

MC102: Algoritmos e Programação de Computadores (turmas 4,5,6,7)

1 - Introdução

ARTHUR VALENCIO

Pós-doutorando IC/Unicamp

Fapesp CEPID NeuroMat

Campinas, 6 de Março de 2019



>> Para que isso serve?

1. Conhecimento de programação -> oportunidade de emprego

Meu colegas e amigos:



Outras empresas em Campinas:



2. Aperfeiçoamento da sua própria área através de **automação**, **machine learning** etc

3. Porque é super legal...

... construir video-games
... simular sistemas

... prever comportamentos
... construir dispositivos inteligentes (IoT)

4. E por que não construir a sua própria startup? (Movile, Quinto andar)



>>Resumo do curso

Aula 1: Introdução

Aulas 2-3: Variáveis, objetos, atribuição

Aulas 4-5: Operadores lógicos, condicionais

Aulas 6-9: Loops

Aula 10: Listas

Aula 11-12: Strings

Aulas 13-16: Revisão para prova e prova 1

Aula 17: Tuplas e dicionários

Aulas 18-19: Funções

Aula 20: Matrizes

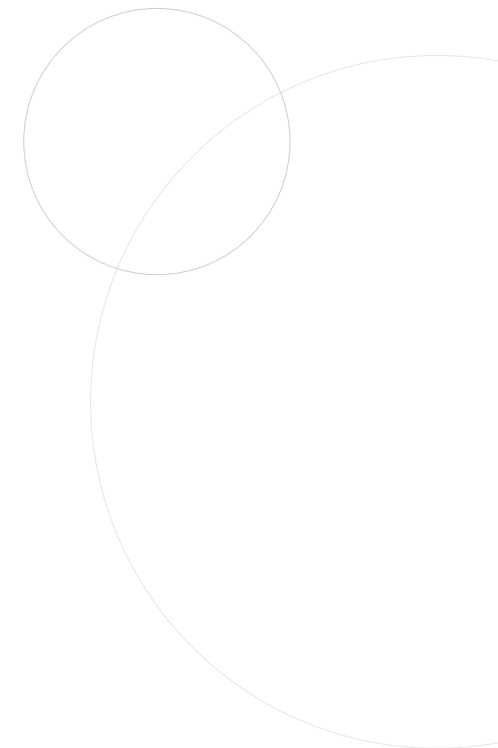
Aula 21: Expressões regulares

Aulas 22-23: Ordenação e busca

Aulas 24-25: Arquivos

Aulas 26 e 27: Recursão e algoritmos recursivos de ordenação

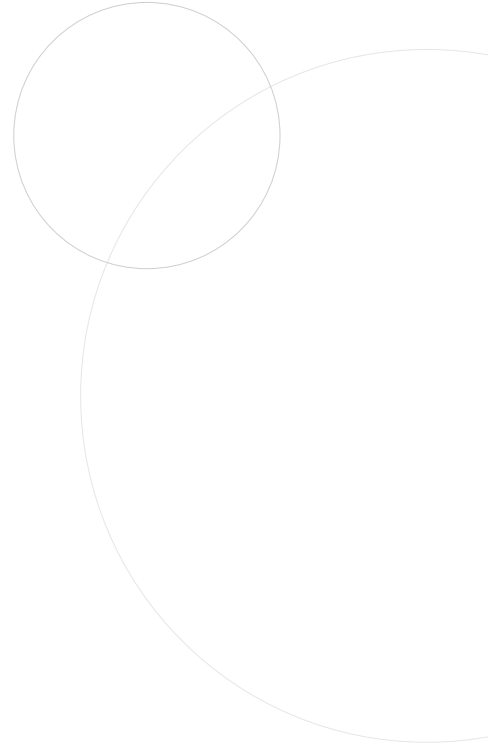
Aulas 28-30: Revisão para prova e prova 2





>>Quiz

Qual é o seu curso?





>>Regras da disciplina

Caso frequência <75%: **Reprovado por frequência**

Caso $M_{provas} < 2,5$ ou $M_{lab} < 2,5$: **Reprovado por nota**

Caso frequência $\geq 75\%$ e $M_{provas} \geq 5$ e $M_{lab} \geq 5$

Aprovado com

$$M_{final} = \max(5; 0,6 \cdot M_{provas} + 0,3 \cdot M_{lab} + 0,1 \cdot M_{atividades})$$

Caso contrário

EXAME

$$M_{preliminar} = \min(4,9; 0,6 \cdot M_{provas} + 0,3 \cdot M_{lab} + 0,1 \cdot M_{atividades})$$

$$M_{final} = (M_{preliminar} + M_{exame})/2$$

Aprovado se $M_{final} \geq 5$

Reprovado por nota se $M_{final} < 5$



>>Regras da disciplina

As tarefas de laboratório serão submetidas pelo sistema SuSy:

<https://susy.ic.unicamp.br:9999/mc102>

As atividades complementares serão submetidas pelo Sistema Moodle:

<http://www.ggte.unicamp.br/ea>

Estes sistemas contam com ferramenta de detecção de plágio.

Os alunos que plagiarem, fizerem a tarefa “em conjunto”, ou qualquer outra tentativa de fraude, receberão **zero na disciplina**.

É considerada tentativa de fraude a submissão de tarefa que passa por todos os testes abertos mas nenhum dos testes fechados do SuSy

Esta é uma disciplina coordenada. Não serei eu a corrigir os exercícios e atividades. Não serei eu a elaborar as questões de prova.

Pedidos de correção de prova levarão à uma re-análise da prova **como um todo**, e as notas podem ser **reduzidas**.



>>Uma breve história da computação

Muito além de Jobs&Gates...

Ada Lovelace



c.1840
1º algoritmo computacional

Alan Turing



c.1940-50
Desenvolvimento da teoria da computação

Grace Hopper



c.1950
Linguagem de programação independente da máquina utilizada

Klára Dan von Neumann

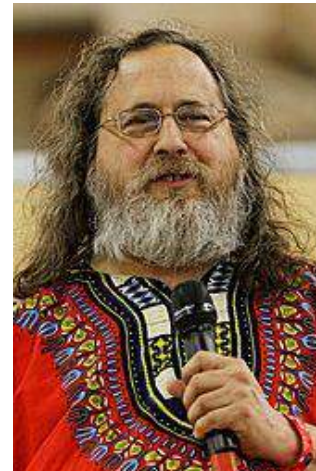


c.1940-60
Programação para previsão do tempo e Projeto Manhattan

Margaret Hamilton **Richard Stallman**



c.1960
Programação do software da Apollo 11



c.1980
GNU – Free software

Tim Berners-Lee



c.1990
HTTP, WWW, W3C
Open data

Larry Page & Sergey Brin

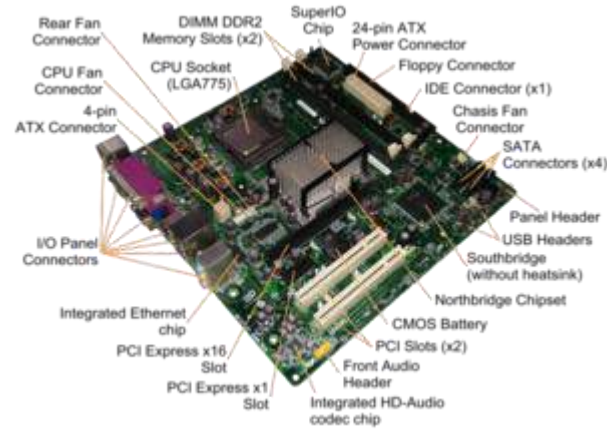


1998-onwards
Google
Alphabet



>>O PC: Componentes básicos

Placa-mãe



CPU



Memória



Disco rígido



Periféricos





>>Unidade de informação

Bit:



'0' ou '1'

- Magnetização (para cima ou para baixo) de um pequeno domínio do material do disco rígido
- Estados de alta/baixa tensão elétrica em uma entrada de transistor no CPU
- Estados de alta/baixa tensão salvos em uma partição da memória enquanto o computador estiver ligado



>>Unidade de informação

SI:

Kilobyte: $KB=10^3B=1000B$

Megabyte: $MB=10^6B=1000KB$

Gigabyte: $GB=10^9B=1000MB$

Terabyte: $TB=10^{12}B=1000GB$

Petabyte: $PB=10^{15}B=1000TB$

Exabyte: $EB=10^{18}B=1000PB$

Zettabyte: $ZB=10^{21}B=1000EB$

Yottabyte: $YB=10^{24}B=1000ZB$

Byte (ISO 2382-1/1993):



256 combinações!

Suficiente para todas as letras, números, caracteres especiais, teclas modificadoras e comandos especiais para a máquina: ASCII

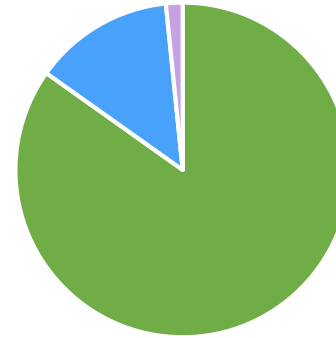
ASCII control characters		ASCII printable characters				Extended ASCII characters									
00	NUL (Null character)	32	space	64	@	96	ˆ	128	Ç	160	à	192	À	224	Ó
01	SOH (Start of Header)	33	!	65	A	97	a	129	Ù	161	á	193	Á	225	Ô
02	STX (Start of Text)	34	"	66	B	98	b	130	Ú	162	â	194	Â	226	Õ
03	ETX (End of Text)	35	#	67	C	99	c	131	Û	163	ã	195	Ã	227	Ö
04	EOT (End of Trans.)	36	\$	68	D	100	d	132	Ü	164	ä	196	Ä	228	ß
05	ENQ (Enquiry)	37	%	69	E	101	e	133	Ý	165	å	197	Å	229	¸
06	ACK (Acknowledgement)	38	&	70	F	102	f	134	ÿ	166	æ	198	Æ	230	ˆ
07	BEL (Bell)	39	'	71	G	103	g	135	ÿ	167	¸	199	¸	231	ˆ
08	BS (Backspace)	40	(72	H	104	h	136	ÿ	168	¸	200	¸	232	ˆ
09	HT (Horizontal Tab)	41)	73	I	105	i	137	ÿ	169	¸	201	¸	233	ˆ
10	LF (Line feed)	42	*	74	J	106	j	138	ÿ	170	¸	202	¸	234	ˆ
11	VT (Vertical Tab)	43	+	75	K	107	k	139	ÿ	171	¸	203	¸	235	ˆ
12	FF (Form feed)	44	,	76	L	108	l	140	ÿ	172	¸	204	¸	236	ˆ
13	CR (Carriage return)	45	-	77	M	109	m	141	ÿ	173	¸	205	¸	237	ˆ
14	SO (Shift Out)	46	.	78	N	110	n	142	ÿ	174	¸	206	¸	238	ˆ
15	SI (Shift In)	47	:	79	O	111	o	143	ÿ	175	¸	207	¸	239	ˆ
16	DLE (Data link escape)	48	;	80	P	112	p	144	ÿ	176	¸	208	¸	240	ˆ
17	DC1 (Device control 1)	49	<	81	Q	113	q	145	ÿ	177	¸	209	¸	241	ˆ
18	DC2 (Device control 2)	50	=	82	R	114	r	146	ÿ	178	¸	210	¸	242	ˆ
19	DC3 (Device control 3)	51	>	83	S	115	s	147	ÿ	179	¸	211	¸	243	ˆ
20	DC4 (Device control 4)	52	@	84	T	116	t	148	ÿ	180	¸	212	¸	244	ˆ
21	NAK (Negative acknow.)	53	A	85	U	117	u	149	ÿ	181	¸	213	¸	245	ˆ
22	SYN (Synchronous idle)	54	B	86	V	118	v	150	ÿ	182	¸	214	¸	246	ˆ
23	ETB (End of trans. block)	55	C	87	W	119	w	151	ÿ	183	¸	215	¸	247	ˆ
24	CAN (Cancel)	56	D	88	X	120	x	152	ÿ	184	¸	216	¸	248	ˆ
25	EM (End of medium)	57	E	89	Y	121	y	153	ÿ	185	¸	217	¸	249	ˆ
26	SUB (Substitute)	58	F	90	Z	122	z	154	ÿ	186	¸	218	¸	250	ˆ
27	ESC (Escape)	59	G	91	[123	{	155	ÿ	187	¸	219	¸	251	ˆ
28	FS (File separator)	60	H	92	\	124		156	ÿ	188	¸	220	¸	252	ˆ
29	GS (Group separator)	61	I	93]	125	}	157	ÿ	189	¸	221	¸	253	ˆ
30	RS (Record separator)	62	J	94	^	126	~	158	ÿ	190	¸	222	¸	254	ˆ
31	US (Unit separator)	63	K	95	_			159	ÿ	191	¸	223	¸	255	nbsp
127	DEL (Delete)														



>>Sistemas operacionais

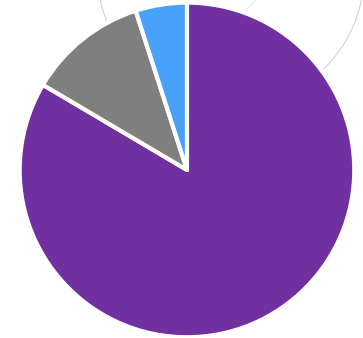


PC



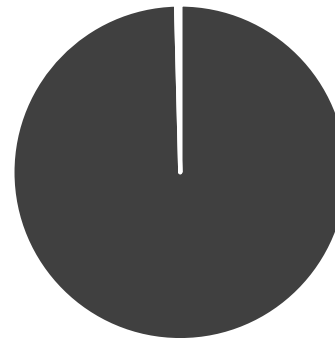
■ Windows ■ MacOS ■ Linux

Smartphones



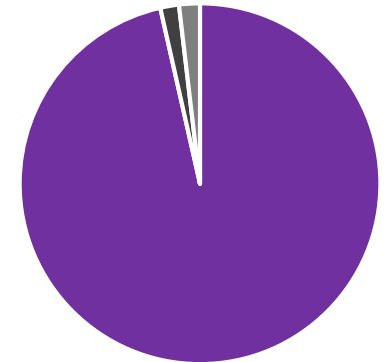
■ Android ■ iOS ■ Others

Supercomputadores



■ Linux ■ Outros Unix

Servidores Web

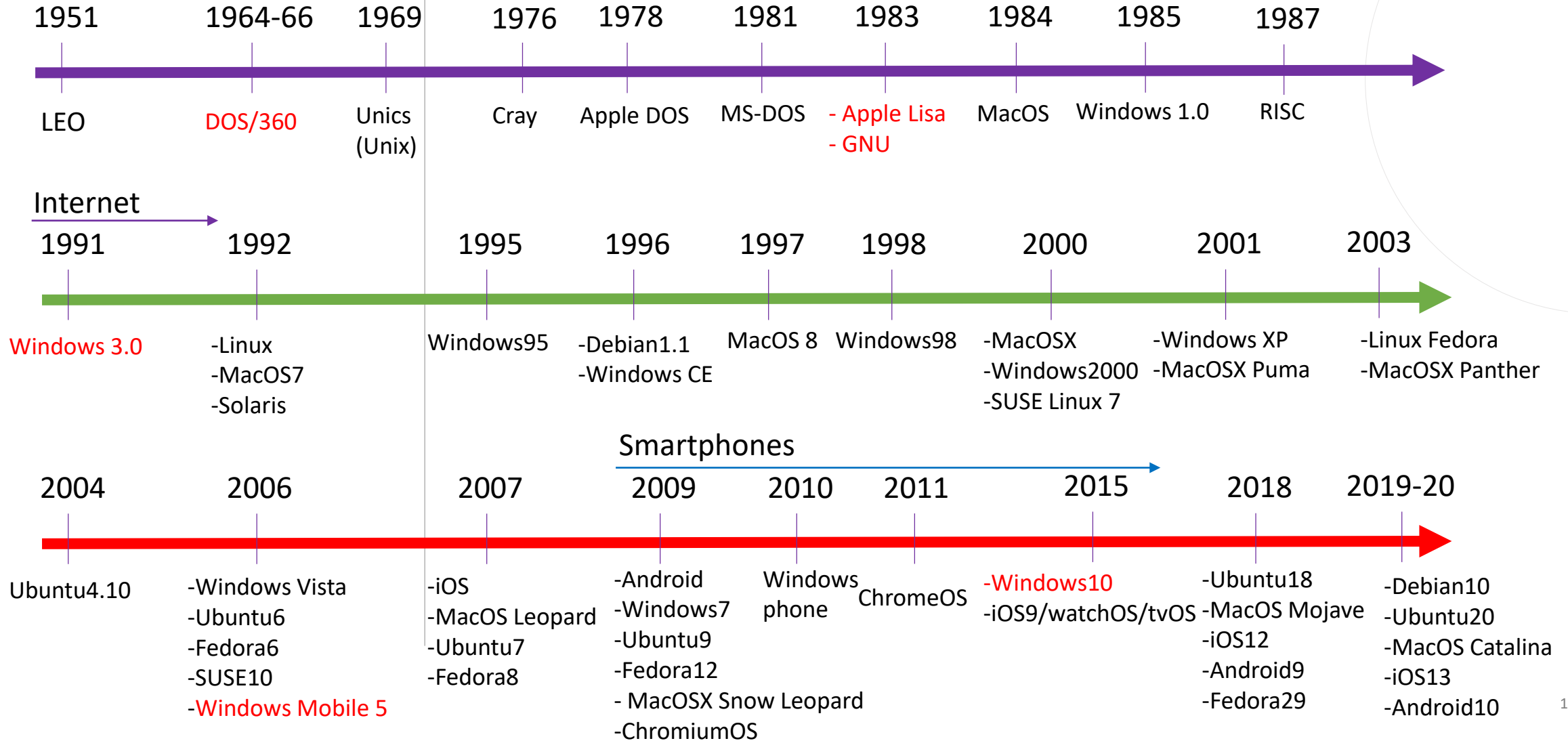


■ Linux ■ Outros Unix ■ Windows



>>Sistemas operacionais

Interface gráfica





>>Linguagens de marcação

LATEX



Adobe PostScript

HTML



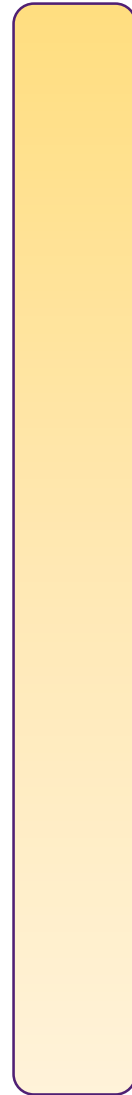


>>Quiz

Você já sabe programar em alguma
linguagem de marcação?



>>Linguagens de programação



Wolfram Mathematica
Matlab, Octave, R, Maple

Python, Ruby

Kotlin/Swift

Java, C#

C++

C / Fortran

Assembly



+ funcio-
nalidades

- funcio-
nalidades



códigos
+ curtos

códigos
+ longos



+ lento

+ rápido



>>Quiz

Você já sabe programar em alguma
linguagem de programação?

>>O que é um algoritmo?

Preâmbulo:
Declaração de
variáveis e funções,
importar
bibliotecas

Instruções
definindo as
operações para
fazer o bolo

- Série de regras (instruções) que definem uma sequência de operações
- Em geral, formalizações requerem que um programa eventualmente finalize em **tempo finito** para ser considerado um algoritmo

Receita de Bolo de Chocolate Simples

Ingredientes:

- 1 xícara de chá de leite
- 1 xícara de chá de óleo de soja
- 2 unidades de ovo
- 2 xícaras de chá de farinha de trigo
- 1 xícara de chá de achocolatado em pó
- 1 xícara de chá de açúcar
- 1 colher de sopa de fermento químico em pó

Preparo:

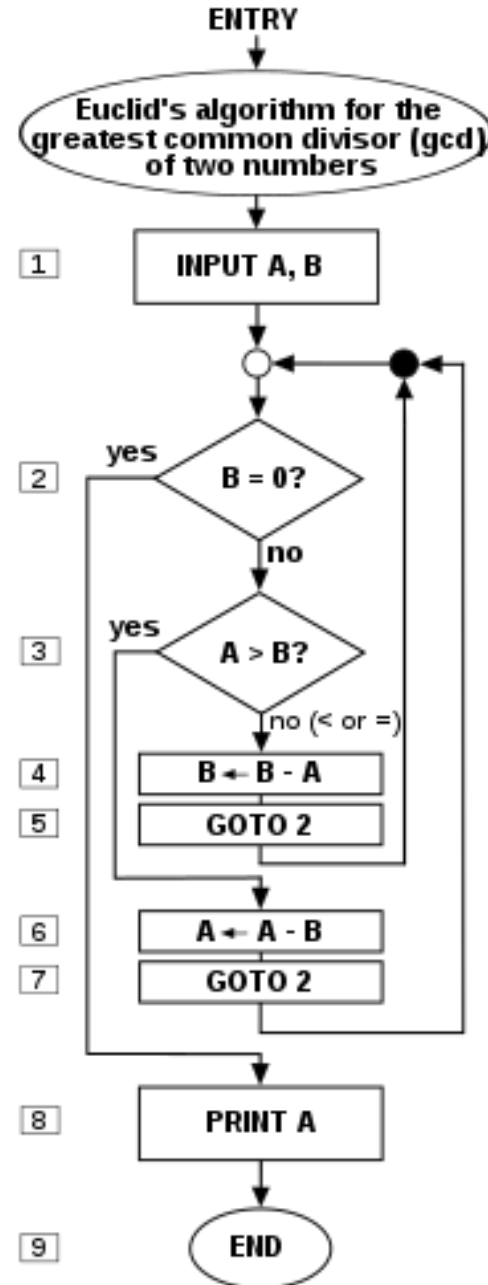
- Coloque os líquidos no liquidificador
- Bata até misturar bem
- Coloque os outros ingredientes, sendo o fermento o último
- Prepare uma forma untada e enfarinhada
- Leve para assar em forno médio



O processo de fabricação do bolo finaliza em **tempo finito**, e com um resultado perfeito para o café da tarde



>>O que é um algoritmo?





>> Sobre o Python



Objetivos:

- Linguagem simples e intuitiva tão poderosa quanto alternativas;
- **Código aberto**, de forma que qualquer um pode contribuir para seu desenvolvimento;
- Código que pode ser facilmente entendível a partir do inglês simples;
- Adequado para tarefas cotidianas, permitindo desenvolver códigos rapidamente.

Comunidade ampla e ativa, com muitas ferramentas disponíveis

3ª maior linguagem no GitHub, e continua aumentando

(1ª e 2ª são Java e JavaScript)



Criado por Guido van Rossum em 1991

Versão 2.7
e
Versão 3.7



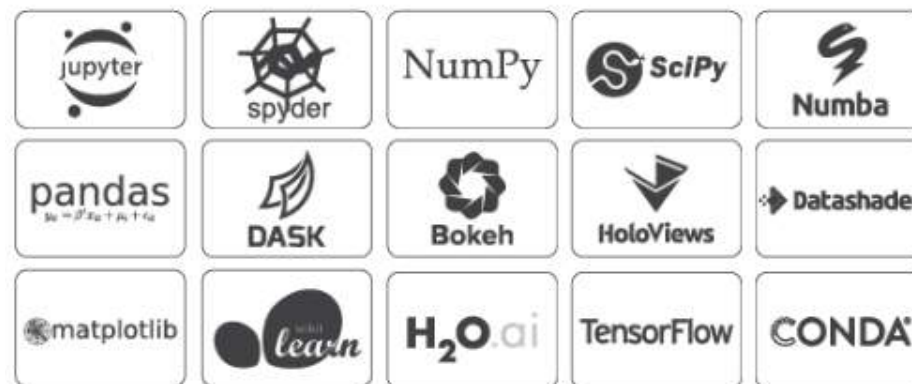
>> Instalando o Python no seu PC

- Mac/Linux vem com Python pré-instalado (veja se está só na versão 2.7, antiga, ou se a versão 3, que usaremos neste curso, está instalada).
- Windows não vem com Python instalado.
- Se necessário, instale a versão desejada em <https://www.python.org/downloads/>

Recomendação: instale o pacote Anaconda

<https://www.anaconda.com/distribution/#download-section>

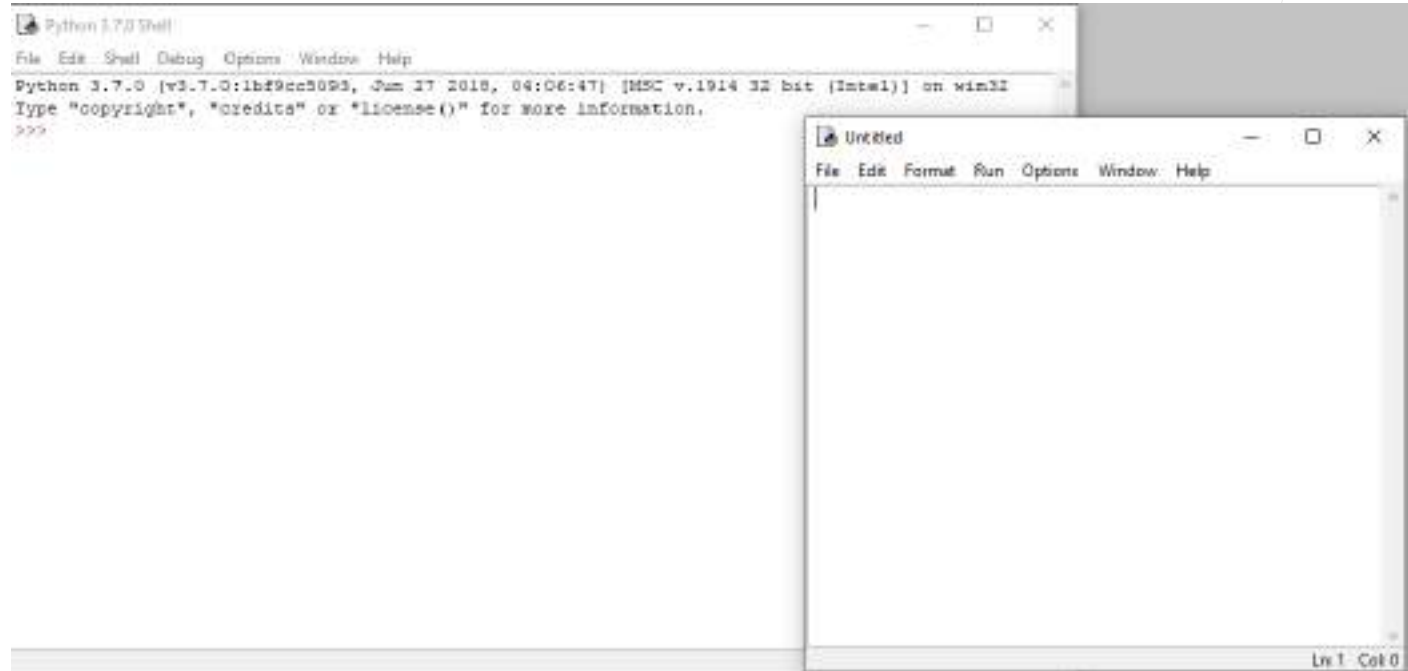
Tal pacote já instala todas estas bibliotecas para análise de dados, processamentos diversos, documentação, geração de gráficos, etc:





>>Ambiente de desenvolvimento (IDE)

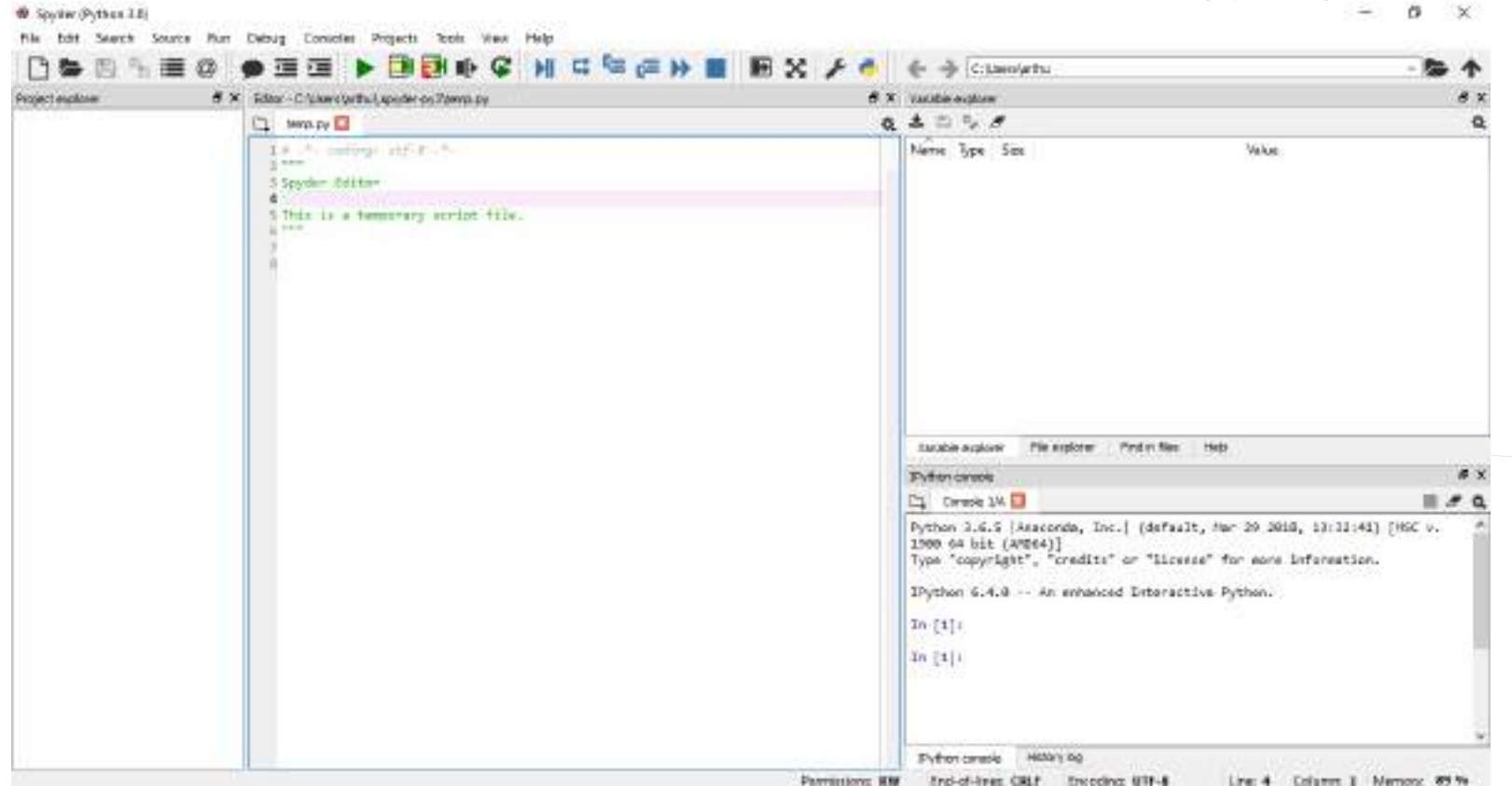
IDLE (IDE leve e de propósito geral)





>>Ambiente de desenvolvimento (IDE)

Spyder (IDE para ciência de dados)

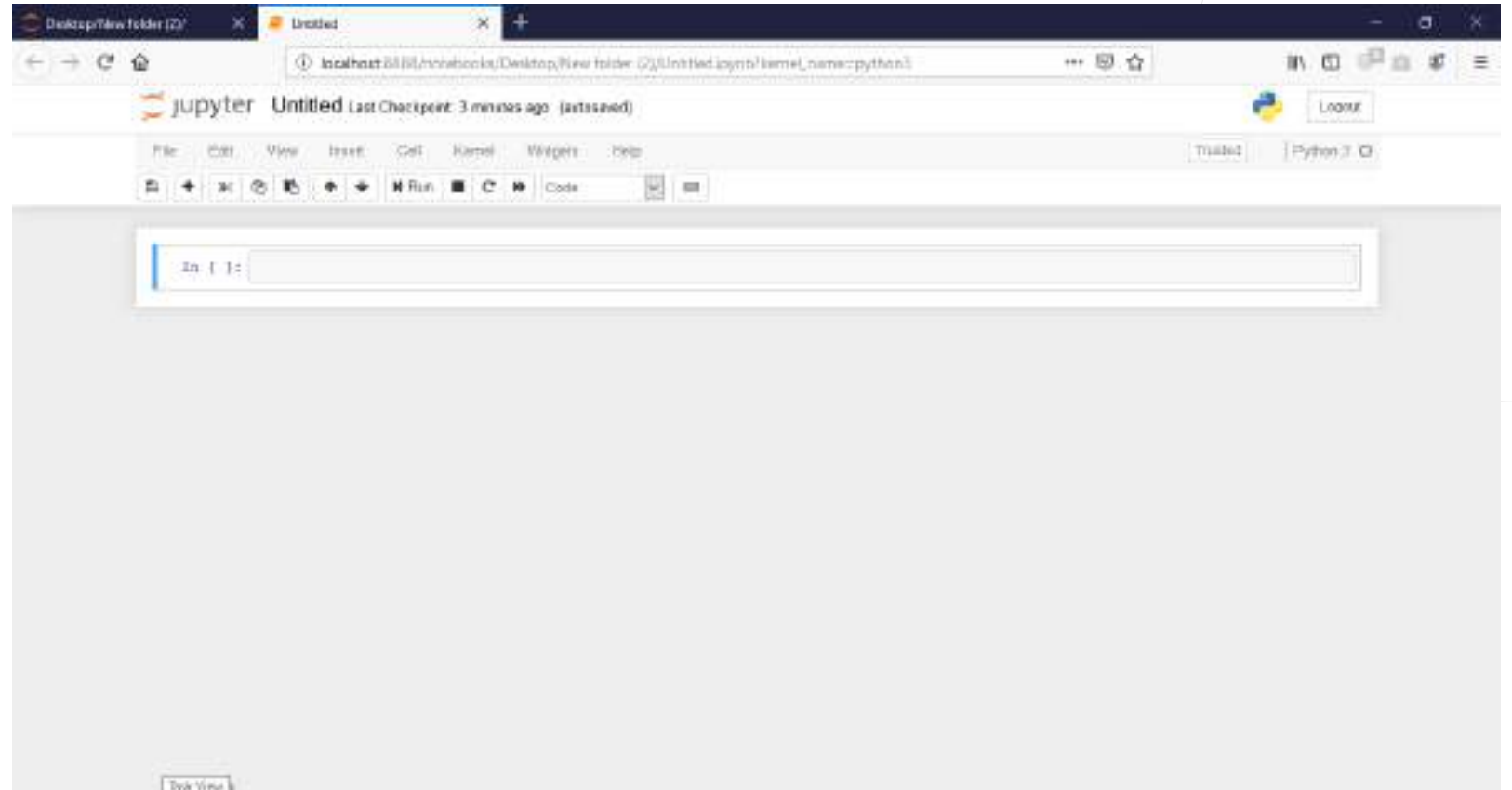




>>Ambiente de desenvolvimento (IDE)

Jupyter

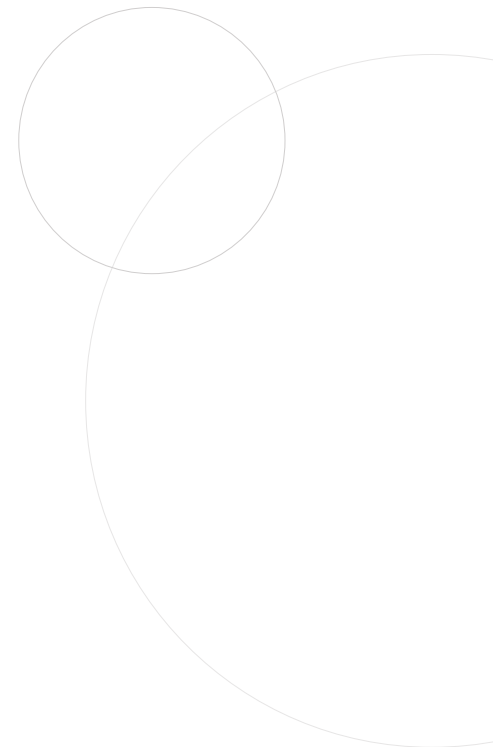
(Desenvolvimento de “cadernos” mesclando código, resultados gráficos, texto e equações. Roda em browser.)





>>Ambiente de desenvolvimento (IDE)

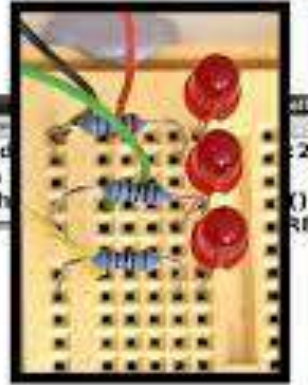
<https://jupyter.org/try>



>>Desafio para curiosos



```
import RPi.GPIO as GPIO
import time
GPIO.setmode(GPIO.BOARD)
GPIO.setup(7,GPIO.OUT)
GPIO.setup(11,GPIO.OUT)
GPIO.setup(13,GPIO.OUT)
for x in range(0,7):
    GPIO.output(7,True)
    time.sleep(.5)
    GPIO.output(7,False)
    GPIO.output(11,True)
    time.sleep(.5)
    GPIO.output(11,False)
    GPIO.output(13,True)
    time.sleep(.5)
    GPIO.output(13,False)
GPIO.cleanup()
```



Fazer programação Python para controlar LEDs com Raspberry Pi



>>Desafio para curiosos

Construir um jogo em Scratch:

<https://scratch.mit.edu/>





>> Grupos



Python Campinas

- 📍 Campinas, Brazil
- 👤 126 members - Public group
- 👤 Organized by Emerson Rodrigues da S. and 1 other

Show 📅 🗨️ 📄



GruPy-Campinas

- 📍 Campinas, Brazil
- 👤 38 members - Public group
- 👤 Organized by Felipe K. and 1 other

Show 📅 🗨️ 📄



Camp Data

- 📍 Campinas, Brazil
- 👤 29 members - Closed group
- 👤 Organized by Felipe K.

Open to all 📅 🗨️ 📄



PyLadies Campinas
@pyladiescampinas
Página inicial
Grupos





>>MOOCs



Python Courses

All Subjects > Computer Programming > Python

[View all edX Courses](#)

Take free online Python courses from institutions like MIT, Microsoft and Georgia Tech. Learn Python programming and advance your career. Join today.

Related Topics - Amazon Web Services | Android Development | Angular | Apache Spark | App Development | Azure | Bootstrap | C | C++ | C# | CSS | Computer Engineering | Code | Data Science | DevOps | Docker | Git | HTML | HTML5 | IoT | Kids Coding Languages | Kubernetes | Natural Language Processing | .NET | Java | JavaScript | R Programming | Node.js | PowerShell | ReactJS | Scala | SQL | Software Engineering | Software Testing | Swift | Technology | TypeScript | Virtual Reality



XSERIES PROGRAM

MITX
Computational Thinking using Python
Learn to think computationally and write programs to tackle...

Current



CREDIT-ELIGIBLE

MITX
Introduction to Computer Science and Programmin...

Current
Starts: January 22, 2020



PROFESSIONAL CERTIFICATE PROGRAM

GTx
Introduction to Python Programming
Learn the fundamentals of computer science in one of the...

Current



PROFESSIONAL CERTIFICATE PROGRAM

IBM
Python Data Science
Launch your career in data science with hands-on Python...

Current



VERIFIED

HarvardX
CS50's Introduction to Computer Science

Current
Self Paced



>>MOOCs

coursera

Explore ▾

python



Browse > Python

Python Courses

All

Degrees & Certificates

Explore 100% online Degrees and Certificates on Coursera



Master of Data Science
HSE University



Master of Computer Science
Arizona State University

Showing 376 total results for "python"

Filter By

Language ▾

Level ▾

Skills ▾

Partner ▾

Learning Product ▾



Python for Everybody

University of Michigan

SPECIALIZATION

★★★★★ 4.8 (91,364) | 1.1M students

Beginner



Python 3 Programming

University of Michigan

SPECIALIZATION

★★★★★ 4.7 (3,507) | 71K students

Beginner



IBM Data Science

IBM

PROFESSIONAL CERTIFICATE



>>MOOCs

coursera

Explore ▾

python



Browse > Python

Python Courses

All

Degrees & Certificates

Explore 100% online Degrees and Certificates on Coursera



Master of Data Science
HSE University



Master of Computer Science
Arizona State University

Showing 376 total results for "python"

Filter By

Language ▾

Level ▾

Skills ▾

Partner ▾

Learning Product ▾



Python for Everybody

University of Michigan

SPECIALIZATION

★★★★★ 4.8 (91,364) | 1.1M students

Beginner



Python 3 Programming

University of Michigan

SPECIALIZATION

★★★★★ 4.7 (3,507) | 71K students

Beginner



IBM Data Science

IBM

PROFESSIONAL CERTIFICATE



>>MOOCs

coursera

Explore ▾

What do you want to learn?



For Enterprise

Browse > Computer Science > Software Development

Introdução à Ciência da Computação com Python Parte 1

★★★★★ 4.5 (3,190 ratings) | 🔥 99%



Fabio Kon

TOP INSTRUCTOR

Enroll for Free

Starts Mar 06

Financial aid available

98,735 already enrolled

Offered by



Included with **coursera** PLUS

[Learn More](#)

About

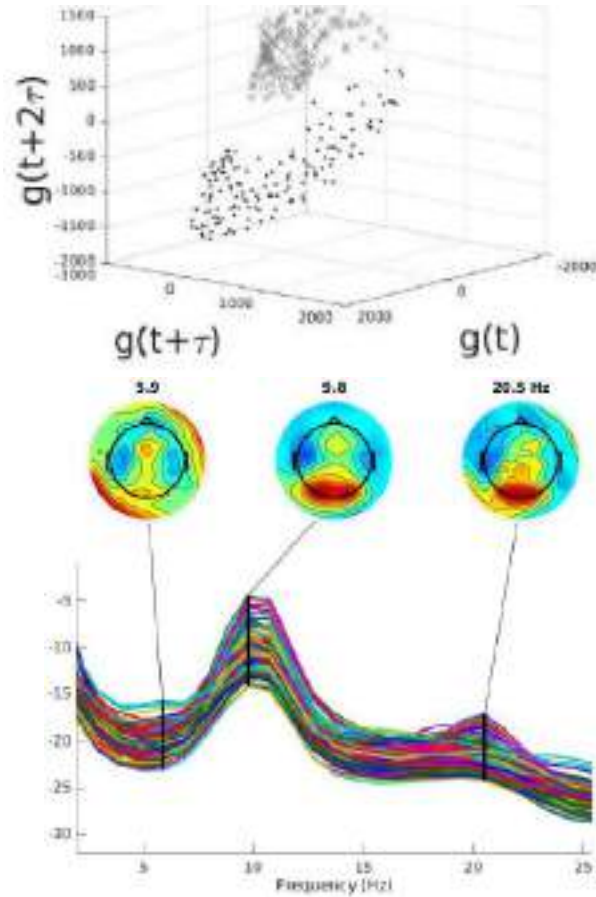
Instructors

Syllabus

Reviews

Enrollment Options

FAQ



That's all folks!



arthur_valencio@physics.org



<http://www.arthurvalencio.com>

<http://www.ic.unicamp.br/~mc102>