



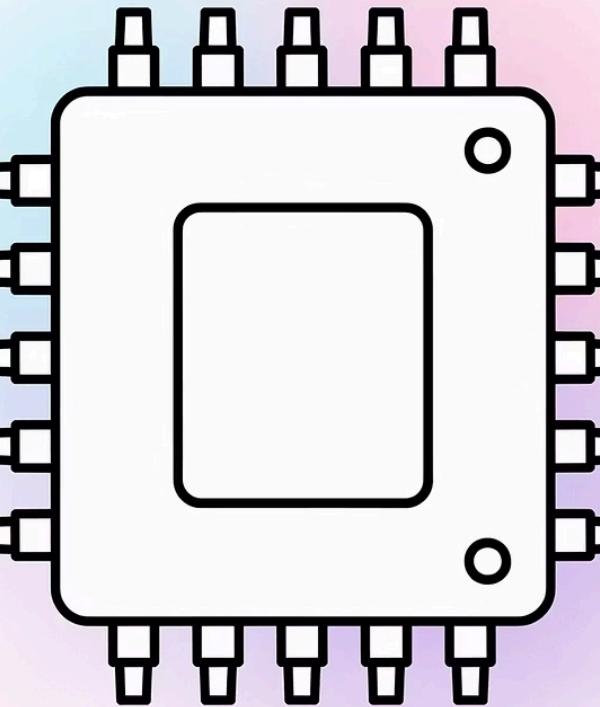
Variáveis: Guardando Informações

Entenda como o computador usa "caixinhas" para guardar valores que mudam durante o algoritmo.

Eduardo Ogasawara

eduardo.ogasawara@cefet-rj.br

<https://eic.cefet-rj.br/~eogasawara>



 CONCEITO FUNDAMENTAL

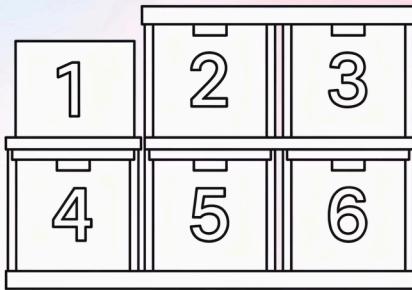
O Computador Precisa Guardar Coisas

Quando fazemos uma conta, precisamos lembrar números. O computador também precisa.

Para isso ele usa **variáveis**.

Variáveis são lugares especiais para guardar valores que podem mudar.

Variável é uma Caixinha



Nome da Caixa

Cada variável tem um nome único que não muda.



Valor Guardado

Dentro da caixa fica um valor que pode ser alterado.



Pode Mudar

O valor dentro pode ser trocado a qualquer momento.

 EXEMPLO PRÁTICO

Exemplo do Mundo Real



1

Caixa "pontos"

Imagine uma caixa chamada "pontos" no seu jogo favorito.

2

Guardar pontuação

Dentro dela guardamos a pontuação atual do jogador.

3

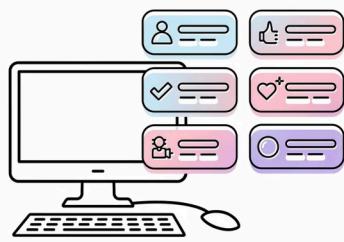
Valor aumenta

Quando o jogador ganha, o valor dentro aumenta.

4

Nome permanece

A caixa continua sendo "pontos", só o número muda.



Variáveis no Computador

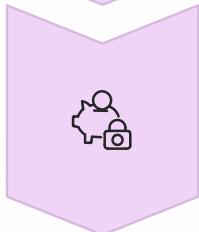
- O computador cria caixinhas digitais
- Ele dá um nome único para cada uma
- Dentro guarda números ou textos
- Essas caixinhas são as variáveis

Guardando um Número



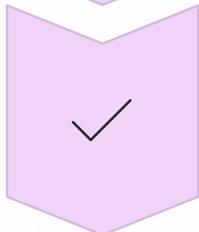
Fazer a conta

Realizamos um cálculo matemático



Guardar resultado

Salvamos o resultado numa variável

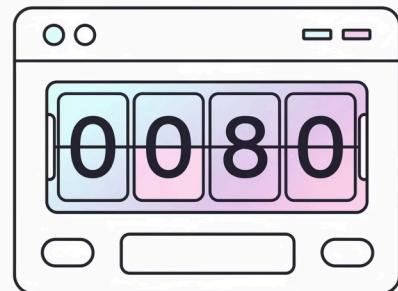


Usar depois

Podemos usar esse valor quando precisar

Isso evita esquecer valores importantes durante o algoritmo.

Mudando o Valor



O computador troca o valor dentro da caixinha sempre que necessário. O nome permanece o mesmo.

★ IMPORTANTE

Usando Variáveis



Somar

Usar valores guardados em operações matemáticas



Comparar

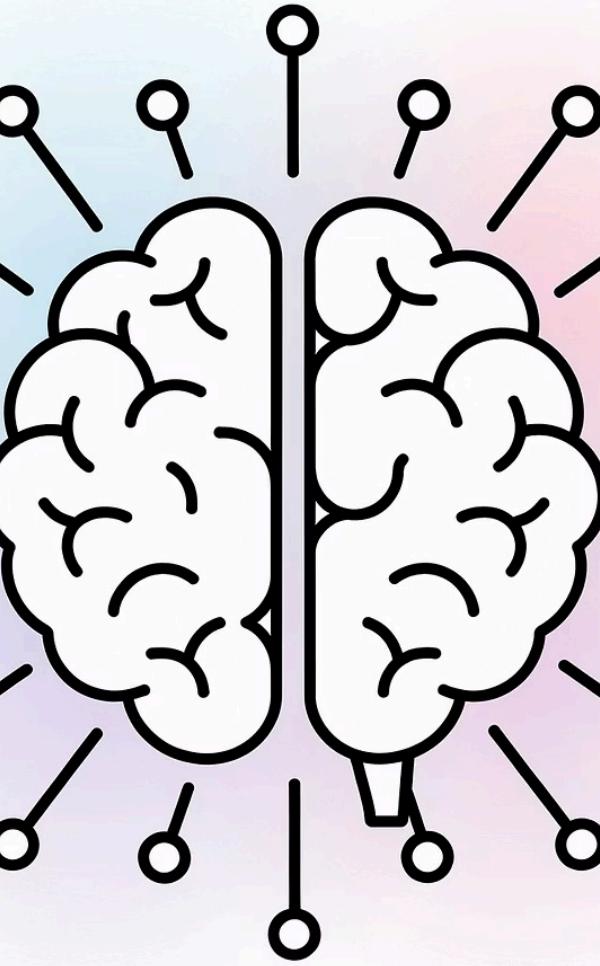
Verificar se um valor é maior ou menor que outro



Mostrar

Exibir o valor guardado na tela para o usuário

Tudo no programa passa pelas variáveis. Elas são muito importantes!



Por Que Precisamos Disso?

Sem variáveis

O computador não lembra de nada. Cada conta se perderia.

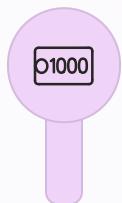
Com variáveis

O programa tem memória e pode realizar tarefas complexas.

Variáveis dão memória ao programa. Isso torna os algoritmos poderosos.

Exemplo Simples

$$\boxed{} + \boxed{} = \boxed{}$$



Valor inicial

Guardamos 10 em uma variável



Operação

Somamos 5 ao valor guardado



Novo valor

O valor vira 15 na mesma variável

O Que Aprendemos

Guardam valores

Variáveis armazenam informações importantes

Têm nomes

Cada variável possui um identificador único

Mudam com tempo

Os valores podem ser alterados durante execução

São essenciais

Fundamentais para criar qualquer programa

EXERCÍCIO 1

Soma de Dois Números

Problema

Leia dois números e mostre a soma.

Plano

1. Ler o primeiro número
2. Ler o segundo número
3. Somar os valores
4. Mostrar o resultado

```
algoritmo "soma"
var
    a, b, s : inteiro
inicio
    leia(a)
    leia(b)
    s <- a + b
    escreva(s)
fimalgoritmo
```

Testando o Programa

01

Entrada

Digitamos: 5 e 7

03

Saída

Mostra na tela: 12

02

Processamento

O computador calcula: $s \leftarrow 5 + 7 = 12$

04

Confirmação

Isso confirma que o algoritmo funciona!

EXERCÍCIO 2

Média de Dois Números

Problema

Leia dois números e mostre a média.

Teste

Entrada: 6 e 8

Cálculo: $m \leftarrow (6 + 8) / 2 = 7$

Saída: 7

```
algoritmo "media"
var
    n1, n2, m : real
inicio
    leia(n1)
    leia(n2)
    m <- (n1 + n2) / 2
    escreva(m)
fimalgoritmo
```

Converter Idade em Meses

Problema

Leia a idade em anos e mostre em meses.

```
algoritmo "idade"
var
    anos, meses : inteiro
inicio
    leia(anos)
    meses <- anos * 12
    escreva(meses)
fimalgoritmo
```

 **Dica:** Multiplicamos por 12 porque cada ano tem 12 meses!



Referências

1

WING, Jeannette M. Computational thinking. Communications of the ACM, New York, v. 49, n. 3, p. 33–35, 2006.

2

PAPERT, Seymour. Mindstorms: children, computers, and powerful ideas. New York: Basic Books, 1980.

3

PÓLYA, George. How to solve it: a new aspect of mathematical method. 2. ed. Princeton: Princeton University Press, 1957.