Introdução à Programação

Aula 01 Introdução

Profa. Andreia Rodrigues



Andreia Rodrigues

andreia.rodrigues@ifce.edu.br

- Bacharel em Ciência da Computação
- Mestrado na área de engenharia de software e pesquisa operacional
- Doutorado na área de engenharia de software (RNFs)
- IFCE desde 2017



Agenda

- ____
 - Disciplina
- Conceitos iniciais
- Algoritmo e Programa
- Linguagens de Programação
- Paradigmas de Programação
- Programação Estruturada
- Compiladores e Interpretadores
- Tipos de Dados Básicos



INTRODUÇÃO A PROGRAMAÇÃO

- Fundamentos da Programação de Computadores
- Aulas Expositivas e Práticas
- Atividades Coletivas e Individuais



Horários, frequência e notas

Quartas-feiras:

Horários: 18:30 às 20:30

Sexta-feira:

Horário: 13:30 às 15:30 ou 16:00 às 18:00

Frequência e notas:

- atividades assíncronas
- atividades síncronas



Segredo do sucesso!

Você já teve algum contato com lógica de programação?



Segredo do sucesso

1. Façam anotações

2. Resolvam todos os exercícios (sala de aula/casa)

3. Estudem todos os dias (Hábito)

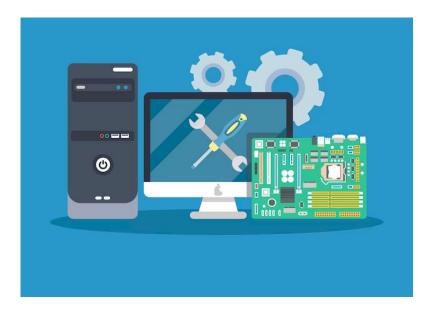
4. Tire suas dúvidas



Computador



Software



Hardware



Etapas do desenvolvimento de software

Análise Codificação Algoritmo Entender o problema Definir a solução Escrever o algoritmo Identificar as entradas para o problema em na linguagem de Definir a estratégia de forma de passos, programação com detalhes sobre escolhida. processamento Identificar as saídas seu processamento, entradas e saídas.



O que é um ALGORITMO?



Algoritmo

"Algoritmo é uma sequência de passos que visa atingir um objetivo bem definido" (FORBELLONE, 1999).".



Algoritmo







Exemplo de Algoritmo

Algoritmo para escovar os dentes:

- Molhar a escova;
- Aplicar pasta de dente;
- 3. Escovar cada dente por um determinado tempo;
- 4. Enxaguar a boca;
- 5. Enxaguar a escova; e
- 6. Guardar a escova.



Exemplo de Algoritmo



Algoritmo para trocar uma lâmpada

- pegar a nova lâmpada
- 2. pegar a escada
- posicionar a escada abaixo da lâmpada que deve ser trocada
- 4. subir a escada de posse da nova lâmpada
- 5. retirar a lâmpada queimada
- 6. colocar a nova lâmpada
- 7. descer da escada
- 8. acionar o interruptor para testar a lâmpada
- 9. guardar a escada
- 10. descartar a lâmpada queimada



Se eu resolver esses problemas de forma diferente?

Formas de representação de um algoritmo

Descrição narrativa

Fluxograma

Pseudocódigo



Exemplo

Escreva um algoritmo que solicite dois números inteiros como entrada e exiba sua soma na tela.

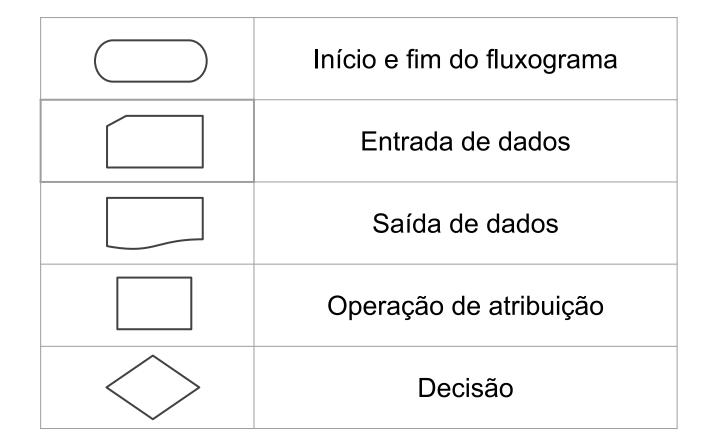


Descrição narrativa

- 1. Receber as duas notas
- 2. Calcular a soma
- 3. Exibir a soma

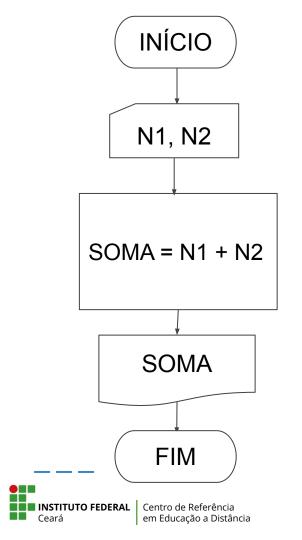


Fluxograma





Fluxograma



Pseudocódigo

```
Algoritmo <nome_do_algoritmo>
<declaração_de_variáveis>
Início
```

<corpo do algoritmo>

Fim



Pseudocódigo

Algoritmo SOMA

Var numero1, numero2, soma: inteiro

Início

ESCREVA 'Digite os números: '

LEIA numero1

LEIA numero2

soma = numero1 + numero2

ESCREVA soma

FIM



Pseudocódigo

```
Algoritmo MEDIA
```

Var nota1, nota2, media: real

```
Início
```

```
ESCREVA 'Digite as notas: '
LEIA nota1
LEIA nota2
media = (nota1 + nota2)/2
Se MEDIA >= 7 então
Escreva 'Aprovado'
Senão
Escreva 'Reprovado'
Fim_se
```

FIM



Programa



Programa

PROGRAMA é um conjunto de instruções usadas por um computador para realizar uma determinada tarefa.



Exemplos de programas













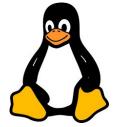
















Linguagens de Programação



Linguagens de Programação

- ____
- Computador é formado de circuitos elétricos;
- Sistema binário:
 - o 0 desligado
 - o 1 ligado
- Existem várias linguagens de programação;



Linguagens de Programação

Segundo a definição da IEEE, "uma linguagem de programação é um conjunto de símbolos e regras sintáticas e semânticas usados para definir um programa de computador".



Tipos de Linguagens de Programação

Alto nível Baixo nível



Tipos de linguagem

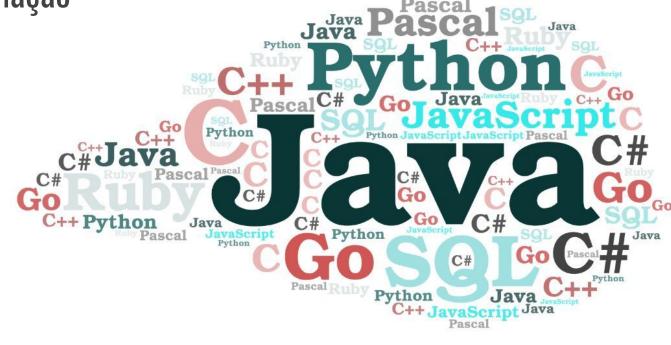
Python

print('Hello, world!')

Assembly

section .text global _start start: mov edx, len mov ecx, msg mov ebx, 1 mov eax, 4 int 0x80 mov eax, 1 int 0x80 section .data msg db 'Hello, world!',0xa len equ \$ - msg

Linguagens de Programação





Exemplos de Linguagens de Programação

С	PYTHON	JAVA	JAVA SCRIPT	SQL
sistemas operacionais, jogos e aplicativos de desktop	ciência de dados, inteligência artificial, aprendizado de máquina e desenvolvimento web.	aplicativos móveis, jogos e aplicações empresariais.	Criar páginas web interativas e dinâmicas.	gerenciar e consultar bancos de dados.

Paradigmas de Programação



Paradigmas de Programação

Conjunto de **conceitos e técnicas** que definem como um programa deve ser **organizado e escrito**.

Esses paradigmas **fornecem uma maneira de pensar** e abordar problemas de programação, e incluem diferentes abordagens para **modelar a solução de um problema**.

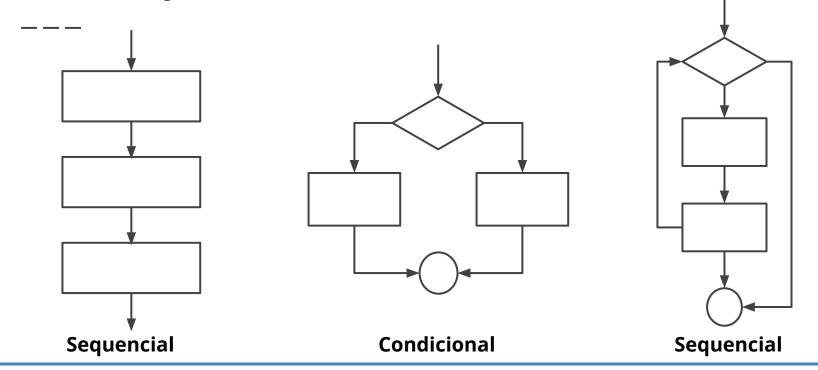


Exemplos de Paradigmas de Programação

- Paradigma Imperativo
- Paradigma Orientado a Objetos
- Paradigma Funcional
- Paradigma Lógico
- Paradigma de Programação Concorrente









A programação estruturada também enfatiza a **modularidade**, dividindo o código em pequenos blocos de construção chamados de **procedimentos** ou **funções**.

Esses blocos de construção podem ser **reutilizados** em diferentes partes do programa, o que ajuda a tornar o código mais eficiente e fácil de manter.

Linguagens de programação que utilizam a programação estruturada incluem **C**, **Pascal** e **Ada**.

A programação estruturada é amplamente utilizada em programação de sistemas operacionais.



Exemplo de Programa Estruturado

Exemplo de programa estruturado em Python que solicita ao usuário um número e verifica se ele é par ou ímpar:

```
num = int(input("Digite um número: "))
if num % 2 == 0:
    print("O número digitado é par.")
else:
    print("O número digitado é impar.")
```



Compiladores x Interpretadores



Se o computador não é capaz de entender as linguagens de programação como os programas funcionam?

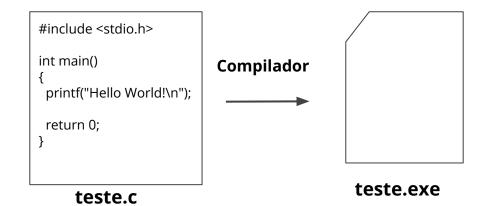


Compilador

Etapas do processo de compilação:

- Análise léxica;
- Análise sintática;
- Análise semântica;
- Otimização de código; e
- Geração de código objeto.

O resultado final é um arquivo executável que pode ser executado diretamente em um computador ou máquina virtual.



Exemplos de Linguagens Compiladas

Alguns exemplos de linguagens compiladas incluem:

- C, C++ e Objective-C
- Fortran
- Swift
- Rust
- Pascal
- Ada



Interpretadores



Interpretador

O interpretador executa o programa **linha por linha**, verificando cada comando e instrução para ver se ele pode ser executado.

Se o comando é válido, o interpretador executa a instrução associada.

Se houver um erro, o interpretador geralmente interrompe a execução do programa e apresenta uma mensagem de erro ao usuário.

Exemplos de Linguagens Interpretadas

Alguns exemplos de linguagens interpretadas incluem:

- Python
- Ruby
- JavaScript
- Perl

Tipos de Dados Básicos



Tipos de Dados

Tipos de dados referem-se aos diferentes tipos de valores que podem ser armazenados e manipulados em um programa de computador.



Exemplos de Tipos de Dados Básicos

- Inteiros: números inteiros, como -10, 0, 42, etc.
- Números de ponto flutuante: números com casas decimais, como 3.14,
 0.5, etc.
- Booleanos: valores lógicos verdadeiros ou falsos, representados por true ou false.
- Caracteres: caracteres únicos, como 'a', 'Z', '?', etc.
- Strings: sequências de caracteres, como "Olá, mundo!" ou "1234".



Tipos de Dados

Cada tino de dado tem um conjunto esne

Cada tipo de dado tem um **conjunto específico de operações** que podem ser realizadas sobre ele.

Por exemplo, é possível adicionar, subtrair, multiplicar e dividir números, enquanto as strings podem ser concatenadas, comparadas e divididas em substrings.



Tipos de Dados Mais Complexos

- Arrays: coleções de valores do mesmo tipo, como [1, 2, 3, 4] ou ["maçã", "banana", "laranja"].
- Ponteiros: valores que apontam para posições de memória, usados para trabalhar com estruturas de dados mais complexas.
- Estruturas: tipos de dados personalizados que contêm vários campos, como um registro em um banco de dados.



Leitura Complementar

- The Art of Computer Programming (A Arte da Programação de Computador) de Donald E. Knuth
- Structured Programming (Programação Estruturada) de Edsger W. Dijkstra
- Compilers: Principles, Techniques, and Tools (Compiladores: Princípios, Técnicas e Ferramentas) de Aho, Lam, Sethi e Ullman
- Concepts of Programming Languages (Conceitos de Linguagens de Programação) de Robert W. Sebesta

Encerramento

- Revisão
- Exercício Complementar
- Próxima Aula

