



**UNIFACS**

# **Programação de Soluções Computacionais**

**Introdução a computadores.  
Linguagens de programação.  
Java.**

- Motivação e discussão inicial
- Programa
- Computador
- Software
- Linguagem de Programação
- Compilador
- Hierarquia de dados
- Java
- Algoritmo

Antes de iniciar vamos assistir a um vídeo ...

[https://www.youtube.com/watch?v=iKKOV4yGI\\_M](https://www.youtube.com/watch?v=iKKOV4yGI_M)

O que mais te impressionou nesse vídeo?  
Qual a sua opinião agora sobre programar?

O que é um programa e como ele funciona?

- Instruções fazem com que os computadores realizem tarefas.
- O software (isto é, as instruções que você escreve) controla o hardware (isto é, os computadores).
- Os computadores processam dados sob o controle de conjuntos de instruções chamados **programas de computador**.
- Esses programas de software orientam o computador por meio de ações ordenadas especificadas por pessoas chamadas **programadores de computador**.



Mas o que é um computador?

Um computador é uma máquina eletrônica lógica

Programável

- Programa

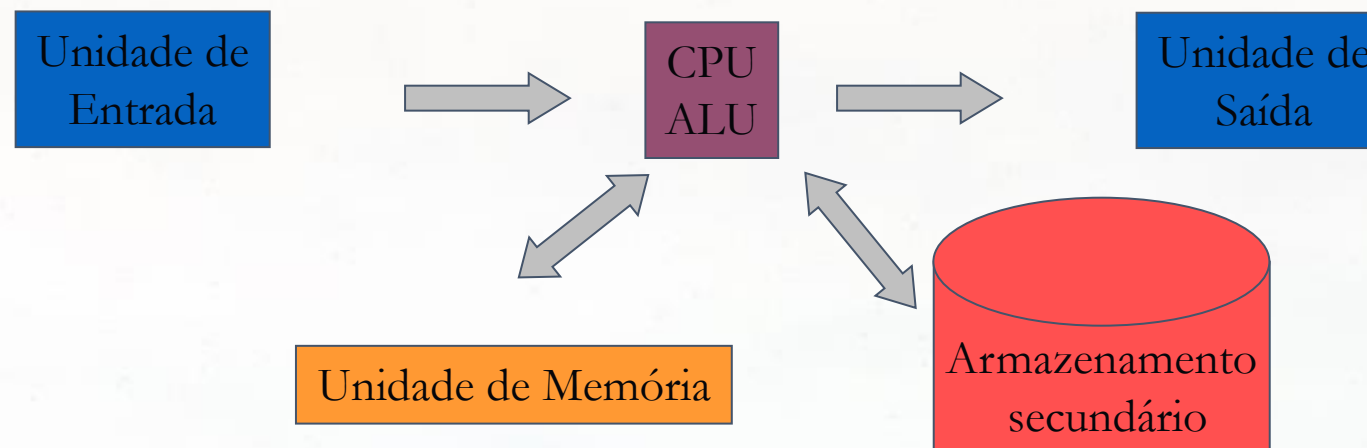
Instruções



## Organização do computador

- Os computadores podem ser visualizados como divididos em várias unidades lógicas ou seções lógicas:

- |                      |                                          |
|----------------------|------------------------------------------|
| ✓ Unidade de entrada | ✓ Unidade de aritmética e lógica (ALU)   |
| ✓ Unidade de saída   | ✓ Unidade de processamento central (CPU) |
| ✓ Unidade de memória | ✓ Unidade de armazenamento secundária    |



Processador: É o centro de controle de todo o sistema, onde os cálculos são feitos e as decisões são tomadas. É esta parte que interpreta, executa e controla todas as instruções do computador.

CPU – Unidade Central de Processamento.

Memória: A CPU usa a memória do computador para guardar as informações com as quais trabalha. Quando o computador é reinicializado ou desligado, estas informações desaparecem.

RAM – Random Access Memory.

Dispositivos de entrada e dispositivos de saída: São todos os meios de comunicação entre o computador e os seus usuários (sejam pessoas ou outras máquinas).

Dispositivos de entrada: aceitam dados e instruções do usuário.

São exemplos o TECLADO, o MOUSE e o SCANNER.

Dispositivos de saída: retornam dados processados para os usuários.

São exemplos o MONITOR e a IMPRESSORA.



Armazenamento: Função de guardar os dados que não estão sendo processados no momento. Mesmo desligando o computador, estas informações não se perdem.

Meios de armazenamento: disco magnético, fita magnética, cd-rom, etc.

Dispositivos de E/S (entrada e saída): unidades de disco, unidades de fita, unidades de cd-rom, etc.

Software: Instruções que as pessoas escrevem para dizer ao hardware o que fazer.

- Um conjunto específico destas instruções é chamado programa.
- Quando o computador está usando um determinado programa, dizemos que ele está rodando ou executando este programa.
- Como o programa informa aos componentes físicos da máquina o que fazer, sem eles o computador seria apenas uma caixa sem utilidade.

Categorias de software:

- Software Básico ou Sistema Operacional
- Software Aplicativo
- Linguagens de Programação

Software Básico ou Sistema Operacional (SO): É composto por um conjunto de programas e rotinas que gerenciam os recursos do hardware do computador de modo a possibilitar sua utilização aumentando sua eficiência e permitindo a comunicação com outros equipamentos.

## Quais as responsabilidades de um SO?

- Tratamento de entradas e saídas de dados;
- Definição de espaço para armazenamento de dados na memória;
- Administração da transição entre programas;
- Controle de autenticação de usuários;
- Gerência de Execução de programas;
- Implementação de segurança as informações.

## Quais os principais S.O. no mercado atualmente?

Windows, AIX, Solaris, Linux, etc.

Software Aplicativo: É um conjunto de programas desenvolvidos para realizar, em combinação com a atividade humana, tarefas ou processos específicos, em geral, relacionados com o processamento de dados para a geração de informações.

Classificação dos softwares aplicativos:

Administrativos	Sistemas de faturamento, contas a pagar, folha de pagamento, controle de estoque, controle de produção, contabilidade, etc
Tecno-científicos	Cálculo de estruturas, planejamento e controle de projetos, pesquisas operacionais, problemas de engenharia, etc
Automação Industrial	Ativação, programação e controle de máquinas e equipamentos, controle de processos, calibração, etc
Automação Comercial	Reserva de passagens, contas correntes, pontos de venda, caixas automáticos, etc
Apoio Educacional	Assistência à instruções, ensino auxiliado pelo computador, etc
Especiais e Científicos	Teleprocessamento, comunicações, militares, pesquisas espaciais, pesquisas meteorológicas, etc
Entretenimento	Jogos, musica, etc

Exemplos?



Linguagens de Programação: Proporcionam ao programador a possibilidade de fornecer instruções ao computador através de comandos ou sentenças, escritas em uma linguagem próxima à linguagem humana, permitindo desta forma a criação de novos softwares.

A linguagem que o computador entende é a linguagem de máquina, composta de 0 e 1.

Linguagem de baixo nível: Linguagem que utiliza código mneumônicos e endereços simbólicos no lugar de código binário. Já é mais evoluída do que a linguagem de máquina, mas ainda está bem próxima desta, sendo de difícil entendimento e requerendo um grande esforço de programação.

Exemplo: linguagem assembler (montadora)

COMPILADORES – programas que traduzem um programa escrito em linguagem de alto nível (código fonte) em instruções de linguagem de máquina (código objeto).



Os itens de dados processados pelos computadores formam uma **hierarquia de dados** que torna-se maior e mais complexa em estrutura à medida que passamos dos itens de dados simples (chamados “bits”) para aqueles mais ricos, como caracteres e campos.

## ***Bits***

- O menor item de dados em um computador.
- Abreviação de “binary digit” (dígito binário).
- Pode assumir um de *dois* valores: valor 0 ou o valor 1.
- As funções impressionantes desempenhadas pelos computadores envolvem apenas as manipulações mais simples de 0s e 1s — *examinar, configurar e inverter o valor de um bit* (de 1 para 0 ou de 0 para 1).



- Dígitos, letras e símbolos especiais são conhecidos como **caracteres**.
- O conjunto de caracteres do computador é o conjunto de todos os caracteres utilizados para escrever programas e representar itens de dados.
- O Java usa caracteres Unicode® que são compostos por um, dois ou quatro bytes (8, 16 ou 32 bits)

5

-

X

9

\*

M

- Assim como caracteres são compostos de bits, **campos** são compostos de caracteres ou bytes.
- Um campo é um grupo de caracteres ou bytes que transmitem um significado.
- Por exemplo, um campo que consiste em dígitos decimais pode representar a idade de uma pessoa

Maria

8.3

25

- Vários campos relacionados podem ser usados para compor um **registro** (implementado como uma class no Java).
- Um **arquivo** é um grupo de registros relacionados.
- Não é incomum uma organização ter muitos arquivos, que contêm alguns bilhões, ou mesmo trilhões, de caracteres de informações.

1234;Maria da Silva;Sistemas de Informação

1234;Maria da Silva; Sistemas de Informação  
4567;João Santos; Ciência da Computação  
8907;Antônio Carlos; Direito

- Um **banco de dados** é uma coleção de dados organizados para fácil acesso e manipulação.
- O modelo mais popular é o *banco de dados relacional*, em que os dados são armazenados em *tabelas* simples.

Matricula	Nome	Curso
1234	Maria da Silva	Sistemas de Informação
4567	João Santos	Ciência da Computação
8907	Antônio Carlos	Direito



- Uma tabela inclui *registros* e *campos*. Por exemplo, uma tabela de alunos pode incluir os campos nome, sobrenome, especialização, ano, número de identificação do aluno e rendimento acadêmico médio.
- Os dados para cada aluno são um registro, e as informações individuais em cada registro são os campos.

Matricula	Nome	Curso
1234	Maria da Silva	Sistemas de Informação
4567	João Santos	Ciência da Computação
8907	Antônio Carlos	Direito

- Linguagem de programação orientada a objetos.
- “Escreva uma vez, execute em qualquer lugar”.
- Utilizado para desenvolver aplicativos corporativos de grande porte, aprimorar a funcionalidade de servidores da web, fornecer aplicativos para dispositivos voltados ao consumo popular (por exemplo, telefones celulares, smartphones, televisão, *set-up boxes* etc.) e para muitos outros propósitos.
- Linguagem-chave para desenvolvimento de aplicativos Android adequados a smartphones e tablets.

- Criando um programa fonte
  - Programas são criados utilizando um programa editor
    - Ex. bloco de notas
  - Abre o editor, digita o programa e salva em uma unidade de memória secundária
  - Programas fonte em Java são criados com a extensão `.java`

- IDEs
  - Fornecem ferramentas que suportam o processo de desenvolvimento de software, como editores e depuradores para localizar erros lógicos
- Ex.
  - Eclipse ([www.eclipse.org](http://www.eclipse.org))
  - NetBeans ([www.netbeans.org](http://www.netbeans.org))
  - IntelliJ IDEA ([www.jetbrains.com](http://www.jetbrains.com))



- Uma vez editado o programa fonte pode ser compilado para gerar o arquivo .class
- Você pode digitar na sua linha de comando do DOS

`Javac programa.java`

- Se o programa estiver correto será gerado o arquivo `programa.class`
- Ou compilar diretamente na sua IDE, caso esteja utilizando uma.
- O compilador Java converte o seu arquivo fonte em `bytecodes` para ser executado pelo `Java Virtual Machine (JVM)`.

A máquina virtual (*virtual machine* — VM) é um aplicativo de software que simula um computador, mas oculta o sistema operacional e o hardware subjacentes dos programas que interagem com ela.

Se a mesma máquina virtual é implementada em muitas plataformas de computador, os aplicativos escritos para ela podem ser utilizados em todas essas plataformas.

Diferentemente das instruções em linguagem de máquina, que são *dependentes de plataforma* (isto é, de hardware específico de computador), instruções bytecode são *independentes de plataforma*. Portanto, os bytecodes do Java são portáteis — sem recompilar o código-fonte, as mesmas instruções em bytecodes podem ser executadas em qualquer plataforma contendo uma JVM que entende a versão do Java na qual os bytecodes foram compilados.

A JVM é invocada pelo comando **java**.

Para executar um aplicativo Java chamado Programa.class, você digitaria em uma janela de comando (ex. o comand do DOS):

Java Programa

A JVM carrega seu programa na memória principal

Verifica os bytecodes (Se são válidos)

A JVM realiza as instruções do seu programa

Outra opção é executar diretamente o programa pela sua IDE de preferência

# ã Fazendo seu primeiro programa Java



UNIFACS

Considerando que o Java já está instalado

Criar uma pasta C:\HelloWorld

Entrar no bloco de notas

Digitar o programa ao lado

Salvar como HelloWorld.java

Acessar a janela de comando ( executar cmd)

Acessa a pasta criada (cd c:\HelloWorld)

Compilar (javac HelloWorld.java)

Será criado o arquivo HelloWorld.class

Executar (java HelloWorld)

```
public class HelloWorld{  
  
    public static void main(String[] args){  
  
        System.out.println("Ola mundo");  
  
    }  
  
}
```

## O que é um algoritmo?

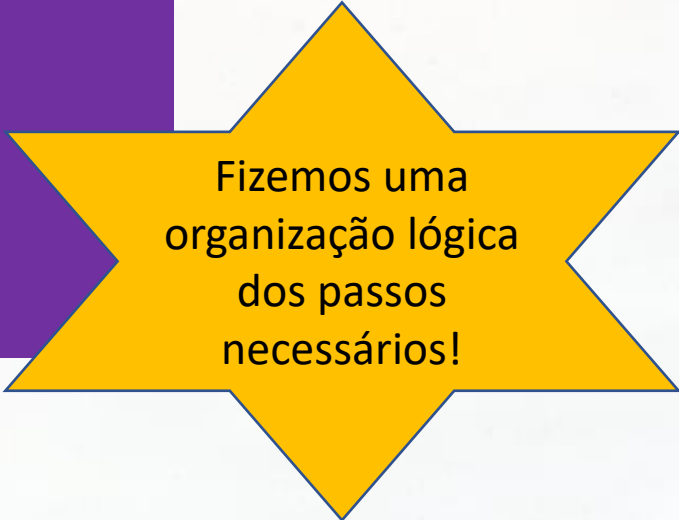
Sequência de instruções finitas e ordenadas de forma lógica para a resolução de uma determinada tarefa ou problema;

Não necessariamente envolve aspectos computacionais. Ex.: Uma receita de bolo, trocar um pneu de carro, trocar uma lâmpada, manual de instruções;



Você sabe fritar um ovo? Quais seriam os passos?

1. Retirar o ovo da geladeira
2. Colocar a frigideira no fogo
3. Colocar óleo
4. Esperar até o óleo ficar quente
5. Quebrar o ovo separando a casca
6. Colocar o conteúdo do ovo na frigideira
7. Esperar um minuto
8. Retirar o ovo da frigideira
9. Apagar o fogo



Fizemos uma  
organização lógica  
dos passos  
necessários!

## O que é lógica?

Em termos usuais – falamos de lógica no nosso dia-a-dia...

- ✓ Fulano é lógico.
- ✓ Eu sabia que isso aconteceria. Era lógico!
- ✓ A lógica do seu raciocínio é estranha.

⇒ **LÓGICA** é a forma de ordenar os pensamentos para solucionar um problema ou executar uma tarefa.

**Exemplos:**

- João é pai de Maria
- Pedro é pai de João
- Portanto, Pedro é avô de Maria

## ⇒ E o que é **LÓGICA DE PROGRAMAÇÃO**?

É o ato de programar o computador para que ele cumpra determinada tarefa. Para isso então é necessário usar lógica, ou seja, ordenar os pensamentos e identificar os passos para a solução do problema que será realizado pela máquina.

### Analizando a nossa sequencia de passos inicial...

1. Retirar o ovo da geladeira
2. Colocar a frigideira no fogo
3. Colocar óleo
4. Esperar até o óleo ficar quente
5. Quebrar o ovo separando a casca
6. Colocar o conteúdo do ovo na frigideira
7. Esperar um minuto
8. Retirar o ovo da frigideira
9. Apagar o fogo

### Essa sequencia é um algoritmo?

Possui uma sequência de instruções finitas e ordenadas?

R.: SIM. Instruções de 1 a 9.

O objetivo da sequência foi atingido?

R.: SIM. O ovo foi frito.

Logo, é um algoritmo!

### O que podemos concluir sobre algoritmos?

- Algoritmo é uma espécie de passo a passo;
- Algoritmos realizam uma saída (ex.: Um ovo frito) a partir de uma determinada entrada (um ovo) através de uma sequência de passos;
- Os passos tem que ser executados um após o outro;
- Um algoritmo está correto quando sua sequência de instruções resulta em uma saída esperada.
- Podem existir um ou mais algoritmos para atingir um resultado, desde que o resultado (a saída) seja o mesmo!

O algoritmo não é a solução do problema, mas o caminho que leva a mesma!

Construa uma sequencia de passos para

1. Fazer um bolo
2. Trocar uma lâmpada
3. Trocar o pneu de um automóvel
4. Ir da sua casa até a universidade
5. Realizar um saque no caixa automático de um banco
6. Alugar um livro em uma biblioteca
7. Escovar os dentes

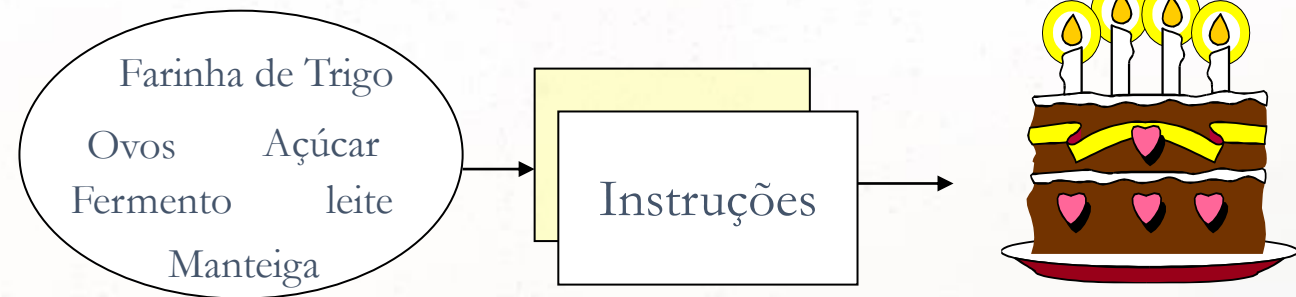


# Algoritmo para fazer um bolo



UNIFACS

- 1) Colocar dois ovos em uma batedeira
- 2) Colocar uma xícara de açúcar,
- 3) Colocar duas colheres de manteiga,
- 4) Bater os ingredientes por 5 minutos
- 5) Colocar uma xícara de leite de coco,
- 6) Colocar 3 xícaras de farinha de trigo
- 7) Misturar os ingredientes
- 8) Colocar uma colher de fermento e misturar levemente
- 9) Colocar em tabuleiro e assar em fogo médio.



Leitura do capítulo 2 do livro texto de Deitel

## Referências

1. Deitel. Java Como Programar, ed.10 (Capítulo 1)
2. Alexandra Kapusinski. “Histórico e Evolução dos Computadores”. (disponível na web)
3. Alcides Calsavara. “Histórico e Evolução dos Computadores”. (disponível na web)
4. Ivan Saraiva Silva & Milano Gadelha. “Organização e Arquitetura de Computadores I – Introdução”. (disponível na web)

ecossistema  
ãnima

# Obrigado(a)!

Prof. Artur Henrique Kronbauer  
[artur.kronbauer@animaeducacao.com.br](mailto:artur.kronbauer@animaeducacao.com.br)



**UNIFACS**