * pi))) * exp(-(1/162) * (x - 100)^2) }
integrate(fx, 85, 105)
0.6629523 with absolute error < 7.4e-15
```

# Linguagem R

## Análise de regressão linear

```
#  
# Ajustes de curvas a dados  
#  
area <- c(300, 350, 961, 295, 332, 47, 122, 11, 53, 2749) # Cria o  
vetor area  
print(area) # imprime o vetor area  
[1] 300 350 961 295 332 47 122 11 53 2749  
summary(area) # Calcula média, desvio padrão e outras quantidades  
estatísticas do vetor area  
Min. 1st Qu. Median Mean 3rd Qu. Max.  
11.00 70.25 297.50 522.00 345.50 2749.00  
riqueza <- c(1, 7, 20, 7, 8, 4, 8, 3, 5, 23) # Cria o vetor riqueza  
print(riqueza) # imprime o vetor riqueza  
[1] 1 7 20 7 8 4 8 3 5 23  
modelo1 <- lm(riqueza~area) # Monta um modelo linear de mínimos  
quadrados riqueza = A*area + B  
print(modelo1) # Imprime o modelo ajustado  
Call: lm(formula = riqueza ~ area)  
Coefficients: (Intercept) area  
4.676806 0.007516
```

# Linguagem R

## Análise de regressão linear

```
summary(modelo1)
```

Call:

```
lm(formula = riqueza ~ area)
```

Residuals:

```
Min 1Q Median 3Q Max -5.9315 -1.5771 -0.1912 0.6475 8.1006
```

Coefficients:

Estimate	Std. Error	t value	Pr(> t )
----------	------------	---------	----------

```
(Intercept) 4.676806 1.451534 3.222 0.01220 *
```

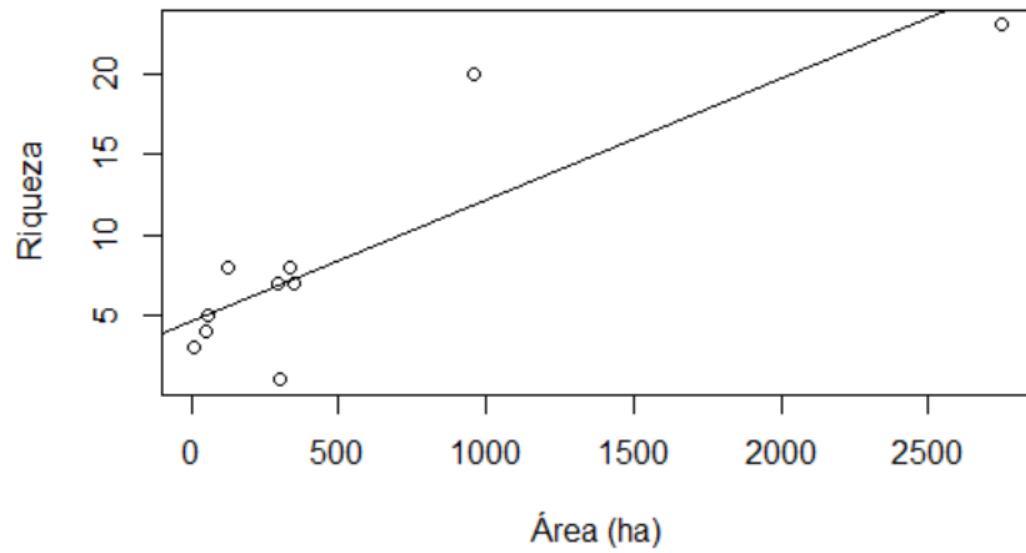
```
area 0.007516 0.001538 4.888 0.00121 **
```

--- Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1 Residual standard error: 3.824 on 8 degrees of freedom

Multiple R-squared: 0.7491, Adjusted R-squared: 0.7178

F-statistic: 23.89 on 1 and 8 DF, p-value: 0.001213

```
plot(area, riqueza, xlab="Área (ha)", ylab="Riqueza") # Gera o gráfico de area vs  
riqueza  
abline(modelo1)
```



# Linguagem R

## Leitura e escrita de dados externos

```
#  
# Leitura e escrita de dados externos  
#  
getwd() # Mostra o caminho de trabalho no seu computador  
[1] "C:/Users/RochaGB/Documents"  
gbmam93 <- read.table(file="gbmam93.txt",header=T,row.names=1,sep=",") # Ler o  
arquivo gbmam93.txt do diretório de trabalho  
print(gbmam93) # imprime os dados lidos externamente  
a b c d e f g h i j k l m n o p q r s  
.....  
#  
# Comandos uteis  
#  
save.image(file= "Todos_comandos.RData") # Salva todos comando digitados  
rm(list=ls()) # Limpa todo conteúdo da memória  
my.d <- 'C:/Minha Pesquisa/'  
setwd(my.d) # Define seu diretório de trabalho
```

<https://octave.sourceforge.io/>

The screenshot shows a Microsoft Edge browser window with multiple tabs open at the top. The active tab is for the Octave Forge homepage at [octave.sourceforge.io](https://octave.sourceforge.io/). The page title is "Octave Forge". Below the title, there's a section titled "Extra packages for GNU Octave" with a bulleted list of features:

- Central location for collaborative development of packages for [Octave](#)
- Expands Octave's core functionality by providing field specific features via Octave's package system
- Image and signal processing, fuzzy logic, instrument control, and statistics... and more!

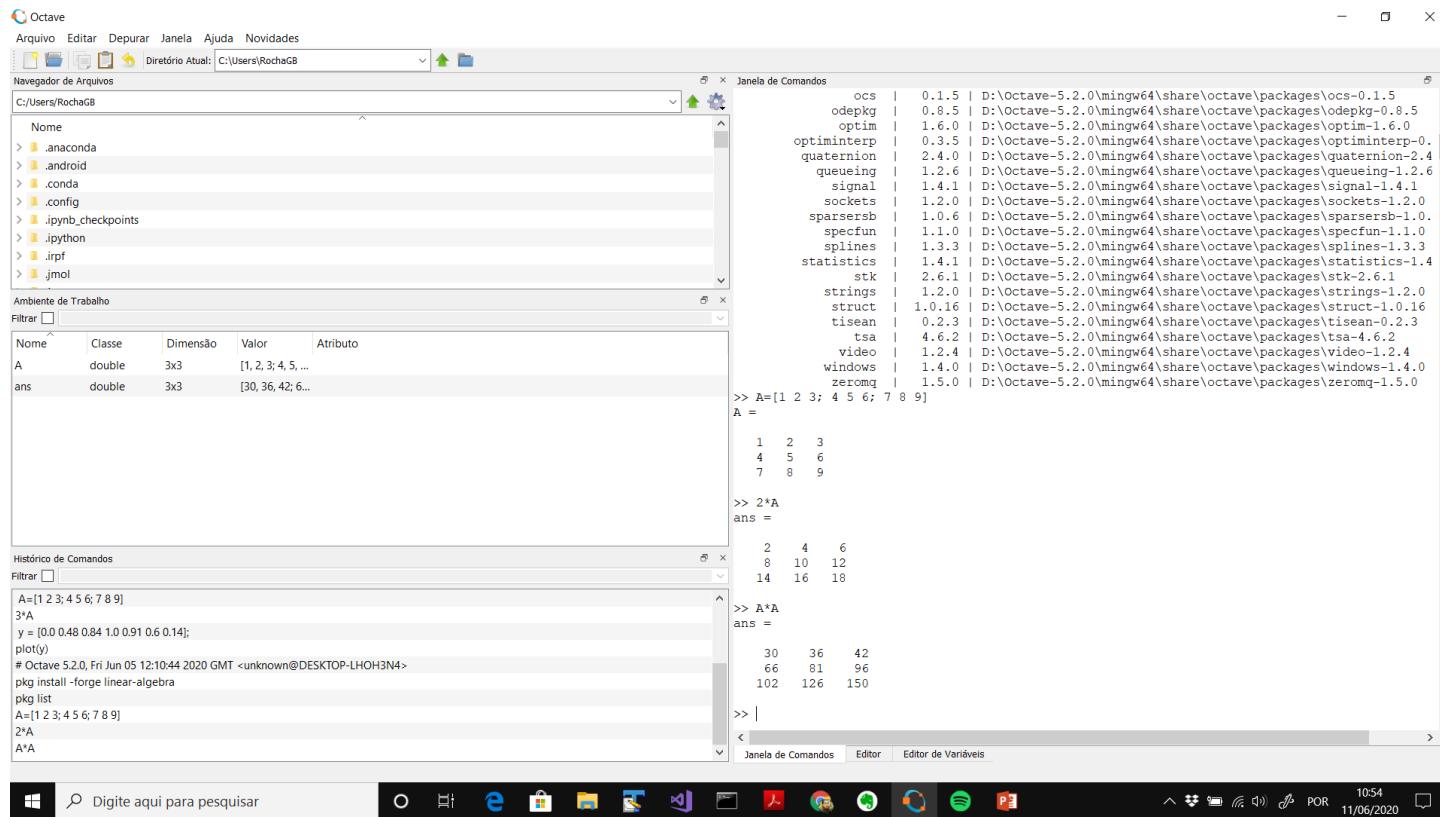
Next to the list is a circular icon composed of six hexagons, each containing a different icon related to scientific computing.

Below this section is a heading "Installation". A note says: "Refer to the [GNU Octave web site](#) for information about how to install GNU Octave itself on your specific OS. Many OS distributions of Octave also provide Octave Forge packages." There are two callout boxes: one for "Display already installed packages" and another for "Individual package installation".

The bottom of the screen shows the Windows taskbar with various pinned icons and the date/time "11/14 11/06/2020".

# Octave

<https://octave.sourceforge.io/>



# Octave – **Tutoriais**, **Livros**, **Videos** e **links úteis**

- <http://www.rodrigofernandez.com.br/ecomp/ref/octave-final.pdf>
- <https://www.casadocodigo.com.br/products/livro-octave>
- <https://www.ufrgs.br/reamat/CalculoNumerico/index.html>
- [http://www2.fct.unesp.br/docentes/carto/galo/\\_Softwares\\_Free/Octave/Tutorial\\_GNU\\_Octave\\_Unesp.pdf](http://www2.fct.unesp.br/docentes/carto/galo/_Softwares_Free/Octave/Tutorial_GNU_Octave_Unesp.pdf)
- [https://www.youtube.com/watch?v=MoKKDVAMjDQ&list=PLUfT\\_zSB\\_TFTFhB87Boy3RTzB0CpcMJ4C](https://www.youtube.com/watch?v=MoKKDVAMjDQ&list=PLUfT_zSB_TFTFhB87Boy3RTzB0CpcMJ4C)
- <https://octave-online.net/>
- [https://octave.sourceforge.io/list\\_functions.php?sort=package](https://octave.sourceforge.io/list_functions.php?sort=package)

# Octave

## Operações algébricas e de álgebra linear com Vetores e matrizes

```
>> A=[1, 2, 3; 4, 5, 6; 7, 8, 9] # Entrando uma matriz e salvando na varivel A
```

```
A =
```

```
1 2 3  
4 5 6  
7 8 9
```

```
>> B = A' # Transposta de A
```

```
B =
```

```
1 4 7  
2 5 8  
3 6 9
```

```
>> A*A # Produto de duas matrizes
```

```
ans =
```

```
30 36 42  
66 81 96  
102 126 150
```

```
>> trace(A)
```

```
ans = 15
```

```
>> [v,x]=eig(A) # Retorna os autovetores em v e os autovalores em x
```

```
v =
```

```
-0.231971 -0.785830 0.408248  
-0.525322 -0.086751 -0.816497  
-0.818673 0.612328 0.408248
```

```
x =
```

```
Diagonal Matrix
```

```
1.6117e+01 0 0  
0 -1.1168e+00 0  
0 0 -1.3037e-15
```

## Operações algébricas e de álgebra linear com Vetores e matrizes

```
>> V = [1 2 3 4 5 6 7 8 9 10]; # Define o vetor V  
  
>> std(V), mean(V), sum(V) # Calcula o desvio padrao, media e soma dos elementos do vetor V  
  
ans = 3.0277  
ans = 5.5000  
ans = 55  
>> C = 1:1:5 # Cria um vetor começando em 1 e terminando em 5 com passo igual a 1  
  
C =  
  
1 2 3 4 5  
  
>> C2 = 1:0.5:5 # Cria um vetor começando em 1 e terminando em 5 com passo igual a 0.5  
  
C2 =  
  
1.0000 1.5000 2.0000 2.5000 3.0000 3.5000 4.0000 4.5000 5.0000  
  
>> C3 = 5:-1:1 # Cria um vetor começando em 5 e terminando em 1 com passo igual a -1  
  
C3 =  
  
5 4 3 2 1  
  
>>
```

# Octave

## Operações algébricas e de álgebra linear com Vetores e matrizes

```
>> k = linspace (0, 1, 6) # Gera um vetor começando em 0 e terminando em 1 de seis elementos regularmente espacados
```

```
k =
```

```
0.00000 0.20000 0.40000 0.60000 0.80000 1.00000
```

```
>> A = rand(3) # Gera uma matriz 3x3 aleatória
```

```
A =
```

```
0.42528 0.38831 0.60530  
0.81902 0.81261 0.17516  
0.21806 0.79886 0.98786
```

```
>> B = A(1:3,2) # Gera uma submatriz da matriz A pegando os elementos da coluna 2.
```

```
B =
```

```
0.38831  
0.81261  
0.79886
```

```
>> D = rand(3)
```

```
D =
```

```
0.095186 0.443786 0.107241  
0.128126 0.425546 0.386941  
0.079726 0.962976 0.306773
```

```
>> E = A + D # Soma as matrizes A e D
```

```
E =
```

```
0.52046 0.83210 0.71254  
0.94715 1.23816 0.56210  
0.29779 1.76183 1.29464
```

# Octave

## Definição de Funções matemáticas e integrais numéricas

```
>> function y=f(x)
```

```
y = x^2;
```

```
endfunction
```

```
>> f(10)
```

```
ans = 100
```

```
>> [v, ier, nfun, err]=quad("f", 0, 3)
```

```
v = 9.0000
```

```
ier = 0
```

```
nfun = 21
```

```
err = 9.9920e-14
```



Definição de uma função



Valor da função em  $x = 10$

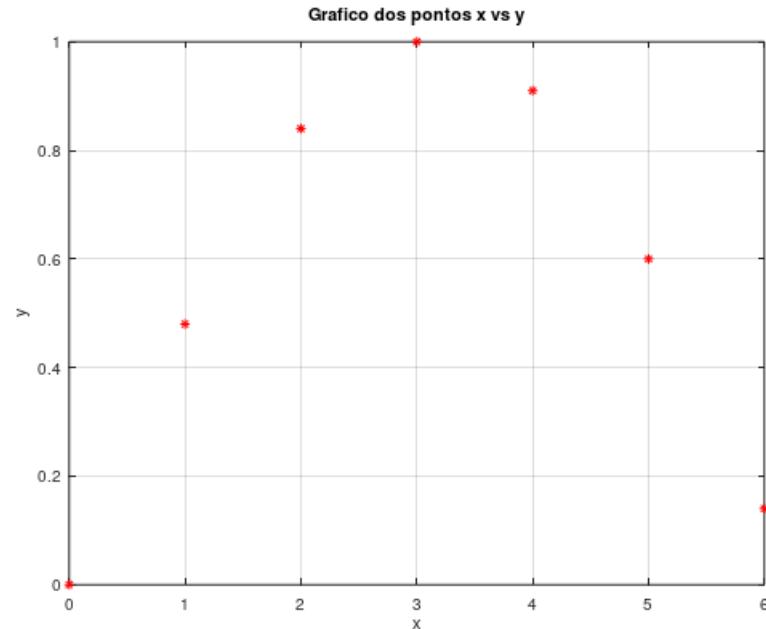


Cálculo da integral

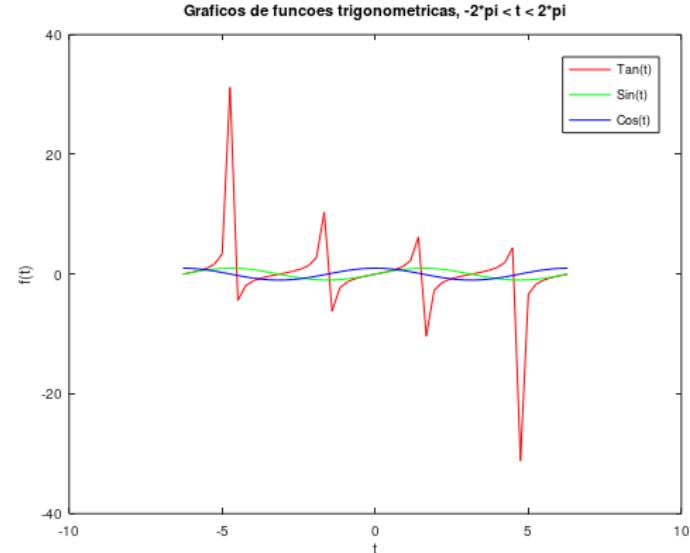
# Octave

## Gráficos de funções matemáticas e de dados

```
>> y = [0.0 0.48 0.84 1.0 0.91 0.6 0.14];  
  
>> x = [0.0 1.00 2.00 3.0 4.00 5.0 6.00];  
  
>> plot(x,y,'*r') # Grafico dos pontos x e y  
>> grid('on');  
>> xlabel 'x';  
>> ylabel 'y';  
>> title 'Grafico dos pontos x vs y';
```



```
>> t = linspace(-2*pi,2*pi,50);  
>> y1=tan(t);  
>> y2=sin(t);  
>> y3=cos(t);  
>> plot(t,y1,"r");  
>> hold on  
>> plot(t,y2,"g");  
>> plot(t,y3,"b");  
>> legend('Tan(t)','Sin(t)','Cos(t)');  
>> xlabel 't';  
>> ylabel 'f(t)';  
>> title 'Graficos de funcoes trigonometricas, -2*pi < t < 2*pi';
```



## Análise de regressão linear

$$\mathbf{b} = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{Y} \quad (\text{vetor dos coeficientes})$$

$$R^2 = 1 - \frac{\sum_{i=1}^n (Y_i^{calc} - Y_i^{exp})^2}{\sum_{i=1}^n (Y_i^{exp} - \bar{Y})^2}$$

Material para consulta sobre o método de ajuste de curvas por mínimos quadrados.

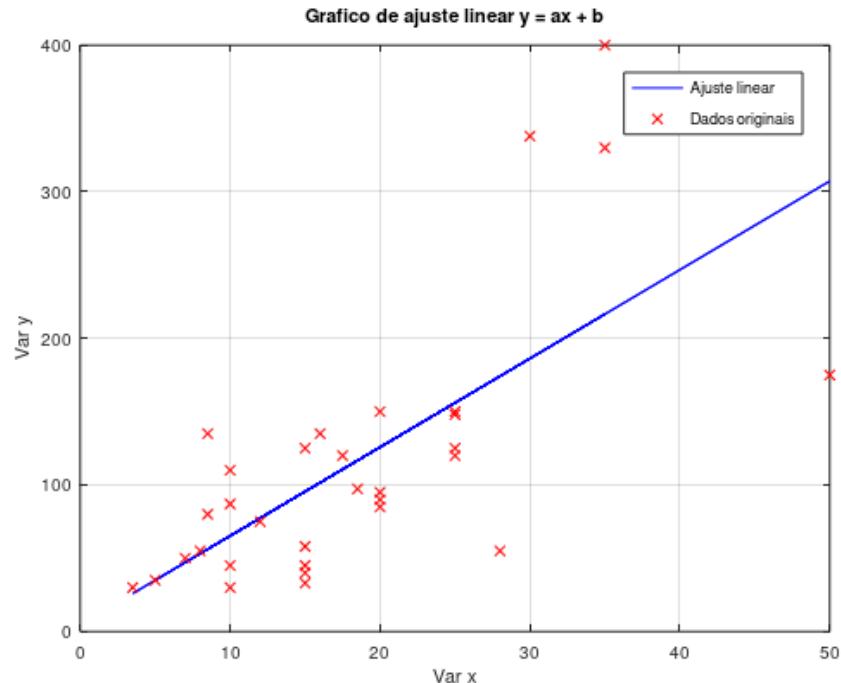
- [http://pt.wikipedia.org/wiki/M%C3%A9todo\\_dos\\_m%C3%ADnimos\\_quadrados](http://pt.wikipedia.org/wiki/M%C3%A9todo_dos_m%C3%ADnimos_quadrados)
- [http://www.mat.ufmg.br/gaal/aplicacoes/quadrados\\_minimos.pdf](http://www.mat.ufmg.br/gaal/aplicacoes/quadrados_minimos.pdf)
- [https://files.cercomp.ufg.br/weby/up/56/o/Q2\\_14-2\\_Aula\\_P05\\_RLM.pdf](https://files.cercomp.ufg.br/weby/up/56/o/Q2_14-2_Aula_P05_RLM.pdf)
- <https://machinelearningbrasil.wordpress.com/2014/12/23/regressao-linear-com-octave/>

# Octave

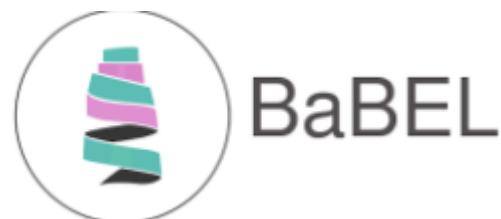
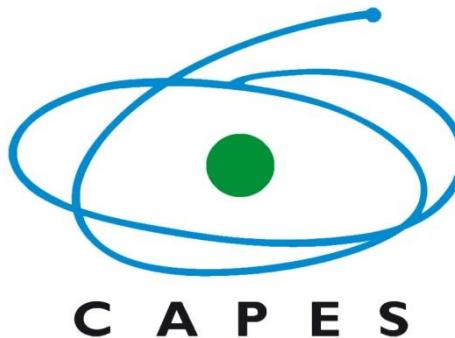
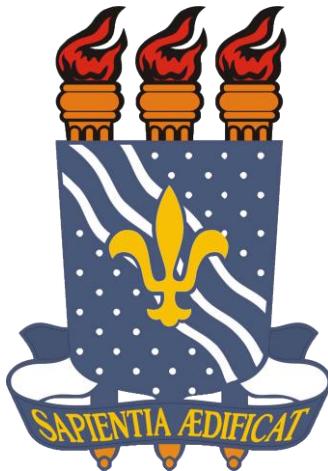
## Análise de regressão linear

```
>> y = load('slr11l2.txt');  
>> x = load('slr11l1.txt');  
          ←  
>> rowX = size(x)(1);  
>> rowY = size(y)(1);  
>> X(1:rowX,1) = 1.0; # Adiciona uns na coluna 1 de X  
>> X(1:rowX,2) = x(1:rowX); # Cria uma matriz onde a primeira coluna eh de uns e a segunda coluna tem os dados em x  
>> w = (inv(X'*X))*X'*y; # Calcula os coeficientes do ajuste linear y = ax + b  
>> yest = X*w;  
>> erro = y - yest; # Calcula o erro do ajuste dos dados  
>> correl = 1.0 - (sum((y - yest).^2)/sum((y-mean(y)).^2)); # Calcula o coeficiente de correlacao do  
ajuste  
>> correl  
correl = 0.46786  
>> plot(x, yest, '-b');hold on; xlabel('Var x'); ylabel('Var y'); plot(x,y,'xr'); # Plota os valores  
de x e yest  
>> legend('Ajuste linear','Dados originais');  
>> grid('on')  
>> title 'Grafico de ajuste linear y = ax + b';
```

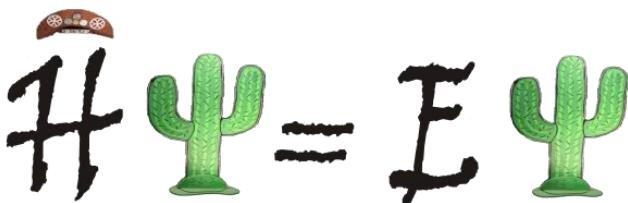
## Leitura de dados externos



# Agradecimentos



<http://babel.dcc.ufmg.br/>



Grupo de Química Computacional mais Oriental das Américas  
I.Q.Q.C. - UFPB - João Pessoa - PB

**Edital Biologia Computacional 51/2013,  
processo AUXPE1375/2014 - CAPES**





# Universidade Federal da Paraíba

## Departamento de Química

Laboratório de Química Quântica Computacional

Prof. Gerd Bruno da Rocha

(gbr@quimica.ufpb.br, gerd.rocha@gmail.com)

**www.quantum-chem.pro.br**





# Universidade Federal da Paraíba

## Departamento de Química

Laboratório de Química Quântica Computacional

Prof. Gerd Bruno da Rocha

(gbr@quimica.ufpb.br, gerd.rocha@gmail.com)

**www.quantum-chem.pro.br**

The screenshot shows the homepage of the research group. It features the UFPB logo at the top left, followed by the group's name and contact information. Below this is a navigation menu with links to Home, Research, Publications, Members, Collaborators, Links, About Us, and Sponsors. The main content area contains a diagram illustrating the NAMD/ORCA input process, showing the interaction between QM and MM regions. A welcome message from Gerd Bruno Rocha is displayed on the right, along with a brief description of their research focus.

Research Group on Quantum Chemistry  
Applied to Biological Systems  
Gerd Bruno Rocha  
DQ - CCEN - UFPB - Brazil

HOME RESEARCH PUBLICATIONS MEMBERS COLLABORATORS LINKS ABOUT US SPONSORS

WELCOME TO:  
the Group Webpage of Gerd Rocha

We are a Theoretical Quantum Chemistry Group in the Department of Chemistry at UFPB.

Our research focuses on the development of new quantum chemistry methods, high performance computing in chemistry and molecular modeling of organic, inorganic and biological systems.

More information can be found through the links in the upper menu.

NAMD/ORCA input:  
NAMD QM-MM configuration file  
PSF (topology)  
PDB (structure with identified QM atoms)

QM atoms +  
QM atoms: gradients and partial charges ( $r < r_{\max}$ )  
MM atoms: gradients due to Coulomb interactions

QM atoms: coordinates and elements  
MM atoms: coordinates and point charges

Link atom  
C([H])=C  
QM region MM region

Charge redistribution scheme  
(carbon "C" charge is distributed to preserve local polarization)

**THE END**  
Thank you for your  
attention!