

Algoritmos e Estruturas de Dados

(Aula 2 - Introdução C/C++)

Prof. Me. Diogo Tavares da Silva
contato: *diogotavares@unibarretos.com.br*

A Linguagem C/C++

Visão geral da linguagem:

```
#include <nome_da_biblioteca>
int main()
{
    bloco_de_comandos;
    return 0;
}
```

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

int main()
{
    cout << "Ola Mundo!" << endl;
    return 0;
}
```

A Linguagem C/C++

- Visão geral da linguagem:
 - Linguagem estática e fortemente tipada
 - Variável declarada antes de usada
 - sem conversão de tipos
 - Tipos primitivos:

Tipo	Faixa de valores	Tamanho (aproximado)
char	-128 a 127	8 bits
unsigned char	0 a 255	8 bits
int	-32.768 a 32.767	16 bits
unsigned int	0 a 65.535	16 bits
short int	-32.768 a 32.767	16 bits
long	-2.147.483.648 a 2.147.483.647	32 bits
unsigned long	0 a 4.294.967.295	32 bits
float	3.4×10^{-39} a 3.4×10^{39}	32 bits
double	1.7×10^{-308} a 1.7×10^{308}	64 bits
long double	3.4×10^{-4932} a 1.1×10^{4932}	80 bits

A Linguagem C/C++

- Visão geral da linguagem:
 - Operadores da linguagem:
 - Aritméticos

Operador	Exemplo	Comentário
=	<code>x = y</code>	O conteúdo da variável Y é atribuído à variável X (A uma variável pode ser atribuído o conteúdo de outra, um valor constante ou, ainda, o resultado de uma função).
+	<code>x + y</code>	Soma o conteúdo de X e de Y.
-	<code>x - y</code>	Subtrai o conteúdo de Y do conteúdo de X.
*	<code>x * y</code>	Multiplica o conteúdo de X pelo conteúdo de Y.
/	<code>x / y</code>	Obtém o quociente da divisão de X por Y. Se os operandos são inteiros, o resultado da operação será o quociente inteiro da divisão. Se os operandos são reais, o resultado da operação será a divisão. Por exemplo: <code>int z = 5/2;</code> → a variável z receberá o valor 2. <code>float z = 5.0/2.0;</code> → a variável z receberá o valor 2.5.
%	<code>x % y</code>	Obtém o resto da divisão de X por Y.

A Linguagem C/C++

- Visão geral da linguagem:
 - Operadores da linguagem:
 - Aritméticos

Operador	Exemplo	Comentário
<code>+=</code>	<code>x += y</code>	Equivale a <code>X = X + Y</code> .
<code>-=</code>	<code>x -= y</code>	Equivale a <code>X = X - Y</code> .
<code>*=</code>	<code>x *= y</code>	Equivale a <code>X = X * Y</code> .
<code>/=</code>	<code>x /= y</code>	Equivale a <code>X = X / Y</code> .
<code>%=</code>	<code>x %= y</code>	Equivale a <code>X = X % Y</code> .
<code>++</code>	<code>x++</code>	Equivale a <code>X = X + 1</code> .
<code>++</code>	<code>y = ++x</code>	Equivale a <code>X = X + 1</code> e depois <code>Y = X</code> .
<code>++</code>	<code>y = x++</code>	Equivale a <code>Y = X</code> e depois <code>X = X + 1</code> .
<code>--</code>	<code>x--</code>	Equivale a <code>X = X - 1</code> .
<code>--</code>	<code>y = --x</code>	Equivale a <code>X = X - 1</code> e depois <code>Y = X</code> .
<code>--</code>	<code>y = x--</code>	Equivale a <code>Y = X</code> e depois <code>X = X - 1</code> .

A Linguagem C/C++

- Visão geral da linguagem:
 - Operadores da linguagem:
 - Comparativos

Operador	Exemplo	Comentário
==	x == y	O conteúdo de X é igual ao conteúdo de Y.
!=	x != y	O conteúdo de X é diferente do conteúdo de Y.
<=	x <= y	O conteúdo de X é menor ou igual ao conteúdo de Y.
>=	x >= y	O conteúdo de X é maior ou igual ao conteúdo de Y.
<	x < y	O conteúdo de X é menor que o conteúdo de Y.
>	x > y	O conteúdo de X é maior que o conteúdo de Y.

A Linguagem C/C++

- Visão geral da linguagem:
 - Operadores da linguagem:
 - Lógicos
 - Operador **E**
 - **&&**
 - Operador **OU**
 - **||**
 - Operador **NÃO**
 - **!**
 - **NÃO EXISTE VARIÁVEL LÓGICA EM C/C++**
 - Falso → 0, null ou ""
 - Verdadeiro → qualquer outro valor não nulo

A Linguagem C/C++

- Leitura e escrita de dados (em C++)
 - Leituras/escritas por objetos “stream” (fluxos)
 - **cin >>**
 - leitura de dados
 - **cout <<**
 - saída de dados

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

int main()
{
    int a,b;
    cout << "Valor de a:";
    cin >> a;
    cout << "Valor de b:";
    cin >> b;
    cout << "soma de a+b = " << a+b << endl;
    return 0;
}
```


A Linguagem C/C++

- Leitura e escrita de dados
 - Leituras/escritas por objetos “stream” (fluxos)
 - **cin >>**
 - leitura de dados
 - **cout <<**
 - saída de dados
- Mais sobre:
 - <https://www.inf.pucrs.br/~pinho/PRGSWB/Streams/streams.html>

A Linguagem C/C++

- Leitura e escrita de dados
- Em C:
 - Uso da biblioteca **<stdio.h>**
 - **#include <stdio.h>**
- Principalmente:
- puts()
- scanf()
 - <https://petbcc.ufscar.br/stdiofuncoes/#scanf>
- printf()
 - <https://petbcc.ufscar.br/stdiofuncoes/#printf>
- saiba mais:
 - https://www.tutorialspoint.com/cprogramming/c_input_output.htm

A Linguagem C/C++

- estruturas de decisão
- **if-else**

```
if (condição)
{
    comando1;
    comando2;
}
else
{
    comando3;
    comando4;
}
```

A Linguagem C/C++

- estruturas de decisão
- **switch-case**

```
switch (variável)
{
    case valor1: lista de comandos;
                break;
    case valor2: lista de comandos;
                break;
                ....
    default: lista de comandos;
}
```

A Linguagem C/C++

- estruturas de controle
- **for**

```
for (i = valor_inicial; condição; incremento ou decremento de i)
comando;
```

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

```
int main()
{
    for (int i=0; i<10; i++){
        if (i % 2 == 0){
            cout << "i: " << i << ". Número par!" << endl;
        }else{
            cout << "i: " << i << ". Número ímpar!" << endl;
        }
    }
}
```

A Linguagem C/C++

- estruturas de controle
- **while**

```
while(condição)
{ comando1;
  comando2;
  comando3;
  ...
}
```

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

int main()
{
    int soma = 0;
    int num = 1;

    while (num <= 10) {
        soma = soma+num;
        num++;
    }
}
```



A Linguagem C/C++

- estruturas de controle
- **do-while**

```
do
{
    comandos;
}
while (condição);
```

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

int main()
{
    int num;
    do{
        cout << "Digite um número menor que 10: ";
        cin >> num;
    }while (num > 10);
}
```

Próxima aula

- funções
- vetores e matrizes