



# **Engenharia de Computação**

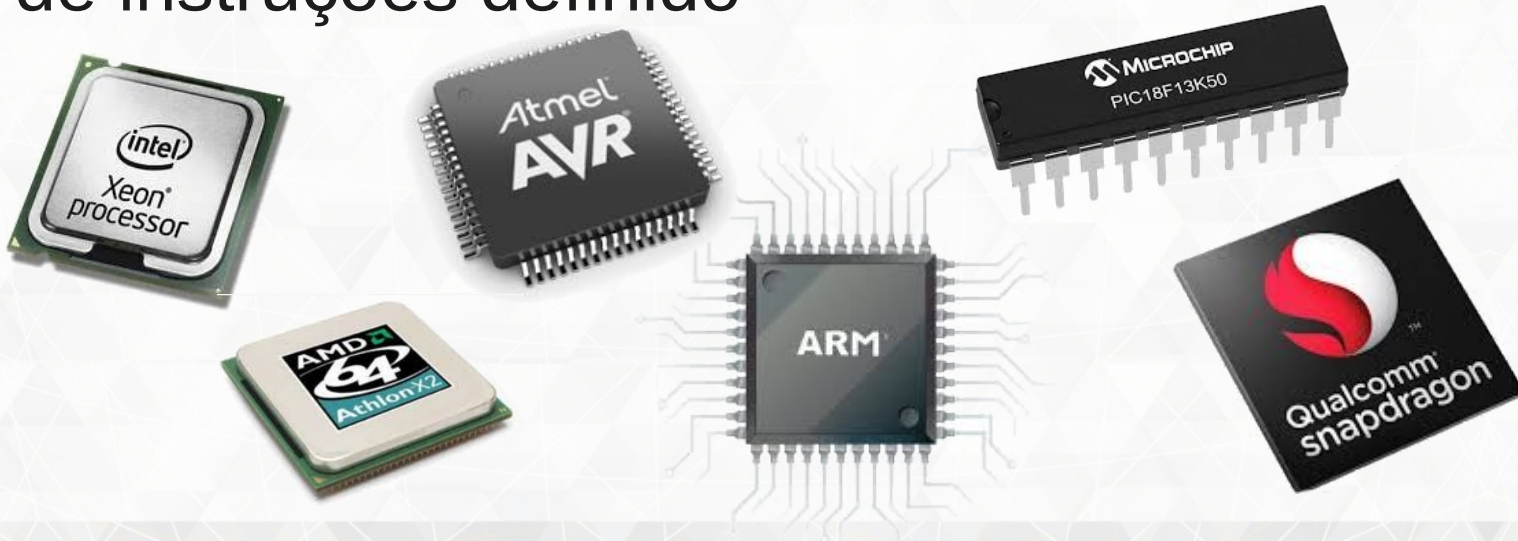
## **Fundamentos de Programação**

### **Aula 02 – Linguagem C**

**Prof. Muriel de Souza Godoi**  
muriel@utfpr.edu.br

# Linguagem/Código de Máquina

- Instruções em código de máquina
  - Consiste de uma sequência de bytes, que representam uma lista de instruções para ser executado por um determinado processador
- Cada família de processador possui um conjunto padrão de instruções definido



# Exemplo de Código de Máquina

- Programa em código de máquina x64 que mostra a mensagem Hello World na tela:

```
objdump -s -d hello64 , arquivo executável com o binário do código de máquina x64  
mostrado em hexadecimal. Ao lado direito mostra o binário convertido em Ascii
```

Conteúdo da seção .text:

```
4000b0: 48 c7 c0 01 00 00 00  c7 c7 01 00 00 00 48 c7  
4000c0: c6 de 00 60 00 48 c7  0d 00 00 00 0f 05 48 c7  
4000d0: c0 3c 00 00 00 48 c7  00 00 00 00 0f 05
```

```
H.....H.....H.  
...`.H.....H.  
.<...H.....
```

Conteúdo da seção .data:

```
6000de: 48 65 6c 6c 6f 20 57  6f 72 6c 64 21 0a
```

```
Hello World!.
```

Desmontagem da seção .text:

```
00000000004000b0 <_start>:
```

```
4000b0: 48 c7 c0 01 00 00 00      mov     $0x1,%rax  
4000b7: 48 c7 c7 01 00 00 00      mov     $0x1,%rdi  
4000be: 48 c7 c6 de 00 60 00      mov     $0x6000de,%rsi  
4000c5: 48 c7 c2 0d 00 00 00      mov     $0xd,%rdx  
4000cc: 0f 05                     syscall  
4000ce: 48 c7 c0 3c 00 00 00      mov     $0x3c,%rax  
4000d5: 48 c7 c7 00 00 00 00      mov     $0x0,%rdi  
4000dc: 0f 05                     syscall
```



# Linguagem mais alto nível

- Linguagem Assembly (montagem) possibilita códigos mais rápidos e enxutos. Porém:
  - Difícil programação
  - Dependente do hardware
  - Suscetível a erros
  - Códigos extensos e difíceis de manter
- Necessidade de linguagens de alto nível para abstrair a programação do código de máquina
  - **Linguagem C**

# Linguagem C: Origens

- A linguagem C nasceu na década de 70 (Dennis Ritchie);
- A linguagem C é derivada da linguagem B, criada por Ken Thompson. A linguagem B, por sua vez, veio da linguagem BCPL, inventada por Martin Richards;
- O C é uma linguagem genérica utilizada para a criação de programas diversos:
  - Planilhas e editores de textos, sistemas operacionais, programas de automação, banco de dados, automação industrial, dentre outros.

# Características

- Comumente chamada de linguagem de “médio nível”
  - Combina elementos de alto nível com a funcionalidade do assembly
  - Manipula bits, bytes e endereços
- Linguagem altamente portátil
  - Possível adaptar o mesmo software a diferentes tipos de computadores
  - Desde que se siga os padrões:
    - C K&R (1978)
    - ANSI C, C89, C90, C99, C11, C18



# Linguagem C

- Arquivo com Código Fonte (extensão .c)
  - Representa o Algoritmo desenvolvido
  - Instruções escritas na linguagem de programação C
  - É utilizado pelo Compilador C para transformar todas as instruções da linguagem C em código de máquina (arquivo executável)
- Compilador C
  - Existem vários, no caso usaremos o GNU C (gcc)
  - <http://gcc.gnu.org/>

# Estrutura Básica

- Um programa em C consiste de uma ou várias funções:

```
tipo_saida nome_funcao (parametros){  
    declaracao_variaveis  
    instrucoes  
    retorno;  
}  
//final do bloco da função
```

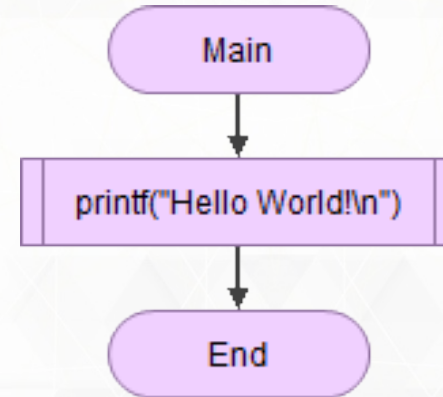


# Exemplo de Programa em C

- Primeiro programa em C

```
#include <stdio.h>
#include <stdlib.h>

int main(){
    printf("Hello World\n");
    return 0;
} //main
```



- Para exemplificar, salve esse código fonte acima em um arquivo hello.c com o editor de texto
- Para compilar no terminal ou prompt execute o comando:



```
gcc -Wall -o hello hello.c
```

# Exemplo de Programa em C

- Inclusão de bibliotecas das funções que serão utilizadas:

```
#include <stdio.h>
```

- Função principal (primeira a ser executada):

```
int main()
```

- Estabelece início e fim da função: { e }
- Instrução: `printf("Hello world!\n");`
- Retorno da função: `return 0;`
- Parênteses após o nome indicam que esta é uma função;
- Observe o ; no final de algumas linhas
- Qualquer nome exceto *main*;

# Exibindo Informações

- Função **printf**
  - Usada para **escrever** (imprimir) uma mensagem na tela
  - A mensagem deve estar entre aspas duplas
  - O **\n** insere uma quebra de linha

```
printf("<Seu Nome>\n");  
printf("Meu primeiro programa \n");  
printf("na linguagem C \n");
```



# Caracteres de Escape

Caractere	Significado
\a	Caractere (invisível) de aviso sonoro (bip)
\n	Caractere (invisível) de nova linha
\t	Caractere (invisível) de tabulação horizontal
\\	Caractere de barra invertida '\'
\'	Caractere de aspas simples
\"	Caractere de aspas duplas
\?	Sinal de Interrogação

# Comentários

- Em regra, são utilizados para a documentação do código;
- Objetiva facilitar o entendimento do programa por outros programadores ou o próprio criador;
- Os comentários em linguagem C podem ser feitos de duas formas:
  - `/* <comentário> */` - para um determinado trecho de código
  - `//` - para uma única linha de código, sendo colocado em seu início