

Universidade Federal de Viçosa Departamento de Informática Centro de Ciências Exatas e Tecnológicas



INF 100 – Introdução à Programação

Apresentação e Conceitos Básicos

A disciplina

- Aulas teóricas
- Aulas práticas → turmas "embaralhadas
- Monitoria: horário a definir
 - Link disponível no PVANet
- Calendário, <u>avisos</u>, notas → PVA-NET
- Acesso ao material das aulas práticas (roteiros, arquivos entregues, notas):
 - Link disponível no PVANet





A disciplina

- Professores (turmas teóricas):
 - Mauro Rocha (coordenador da disciplina) T3
 - 3899-1775
 - mnrocha.ufv@gmail.com
 - Marcos Henrique Ribeiro T1 e T5
 - mhfribeiro@gmail.com
 - Lucas Vegi T2
 - lucasvegi@gmail.com
 - Levi Lelis T4
 - levilelis@gmail.com
 - Vladimir (turma especial em inglês: T6)
 - vladimir.iorio@gmail.com





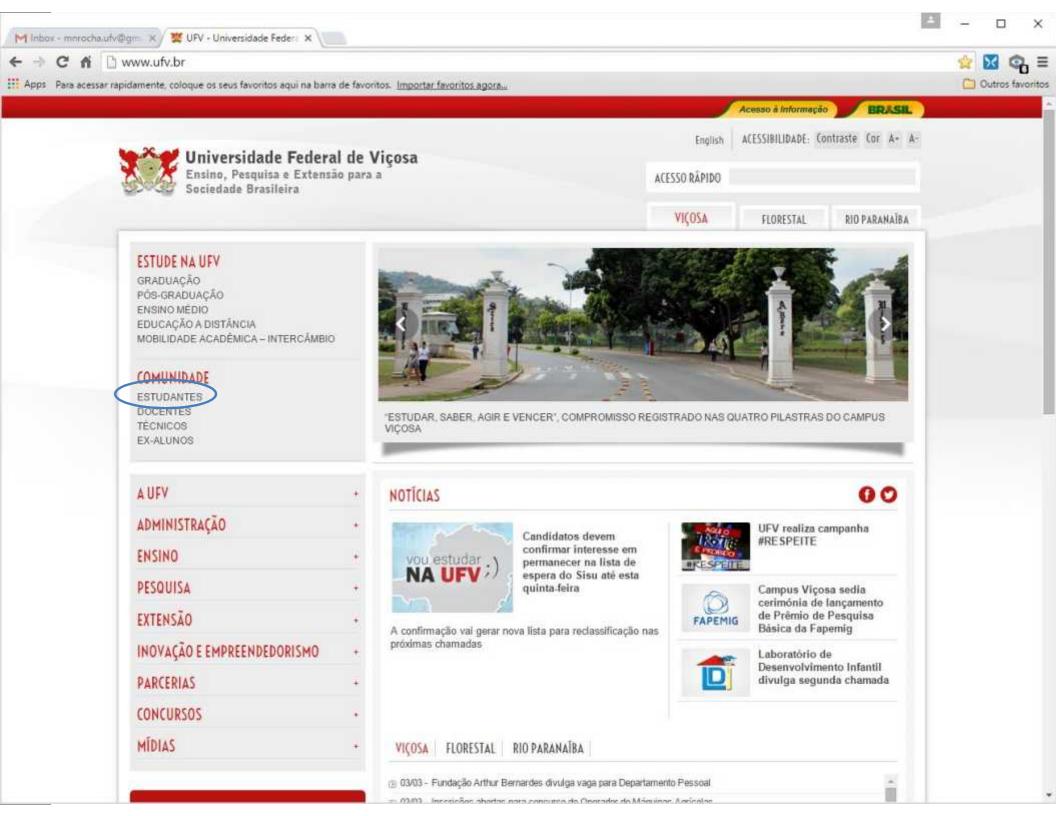


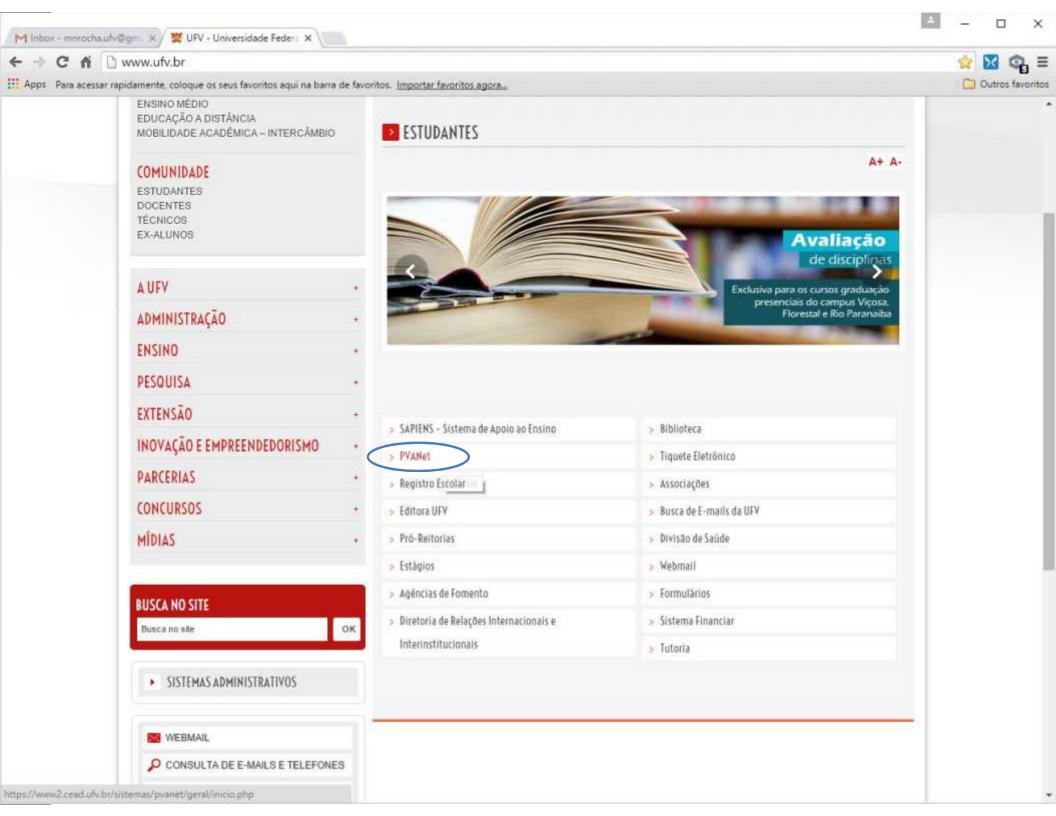
Material

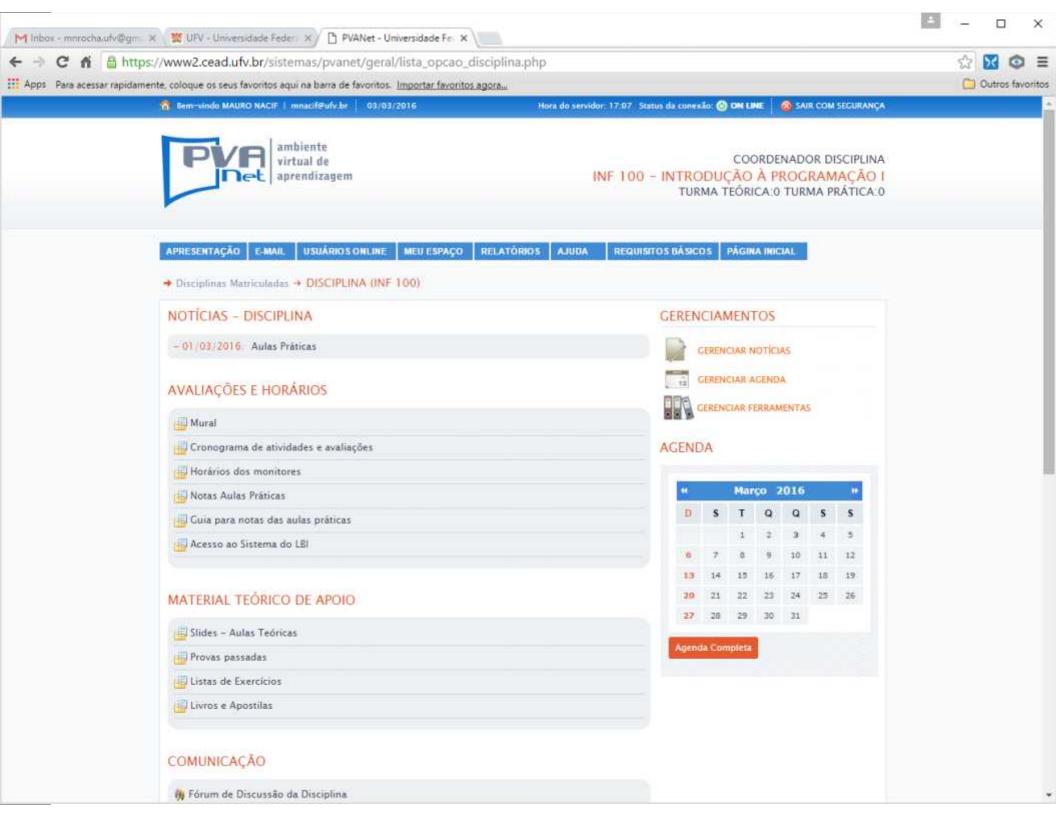
- Slides de Aula (disponíveis no PVANet)
- Alguns livros grátis:
 - Links disponíveis no PVANet
- Muito material completar na Internet (aulas, cursos, apostilas e livros online)

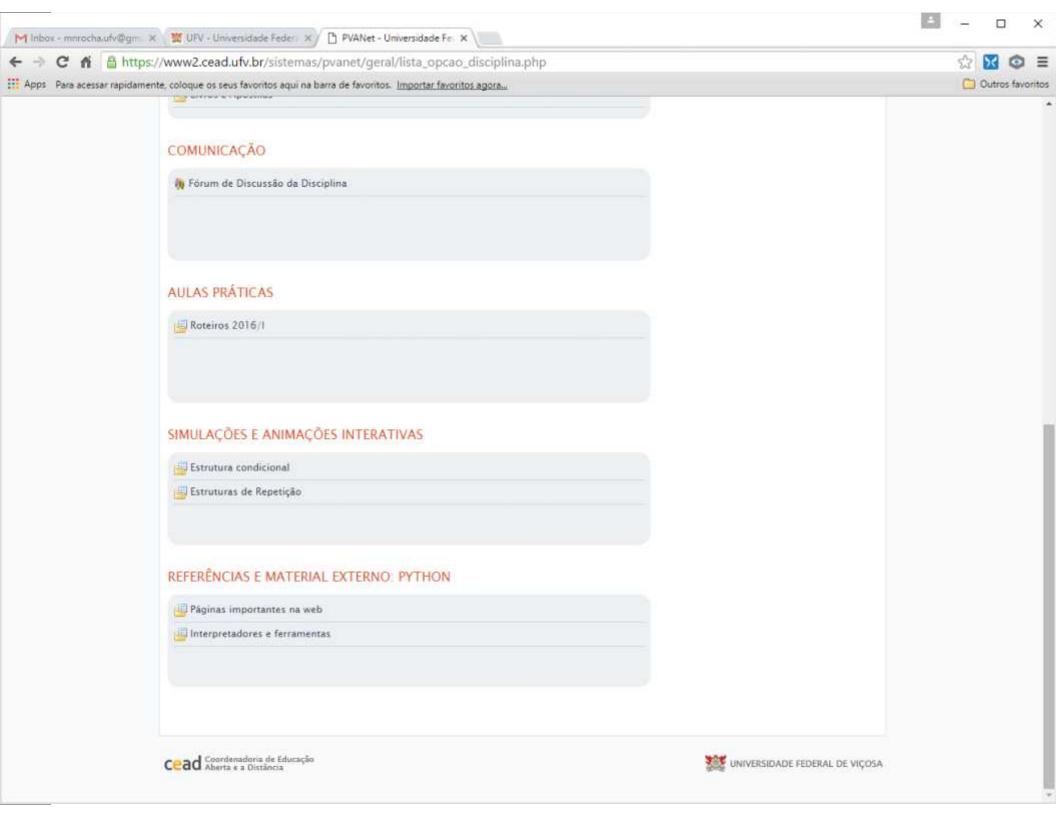










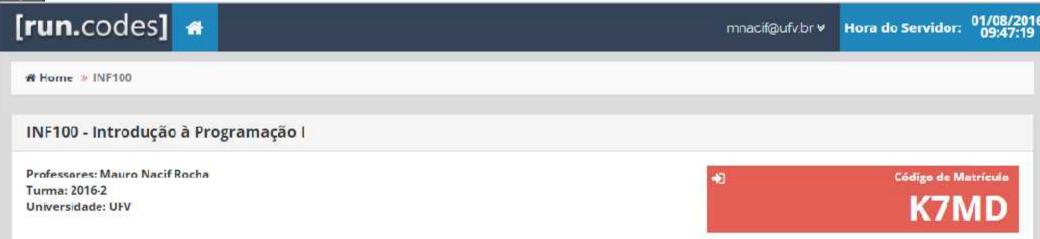


Exercícios Online



Exercícios Online

0



Exe	rcícios					
No.	Exercício	Status	Casos Corretos	Nota	Prazo de Entrega	Ações
1	Exercício A.1	Não Entregue	0/2	0	15/12/2016 18:00:00	Ver Detalhes
2	Exercício A.2	Não Entregue	0/2	0	15/12/2016 18:00:00	Ver Detalhes
3	Exercício A.3	Não Entregue	0/2	0	15/12/2016 18:00:00	Ver Detalhes
4	Exercício A.4	Finalizado	2/2	10.00	15/12/2016 18:00:00	Ver Detalhes
5	Exercício A.5	Finalizado	2/2	10.00	15/12/2016 18:00:00	Ver Detalhes

Exercícios Online







Avaliação

- 2 provas de 35 pontos (Matéria cumulativa)
- 30 pontos em aulas práticas, distribuídos ao longo das diversas aulas práticas
- Algumas aulas práticas possuem trabalhos que valem mais pontos (sem consulta).
- NÃO haverá prova substitutiva
- Segunda chamada → falta justificada (matéri toda)





Cronograma

Planejamento: Sincronia entre turmas, aulas teóricas e práticas.

	Período: 2016/2				T	Auda	Pontos			
Samana	de		TER	QUA	QUI	SEX		Tema / Conteúdo	Aula Prática	por Prática
Semana	ue	а	Р	Р	Т	Т	PROVA	Conteudo		
1	01 ago	05 ago						1	-	-
2	08 ago	12 ago						2	1	1
3	15 ago	19 ago						3	2	1
4	22 ago	26 ago						3	3	1
5	29 ago	02 set						4	4	1
6	05 set	09 set		Feriado				5	5	1
7	12 set	16 set						6	6	4
8	19 set	23 set					Prova 1	Exerc.	7	4
9	26 set	30 set				Feriado		7	8	1
10	03 out	07 out						7	9	1
11	10 out	14 out		Feriado				8	10	1
12	17 out	21 out	SIA	SIA	SIA			8	11	1
13	24 out	28 out				Recesso		Exerc.	12	4
14	31 out	04 nov		Feriado				9	13	1
15	07 nov	11 nov						10	14	3
16	14 nov	18 nov	Feriado					Exerc.	15	1
17	21 nov	25 nov					Prova 2	Exerc.	16	4
18	28 nov	02 dez					2a cham.		Repositiva	30
19	05 dez	09 dez					Final			





Cronograma

۱ ا	Conteúdo	Tema	
	1	Apresentação / Introdução	±,
	2	Algoritmos, estrutura do programa, E/S, variáveis, programas simples	·
'	3	Comandos de seleção if, if/else, ifs aninhados, elif	Prova 1
	4	Comandos de repetição: while, break	Flova
	5	Comandos de repetição: for	
	6	Comandos de repetição: exercícios	
۱ ا	7	Arranjos 1D (listas, vetores)	
	8	Arranjos 2D (matrizes)	Prova 2
	9	Funções 1 (conceitos gerais, escopo)	FIOVAZ
	10	Funções 2 (arranjos como parâmetros)	





Cronograma

INF100 – Previsão de Calendário Semestre Letivo 2016/II

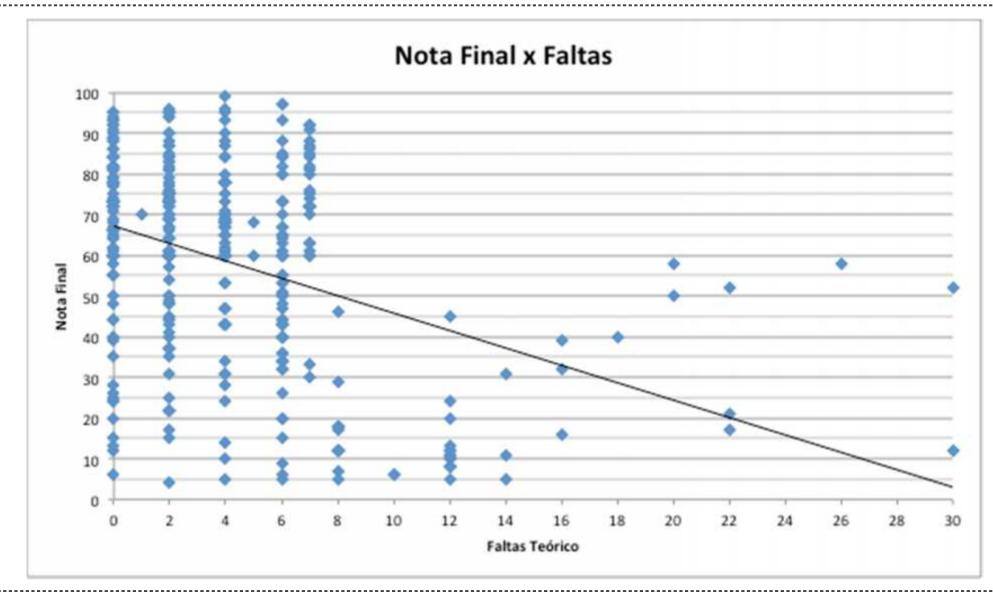
Prática	Semana	Valor (em
Fratica	Semana	pontos)
P1	08 ago	1
P2	15 ago	1
P3	22 ago	1
P4	29 ago	1
P5	05 set	1
P6	12 set	3
P7	19 set	4
P8	26 set	1
P9	03 out	1
P10	10 out	1
P11	17 out	1
P12	24 out	4
P13	31 out	1
P14	07 nov	4
P15	14 nov	1
P16	21 nov	4
repositiva	28 nov	4 ou 1

Provas Teóricas	Data	Horário	
Primeira Prova	24/set/2016	10:00	
Segunda Prova	26/nov/2016	10:00	
2ª chamada	03/dez/2016	10:00	





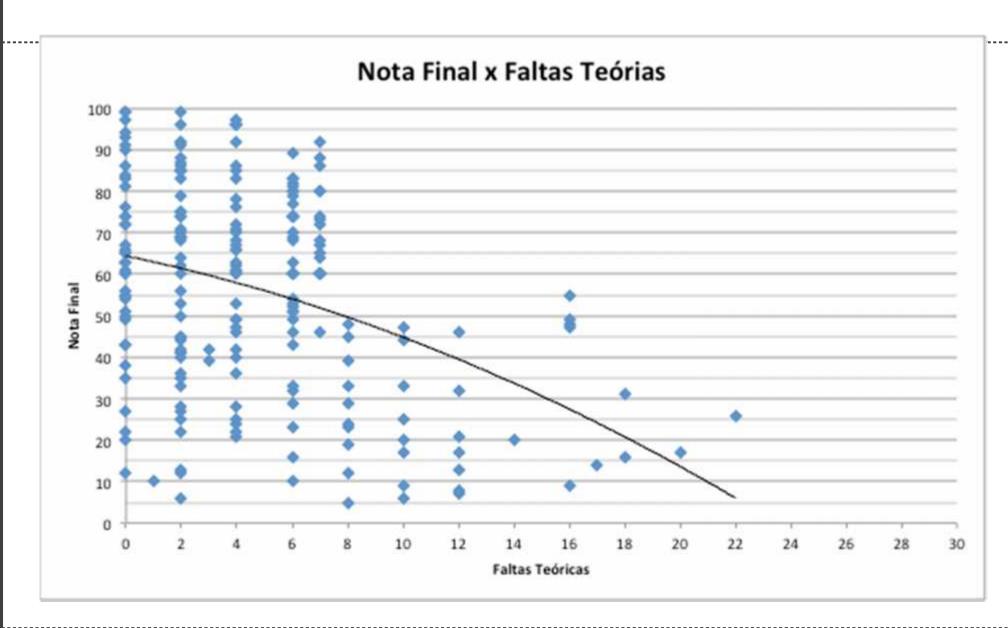
Dados de 2015-I







Dados de 2015-II

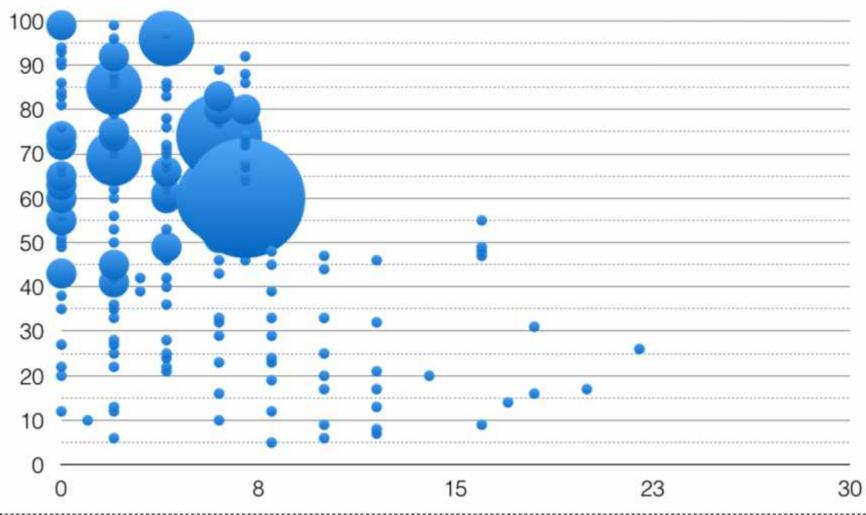






Dados de 2015-II

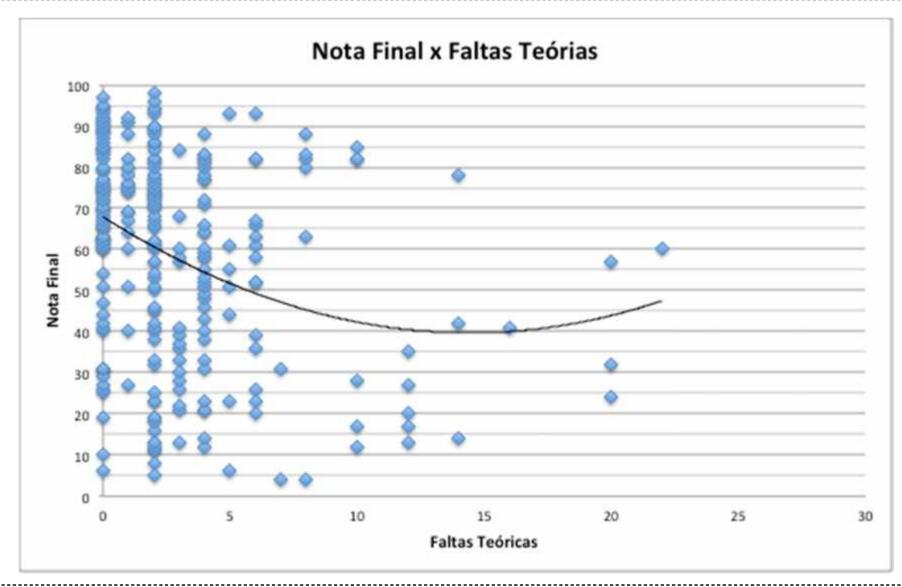
Nota Final x Faltas Teóricas







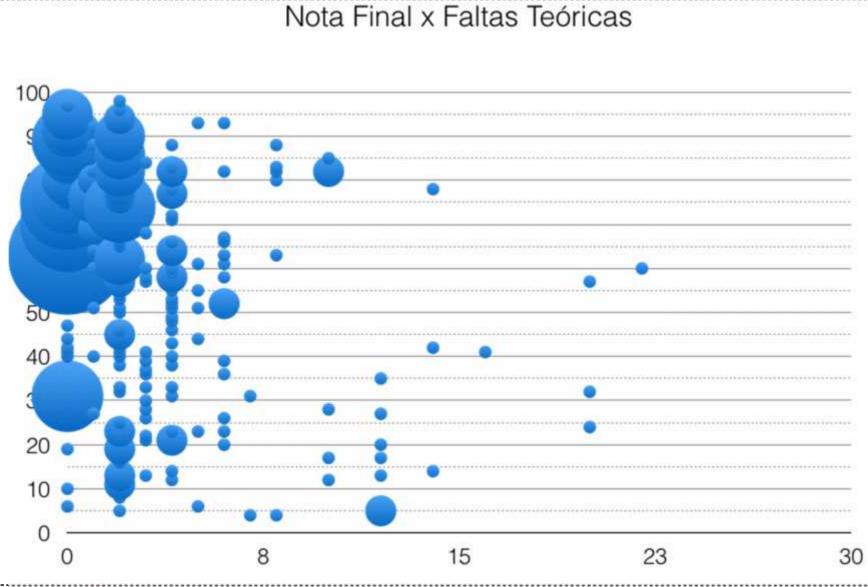
Dados de 2016-l







Dados de 2016-l





Objetivos

- Para que programar?
- Dispositivos controlados por software em diversas áreas.
- Lógica de programação → capacidade de aprender a programar utilizando diversas ferramentas como Excel, Matlab, etc.
- Noção de custos e dificuldade para desenvolver programas.
- Vocabulário "básico" para se comunicar com desenvolvedores de software.





Objetivos

- Desenvolver um "raciocínio computacional".
- Conseguir dominar os elementos construtivos mais simples da programação para escrever programas pequenos.
- Conseguir ler e entender programas pequenos escritos por outras pessoas.





Objetivos

- Usar o computador de forma mais efetiva para a automação de tarefas.
- Conseguir mapear problemas científicos simples para dentro do escopo computacional (da programação).
- Obter vantagem competitiva: IC, estágios, pósgraduação, mercado de trabalho.



O que é computador?

"uma máquina capaz de variados tipos de tratamento de informações ou processamento de dados."







Calculadora x Computador?



- Programação fixa (circuitos eletrônicos) – fixed-program computer
- Entrada e saída fixos (teclado / display)



- Programação aberta stored-program computer
- Entradas e saídas diversas
- Capacidade de "comunicação" com outros computadores





O que é computador?

Uma calculadora um pouco mais capaz...

... mas ainda tão "burro" quanto!







O que é computação ou computar?

O que é Conhecimento?

Conhecimento declarativo. Ex.: definição de √x

$$\sqrt{x} = y, \quad y^2 = x, \, y \ge 0$$

Mas isso não nos ajuda a obter o valor de y.

– Conhecimento imperativo Como obter (computar) $\sqrt{\chi}$?





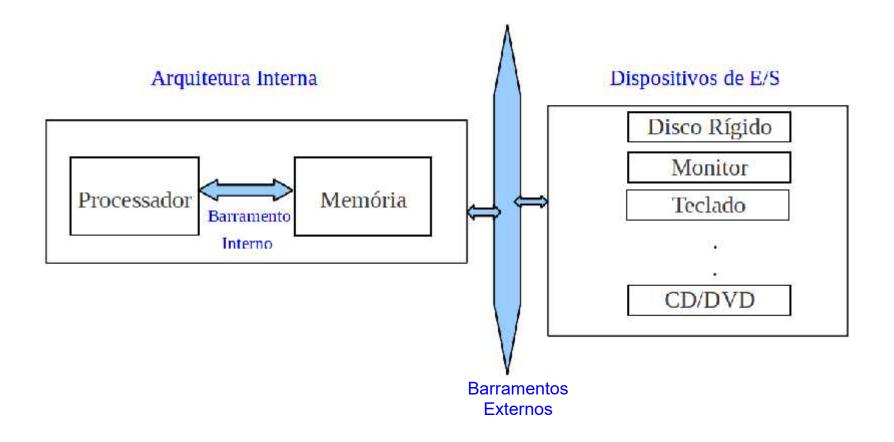
Softwares

- Programas de computador;
- Sequência finita de instruções bem definidas e não ambíguas → algoritmo (como uma "receita de bolo").
- É o que faz o computador funcionar ou realizar qualquer coisa útil que não seja só dissipar energia.
- Podem ser vistos como uma sequência de comandos que detalham todas as ações que um computador deve realizar.
- Um computador executa precisamente o que está determinado em um programa.





Funcionamento Básico (arquitetura) de um Computador







- Software
- O computador processa suas instruções passo a passo, começando pela primeira instrução e seguindo a ordem pré-estabelecida no programa;
- Duas grandes categorias de software
 - Sistema Operacional
 - Aplicativos





- Software Básico = Sistema Operacional
 - Exemplos
 - Windows (10, 8, 7, Vista, 2003, XP, 2000, ME, 98 etc.)
 - Linux (Debian, Ubuntu, Mint, Red Hat, Mandriva etc.)
 - Unix
 - Android
 - iOS
 - MAC OSX
 - etc.





- Software Básico = Sistema Operacional
 - Interface Gráfica x Modo Texto













- Software Básico = Sistema Operacional
 - Controla o funcionamento do computador;
 - Reconhece e controla o funcionamento do hardware;
 - Possibilita que outros programas aproveitem os recursos do monitor gráfico, disco rígido, CD, impressora, câmera etc. sem precisar saber dos detalhes de funcionamento desses dispositivos;
 - Realiza a interação com o usuário;
 - Torna o uso do computador mais fácil.



Aplicativos

- Softwares desenvolvidos para realizarem uma tarefa ou um conjunto de tarefas relacionadas a uma área ou finalidade específica:
 - Edição de textos
 - Planilhas eletrônicas
 - Cálculos estruturais (Engenharia Civil)
 - Análise de riscos (Economia)
 - Sequenciamento de DNA (Biologia)
 - Jogos
 - etc.





- O que preciso para produzir Aplicativos?
 - Conhecer uma linguagem de programação;
 - Um editor de textos para que se escreva o código do programa na linguagem escolhida;
 - Um compilador ou interpretador;



Linguagens de Programação

Alto Nível x Baixo Nível

Uso Geral x Específica

Interpretada x Compilada





Linguagem Python

Alto Nível x Baixo Nível

Uso Geral x Específica

Interpretada x Compilada





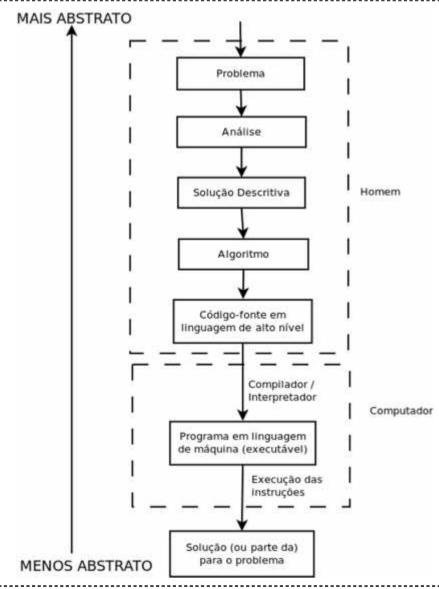
Conceitos Básicos Complementares

- Interpretadores e Compiladores
 - São responsáveis por converter um código escrito em uma linguagem de alto nível (ex.: Python, C++, Java, C# etc.) em um código binário executável (instruções mais simples, que a CPU entende);
 - São específicos para a linguagem para o qual foram projetados para compilar e também para o SO e o hardware onde estão sendo executados.
 - Portanto, para uma mesma linguagem, existem compiladores diferentes, em SOs diferentes.





Introdução à Programação de Computadores









Universidade Federal de Viçosa Departamento de Informática Centro de Ciências Exatas e Tecnológicas



INF 100 – Introdução à Programação

Algoritmos e Programas Básicos

Exercício

Considere a Equação de Torricelli:

$$v_F^2 = v_0^2 + 2ad$$

Um carro parte do repouso e, após percorrer 100m, já está a 200 km/h. Determine a aceleração média desse carro.





Algoritmos

- Uma sequência de ações executáveis.
 - Receita de bolo;
 - Instrução para montagem de equipamentos eletrônicos;
 - Instrução para uso de medicamentos;
 - etc.





Algoritmo = Receita

INGREDIENTES

Dados

- 2 xícaras de açúcar
- 3 xícaras de farinha de trigo
- 4 colheres de margarina bem cheias
- 3 ovos
- 1 1/2 xícara de leite aproximadamente
- 1 colher (sopa) de fermento em pó bem cheia

MODO DE PREPARO

O que fazer com os dados

- 1. Bata as claras em neve
- Reserve
- 3. Bata bem as gemas com a margarina e o açúcar
- 4. Acrescente o leite e farinha aos poucos sem parar de bater
- 5. Por último agregue as claras em neve e o fermento
- Coloque em forma grande de furo central untada e enfarinhada
- 7. Asse em forno médio, pré aquecido, por aproximadamente 40 minutos
- 8. Quando espetar um palito e sair limpo estará assado





Algoritmo = Receita

- Um bom cozinheiro, conhecendo apenas alguns poucos comandos (misture, bata, separe, cozinhe, repita etc.), pode produzir uma infinidade de pratos diferentes.
- O mesmo ocorre com a programação de computadores: com alguns poucos comandos, podemos fazer com que o computador faça praticamente qualquer coisa.
- Obs.: existem algumas coisas que [ainda] não conseguimos fazer através da computador.





Algoritmos e Programas

- Assim como na linguagem natural, para que o computador entenda o que queremos que ele faça, precisamos estabelecer regras de:
 - Sintaxe
 - Semântica
 - Lógica





Sintaxe

Ovus Fornum Letche

Ovos Forno Leite

 Palavras sintaticamente corretas ainda não significam uma "frase inteligível".





Semântica

 Bater duas xícaras de farinha sobre forno aquecido a 200 °C.

 Frase sintaticamente correta e completa (sujeito, verbo etc.), mas sem sentido semântico.





Lógica

- Considere a seguinte sequência de instruções:
 - Quebre dois ovos;
 - Mexa bem;
 - Adicione um litro de gasolina;
 - Asse a 350 °C por 45 minutos;
 - Adicione três xícaras de farinha.





Sintaxe, Semântica e Lógica

- Os erros de sintaxe são facilmente identificados pela ferramentas de programação (compiladores e interpretadores)
- Alguns erros de semântica também podem ser identificados automaticamente.
- Os erros de lógica são muito mais difíceis de identificar. Em geral isso depende inteiramente do programador. São os erros que produzem as consequências mais imprevisíveis.





Tipos de Dados Simples em Python

Texto (strings)

```
'Digite um número: '
'José da Silva'
'123'
"Mauro's Pizza"
'M'
```





Tipos de Dados Simples em Python

- Números
 - Valores inteiros

123 62000 -30

Valores reais ou de "ponto flutuante"

12.3 -0.145 1.52E+07

 $^{-1}$,52×10⁷





Tipos de Dados Simples em Python

Valores lógicos (booleanos)

False (Falso)

True (Verdadeiro)





Entrada e Saída Básica em Python

- Nesta disciplina: programas simples que usam interface de texto.
- Comando print: permite enviarmos qualquer informação (texto, números etc.) para a <u>tela</u>.





Escrevendo na Tela

```
print('Carlos da Silva', 'casou-se.')
print('Carlos da Silva', 'casou-se.', sep=':')
print('Carlos da Silva\ncasou-se')
print('Carlos da Silva', end=' ');
print('casou-se com Maria')
print('2 / 3 = ', 2/3)
```





Exemplos de Erro de Sintaxe

```
Print('Carlos da Silva')
print('Carlos da Silva'
print( Carlos da Silva )
```





Exemplo de Erro de Semântica

```
'Carlos' + 5
```

'a' - 'b'

Obs.: 'Carlos' * 5 funciona em Python!





Exemplo de Erro de Lógica

Do exercício anterior (Torricelli):

$$a = \frac{200^2}{2 \times 100}$$

• Em Python:

