



UNIVERSIDADE FEDERAL DO PIAUÍ – UFPI  
Campus Senador Helvídio Nunes de Barros - CSHNB  
Curso Bacharelado em Sistemas de Informação

# ALGORITMOS E PROGRAMAÇÃO I

Glauber Dias Gonçalves  
ggoncalves@ufpi.edu.br

# CONTEÚDO

- Conceitos Básicos
  - Sistemas de computação
  - Programação de computadores

# SISTEMAS DE COMPUTAÇÃO

- Sistemas computacionais podem ser organizados em camadas:

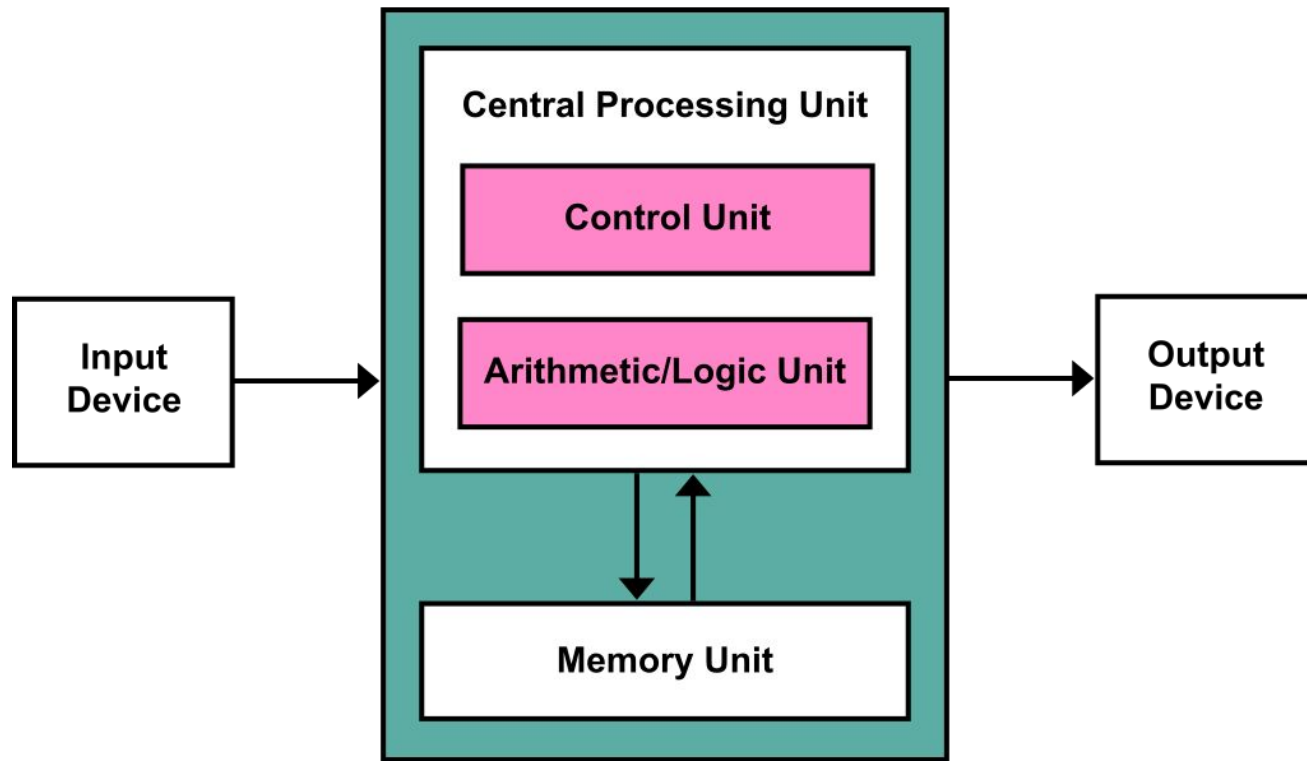


# HARDWARE



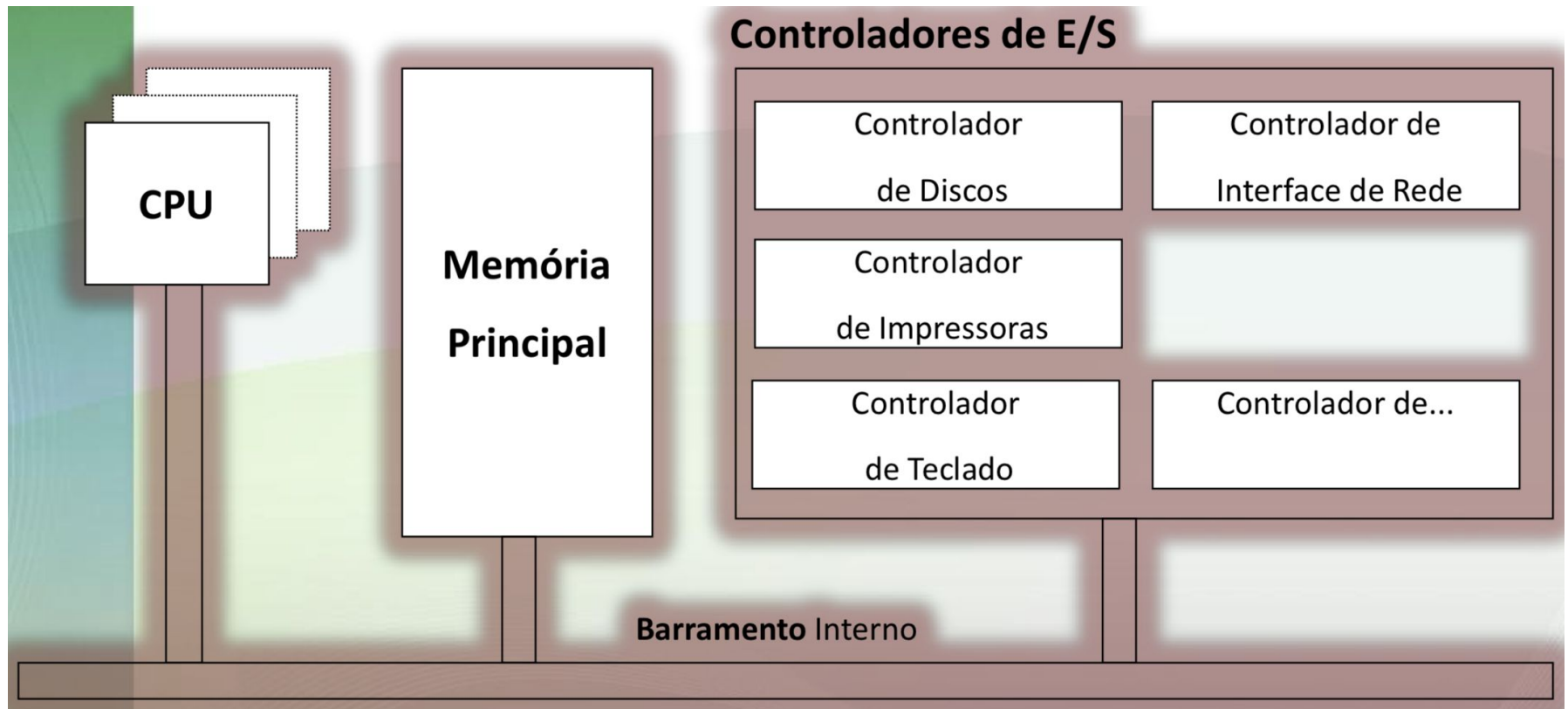
# HARDWARE

- Arquitetura de Von Neumann (visão lógica)



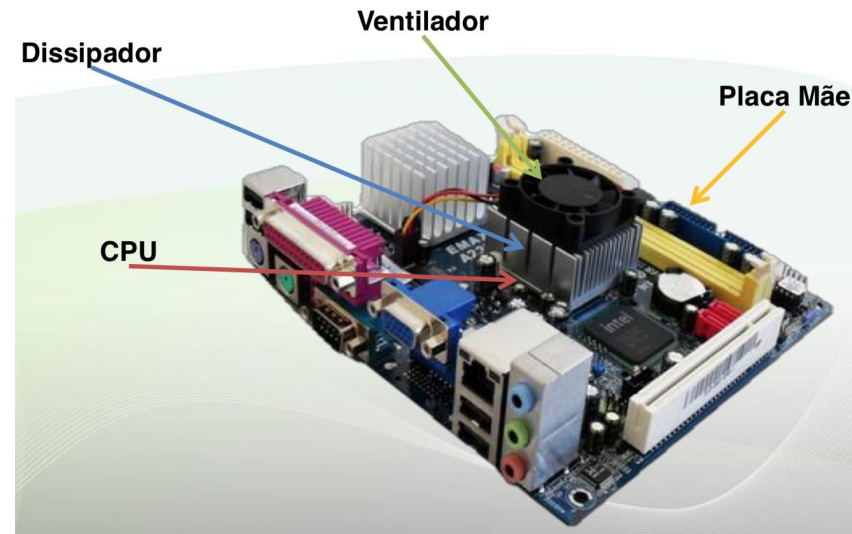
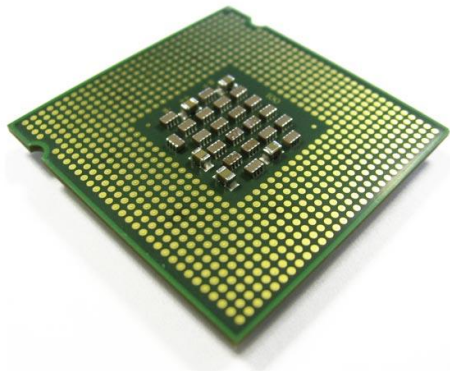
# HARDWARE

- Arquitetura de Von Neumann (implementação)



# HARDWARE

- Unidade Central de Processamento (CPU)
  - Interpreta e executa instruções de programas;
  - Auxiliado pela memória e dispositivos de E/S



# HARDWARE

- Memória: armazena dados dos programas em execução na CPU:
  - Primária: maior rapidez, custo e proximidade da CPU, por exemplo, RAM (random access memory).
  - Secundária: menor rapidez, custo e proximidade da CPU, por exemplo fitas magnéticas, discos magnéticos ou ópticos.

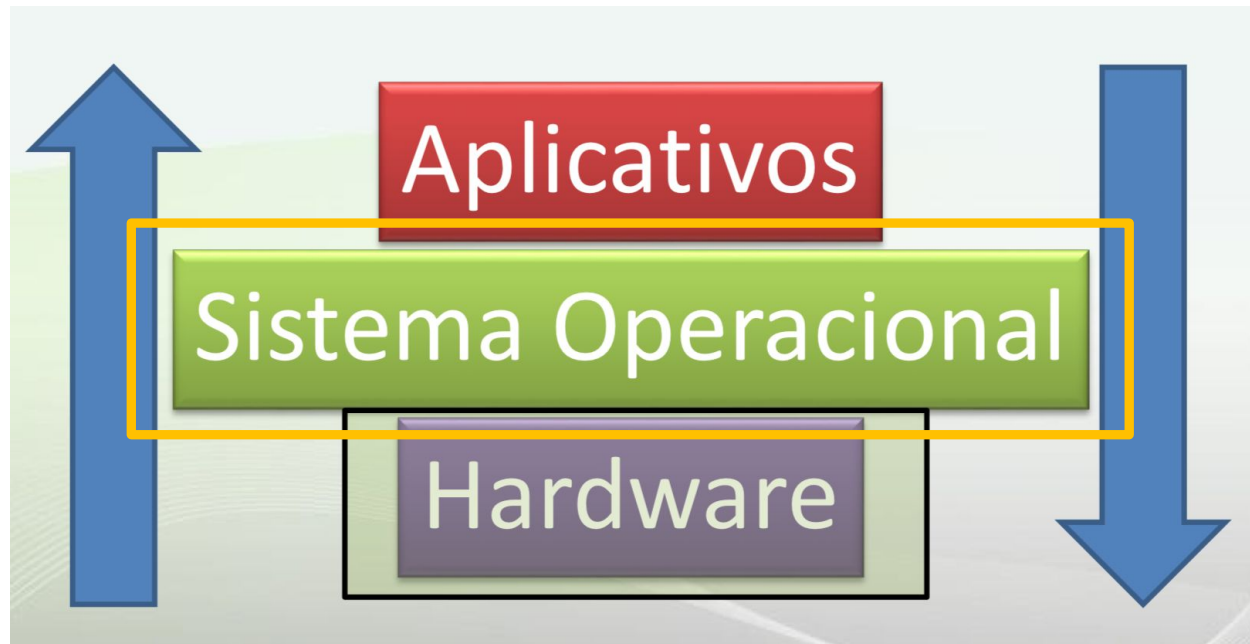


# HARDWARE

- Dispositivos de Entrada e Saída (E/S):
  - As unidades de entrada de dados realizam a interface entre homem e máquina, por exemplo, o teclado, o mouse, touch screen, etc.
  - As Unidades de saída de dados devolvem a informação processada entre a máquina e o homem, por exemplo, a impressora, o vídeo.

# SISTEMAS DE COMPUTAÇÃO

- Sistemas computacionais podem ser divididos em camadas:



# SISTEMA OPERACIONAL (SO)

- Abstrair detalhes de operação do hardware
  - Utilização da CPU por múltiplas aplicações
  - Utilização dos dispositivos E/S (*driver*)
  - Ocupação de memória primária ou secundária, etc.
- Os primeiros computadores não tinham SO :(
- Objetivo: usuário focar em suas aplicações :)

# SISTEMA OPERACIONAL (SO)

- Quantos SOs você conhece?

# SISTEMA OPERACIONAL (SO)

- Quantos SOs você conhece?



MacOS



- Qual o seu(s) SO(s)? Por que?

# SISTEMAS DE COMPUTAÇÃO

- Sistemas computacionais podem ser divididos em camadas:



# APLICATIVOS

- Executa tarefas específicas como o processar texto, reprodução de áudio, etc.
- Desenvolvidos através de linguagens de programação
- Linguagens de programação também são aplicações (ferramentas de desenvolvimento!)

# APLICATIVOS

- Quais os seus aplicativos preferidos?



# APLICATIVOS

- Quais os seus aplicativos preferidos?



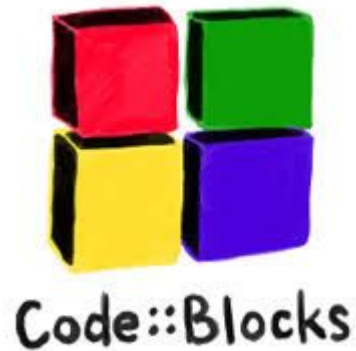
# APLICATIVOS

- E esses aqui? Quem utiliza com frequência?



# APLICATIVOS

- Que tal agora esses?



# CONTEÚDO

- Conceitos Básicos

- Sistemas de computação



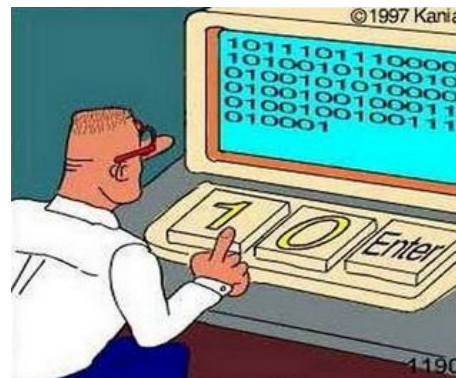
- **Programação de computadores**

# LINGUAGENS DE PROGRAMAÇÃO

- Linguagens de Baixo Nível:
  - mais próxima da máquina
  - pouco expressiva e legível
  - exemplo: Linguagem de Máquina
- Linguagens de Alto Nível:
  - mais próxima do ser humano;
  - alto poder de expressividade;
  - exemplo: C

# LINGUAGEM DE MÁQUINA

- Códigos binários
  - **ausência e presença** de energia.
- Muito difícil de ser compreendida e utilizada pelos seres humanos.



# LINGUAGEM DE MÁQUINA

- Formada por um conjunto de **bits**
- Um bit é representado numericamente por:
  - 0 (zero) ou 1 (um).
  - Exemplo: **0100 0001** (letra **A**)
- A CPU processa conjunto de bits

# LINGUAGEM DE MÁQUINA

- Exemplo: 1011000001100001 ?

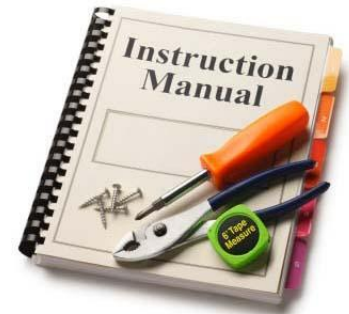
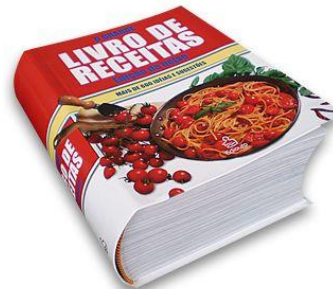


# LINGUAGEM DE MÁQUINA

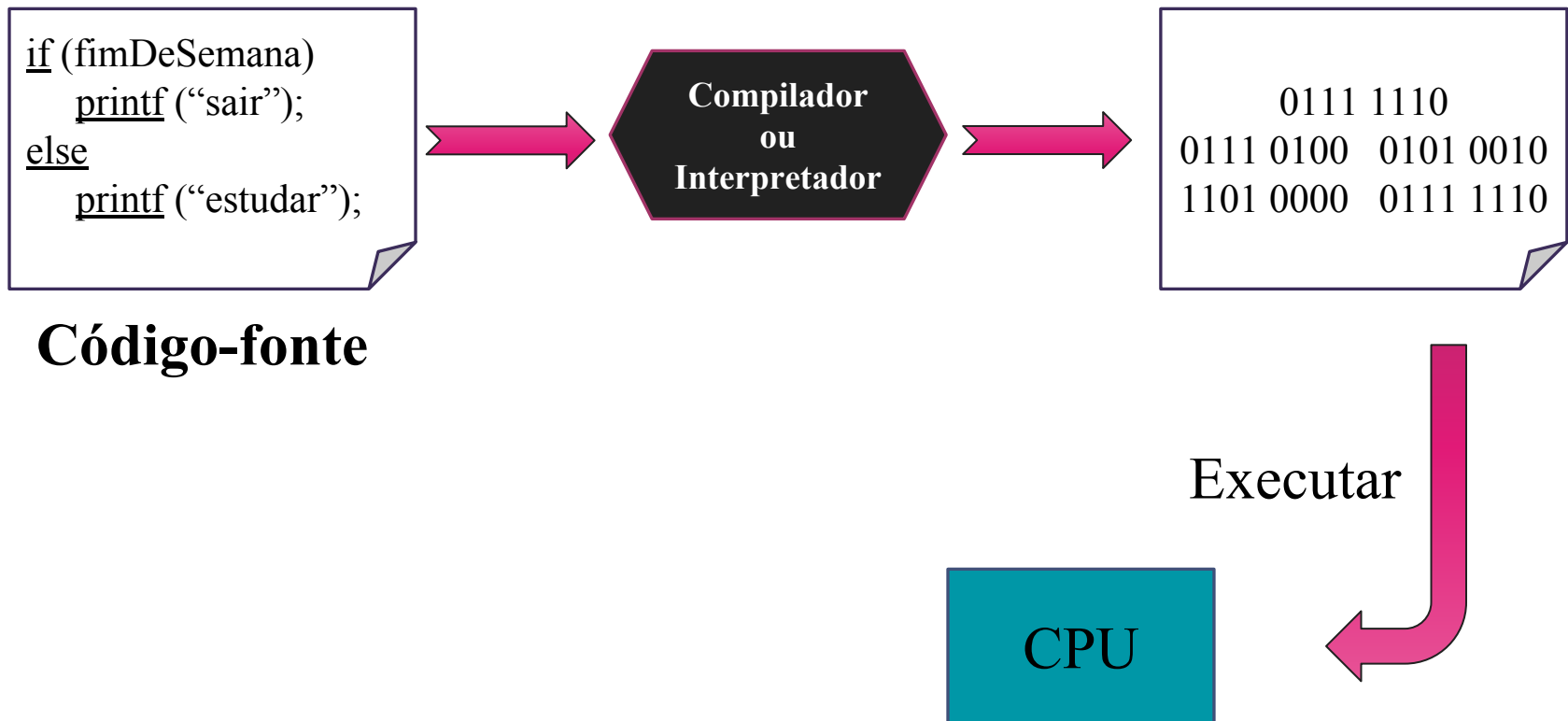
- Exemplo: 1000 000010 100000
  - Código de operação (soma), valor e registrador
- Outro exemplo: 1001 000001 100001
  - Código de operação (mov), valor e registrador
- **Fácil, não é?**
  - Memorizar vários códigos de operações:
    - Soma, subtração, multiplicação, carregamento de memória (mov)
  - Memorizar endereços de memória e valores (binários)

# LINGUAGEM DE ALTO NÍVEL

- Melhor compreendida pelos seres humanos
- Utilizadas para codificar algoritmos
- Algoritmos: processos sistemático de resolver problemas
  - seqüência de passos (instruções ou comandos) finitos e bem definidos.



# LINGUAGEM DE ALTO NÍVEL



# LINGUAGEM DE ALTO NÍVEL

- Representações:
  - linguagem natural;
  - fluxogramas;
  - linguagem algorítmica.

# LINGUAGEM DE PROGRAMAÇÃO

- Representações:
  - **linguagem natural;**
  - fluxogramas;
  - linguagem algorítmica.

# ALGORITMO: FRITAR UM OVO

1. Pegar frigideira, ovo, óleo e sal;
2. Colocar óleo na frigideira;
3. Acender o fogo;
4. Colocar a frigideira no fogo;
5. Esperar o óleo esquentar;
6. Colocar o ovo na frigideira;
7. Retirar o ovo quando pronto.

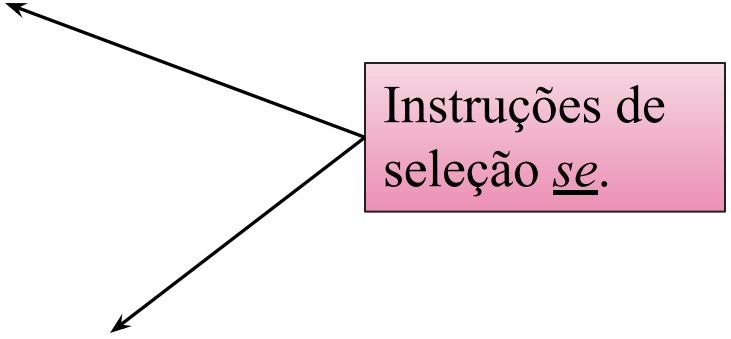


Seqüência de instruções.

# **ALGORITMO: TROCAR LÂMPADAS**

# ALGORITMO: TROCAR LÂMPADAS

1. **Se** (lâmpada estiver alta)  
pegar escada;
2. Pegar lâmpada nova;
3. **Se** (lâmpada estiver quente)  
pegar pano;
4. Tirar lâmpada queimada;
5. Colocar lâmpada nova.



Instruções de  
seleção se.



# **ALGORITMO: FAZER UM BOLO**

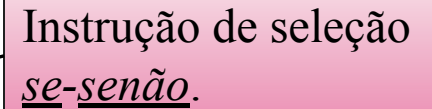
# ALGORITMO: FAZER UM BOLO

1. Pegar ingredientes;
2. Se (roupa for branca)  
colocar avental;
3. Se (tiver batedeira)  
bater ingredientes na batedeira;

## Senão

- bater ingredientes à mão;
4. Colocar a massa na forma;
5. Colocar forma no forno;
6. Esperar o bolo assar;
7. Retirar o bolo do forno

Instrução de seleção  
se-senão.



# **ALGORITMO: DESCASCAR BATATAS**

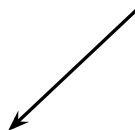
35

# ALGORITMO: DESCASCAR BATATAS

36

1. Pegar faca, bacia e batatas;
2. Se (roupa for branca)  
colocar avental;
3. **Enquanto** (houver batatas)  
descascar uma batata;
4. Limpar e guardar faca e bacia.

Instrução de repetição  
enquanto.



# ALGORITMO

- Deve-se analisar todas as possibilidades possíveis no enunciado do problema.



# SUMÁRIO

- Sistemas de computação:
  - Hardware - SO - Aplicações (software)
- Programação de computadores:
  - Linguagens de Programação
    - codifica algoritmos
    - baixo nível
    - alto nível
- Lista de atividades 01 (No Sigaa)