

# *Técnicas de Programação*

**Luiz Fernando Carvalho**

luizfcarvalhoo@gmail.com

# Algoritmos Computacionais

# Lógica de Programação



# Linguagem de Programação



Aplicação/Programa

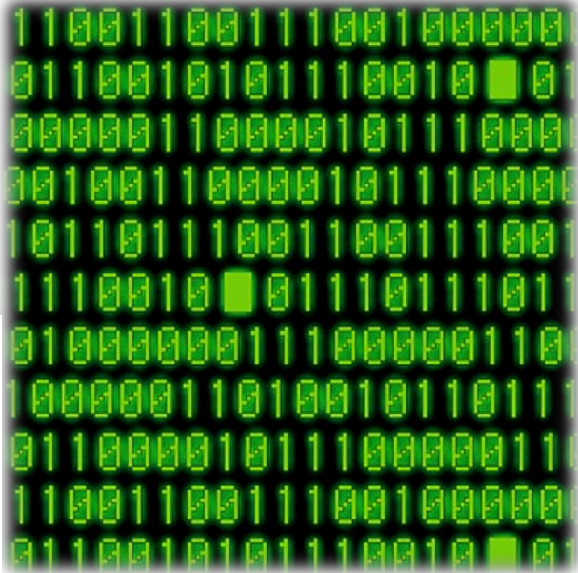


# Linguagem de Programação

- Uma linguagem de programação é um conjunto de ferramentas, regras de sintaxe e símbolos ou códigos que nos permitem escrever programas de computador;
- A primeira e mais primitiva linguagem de computador é a própria linguagem máquina (0's e 1's);
- Antigamente, um programa era difícil, longo e principalmente caro de construir;
- Era também difícil de ser entendido por outros programadores;
- Essa complexidade levou à necessidade de desenvolver novas técnicas e ferramentas.

# Linguagem de Programação

- A resolução do problema passou pela criação de uma linguagem em que os **códigos numéricos** foram **substituídos** por mnemônicos.
  - **Mnemônicos** são palavras usadas para facilitar a memorização de instruções de operações entre dados.



```
#include <stdio.h>
#include <stdlib.h>
#include <sys/types.h>
#include <arpa/inet.h>

void serveur1(portServ ports)
{
    int sockServ1, sockServ2, sockClient;
    struct sockaddr_in monAddr, addrClient, addrServ2;
    socklen_t lenAddrClient;

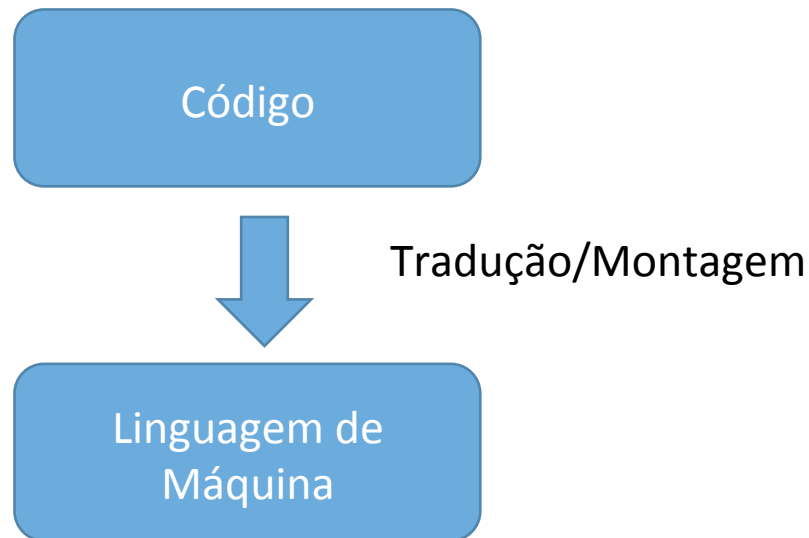
    if ((sockServ1 = socket(AF_INET, SOCK_STREAM, 0)) == -1) {
        perror("Erreur socket");
        exit(1);
    }
    if ((sockServ2 = socket(AF_INET, SOCK_STREAM, 0)) == -1) {
        perror("Erreur socket");
        exit(1);
    }

    bzero(&monAddr, sizeof(monAddr));
    monAddr.sin_family = AF_INET;
    monAddr.sin_port = htons(ports.port1);
    monAddr.sin_addr.s_addr = INADDR_ANY;
    bzero(&addrServ2, sizeof(addrServ2));
```

# Tradução das linguagens

- Montagem

- O processo de montagem traduz um programa escrito numa Linguagem de Programação em programa equivalente em linguagem máquina.



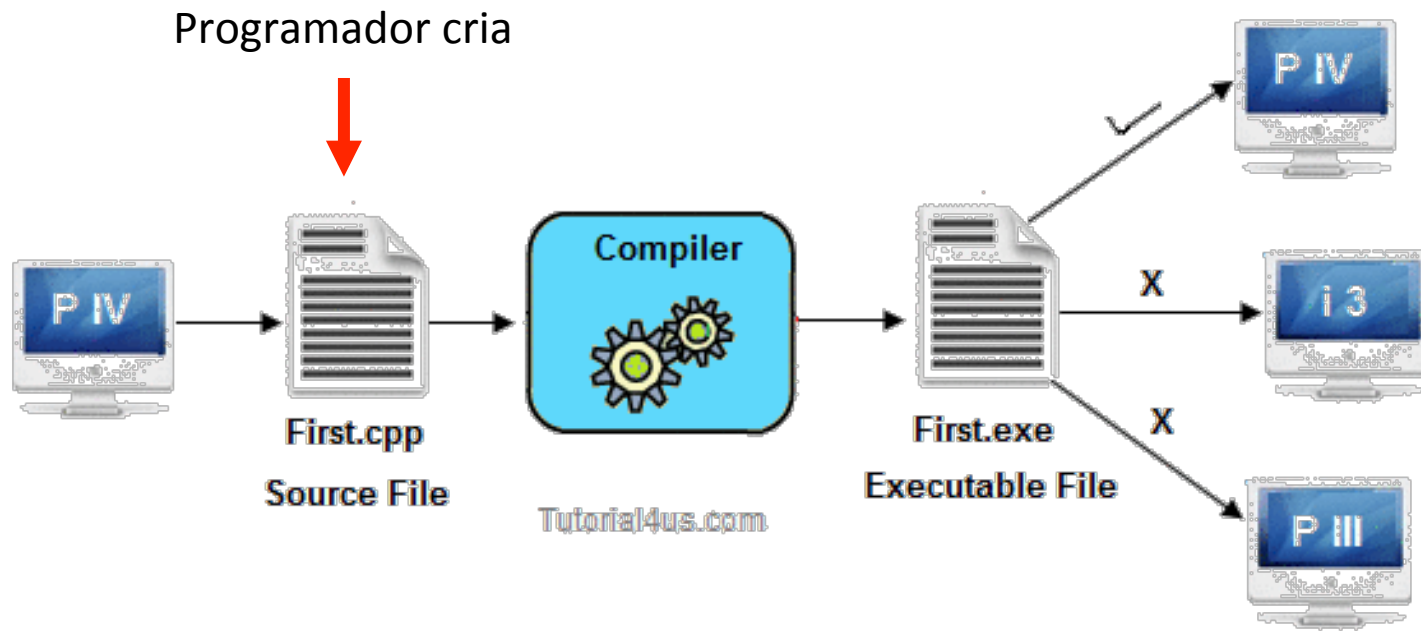
# Tradução das linguagens

- A Tradução/Montagem pode ser feito por duas classes de programas
  - Compiladores;
  - Interpretadores;

# Compilador

- Converte uma linguagem (Linguagem Fonte) de fácil escrita e leitura para os programadores, numa linguagem (Linguagem alvo ou objeto) que possa ser executada pelas máquinas.
- O código executável gerado pelo compilador é **dependente do sistema operacional** e da linguagem de **máquina** para o qual o código fonte foi traduzido.
  - Um programa compilado para uma máquina X, só funcionará corretamente em computadores que possuam sistema operacional e hardware equivalentes aos de X.
- A enorme variedade de compiladores existentes é bem vinda, visto que existem milhares de linguagens fonte, e as linguagens alvo são também muito variadas.

# Compilador





# Compilador

- Todo o código é analisado antes de gerar o código de máquina;
  - Reserva espaço na memória para as variáveis;
  - Verifica se existem erros:
    - Escrita errada de comandos;
    - Construção errada de comandos;
    - Comandos que não fazem sentido.
- Caso algum erro for encontrado, o compilador alertará o programador e não gerará o código de máquina;
- Depois de compilado, o código de máquina pode ser executado sem a necessidade de recompila-lo.
  - Basta executar o arquivo .exe (no Windows);

# Interpretadores

- O funcionamento dos interpretadores é muito parecido ao dos compiladores, mas...
- O interpretador **traduz** o código **linha a linha**.
- O código fonte não é totalmente traduzido antes de ser executado.
- Não existem fases distintas nem se produz código intermediário.
- Passa o **tempo todo lendo e traduzindo** código a medida que os comandos vão sendo executados.
  - Por exemplo: Navegadores;
  - Excel, Word, etc...
  - Python;

# Comparação

	Vantagem	Desvantagem
Compiladores	Execução mais rápida	Várias etapas de tradução
	Impossibilita que o código-fonte seja visualizado	Não permite a visualização do código-fonte
	Permite a otimização do código-fonte por parte do compilador	Processo de correção de erros é mais demorado
Interpretadores	Depurar o programa é mais simples	Execução do programa é mais lenta
	Geralmente consome menos memória	Necessita sempre ser lido do código-fonte original
	Resultado imediato do programa desenvolvido	Necessário fornecer o programa fonte ao utilizador

# Linguagem de Programação

- Foram desenvolvidas diversas linguagens de programação:
  - FORTRAN (1957)
  - ALGOL (1958)
  - COBOL (1959)
  - PASCAL (1963)
  - BASIC (1965)
  - ADA (1968)
  - C (1972) e mais tarde o C++ (1986)
  - Python (1991)
  - Java (1995)
  - Etc....
- Estas novas linguagens foram afastando cada vez mais o programador do nível de máquina.