



A Tecnologia e o Futuro num só Centro



# 0809 - Programação em C/C++ - Fundamentos

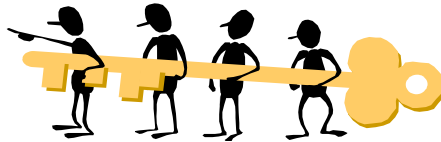
Luís Reis

Data: 15/11/2022

# Conceitos Básicos

- Estrutura básica de um programa
- Variáveis e tipos de dados
- Comandos básicos
- Estruturas de controlo de fluxo

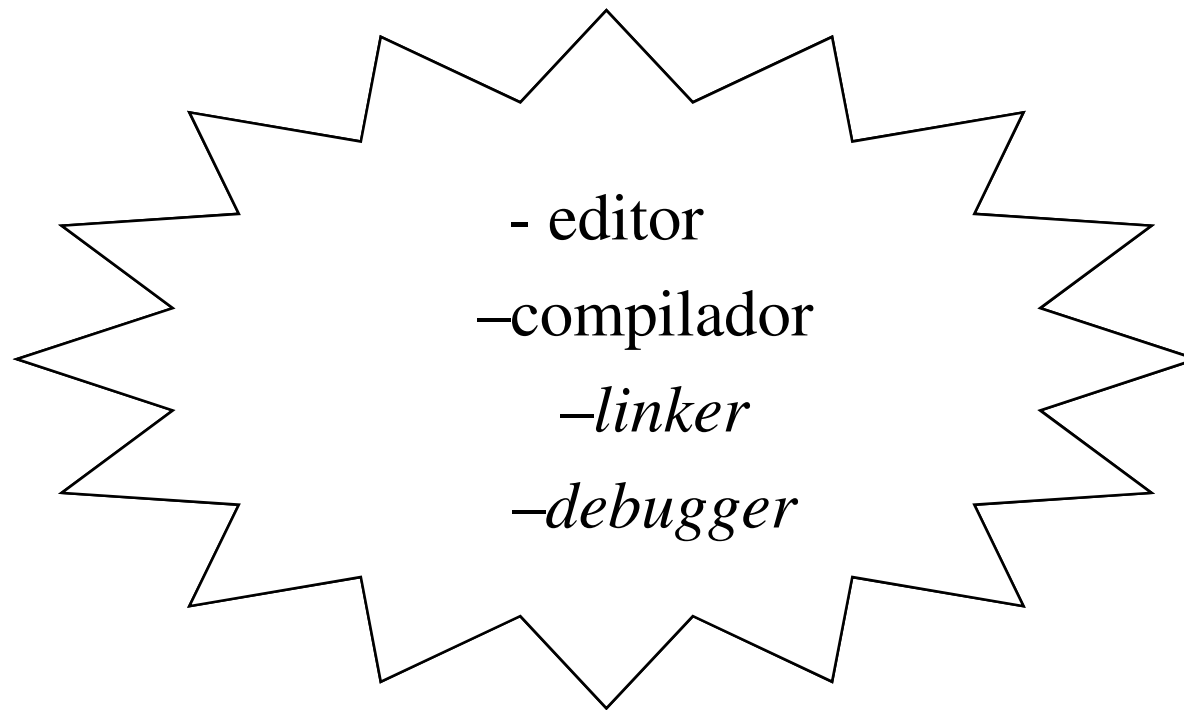
# 1º Programa em C++



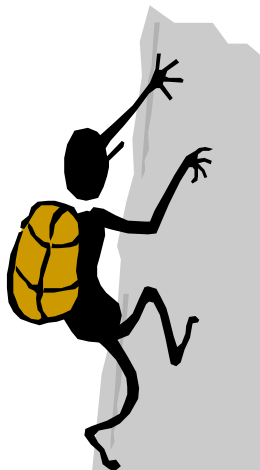
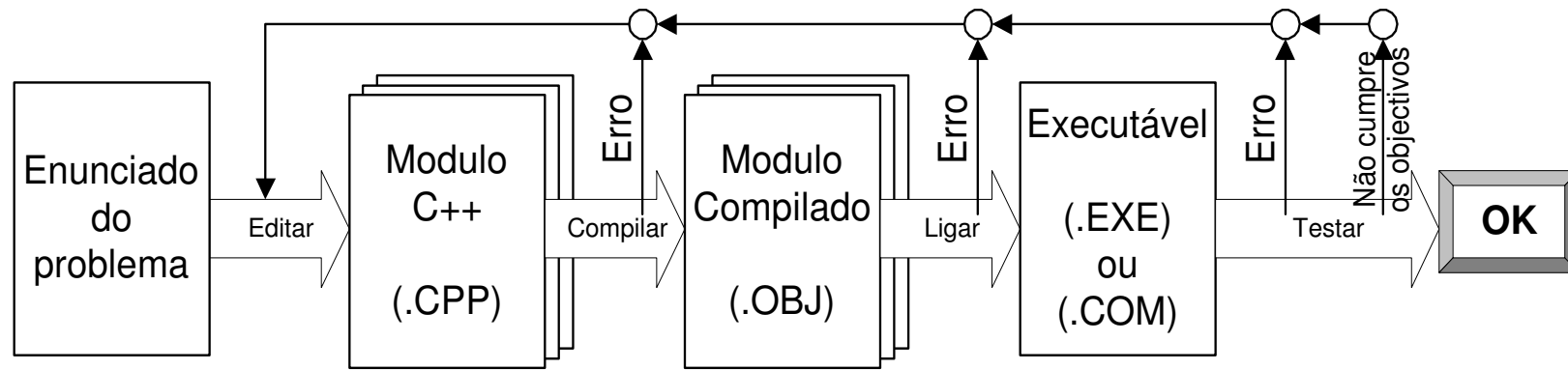
```
#include <iostream>  
int main() {  
  // Mensagem a mostrar no ecrã.  
  cout << "Hello, World!\n";  
  return 0;  
}
```

- Um programa em C++ é constituído por:
  - Várias funções, das quais uma obrigatoriamente tem que se chamar main.
    - A função main(), como qualquer outra é definida :
      - por um cabeçalho constituído por:
        - tipo de dados que a função devolve.
        - o seu nome.
        - parâmetros formais que recebe.
      - por um corpo (definido entre "{ }") com :
        - declarações
        - definições
        - instruções
        - comentários
  - Inclusão de ficheiros *header* com:
    - protótipos de funções, macros, instruções, declarações de tipos

# Ambiente de desenvolvimento



# Fases de geração de um programa



Sempre que um programa for constituído por vários módulos, torna-se necessário criar um projecto, indicando quais os módulos que devem ser ligados, na fase de criação do programa final.

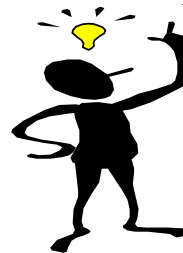


# Declarações e definições - variáveis

Em C/C++ qualquer identificador tem que ser declarado antes de ser usado.

**Declarar** uma entidade, consiste exclusivamente em **anunciar a sua existência**, explicitando-lhe o **nome** e o **tipo**.

No entanto, a maior parte das declarações, são também definições, isto é, definem a entidade que passa a ser designada por esse nome.



```
int y = 10; // variável y do tipo inteiro  
char c = 'k'; // variável c do tipo char, iniciada com o caracter 'k'
```

## Tipos primitivos de dados e sua representação

Tipos fundamentais	Significado
int	inteiro
char	carácter
float	vírgula flutuante ( <i>floating point</i> ) de precisão simples
double	vírgula flutuante de precisão dupla

Qualificadores	Aplica-se a	Representa
short	int	menor dimensão
long	int ; double	maior dimensão
signed	char; int	com sinal
unsigned	char; int	valor sem sinal



A partir dos **tipos de variáveis básicos**, definem-se outros tipos de variáveis ditos **tipos derivados**, tais como *arrays*, **enumerados**, **apontadores**, **referências**, **estrutura** e **uniões** desses tipos fundamentais (ou básicos).

# Alcance e espaço de memória ocupado

Tipo	Ambiente 16 bits	Ambiente 32 bits
signed int	2 bytes (de -32768 a +32767)	4 bytes (- 2 147 483 648 a + 2 147 483 647)
unsigned int	2 bytes (de 0 a 65535)	4 bytes (de 0 a 4 194 967 295)
float	4 bytes (de 3.4E-38 a 3.4E+38)	4 bytes (de 3.4E-38 a 3.4E+38)
signed char	1 byte (-128 a +127)	1 byte (-128 a +127)
unsigned char	1 byte (de 0 a 255)	1 byte (de 0 a 255)
short ou short int	2 bytes (Idêntico a int)	2 bytes (de -32768 a +32767)
unsigned short ou unsigned short int	2 bytes (Idêntico a unsigned int)	2 bytes (de 0 a 65535)
long ou long int	4 bytes (- 2 147 483 648 a + 2 147 483 647)	4 bytes (- 2 147 483 648 a + 2 147 483 647)
unsigned long int ou unsign long	4 bytes (0 a 4 194 967 295)	4 bytes (0 a 4 194 967 295)
double ou long double	8 bytes (1.7E-308 a 1.7E+308)	8 bytes (1.7E-308 a 1.7E+308)



## Representação de dados formatados no ecrã (*cout*)

```
#include <iostream>
using namespace std;

int main() {
    int int_num = 255;
    float float_num = 99.99895;
    // long int por omissão.
    long big_num = 1250500750.75;
    cout << "12345678901234567890\n";
    cout << int_num << endl;
    cout << big_num << endl;
    cout << float_num << endl;
}
```

Qual será o *output* deste programa



```
12345678901234567890
255
1250500750
99.9989
```

# Operadores e expressões

Como constituintes de um programa, constam também **operadores aritméticos, lógicos e relacionais** ( +, -, \*, /, = ).

Com **variáveis e operadores**, constroem-se **expressões** a que ficam associados valores. Expressões operam sobre variáveis de forma a produzir novos valores.



*Cálculo da área de um círculo*

```
#include <iostream> // Directiva para o pré processador.
Int main() {
    // Declaração (definição e iniciação) de uma constante do tipo float.
    const float PI = 3.14159265;
    int raio; // Declaração (definição) de um inteiro.
    float area; // Declaração (definição) de um float.

    // Instruções de saída e entrada de dados
    cout << "Qual o raio do círculo? ";
    cin >> raio;
    area = PI * raio * raio ; // Aqui têm que existir conversões de tipos.
    cout << "A area e" << area << endl; // Escrita do área no ecrã.
}
```

# Atividade 1

- Crie um programa que receba dois números inteiros do teclado, guarde os mesmos em duas variáveis diferentes, some os dois números e imprima o resultado no ecrã

Output: "O resultado de  $N1 + N2 = \text{resultado}$ "

- Crie um programa que permita calcular a área de um quadrado (LXL);
- Crie um programa que receba do teclado um valor de temperatura em graus Celsius e converta para Fahrenheit, imprimindo o resultado da conversão no ecrã.
  - Fórmula de conversão -  $F = 1,8C + 32$