



# LÓGICA DE PROGRAMAÇÃO

PROF<sup>a</sup>. M.Sc. JULIANA H Q BENACCHIO

# Diretivas do Pré-processador

- É possível incluir diversas instruções do compilador no código-fonte de um programa em C.
- Embora não sejam realmente parte da linguagem de programação C, expandem o escopo do ambiente de programação em C.
- É um comando que ocorre antes da compilação.

# Diretivas do Pré-processador

- O pré-processador C contém as seguintes diretivas:
  - **#if**
  - **#ifdef**
  - **#ifndef**
  - **#else**
  - **#elif**
  - **#endif**
  - **#include**
  - **#define**
  - **#undef**
  - **#line**
  - **#error**
  - **#pragma**

# Diretivas do Pré-processador

- Todas as diretivas do pré-processador começam com um #
- Cada diretiva do pré-processador deve estar na sua própria linha

# Diretiva #define

- Define um identificador e uma string que o substituirá toda vez que for encontrado no arquivo-fonte.
- O padrão C ANSI refere-se ao identificador como um nome de macro e ao processo de substituição como substituição de macro.
- A forma geral da diretiva é:

```
#define nome_macro string
```

# Diretiva `#define`

- O comando não deve ser finalizado com `;`
- Pode haver qualquer número de espaços entre o identificador e a string, mas, assim que a string começar, será terminada apenas por uma nova linha.
- Exemplo:

```
#define PI 3.14159265
```

# Diretiva #define

- Uma vez que um nome de macro tenha sido definido, ele pode ser usado como parte do programa

```
printf ("O numero Pi vale: %f",PI);  
area = (raio*raio)*PI;
```

# Diretiva `#define`

- Por exemplo, se deseja usar a palavra VERDADEIRO para o valor 1 e a palavra FALSO para o valor 0, você declarará duas macros `#define`

```
#define VERDADEIRO 1
```

```
#define FALSO 0
```



# Diretiva `#define`

- Os programadores C geralmente usam letras maiúsculas para identificadores definidos.
- Essa convenção ajuda qualquer um que esteja lendo o programa a saber de relance que uma substituição de macro irá ocorrer.
- Além disso, é melhor colocar todas os **`#defines`** no início do arquivo ou em um arquivo de cabeçalho separado em lugar de espalhá-los pelo programa.

# Diretiva `#define`

- Substituição de macro é usada mais frequentemente para definir nomes para os números que aparecem em um programa.
- Por exemplo, você pode ter um programa que defina uma matriz e tenha diversas rotinas que acessam essa matriz.
- Em lugar de fixar o tamanho da matriz com um número, você pode definir um tamanho e usar esse nome sempre que o tamanho da matriz for necessário.

# Diretiva #define

- Dessa forma, você só precisa fazer uma alteração e recompilar para alterar o tamanho da matriz.
- Por exemplo:

```
#define TAM 100
```

```
...
```

```
float salario[TAM];
```

```
...
```

```
for (i=0; i<TAM; i++)
```

```
    printf("%f", salario[i]);
```