



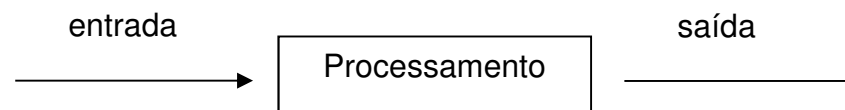
**Técnicas de Programação I**  
**Prof. Dr. Antonio Marcos SELMINI**

[profselmini@uol.com.br](mailto:profselmini@uol.com.br)  
[selmini@fiap.com.br](mailto:selmini@fiap.com.br)

**Introdução a Programação de Computadores**

## Introdução

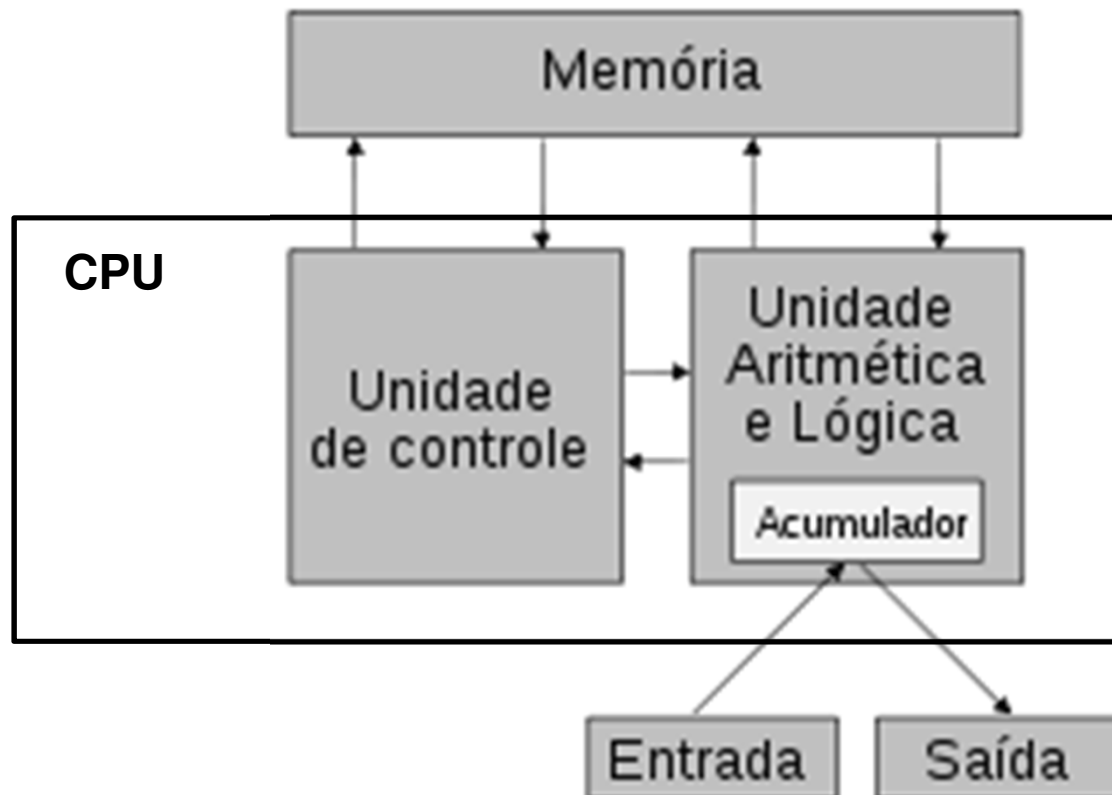
- **O que é um computador?**
  - Máquina que processa dados;
  - Processar significa realizar operações nos dados de entrada de forma a extrair informações;
- **Dado: tudo que pode ser processado;**
- **Informação: é o resultado do processamento, descreve um domínio físico ou abstrato;**
- **Modelo de processamento de dados:**



## Organização de computadores

- Basicamente todo computador pode ser dividido em seis unidades lógicas:
  - Unidade de entrada;
  - Unidade de saída;
  - Unidade de memória;
  - Unidade lógica e aritmética (ULA);
  - Unidade central de controle (UC);
  - Unidade central de processamento (UCP ou CPU – *central processing unit*);

## Organização de computadores



## Organização de computadores – unidade de entrada

- É a seção ‘receptora’ do computador;
- Ela obtém informações de vários dispositivos de entrada e as coloca à disposição das outras unidades, para processamento;
- A maioria das informações é inserida nos computadores por meio de dispositivos como teclados e mouses;
- Outras unidades de entrada: comandos de voz, varredura de imagens e códigos de barras, leitura de dispositivos de armazenamento secundários e recebimento de informações pela Internet (por exemplo, quando você baixa vídeos do YouTube™, e-books da Amazon e outros meios similares).

## Organização de computadores – unidade de saída

- É a seção de ‘expedição’ do computador;
- Pega as informações que foram processadas e as coloca em vários dispositivos de saída para torná-las disponíveis ao uso fora dele;
- A maior parte das informações que saem de computadores hoje em dia é exibida em telas, impressoras, tocadas em players de áudio (como os populares iPods® da Apple®) ou usada para controlar outros dispositivos;
- Os computadores também podem enviar suas informações para redes como a internet.

## Organização de computadores – unidade de memória

- Frequentemente chamada de **memória primária**, é a seção de ‘armazenamento’, com capacidade relativamente baixa;
- Retém informações obtidas por meio da unidade de entrada (que serão processadas), e também informações já processadas;
- As informações na unidade de memória são voláteis: elas normalmente se perdem quando o computador é desligado.

## **Organização de computadores – unidade lógica e aritmética (ULA)**

- **É a seção de ‘processamento’ do computador;**
- **Ela é responsável por executar cálculos como adição, subtração, multiplicação e divisão;**
- **Nos sistemas atuais, a ULA está próxima a UC, dentro da CPU;**



## **Organização de computadores – unidade central de processamento (CPU)**

- **É a seção ‘administrativa’ do computador;**
- **Ela coordena e supervisiona o funcionamento das outras seções;**
- **A CPU diz à unidade de entrada quando as informações devem ser lidas para a unidade de memória, diz à ULA quando as informações da unidade de memória devem ser utilizadas em cálculos e diz à unidade de saída quando enviar as informações da unidade de memória para certos dispositivos de saída;**

## Organização de computadores – unidade central de processamento

- Muitos dos computadores de hoje possuem várias CPUs;
- Podem realizar muitas operações simultaneamente — esses computadores são chamados de multiprocessadores;
- Um processador multi-core executa o multiprocessamento em um único chip integrado — por exemplo, um processador dual-core tem duas CPUs e um processador quad-core tem quatro CPUs;
- **Unidade de controle (UC):** é responsável por buscar um programa na memória, instrução por instrução, e executá-lo sobre os dados de entrada.

## Lógica de programação

- Em desenvolvimento de sistemas computacionais muito se fala em lógica de programação, mas o que é lógica de programação?
- Lógica é o que faz sentido;
- Ao escrever um programa devemos tomar cuidado para que os passos para resolver o problema tenham uma certa ordem de execução e também uma certa coerência, ou seja, tem que fazer sentido, tem que ter lógica;
- A lógica de programação é a técnica de encadear pensamentos para atingir um determinado objetivo;

## Lógica de programação

- O encadeamento gera uma sequência lógica;
- É por meio da lógica de programação que definimos a sequência lógica de passos (instruções) que o computador irá seguir para resolver um determinado problema;

## Teste de lógica

- **João é menor que Pedro. Pedro é menor que José. Podemos afirmar então que:**
  - a) João é maior que José.
  - b) João é igual a José.
  - c) José é maior que João.
  - d) Não podemos afirmar nada.
  
- **Qual o próximo número da série: 0,25 0,125 0,0625?**
  
- **Qual o próximo número da série: 1 1 2 3 5 8 ?**

## Algoritmos

- A palavra algoritmos é o termo central na ciência da computação e em programação de computadores;
- Um algoritmo é uma sequência lógica e finita de instruções para se atingir a solução de um problema;
- Exemplos de algoritmos no dia-a-dia: receita de bolo, tomar banho, trocar uma lâmpada, etc...;
- Um algoritmo não representa, necessariamente, um programa de computador, mas um programa de computador é um algoritmo que foi escrito em uma linguagem de programação;

## Exemplo de algoritmos

### ■ Algoritmo para calcular a média de três valores inteiros:

**Início**

**Fornecer o primeiro valor inteiro;  
Fornecer o segundo valor inteiro;  
Fornecer o terceiro valor inteiro;  
Somar os três valores;  
Pegar o resultado da soma e dividir por 3;  
Mostrar o resultado obtido;**

**Fim**

## Algoritmos

- De acordo com a figura do [slide 2](#), um algoritmo apresenta três partes: entrada, processamento e saída;
- Quais são as entradas do exemplo apresentado?
- Qual é o processamento?
- Qual é a saída?



## Exemplo de algoritmos

- De forma resumida, temos:

**Início**

Fornecer o primeiro valor inteiro;  
Fornecer o segundo valor inteiro;  
Fornecer o terceiro valor inteiro;

} **entrada**

Somar os três valores;

Pegar o resultado da soma e dividir por 3;

} **processamento**

Mostrar o resultado obtido;

} **saída**

**Fim**

## Exemplo de algoritmos

- **Analise o problema abaixo e descubra qual a cor do vestido de cada senhora:**
  - **Três senhoras: dona Branca, dona Rosa e dona Violeta – passeavam pelo parque quando dona Rosa disse:**
  - **Não é curioso que estejamos usando vestidos de cores branca, rosa e violeta, embora nenhuma de nós esteja usando um vestido de cor igual ao seu próprio nome?**
  - **Uma simples coincidência – respondeu a senhora com o vestido violeta**

## Exemplo de algoritmos

- Um homem precisa atravessar um rio com um barco que possui capacidade apenas para carregar ele mesmo e mais um de seus três pertences, que são: um lobo, uma cabra e um pacote de alfafa. Em cada viagem só poderá ir o homem e apenas um de seus pertences. A seguinte regra deverá ser respeitada: o lobo não pode ficar sozinho com a cabra e nem a cabra sozinha com o pacote de alfafa. Escreva um algoritmo para fazer a travessia dos pertences que estão em uma margem do rio para a outra (**uma travessia segura**).

## Exemplo de algoritmos

- **Criar um algoritmo que mova três discos de uma torre de Hanói, que consiste em três hastes (A, B e C), uma das quais serve de suporte para três discos de tamanhos diferentes (1, 2 e 3), os menores sobre os maiores. Pode-se mover um disco de cada vez para qualquer haste, contanto que nunca seja colocado um disco maior sobre um menor. O objetivo é transferir os três discos para outra haste. Escreva um algoritmo para resolver o problema apresentado.**

## Representação de algoritmos

- Um algoritmo pode ser representado basicamente de quatro formas diferentes:
  - Descrição narrativa;
  - Pseudocódigo;
  - Gráficos: fluxogramas e Nassin-Shneiderman-Chapin;
  - Linguagens de programação.

## Representação de algoritmos – descrição narrativa

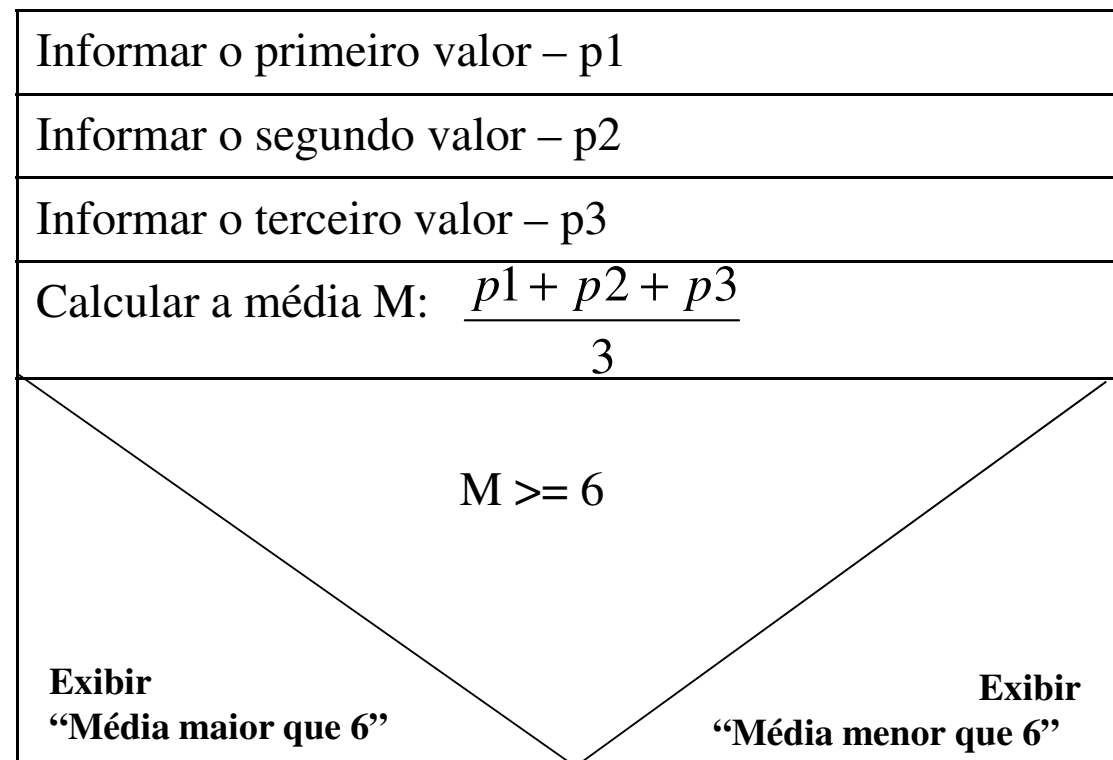
- Nesta forma de representação, os algoritmos são expressos diretamente em linguagem natural (linguagem falada);
- Exemplo:
  - Obter os três valores inteiros;
  - Somar os três valores;
  - Dividir o resultado da soma por três;
  - O resultado da divisão é a média entre os três valores informados.

## Representação de algoritmos – linguagem de programação

- Representa uma forma de escrever os algoritmos para que um computador possa entender e executar as instruções;
- Exemplo de um código em linguagem C:

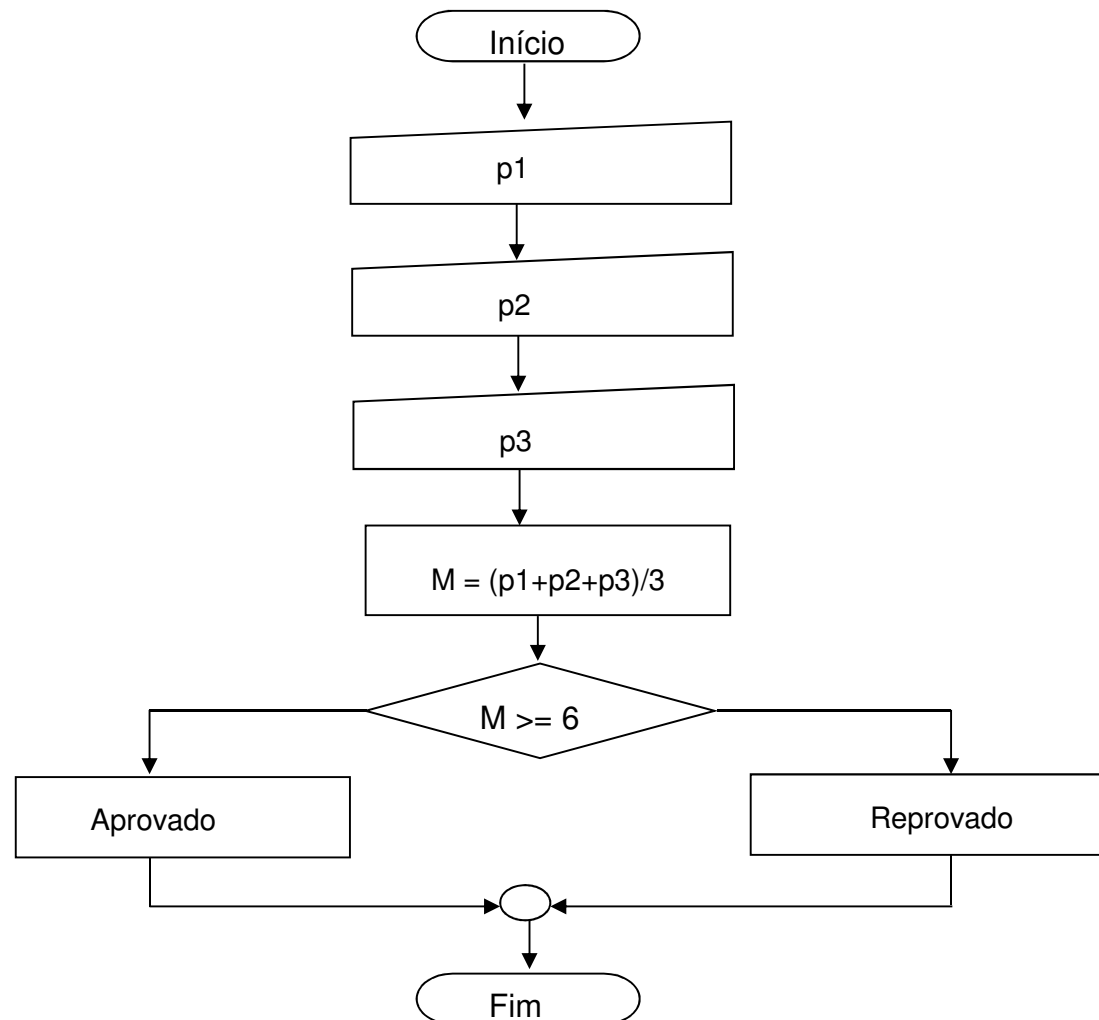
```
#include <stdio.h>
#include <stdlib.h>
int main() {
    int p1, p2, p3;
    float m;
    printf("Informe o primeiro valor: ");
    scanf("%d", &p1);
    printf("Informe o segundo valor: ");
    scanf("%d", &p2);
    printf("Informe o terceiro valor: ");
    scanf("%d", &p3);
    m = (p1+p2+p3)/3;
    printf("O resultado é %f", m);
    system("pause");
    return(0);
}
```

## Representação de algoritmos – diagrama de Nassin-Shneiderman-Chapin





## Representação de algoritmos - fluxograma



## Representação de algoritmos - pseudocódigo

- É uma forma simples de representar algoritmos;
- A pseudolinguagem usa a ideia de uma linguagem de programação, mas as instruções são escritas em linguagem natural;
- Vantagens: foco no desenvolvimento da lógica sem regras drásticas a serem memorizadas;
- Não há convenções e nem padronizações;

## Representação de algoritmos - pseudocódigo

### ■ Exemplo:

Início

```
inteiro p1, p2, p3
real m
imprimir("Informe o primeiro valor")
ler("%d", &p1)
imprimir("Informe o segundo valor")
ler("%d", &p2)
imprimir("Informe o terceiro valor")
ler("%d", &p3)
m = (p1+p2+p3)/3
imprimir("a média é ", m)
```

Fim

- Inicialmente adotaremos a pseudolinguagem como forma de escrever nossos algoritmos;

## Linguagem de programação

- Os primeiros programas de computadores eram escritos em linguagem de máquina;
- Linguagem de máquina representa um conjunto de instruções que todo computador possui e, que seu processador é capaz de executar;
- Exige conhecimento sobre o funcionamento da máquina para escrever programas corretamente;
- Para facilitar a programação de computadores surgiram as linguagens de programação;

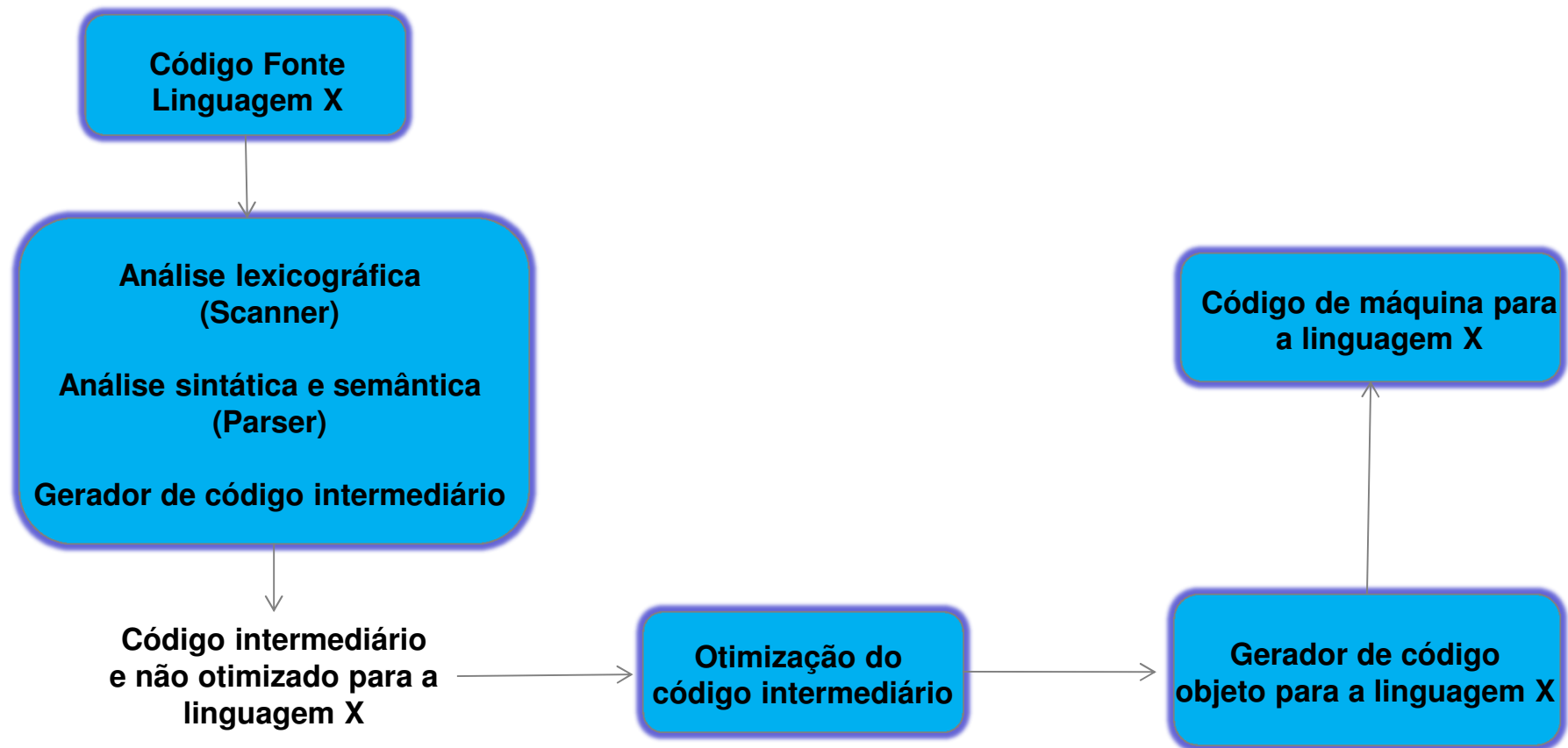
## Linguagem de programação

- Uma linguagem de programação nada mais é do que um conjunto de regras sintáticas e semânticas utilizadas para escrever instruções para que o computador possa executar;
- O computador entende apenas a sua linguagem nativa: linguagem de máquina;
- Para que um programa possa ser executado, o código fonte escrito em qualquer linguagem de programação deve ser traduzido para a linguagem de máquina;
- O processo de tradução é realizado por um programa ou vários programas;

## Linguagem de programação

- O processo de tradução pode ser por compilação ou interpretação;
- Na compilação o código fonte é convertido integralmente para código de máquina e depois executado. A execução é mais rápida, mas o código é dependente da plataforma;
- Linguagens compiladas: C, C++, C#, VB, Dephi, etc...;

## Linguagem de programação – processo de compilação



## Linguagem de programação

- No processo de interpretação, o interpretador traduz cada linha do código fonte para código de máquina e o executa imediatamente;
- A tradução e a interpretação ocorrem juntas;
- O código gerado é portátil, ou seja, não depende de sistema operacional;
- Por outro lado, a execução é mais lenta em relação ao processo de compilação;
- Exemplo de linguagens interpretadas: Java, C#, Python, PHP, etc...;



## Linguagem de programação

- As diversas linguagens de programação podem ser classificadas em **baixo nível** e **alto nível**;
- As linguagens de alto nível foram desenvolvidas com o propósito comercial, facilitando e agilizando o processo de desenvolvimento;
- Exemplo de linguagens de alto nível: Java, C#, Dephi, VB, PHP, etc...;
- As linguagens de baixo nível são linguagens onde as instruções são baseadas em linguagem de máquina;
- Exemplo: assembly (não assembler);

## Curiosidades

- O primeiro programa de computador foi criada por **Ada Lovelace**, filha do escritor inglês **Lord Byron**. Foi utilizado na calculadora programável criada por **Charles Babbage**;
- Uma das primeiras linguagens de programação para computadores foi **Plankalkül**, criada na Alemanha Nazista;
- A primeira linguagem de alto nível amplamente usada foi **Fortran** (criada em 1954);
- O primeiro compilador foi escrito por **Grace Hopper** em 1952 para a linguagem de programação **A-0**;

## Curiosidades

- Em 1957 foi criada **B-0**, sucessora de **A-0**, que daria origem a **Flow-Matic** (1958), antecessora imediata de **Cobol**, de 1959;
- A linguagem de programação **Simula 67** introduz o conceito de classes;

## Bibliografia

- DEITEL, Paul, DEITEL, Harvey. C – Como Programar. 6ª edição. São Paulo: Pearson Prentice Hall, 2011.