

Introdução a algoritmos

Prof. Fernando Figueira
(adaptado do material do Prof. Rafael Beserra Gomes)

UFRN

Material compilado em 26 de agosto de 2025. Licença desta apresentação:



<http://creativecommons.org/licenses/>

Conceitos iniciais de algoritmos

Objetivo

Sequência ordenada e não ambígua de passos que levam à solução de um dado problema [TREMBLAY, 1979].

Conceitos iniciais de algoritmos

Exemplos:

- Como ir do IMD até o Natal Shopping?
- Tarefas do studio.code.org

- Quais são os passos possíveis no algoritmo?
- Existe um único algoritmo para finalizar cada etapa?

Algoritmos computacionais

Algoritmos computacionais

Algoritmos computacionais

Algoritmo computacional

Um algoritmo que pode ser traduzido em uma sequência de instruções que pode ser executado em um computador.

Algoritmos computacionais

- Os algoritmos serão expressos de acordo com o que o computador pode executar
- Vamos conhecer um pouco mais sobre o computador!

Sistemas computacionais

Introdução

Adquire dados

Armazena dados

Processa dados

Exibe dados

Introdução

Adquire dados

Obtemos informações do mundo através de **dispositivos de entrada**.

Armazena dados

Processa dados

Exibe dados

Introdução

Adquire dados ←

Armazena dados

Processa dados

Exibe dados

Obtemos informações do mundo através de **dispositivos de entrada**:

- Teclado (texto)
- Câmera (imagens)
- Scanner (imagens)
- Leitor de digitais (imagens)
- Microfone (som)
- Mouse (coordenada, botões)
- Eletrocardiógrafo (atividade elétrica no coração)
- Touch Screen

Introdução

Adquire dados

Armazena dados

← Através das memórias é possível armazenar dados.

Processa dados

Exibe dados

Introdução

Adquire dados

Armazena dados ←

Processa dados

Exibe dados

Através das memórias é possível armazenar dados:

Memória volátil

Os dados são perdidos quando a energia cessa

- Mem. RAM, mem. cache, registradores (memória principal)

Memória não volátil

- HD, cartão de memória, CD, DVD, Pen-Drive (memória secundária)

Introdução

Adquire dados

Armazena dados

Processa dados

Exibe dados

O processamento dos dados é usualmente realizado por um ou mais processadores (CPU).

O processador contém registradores e efetuam nestes operações básicas como: adição, subtração, multiplicação.

Introdução

Adquire dados

Armazena dados

Processa dados

Exibe dados

O computador utiliza **dispositivos de saída** para que esses dados sejam compreensíveis para o ser humano.

Introdução

Adquire dados

Armazena dados

Processa dados

Exibe dados



O computador utiliza **dispositivos de saída** para que esses dados sejam compreensíveis para o ser humano:

- Impressora
- Monitor
- Alto-falante

Sistema operacional

Sistema Operacional

O sistema operacional (SO) é um conjunto de softwares que controla os recursos do computador e oferece serviços básicos para qualquer aplicativo. Entre os sistemas operacionais mais conhecidos estão:

- Windows
- Linux (diversas distribuições como Ubuntu)
- Android (baseado em Linux)
- Mac OS

Sistema operacional

Programas

Programas (software) são sequência de instruções que podem ser executadas em um processador.

Exemplos:

Processador de texto Microsoft Word, OpenOffice Writer, Latex

Editor de imagens Adobe Photoshop, Gimp

Editor de vídeos Adobe After Effects, Sony Vegas

Científicos Matlab, Geogebra

Jogos Stunts, Sim City, Super Mario, Street Fighter

Animação 3D Studio, Maya, Blender, Adobe Flash

Navegadores Internet Explorer, Mozilla Firefox, Opera, Google Chrome

Programas mais simples cat, calculadora, grep, head, ipconfig (ifconfig)

Você é capaz de responder com que tipo de informações cada programa desse trabalha?

Memória

Como essas informações são representadas em um computador?

Memória

Os dados na memória são armazenados em sequências de bits

0	1	0	0	1	1	0	0	0	1	0	0	1	1	0	0	0	1	0	0	1	1	0	0	0	1	0	0	0	0	0
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

8 bits = 1 byte

1024 bytes = 1 kilobyte (kB)

1024 kilobytes = 1 megabyte (MB)

1024 megabytes = 1 gigabyte (GB)

Memória

Um tipo de dado corresponde a uma sequência de bits de tamanho fixo com uma interpretação específica.

número inteiro = 4 bytes

número real = 4 bytes padrão IEEE 754

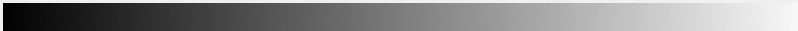
um caractere ASCII = 1 byte por exemplo: A = 65 (01000001)

um caractere Unicode = 4 bytes cerca de 107 mil caracteres

Memória

Imagem

Uma imagem digital é composta por pixels.



Preto (valor 0)

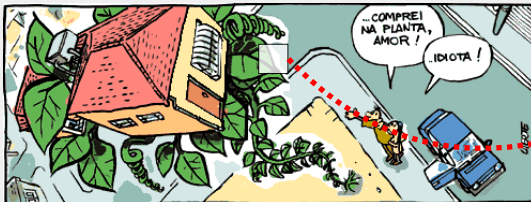
Cinza (valor 127)

Branco (valor 255)

Memória

Imagem colorida

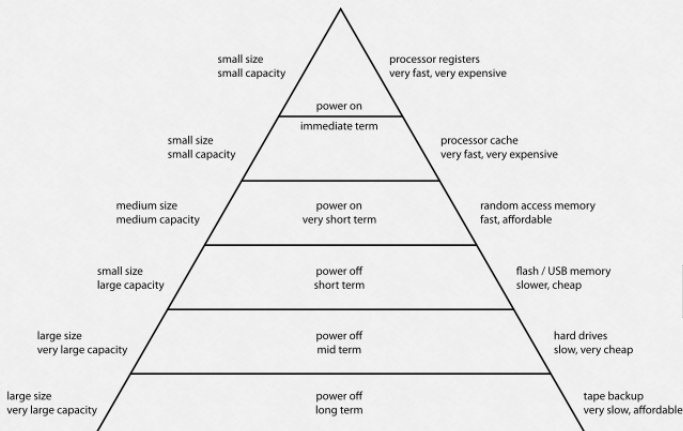
Em uma imagem colorida, cada pixel pode ser descrito como a composição de 3 canais: R (red) , G (green), B (blue)



Componente vermelho
Componente verde
Componente azul

Memória

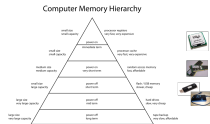
Computer Memory Hierarchy



Introdução a algoritmos

Sistemas computacionais

Memória



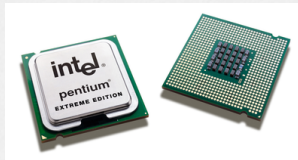
1. Note que quanto mais rápida a memória, mais cara. Por esse motivo, as memórias mais rápidas também possuem menor capacidade de armazenamento.

CPU

A CPU (unidade central de processamento) é um dos responsáveis pelo processamento dos dados em um computador.

A cada ciclo de clock uma instrução é executada na CPU.

O conjunto de instruções disponível depende do modelo de processador.

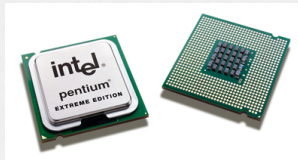


CPU

A CPU (unidade central de processamento) é um dos responsáveis pelo processamento dos dados em um computador.

A cada ciclo de clock uma instrução é executada na CPU.

O conjunto de instruções disponível depende do modelo de processador.



CPU

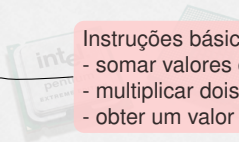
A CPU (unidade central de processamento) é um dos responsáveis pelo processamento dos dados em um computador.

A cada ciclo de clock uma instrução é executada na CPU.

O conjunto de instruções disponível depende do modelo de processador.

Instruções básicas como:

- somar valores de dois endereços
- multiplicar dois números
- obter um valor da memória

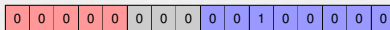


CPU

Quais são as instruções disponíveis para criar algoritmos no computador?

Linguagem/código de máquina

- Específica para cada arquitetura de computador
- Um exemplo fictício (processador de 16 bits):



- 5 bits: instrução
 - 3 bits: registrador
 - 8 bits: valor
- Por exemplo: 10110 é o código para colocar um valor em um determinado registrador. Os 3 seguintes bits especificam qual registrador. Os demais 8 bits representam o valor a ser armazenado no registrador.

CPU

- Quer saber como funciona o seu processador e o conjunto de instruções? Leia o manual!
<https://software.intel.com/en-us/articles/intel-sdm>
- Programar em linguagem de máquina é um trabalho árduo
- A **linguagem assembly** utiliza uma linguagem mais fácil para escrever programas
- Um *assembler* transforma um código em *assembly* para código de máquina

Introdução a algoritmos

└ Sistemas computacionais

└ CPU

- Quer saber como funciona o seu processador e o conjunto de instruções? Leia o manual!
<https://software.intel.com/en-us/articles/intel-sdm>
- Programar em linguagem de máquina é um trabalho árduo
- A linguagem *assembly* utiliza uma linguagem mais fácil para escrever programas
- Um assembler transforma um código em *assembly* para código de máquina

1. Não vamos programar em linguagem de máquina nessa disciplina.
2. A memória principal armazena tanto dados do programa como os próprios programas
3. Geralmente o HD faz o papel de memória secundária. Geralmente os programas estão armazenados nessa memória, mas quando solicitamos sua execução, as instruções são transferidas para a memória principal (memória RAM).

Linguagens de Programação

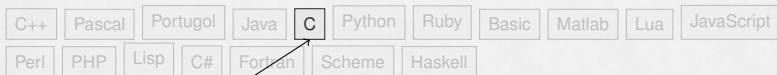
Conceitos

As linguagens de programação facilitam a programação de computadores.

Conceitos



Conceitos



C é uma linguagem compilada

Introdução a algoritmos

└ Linguagens de Programação

└ Conceitos



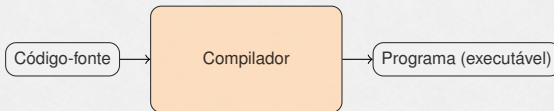
1. Cada linguagem de programação possui suas vantagens e desvantagens e, portanto, a linguagem mais indicada depende do contexto da aplicação.
2. A linguagem C apesar de bastante antiga é ainda uma das mais populares e é excelente para aprender programação.

Sintaxe e semântica

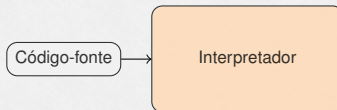
- As linguagens de programação possuem regras de sintaxe e semântica!
- Um algoritmo escrito nessa linguagem de programação só será aceito se estiver completamente de acordo com as regras gramaticais

Linguagens compiladas e interpretadas

Nas **linguagens compiladas**, programas específicos chamados compiladores convertem um código escrito em uma linguagem (código-fonte) para uma sequência de instruções do processador.



Nas **linguagens interpretadas**, um programa interpretador interpreta um código escrito em uma linguagem e executa a sequência de instruções correspondentes.



Introdução a algoritmos

└ Linguagens de Programação

└ Linguagens compiladas e interpretadas

Nas **Linguagens compiladas**, programas específicos chamados compiladores convertem um código escrito em uma linguagem (código-fonte) para uma sequência de instruções do processador.



Nas **Linguagens interpretadas**, um programa interpretador interpreta um código escrito em uma linguagem e executa a sequência de instruções correspondentes.



1. As linguagens interpretadas tendem a ser mais lentas pois várias instruções são executadas com a finalidade de interpretar o código-fonte durante a execução do programa.
2. As linguagens interpretadas têm a grande vantagem de facilitar a portabilidade entre sistemas diferentes.

Linguagens compiladas e interpretadas

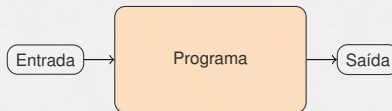
Neste curso utilizaremos como dispositivo de entrada um teclado e como dispositivo de saída o monitor através de um terminal do sistema em modo texto.

Dispositivo de entrada: ☐ Teclado ☐ Scanner ☐ Câmera ☐ Mouse

Dispositivo de saída: ☐ Monitor (terminal) ☐ Impressora ☐ Alto-falante

Entrada e saída de um programa

Cada programa a ser desenvolvido nesse curso terá uma **entrada** fornecido pelo usuário através do teclado e uma **saída** visualizável no monitor.



Exemplo:

Um programa que calcula a média de um aluno dadas as notas das 3 unidades:

Entrada: nota 1, nota 2 e nota 3 (números reais)

Saída: média parcial (número real)

Representação dos algoritmos

Formas para representar um algoritmo:

- Narrativa (linguagem natural)
- Fluxograma
- Pseudo-código

Narrativa

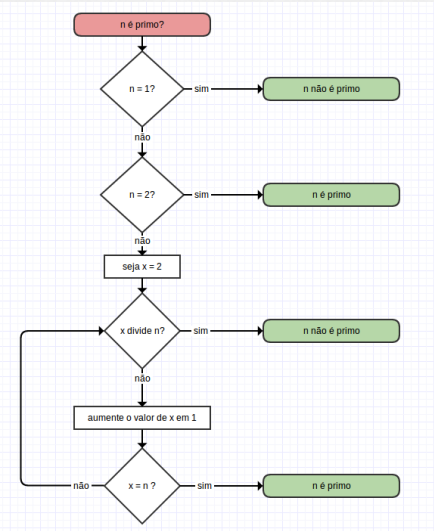
Narrativa:

- Descrição textual em linguagem natural
- Útil para ideias iniciais (lembre-se de que o objetivo é ter um algoritmo escrito em uma linguagem de programação)
- Problema: determine se um número inteiro n maior que 0 é primo
- Solução 1: o número n é primo se ele é diferente de 1 e divisível somente por 1 e por ele próprio
- Solução 2: desconsiderando os casos triviais $n = 1$ e $n = 2$, verifique se há algum número entre 2 e $n-1$ que divide n . Se não houver, então é primo!

Fluxograma

Fluxograma

- Representação gráfica muito boa para visualizar o fluxo lógico



Introdução a algoritmos

Representação dos algoritmos

Fluxograma



1. A representação em fluxograma não é muito comum, pois ocupa muito espaço e é difícil produzi-la.

Pseudo-código

Pseudo-código

- Representação mais próxima de uma linguagem de programação
- Flexibilidade em relação às regras gramaticais da linguagem e uso de linguagem natural

```
se n for 1 então conclua que n não é primo
se n for 2 então conclua que n é primo
caso contrário para cada um dos números de 2 até n-1 faça o seguinte:
    - verifique se esse número divide o número n
    - se dividir então conclua que n não é primo
conclua que n é primo
```