

INTRODUÇÃO AOS SISTEMAS LÓGICOS

ÁREAS DE COMPUTAÇÃO

PROF. ANDRÉ MONTEVECCHI

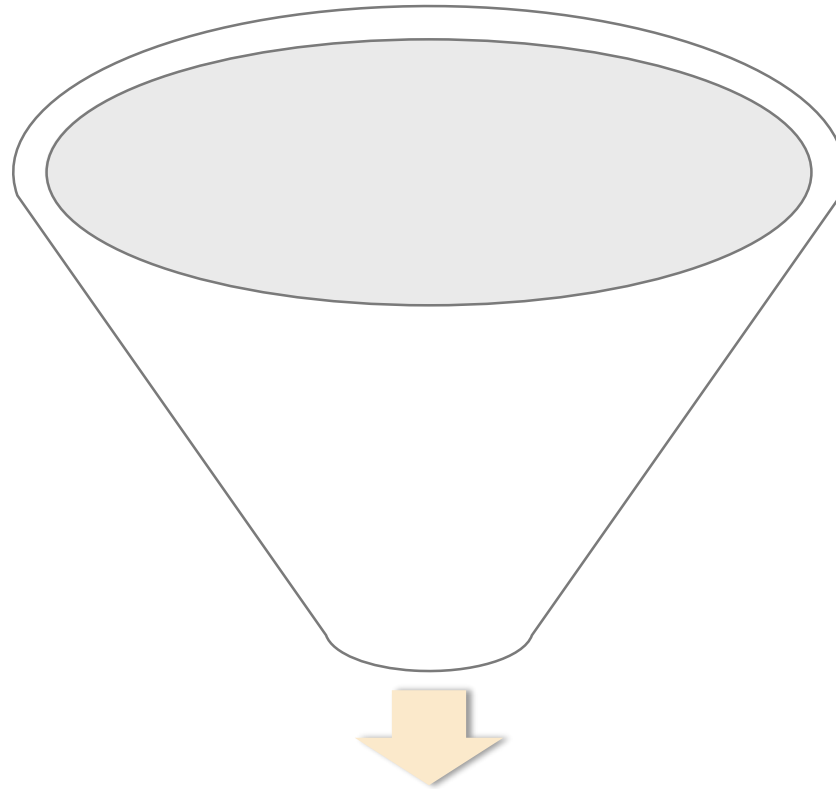
PROFA. ANNA TOSTES

SUMÁRIO

Os Pilares da Computação

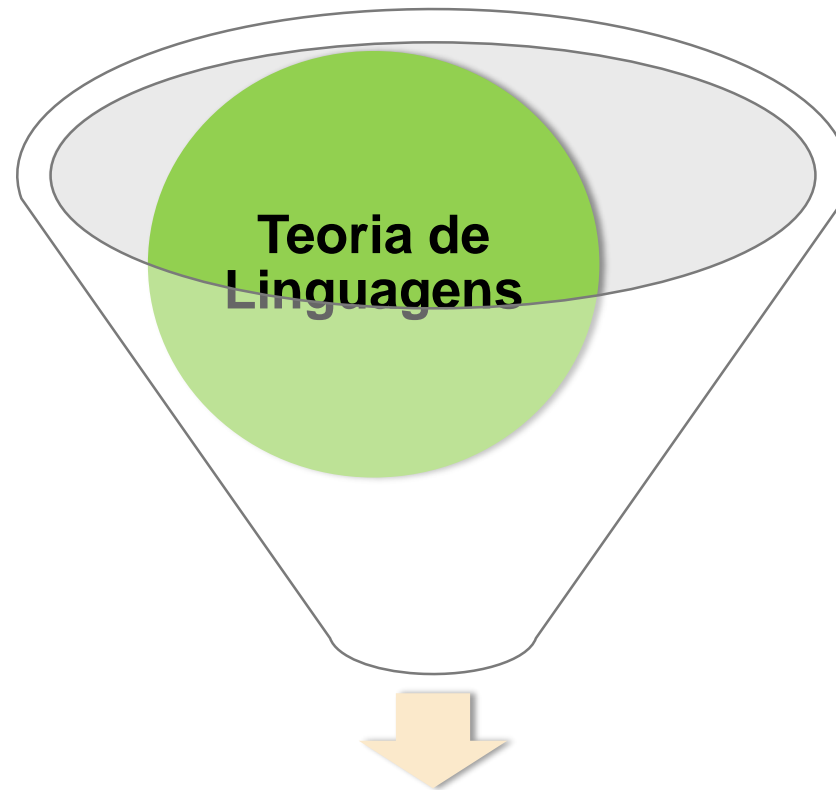
- Algoritmos
- Teoria de Linguagens
- Arquitetura

OS PILARES DA COMPUTAÇÃO



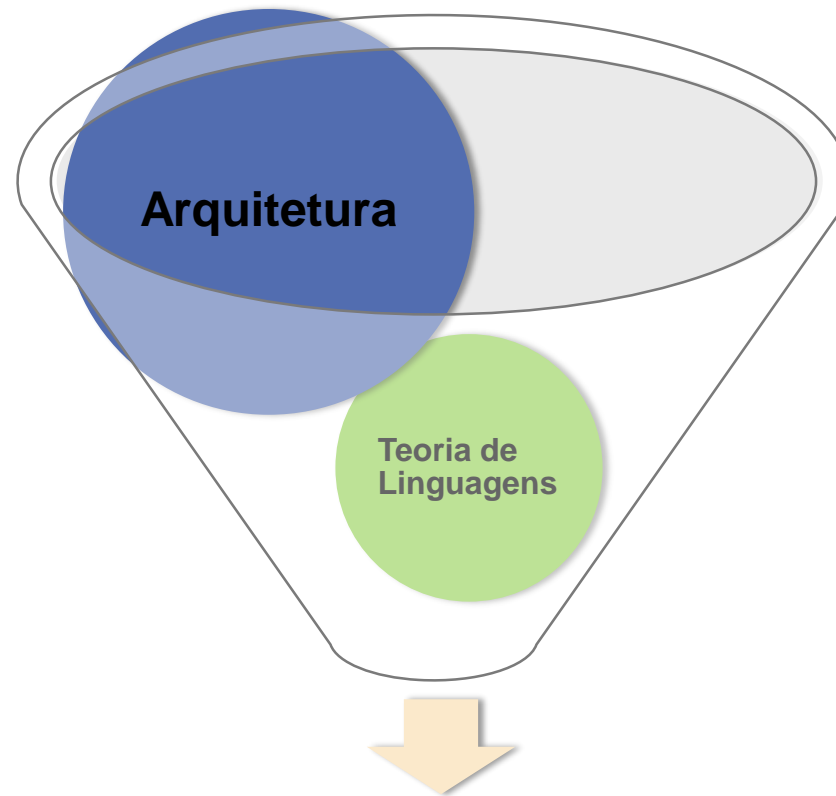
Ciência da Computação

OS PILARES DA COMPUTAÇÃO



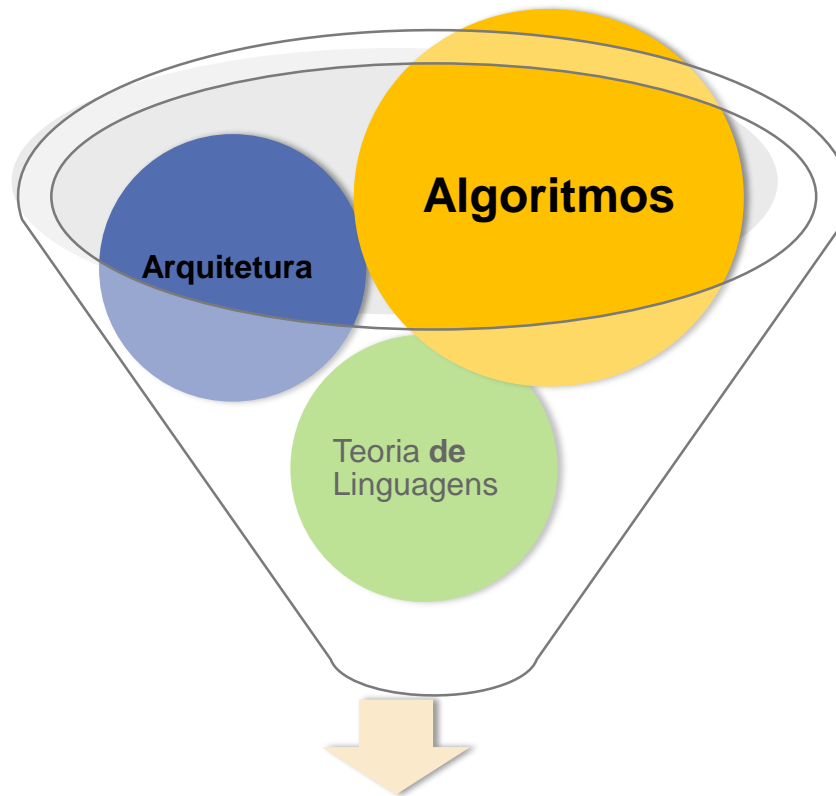
Ciência da Computação

OS PILARES DA COMPUTAÇÃO



Ciência da Computação

OS PILARES DA COMPUTAÇÃO



Ciência da Computação

OS PILARES DA COMPUTAÇÃO

Código fonte

- Criar o algoritmo
- Escrever em uma linguagem de programação

ALGORITMO

Tradução

- Compilação
- Interpretação
- Híbrido

TEORIA DE LINGUAGENS

Executável

- Programa em linguagem de máquina
- Windows: .EXE

ARQUITETURA

ALGORITMOS

No século VII, o matemático indiano *Brahmagupta* explicou pela primeira vez

- Sistema de **numeração** hindu-arábico
- Conceito de **zero** (0)

ALGORITMOS

Em 825, o matemático persa Al-Khowarizmi (Algoritmus, em latim) escreveu o livro *Kitab al jabr w'al muqabala* (“Regras de Restauração e Redução”)

- O nome **algoritmo** vem do seu nome
- Responsável pela difusão do sistema de numeração hindu-arábico no Oriente Médio e Europa
- Desenvolvimento dos primeiros **procedimentos** formalizados

ALGORITMOS

No século XII, houve uma tradução do mesmo livro para o latim:

- *Algoritmi de numero Indorum*

Nesses livros foram apresentados conceitos para **definir sequências de passos para completar tarefas**

- Como aplicações de aritmética e álgebra

ALGORITMOS

Componentes **básicos** de qualquer software

Aprender a programar

- **Abstração** de um programa
- Conhecimento **crucial** para quem deseja desenvolver um software de qualidade

ALGORITMOS

Conceitos

- Desenvolvimento da lógica
- Estruturas de dados
- Técnicas básicas de programação
- Principais problemas e algoritmos
- Análise de complexidade
- Grafos

ALGORITMOS

Exemplo

- Fazer um algoritmo para somar dois números e imprimir o resultado

ALGORITMO

INICIO

// ler dois valores inteiros e

// mostrar sua soma

CRIE as variáveis X_1 , X_2 e Z

$X_1 =$ **LEIA** um valor inteiro

$X_2 =$ **LEIA** um valor inteiro

$Z = X_1 + X_2$

IMPRIMA o valor de Z

FIM

ALGORITMOS

Testes

X_1	X_2	Z

ALGORITMO INICIO

// ler dois valores inteiros e
// mostrar sua soma

CRIE as variáveis X_1 , X_2 e Z

X_1 = LEIA um valor inteiro

X_2 = LEIA um valor inteiro

$Z = X_1 + X_2$

IMPRIMA o valor de Z

FIM

ALGORITMOS

Testes

X_1	X_2	Z
12	-	-

ALGORITMO

INICIO

// ler dois valores inteiros e
// mostrar sua soma
CRIE as variáveis X_1 , X_2 e Z

$X_1 =$ LEIA um valor inteiro

$X_2 =$ LEIA um valor inteiro

$Z = X_1 + X_2$

IMPRIMA o valor de Z

FIM

ALGORITMOS

Testes

X_1	X_2	Z
12	-	-
-	23	-

ALGORITMO

INICIO

// ler dois valores inteiros e
// mostrar sua soma
CRIE as variáveis X_1 , X_2 e Z

X_1 = LEIA um valor inteiro
 X_2 = LEIA um valor inteiro

$Z = X_1 + X_2$

IMPRIMA o valor de Z

FIM

ALGORITMOS

Testes

X_1	X_2	Z
12	-	-
-	23	-
-	-	35

ALGORITMO INICIO

// ler dois valores inteiros e
// mostrar sua soma
CRIE as variáveis X_1 , X_2 e Z

X_1 = LEIA um valor inteiro
 X_2 = LEIA um valor inteiro

Z = $X_1 + X_2$

IMPRIMA o valor de Z

FIM

ALGORITMOS

Testes

X_1	X_2	Z
12	-	-
-	23	-
-	-	35

35

ALGORITMO INICIO

// ler dois valores inteiros e
// mostrar sua soma
CRIE as variáveis X_1 , X_2 e Z

X_1 = LEIA um valor inteiro
 X_2 = LEIA um valor inteiro

$Z = X_1 + X_2$

IMPRIMA o valor de Z

FIM

ALGORITMOS

Estruturas de dados

- Variáveis
- Vetores (*array*)
- Listas encadeadas
- Pilhas e Filas
- Árvores
- Hash
- Heap

ALGORITMOS

Técnicas básicas de programação

- Indução
- Recursividade
- Divisão e conquista
- Programação dinâmica
- Algoritmos gulosos

ALGORITMOS

Principais problemas e algoritmos

- Ordenação
- Pesquisa de dados
- Pesquisa digital
- Problema do caixeiro-viajante
- Problema da mochila
- Problema da grade de horários

ALGORITMOS

Análise de complexidade

ALGORITMO

INICIO

// ler dois valores inteiros e
// mostrar sua soma
CRIE as variáveis X_1 , X_2 e Z

X_1 = **LEIA** um valor inteiro
 X_2 = **LEIA** um valor inteiro

Z = $X_1 + X_2$

IMPRIMA o valor de Z

FIM

1

1

1

1

1

Total: 5 instruções = $O(1)$



ALGORITMOS

Análise de complexidade

ALGORITMO

INICIO

CRIE a variável X

X = 0

ENQUANTO (X < 10), FAÇA

IMPRIMA o valor de Z

X = X + 1

FIM

FIM

1

1

n vezes

1

1

Total: $2 + n \cdot (2) = O(n)$

TEORIA DE LINGUAGENS

Linguagem

- Todo sistema de signos que serve como **meio de comunicação**

Definição da linguagem

- Léxica
- Sintática
- Semântica

Exemplos:

- JAVA, C, C++, PASCAL, COBOL, FORTRAN

TEORIA DE LINGUAGENS

Conteúdo

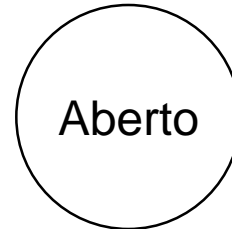
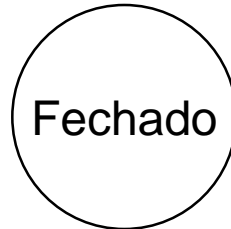
- Máquina de estados
- Autômatos
 - Determinísticos
 - Não determinísticos
- Máquinas de Turing
- Gramática
- Tipos de linguagens

MÁQUINAS DE ESTADOS FINITOS

Exemplos 1: lâmpada

- Dois estados: acesa ou apagada
- Duas ações: acender ou apagar

Exemplos 2: a abertura/fechamento automático de uma porta

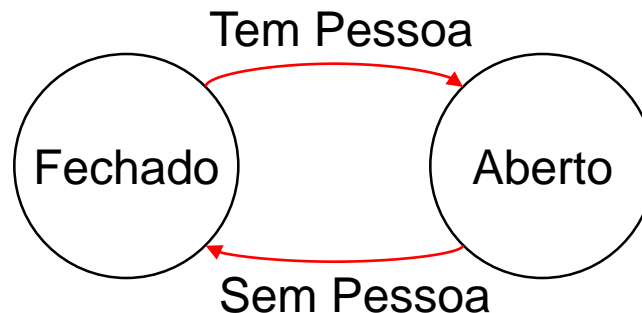


MÁQUINAS DE ESTADOS FINITOS

Exemplos 1: lâmpada

- Dois estados: acesa ou apagada
- Duas ações: acender ou apagar

Exemplos 2: a abertura/fechamento automático de uma porta

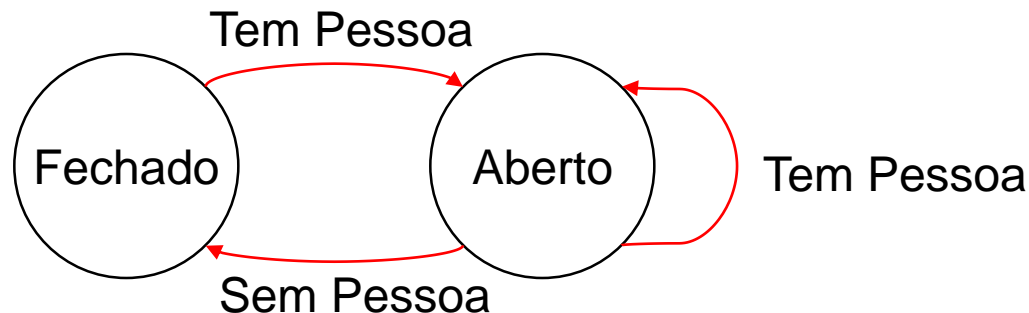


MÁQUINAS DE ESTADOS FINITOS

Exemplos 1: lâmpada

- Dois estados: acesa ou apagada
- Duas ações: acender ou apagar

Exemplos 2: a abertura/fechamento automático de uma porta

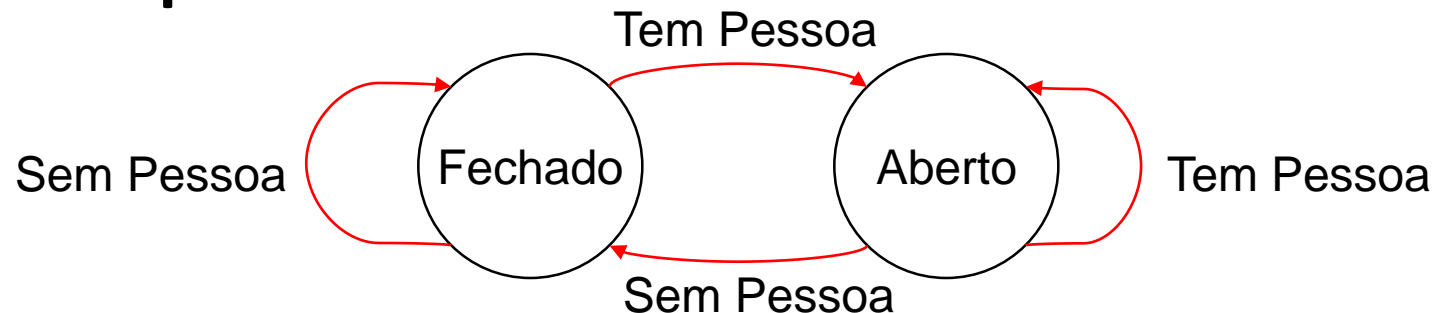


MÁQUINAS DE ESTADOS FINITOS

Exemplos 1: lâmpada

- Dois estados: acesa ou apagada
- Duas ações: acender ou apagar

Exemplos 2: a abertura/fechamento automático de uma porta



ARQUITETURA DE COMPUTADORES

Como projetar as **partes** de um computador visíveis para o programador?

Como **funciona** um computador?

- Processador, memória primária e secundária

ARQUITETURA DE COMPUTADORES

Como os **dados** são **representados**?

- Sistema decimal e binário
- Números inteiros
- Números reais (ponto flutuante)
- Caracteres

Máquina de von Neumann ← Lembraram?

ARQUITETURA DE COMPUTADORES

Saber **como construir** um circuito eletrônico digital (circuitos lógicos)

- Placa mãe, placa de vídeo, placa de rede

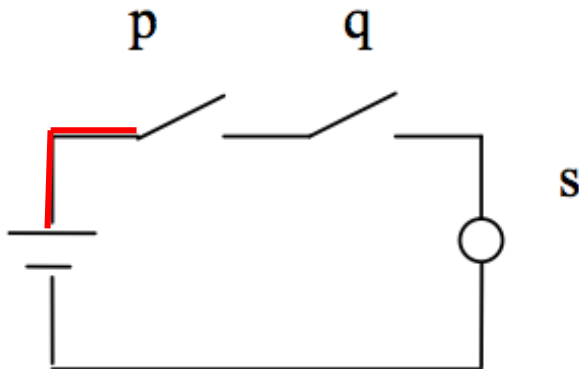
Vantagens de se saber arquitetura para um programador?

- Saber como aproveitar melhor os recursos
- Ter habilidade de otimizar programas
- Programação sequencial X paralela

ARQUITETURA DE COMPUTADORES

Exemplos:

- Analogia com circuitos elétricos:
 - Lâmpada s está acesa ou não?
 - p e q são portas

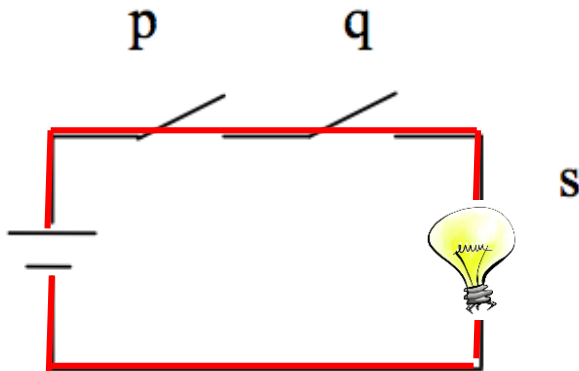


Circuito série (AND)
(Conjunção)

ARQUITETURA DE COMPUTADORES

Exemplos:

- Analogia com circuitos elétricos:
 - Lâmpada s está acesa ou não?
 - p e q são portas

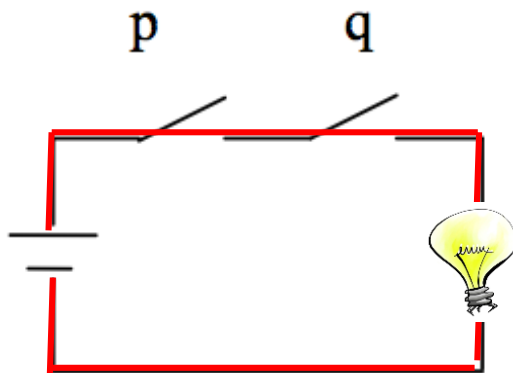


Circuito série (AND)
(Conjunção)

ARQUITETURA DE COMPUTADORES

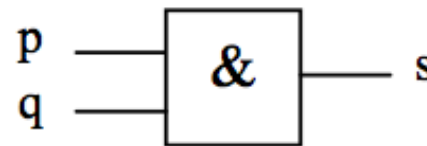
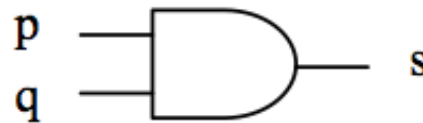
Exemplos:

- Analogia com circuitos elétricos:
 - Lâmpada s está acesa ou não?
 - p e q são portas



Circuito série (AND)
(Conjunção)

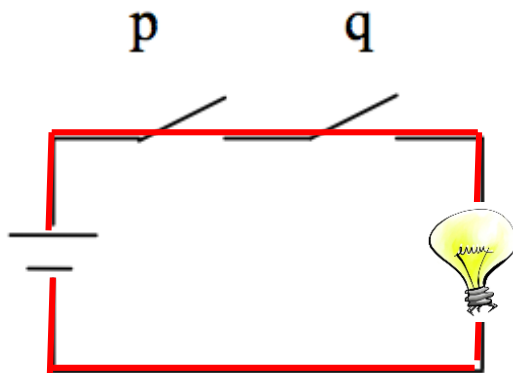
Porta AND (E)



ARQUITETURA DE COMPUTADORES

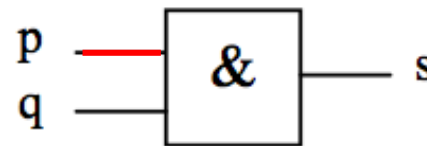
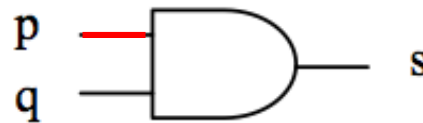
Exemplos:

- Analogia com circuitos elétricos:
 - Lâmpada s está acesa ou não?
 - p e q são portas



Circuito série (AND)
(Conjunção)

Porta AND (E)



E AS POSSIBILIDADES?

Redes de **sensores**

- Detectar fogo na floresta amazônica

Computação **móvel**

- Redes de celulares (3G e 4G)

Computação **ubíqua**

- Supermercado do futuro
- Casa do futuro

EXERCÍCIO



02/02/2014

Prof. André Montevecchi / Profa. Anna Tostes

EXERCÍCIO

1. Identifique e agrupe as disciplinas do seu curso nos três pilares da computação

- Algoritmos
- Teoria de linguagens
- Arquitetura de computadores

- Email: andre.montevercchi@prof.unibh.br