

INTRODUÇÃO À PROGRAMAÇÃO

PROGRAMAÇÃO APLICADA A MATEMÁTICA

Meirylene Avelino

meirylenerea@id.uff.br

PROCESSO DE RESOLUÇÃO DE PROBLEMAS (PRINCÍPIOS DE PÓLYA)

Definição dos requisitos do **problema** (fazer o programa certo)

- Entradas
- Cálculos
- Casos especiais
- Saídas

Desenvolvimento do algoritmo da **solução** (fazer certo o programa)

- Português estruturado
- Pseudocódigo
- Fluxograma

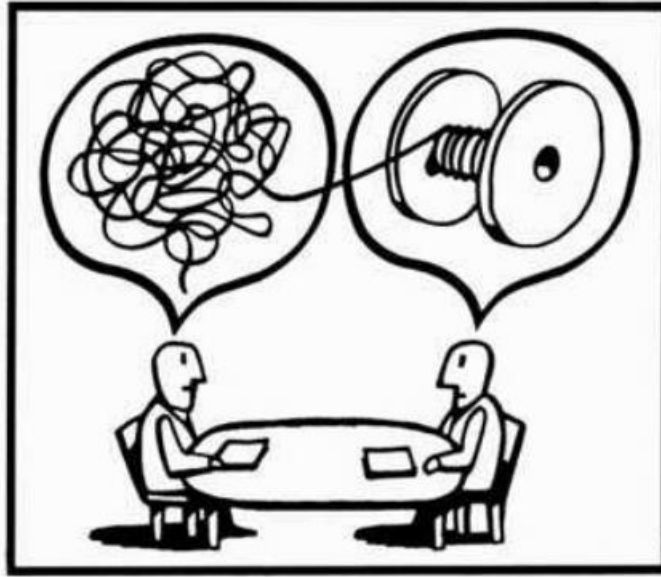
Codificação do programa

- Python

Teste do programa

- Instrução com erro de grafia (defeito na codificação)
- Resultado errado (defeito no algoritmo)

PASSO 1: REQUISITOS



Qual é o problema a ser resolvido?

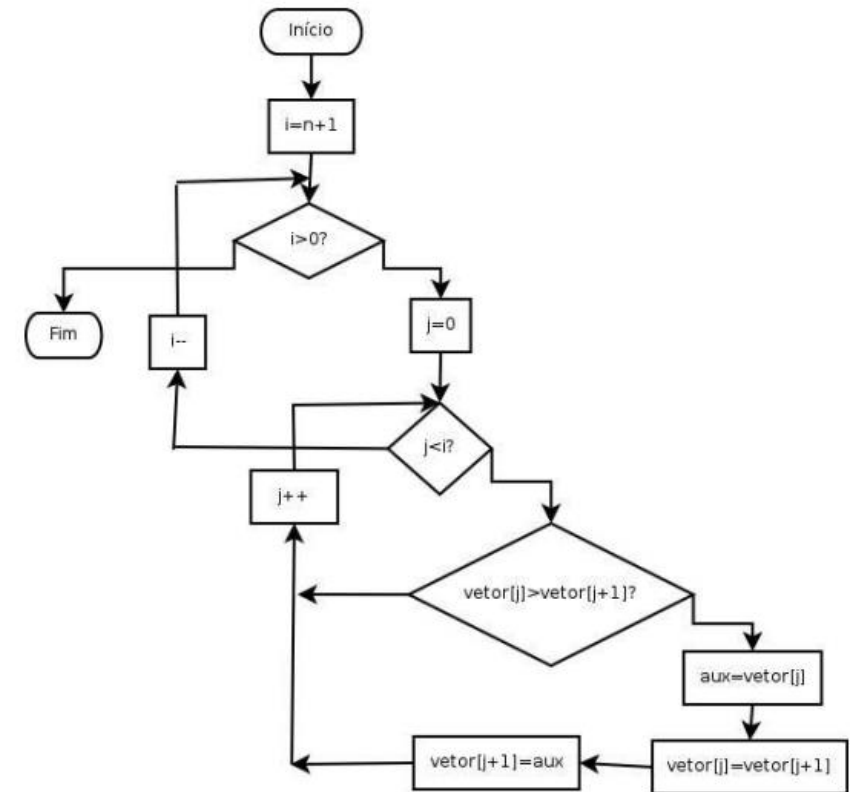
PASSO 2: ALGORITMO

Conjunto de **ações** para a **resolução** de um problema em um **número finito** de passos

Parte mais complexa da programação

Somente iniciar a programação quando

- Souber qual problema deve ser resolvido
- Souber como resolver o problema



PASSO 2: ALGORITMO

Independente de linguagem de programação

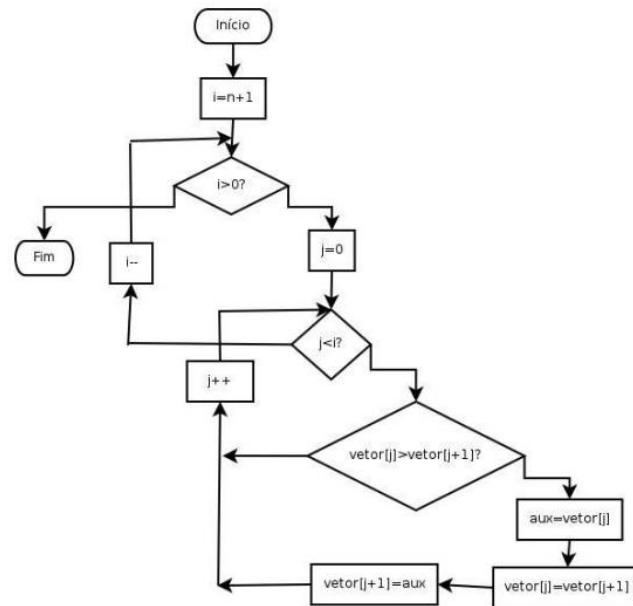
Pode ser implementado em diferentes linguagens



C++

```
#include <algorithm>
using namespace std;

void bubblesort(int a[], int n)
{
    for(int j=0; j<n; j++){
        for(int i=0; i<n-1; i++){
            if(a[i+1] < a[i])
                swap(a[i+1], a[i]);
        }
    }
}
```

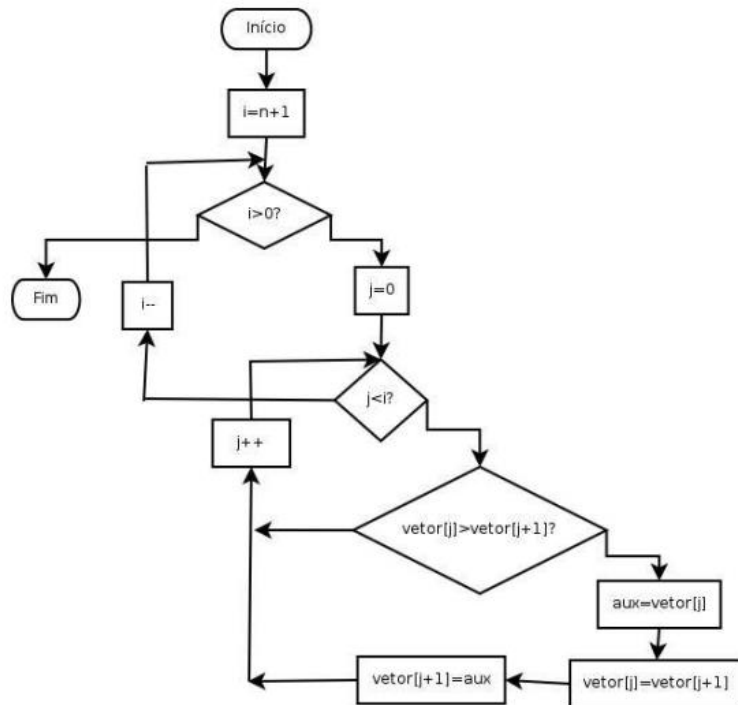


Matlab

```
for(i = 1:n-1)
    for(j = 1:n-i)
        if(x(j) > x(j + 1))
            aux = x(j);
            x(j) = x(j + 1);
            x(j + 1) = aux;
        end
    end
end
```

PASSO 3: CODIFICAÇÃO

A partir do algoritmo, traduzir (implementar) para a linguagem desejada



Implementação

Python

```
def bubble (vetor):
    houve troca = True
    while (houve troca):
        houve troca = False
        for i in range(len(vetor) - 1):
            if (vetor[i] > vetor[i+1]):
                aux = vetor[i+1]
                vetor[i+1] = vetor[i]
                vetor[i] = aux
                houve troca = True
    return v
```

POR QUE NÃO EXECUTAR DIRETAMENTE O ALGORITMO NO COMPUTADOR?

Algoritmo é escrito em linguagem natural

Linguagem natural é muito complexa e pouco precisa

É necessário usar uma linguagem mais simples e precisa, que o computador compreenda

“Calcule cinco mais cinco vezes dez”



PASSO 4: TESTE

O trabalho não termina com o código

Todo código pode ter defeito (bug)

Testar o código é fundamental!



TIPOS DE ERROS

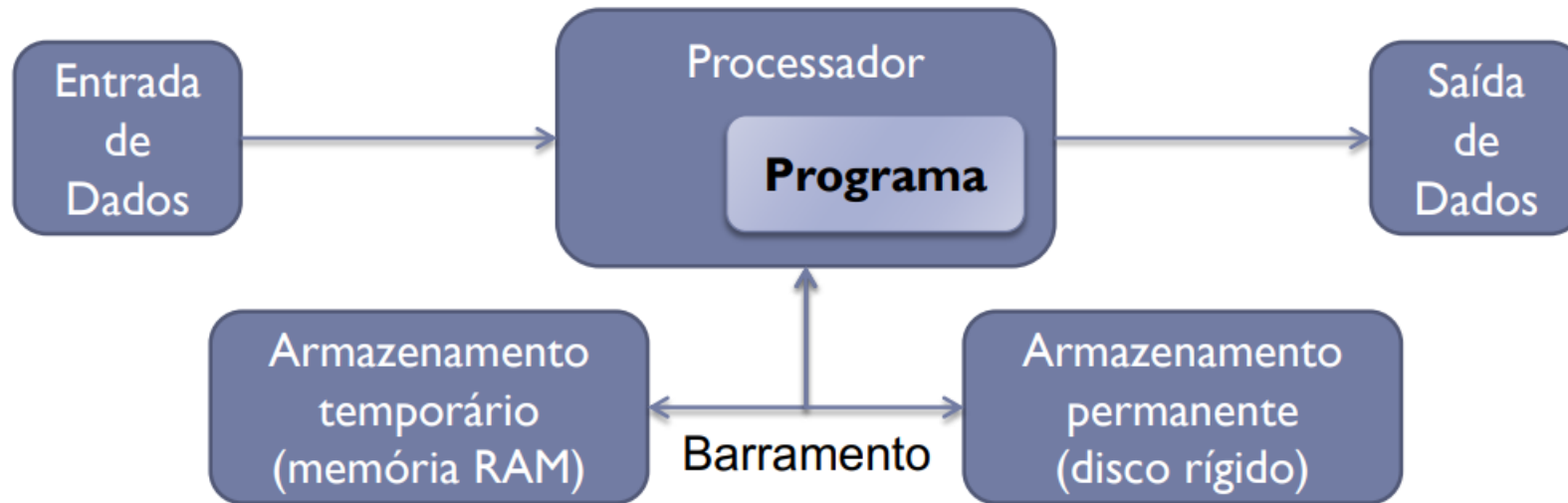
Erro de sintaxe

- Falha na tradução do algoritmo para python
- O compilador vai detectar e dar dicas
- Mais fáceis de corrigir

Erro de lógica

- Resultados diferentes do esperado
- Erro de projeto do algoritmo
- Mais difíceis de corrigir

ARQUITETURA DE UM COMPUTADOR



Entrada	Saída	Armazenamento
Teclado	Vídeo	Memória
Mouse	Impressora	Discos rígidos
Scanner	Auto-Falante	CD/DVD
Webcam		Pen drive

ALGORITMO PARA SOMAR DOIS NÚMEROS



A



B



SOMA

ALGORITMO PARA SOMAR DOIS NÚMEROS

Linguagem Natural:

Leia um valor de entrada

Escreva esse valor na caixa A

Leia um valor de entrada

Escreva esse valor na caixa B

Some o valor da caixa A com o valor da caixa B

Escreva o resultado na caixa SOMA

PARA SABER MAIS

Capítulo 1 do livro Algoritmos e Lógica de Programação. Ed Thomson.

REFERÊNCIAS

Baseado nos materiais da Vanessa Braganholo (UFF)

Alguns exercícios extraídos do livro Furlan, M., Gomes, M., Soares, M., Concilio, R., 2005, "Algoritmos e Lógica de Programação", Editora Thomson.

INTRODUÇÃO À PROGRAMAÇÃO

PROGRAMAÇÃO APLICADA A MATEMÁTICA

Meirylene Avelino

meirylenerea@id.uff.br