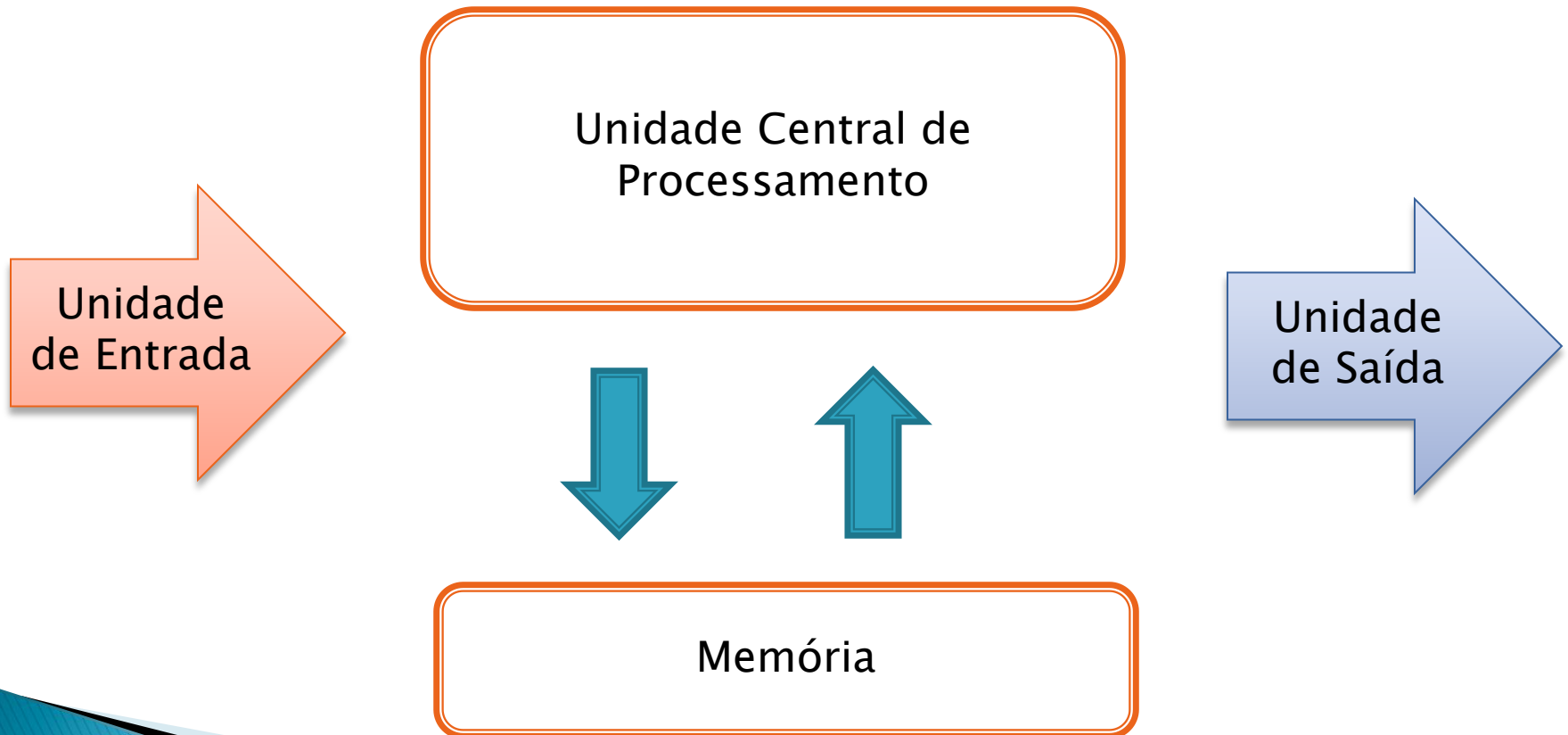


Introdução

Organização Básica de Computador



Unidade de Entrada

► Dispositivos:



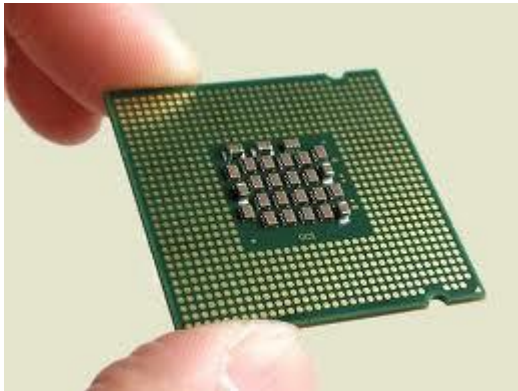
Unidade de Saída

► Dispositivos:



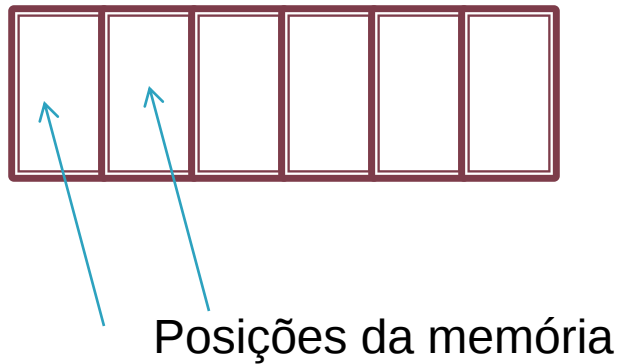
Unidade Central de Processamento

- ▶ Responsável por realizar todo o processamento necessário



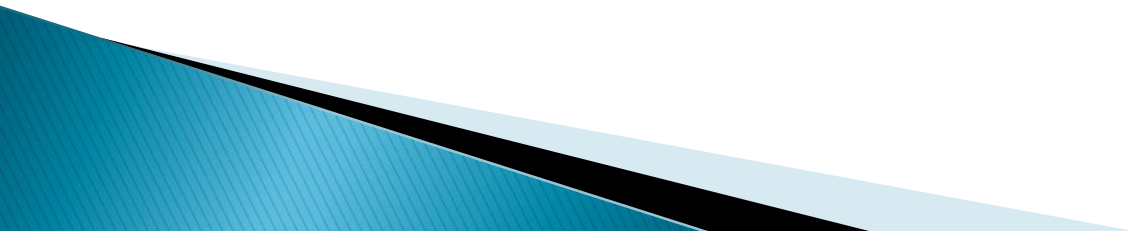
Memória

- ▶ Responsável por armazenar dados e informações que serão utilizados no processamento e nos programas



Linguagem de Máquina

Como podemos nos comunicar com os computadores?



Hardware Não entende Português, muito menos Espanhol!



Por favor haga la suma
de estos números: diez
e sesenta.



E Inglês!?

Please, make the sum of
these numbers: ten and
sixty.



Linguagem de Máquina

- ▶ Há necessidade de enviar comandos que os dispositivos entendam e que possam se comunicar entre si:
 - Unidade de Entrada com Memória
 - Unidade de Processamento com Memória
- ▶ Precisa-se de uma linguagem para haver comunicação!



1010101110101
010111010101

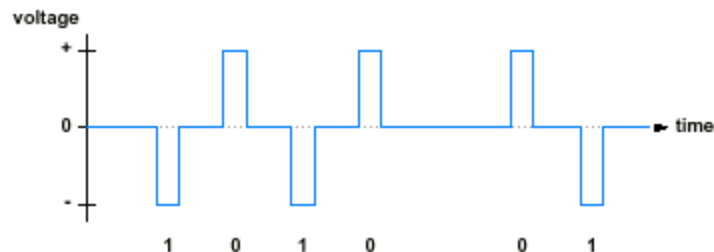
Linguagem dos Seres Humanos

- ▶ Linguagem Escrita e Falada
- ▶ Linguagem Escrita:
 - Parágrafos
 - Períodos
 - Frases
 - Palavras
 - Letras! (Símbolo Básico)
- ▶ Na fala o símbolo básico é o fonema!



Linguagem de máquina

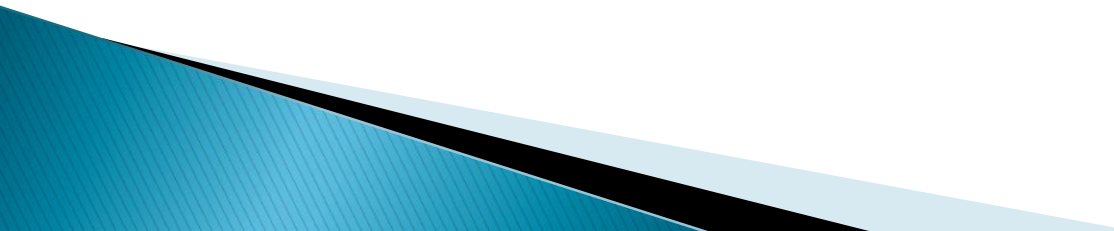
- ▶ Comunicação através de um fenômeno físico
- ▶ Foram escolhidos dois símbolos básicos
 - Facilidade de obter dois estados distintos e não confundíveis
 - EX:
 - Passar corrente elétrica e não passar
 - Estar magnetizado e não estar
 - ---



Linguagem de máquina

- ▶ Símbolos são representados por dois dígitos:
 - 0 (zero)
 - 1 (um)
- ▶ Binary Digit ou Dígito Binário
- ▶ As “palavras” desta linguagem são conjuntos de bits
 - Ex: 10101110001000100001010011

Linguagem de máquina

- ▶ CPU: Contém um pequeno conjunto de instruções
 - * Cada CPU tem um conjunto diferente!
 - ▶ É bastante complicado de escrever em linguagem de máquina
- 

Linguagem de máquina

```
0E3D:0000  CD 20 FF 9F 00 9A F0 FE-1D F0 4F 03 F0 07 8A 03  . .....O.....
0E3D:0010  F0 07 17 03 F0 07 DF 07-01 01 01 00 02 FF FF FF  .....
0E3D:0020  FF FF FF FF FF FF FF FF-FF FF FF FF BD 0D 4C 01  .....L.
0E3D:0030  D0 0C 14 00 18 00 3D 0E-FF FF FF FF FF 00 00 00 00  .....=.....
0E3D:0040  05 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  .....
0E3D:0050  CD 21 CB 00 00 00 00 00-00 00 00 00 00 20 20 20  .!.....
0E3D:0060  20 20 20 20 20 20 20 20-00 00 00 00 00 20 20 20  ....
0E3D:0070  20 20 20 20 20 20 20 20-00 00 00 00 00 00 00 00  .....
0E3D:0080  00 0D 76 2E 65 78 65 0D-44 4F 57 53 5C 73 79 73  ..v.exe.DOWS\sys
0E3D:0090  74 65 6D 33 32 5C 64 6F-73 78 0D 0D 64 61 20 72  tem32\dosx..da r
0E3D:00A0  65 64 65 20 28 63 61 72-72 65 67 61 72 20 61 6E  ede (carregar an
0E3D:00B0  74 65 73 20 64 6F 20 64-6F 73 78 2E 65 78 65 29  tes do dosx.exe)
```

Linguagem de Montagem

- ▶ Também conhecida como assembly
- ▶ Código que tem uma instrução alfanumérica para cada instrução numérica da ling. de máquina.
- ▶ Há um processo de tradução para ser executada (assembler)
- ▶ É uma linguagem de baixo nível!
- ▶ Ainda é muito complexa... Solução?
Linguagens de Alto Nível

Classificação das Linguagens

- ▶ Linguagens de Alto Nível:
 - Linguagem próxima do ser humano
 - Grande maioria é em inglês
 - Alto poder de expressividade
- ▶ Linguagens de Baixo Nível:
 - Mais próxima da máquina
 - Pouco Legível
 - Pouca expressividade
- ▶ “Linguagens de Médio Nível”

Compilação x Interpretação

▶ Compilação:

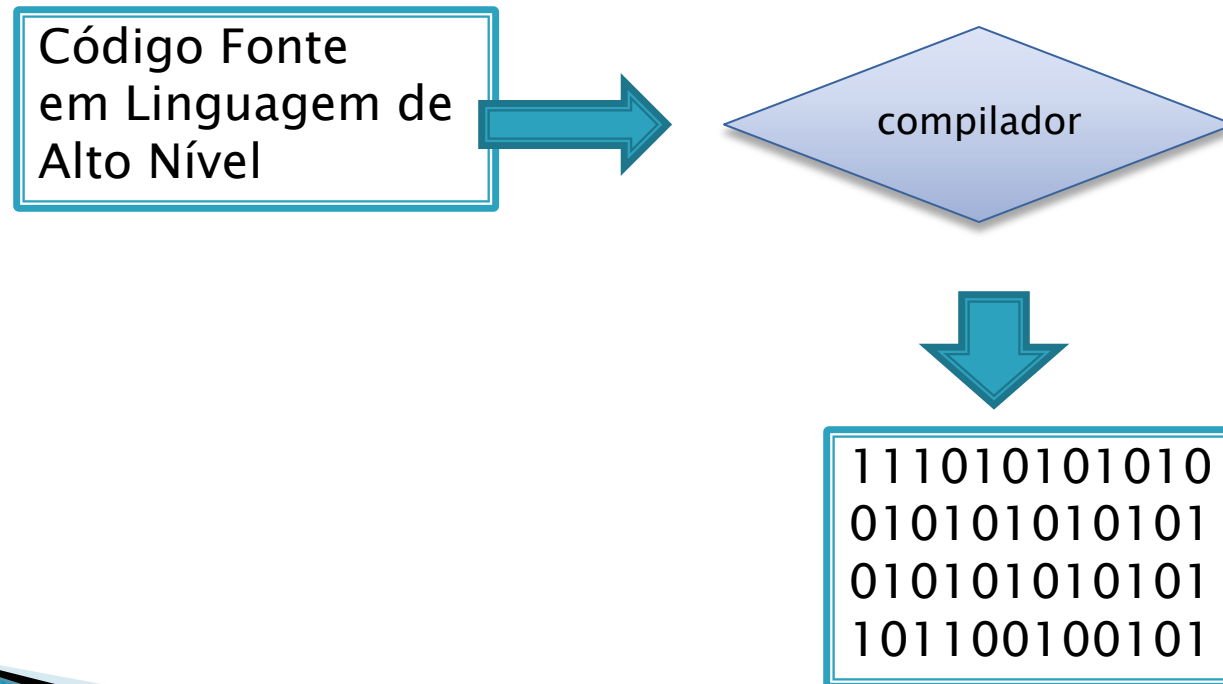
- Código-fonte gera um código em linguagem de máquina: código-objeto
- Só é carregado em memória no momento da execução

▶ Interpretação

- Não gera um código-objeto, as instruções são traduzidas somente no tempo de execução. De instrução a instrução.

Compiladores

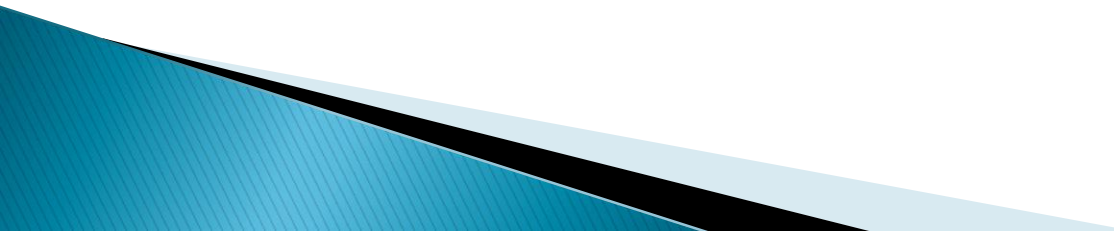
- ▶ Vão servir para transformar o código-fonte em linguagem de máquina



Sistema Operacional

- ▶ Algumas operações podem ser complexas.
- ▶ Os SOs constroem uma camada entre hardware e software para facilitar acesso a recursos.
 - Ex: acesso ao sistema de arquivos, dispositivos de I/O, etc.

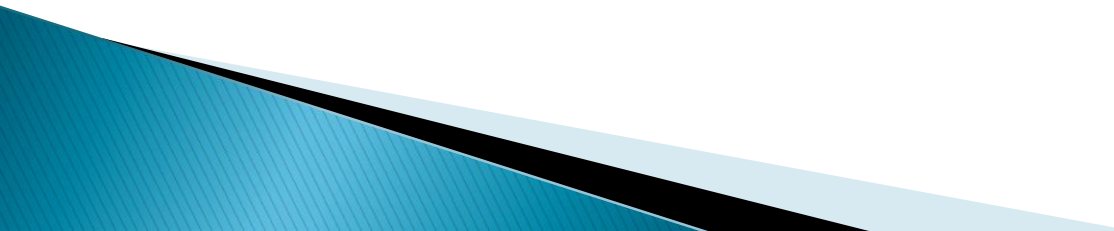
Programas de Computadores

- ▶ É um algoritmo escrito numa linguagem de computador (C, Fortran, Pascal, Delphi, Cobol, Java, C++, .Net, Python, ...)
 - ▶ Este processo é chamado de codificação!
 - ▶ O mais importante de um programa é sua lógica, ou seja, o algoritmo!
- 

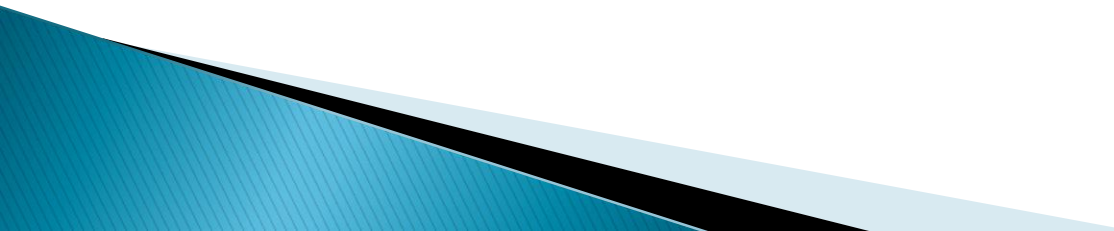
Algoritmo

- ▶ É uma sequência de passos finitos que visam solucionar um problema bem definido
- ▶ Exemplo Clássico:
 - Receita de um bolo

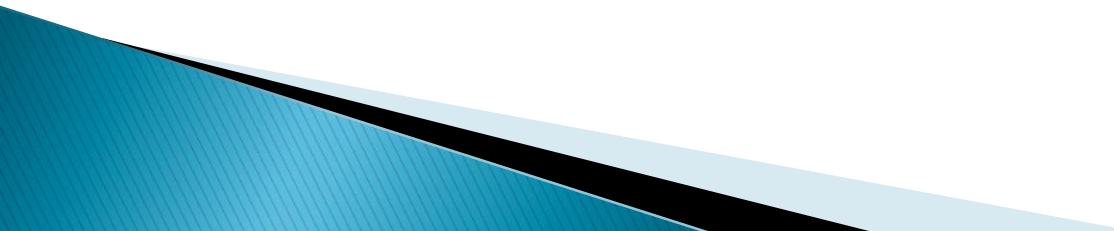
Algoritmo

- ▶ Foco na lógica de construção de algoritmos!
 - ▶ Após o algoritmo pronto, pode-se escrever em qualquer linguagem de programação
 - ▶ Representações
 - Linguagem Natural (*Sequência de passos*)
 - Fluxogramas
 - Português Estruturado
- 

Exemplo: 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
 7. Retirar quanto pronto
- 

Exemplo 2: Trocar Lâmpadas

1. Se (lâmpada estiver fora de alcance)
Pegar escada;
 2. Pegar a lâmpada;
 3. Se (lâmpada estiver quente)
Pegar pano;
 4. Tirar lâmpada queimada;
 5. Colocar lâmpada boa;
- 

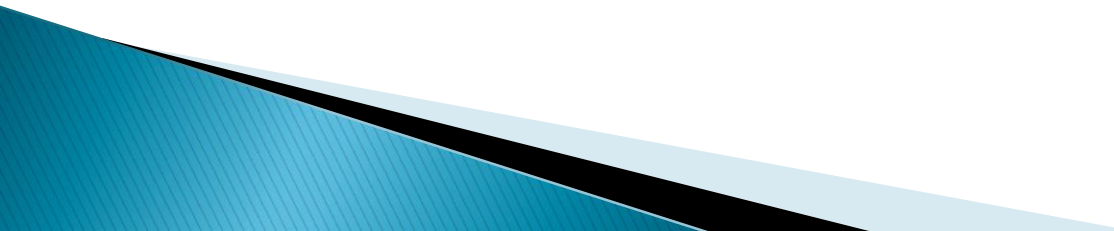
Exemplo 3: Fazer bolo simples

1. Pegar os ingredientes;
2. Se (roupa for branca)
Colocar avental;
3. Se (tiver batedeira)
Bater os ingredientes na batedeira;
4. Senão
Bater os ingredientes à mão;
5. Colocar a massa na forma
6. Colocar a forma no forno
7. Aguardar o tempo necessário
8. Retirar o bolo

Exemplo 4: Batatas

1. Pegar faca, bacia e batatas;
2. Colocar água na bacia;
3. Enquanto (houver batatas)
Descascar batatas

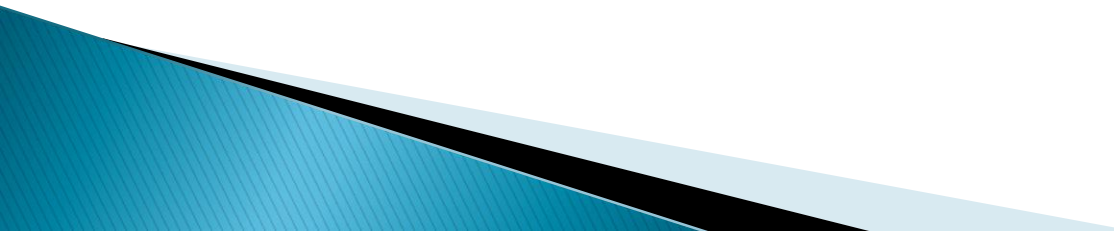
Observação!

- ▶ Cada linha do algoritmo podemos chamar de uma instrução!
 - ▶ Instrução: indica ao computador uma ação elementar a ser executada
 - ▶ Algoritmo = sequência de instruções
- 

Lógica de Programação

- ▶ É a técnica de encadear pensamentos para atingir um determinado objetivo.
- ▶ É uma técnica muito importante para o desenvolvimento de sistemas.

Lógica de Programação

- ▶ O raciocínio pode ser expresso por inúmeros idiomas, sem alterar o sentido.
 - ▶ Da mesma forma, a lógica pode ser representadas em inúmeras linguagens de programação.
 - ▶ Utiliza-se os algoritmos para escapar dos detalhes de cada linguagem.
- 

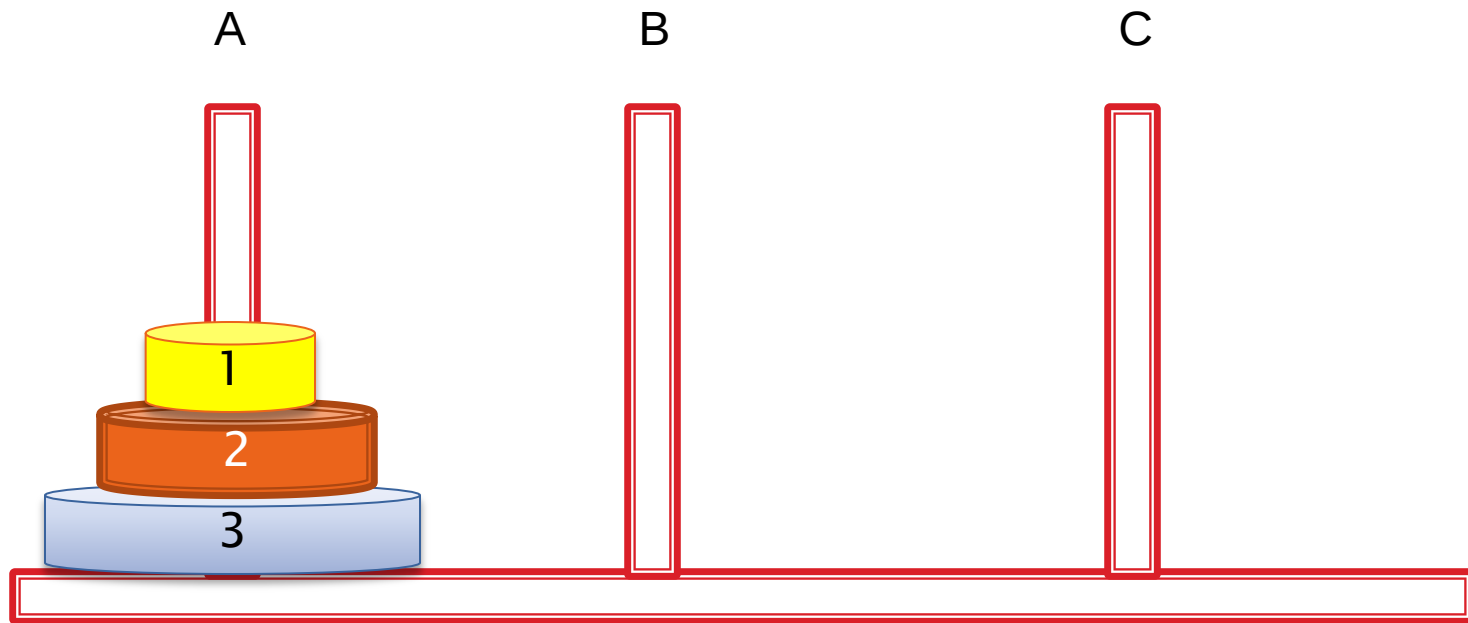
Problema: 3 canibais e 3 missionários

- ▶ Você tem que transportar para a outra margem do rio os três missionários e os três canibais. Obs: Não pode em nenhum momento ter mais canibais do que missionários de uma margem do rio. Sendo que a canoa transporta dois ao mesmo tempo.



Exercício: Torre de Hanói

- ▶ Transferir um disco para outra haste. Não pode colocar um disco maior em cima de um menor



Exercício: Torre de Hanói

- ▶ Transferir um disco para outra haste. Não pode colocar um disco maior em cima de um menor

