

# Introdução à Programação de Computadores

• • •

Natan Sanches

Bacharelado em Ciências da Computação - ICMC/USP

# Conteúdo Programado:

- Introdução e definição de um computador
- Arquitetura primitiva de Von Neumann
- A diferença entre memória Heap e Stack
- O que é um compilador?
- Linguagem compilada x Linguagem interpretada
- Introdução à manipulação de dados com C

# Introdução e definição de um computador

O que torna computadores tão especiais? Ou melhor, o que classifica um dispositivo eletrônico como computador?

# O que é um computador?

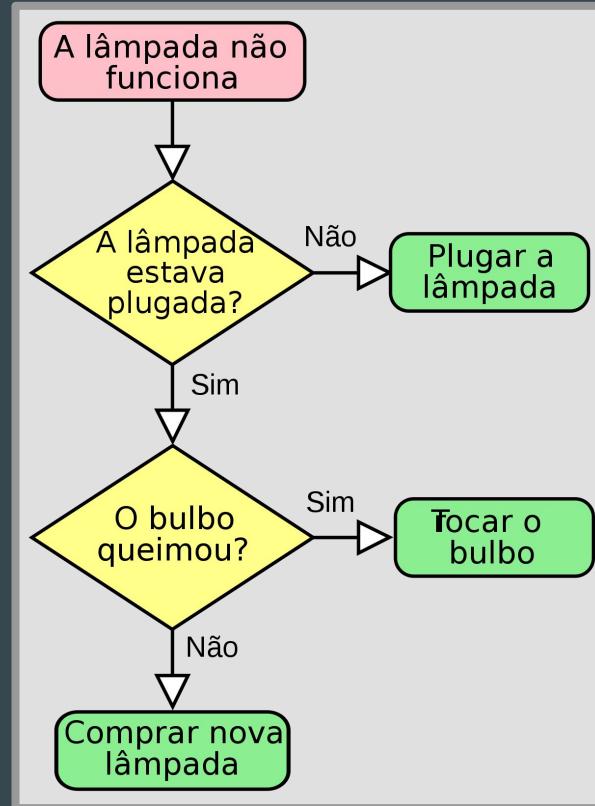
- Um computador é um conjunto de elementos eletrônicos (máquina) capaz de executar variados tipos de algoritmos e tratamento de informações (processamento de dados).
- Uma máquina que pode possuir inúmeros atributos, dentre eles o armazenamento de dados, processamento de dados, cálculo em grande escala, desenho industrial, etc.
- Trata-se de uma definição informal, pois a palavra “computador” possui um conceito matemático rigoroso por trás, utilizado em Teoria da Computação.
- Ícones da *Era da Informação*.

# O que é um algoritmo?

- Na ciência da computação, um algoritmo pode ser descrito como uma série de processos discretos e sequenciais, que visam solucionar um dado problema dado de início.
- Um programa de computador é, em sua essência, um algoritmo que diz ao computador que passos realizar em dada situação.

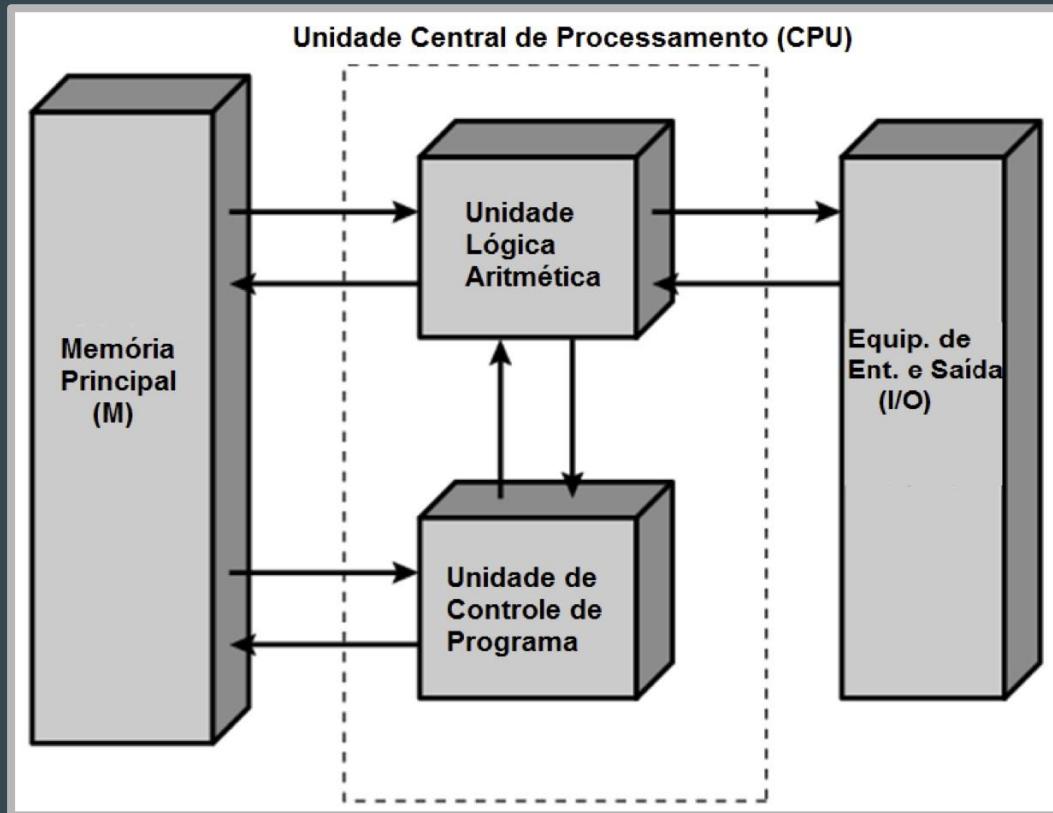
# O que é um algoritmo?

No fluxograma ao lado, a troca de lâmpadas descrita pode ser vista como um algoritmo. Ele também poderia, analogamente, ser escrito como um programa de computador.



# A arquitetura de Von Neumann

Desde a elaboração teórica, até a construção de computadores modernos



## Arquitetura de Von Neumann

Proposição de que os computadores fossem organizados em diversos componentes, onde cada um executa uma ação diferente.

# Importâncias da Arquitetura de Von Neumann

- Introdução da utilização da *base binária* para o trabalho interno do hardware com números.
- Ancestral das arquiteturas modernas, que marcou o surgimento do que conhecemos hoje como computadores.
- Todos os componentes que formam o funcionamento do computador se tornam alinhados com a CPU, evitam-se problemas como dessincronização.

Dúvidas?

# A segmentação da Memória RAM

A memória RAM não é homogênea: isto é, possui segmentos cujo sistema operacional delimita para seu funcionamento harmônico com o restante dos processos.

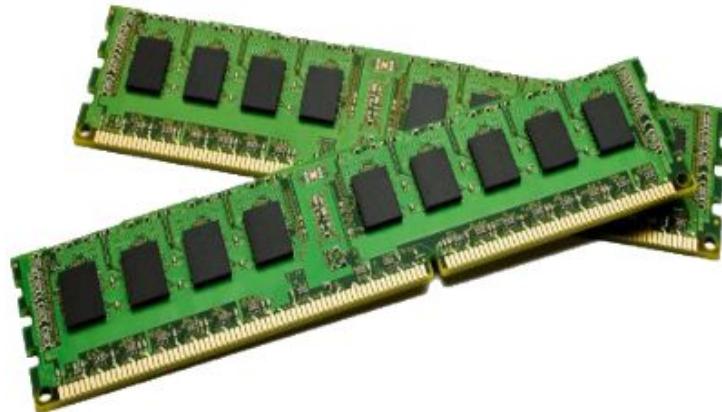
O que é memória?

# O que é memória?

- Memória é um dos componentes internos do computador que permitem o armazenamento de dados (em forma de bits).
- Suas principais características são: volatilidade, velocidade de acesso, capacidade de armazenamento e possibilidade de leitura, escrita ou ambos.
- A velocidade de acesso costuma ser uma grandeza inversamente proporcional à capacidade de armazenamento, uma vez que em memórias com menor capacidade é possível um acesso mais rápido dos dados internos que a compõem.

# Memória RAM

- Memória volátil
- Pouco armazenamento
- Permite leitura/escrita
- Acesso rápido e aleatório



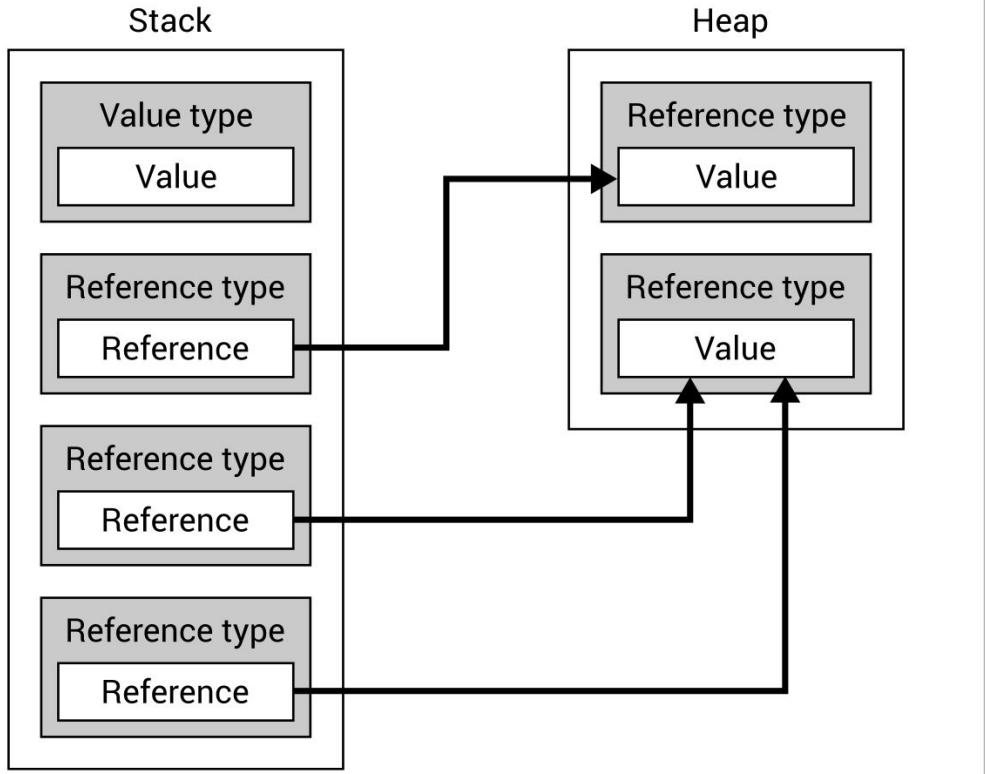
# Segmentação da Memória RAM

## Memória *Stack*

- Partição ‘limitada’ da memória RAM (normalmente, em torno de 4 Megabytes).
- Alocada pelo sistema operacional ao inicializar o programa.
- Armazena em formato de pilha.

## Memória *Heap*

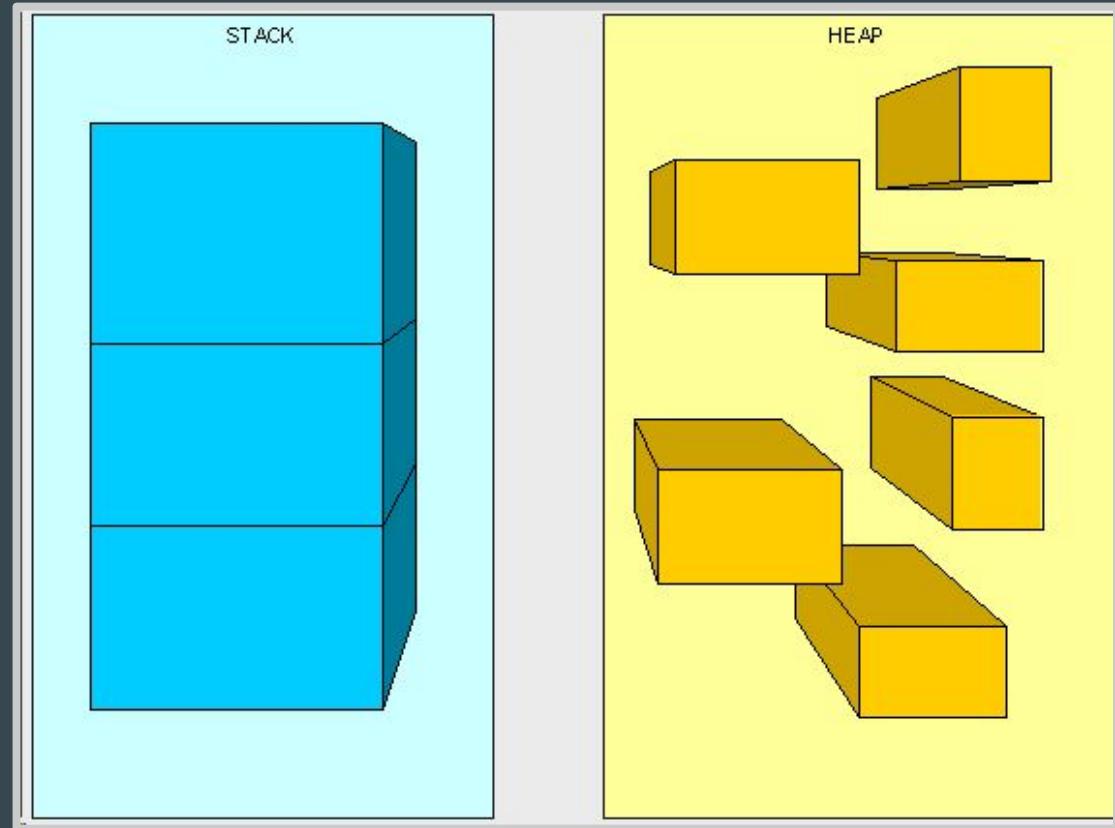
- Partição majoritária da memória RAM, externa à memória Stack.
- Alocada e administrada pelo desenvolvedor no desenvolvimento do código.
- Armazenamento livre.



## Integração das memórias *Heap* e *Stack*

Em C, a relação da memória Heap com a Stack é ponteada pela utilização de variáveis ponteiros (ou *pointers*).

Exemplificação do processo de organização de ambos os segmentos.



Diferenciação entre os segmentos de memória

# A interpretação do computador à programas

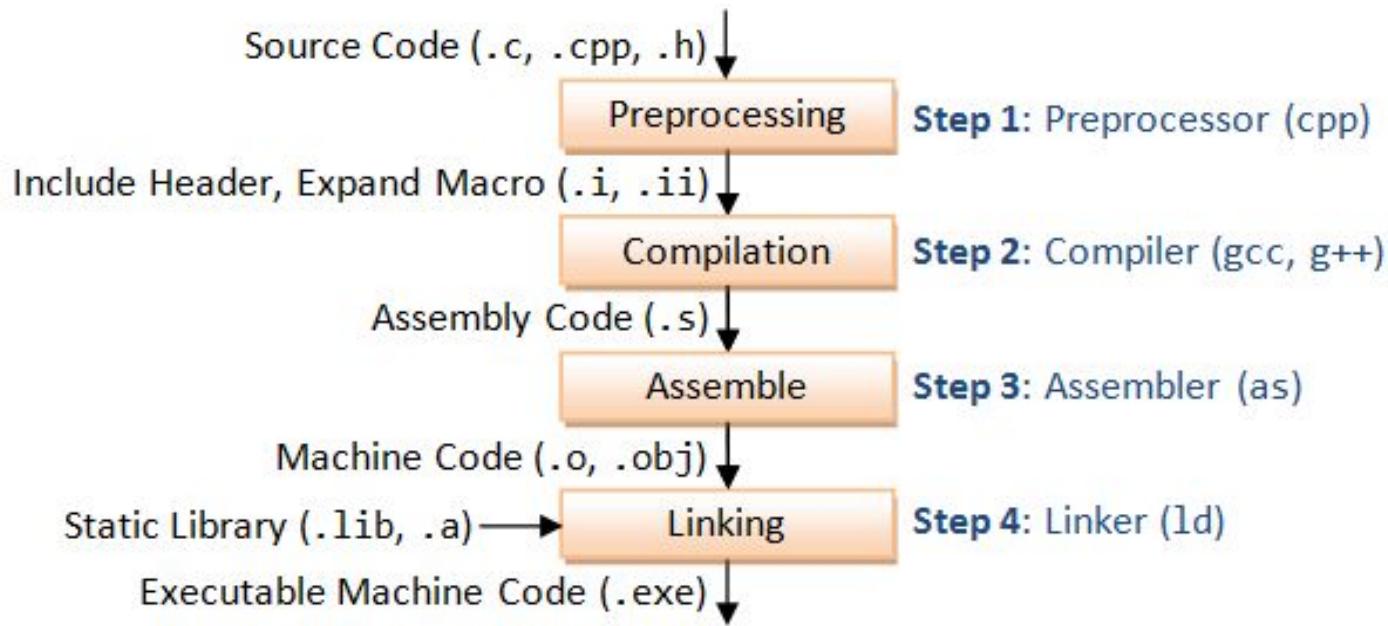
Como um computador reconhece as instruções de um código? É possível que um computador leia símbolos alfanuméricicos?

# ~ NÃO!

Não é possível para um computador ler símbolos alfanuméricos diretamente.

# O intermediário entre o hardware e o usuário

- O intermediário que permite a comunicação entre o hardware e o usuário se denomina **compilador**.
- Presente em todas as linguagens de programação que são *compiladas*.
- A função de um compilador é traduzir, a partir de uma linguagem de alto nível, o código-fonte de um arquivo para linguagem de máquina, que pode ser interpretada por uma CPU.
- Os compiladores também são escritos em linguagens de alto nível.



Processo de compilação de um código em C

# Linguagem compilada x Linguagem interpretada

## Linguagem compilada

- O compilador deve conhecer o tipo da variável antes da compilação.
- Todo o texto do código é traduzido para linguagem de máquina.
- Gera arquivos executáveis.

## Linguagem interpretada

- Possui a vantagem das variáveis não-tipadas.
- O código-fonte é passado por um interpretador e executado no próprio.
- Não necessita de compilação.

Dúvidas?

# Introdução à Programação em C