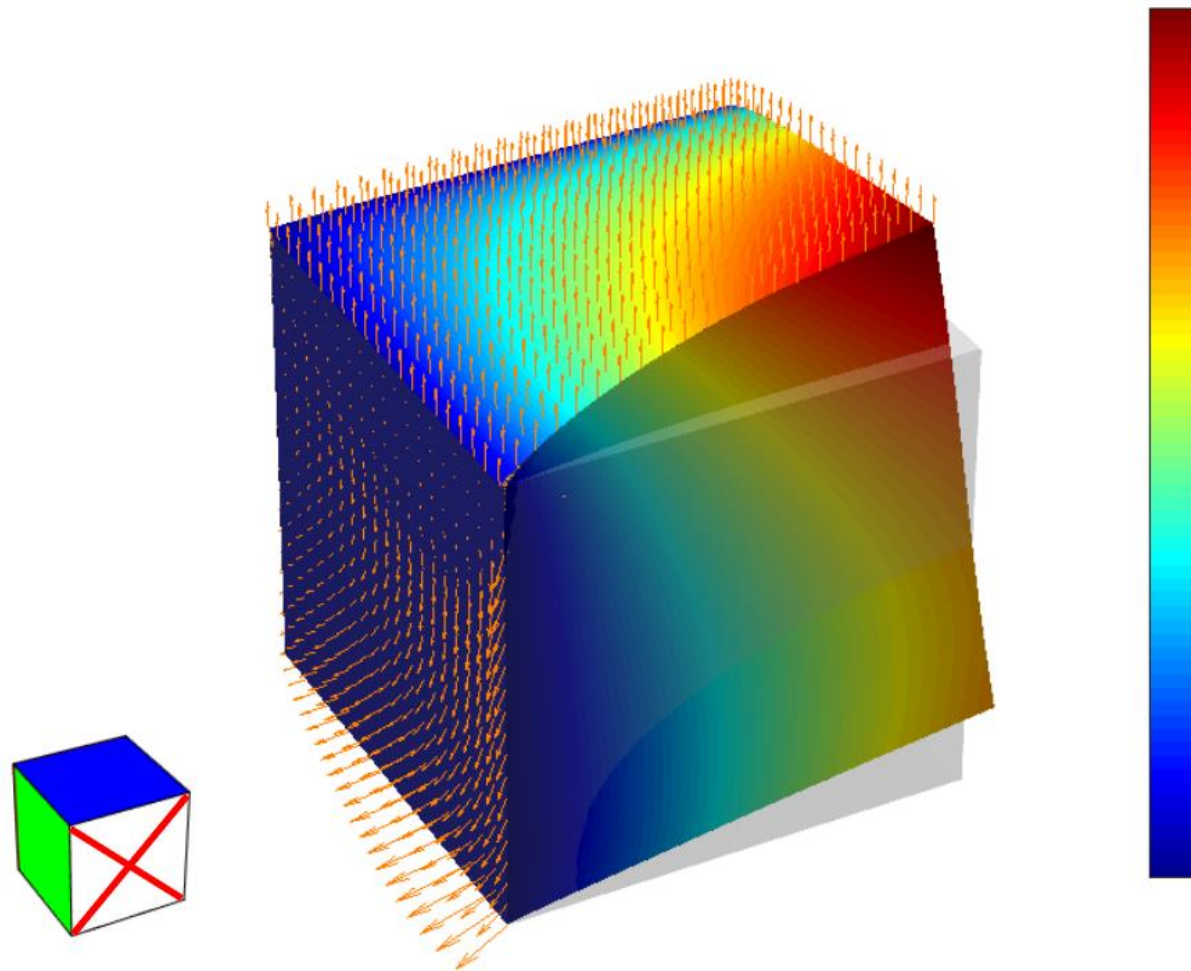


# Engenharia Elétrica/Mecânica

## Métodos Numéricos Computacionais



**Prof. Márcio Artacho Peres**  
[marcio.peres@ifms.edu.br](mailto:marcio.peres@ifms.edu.br)

# Engenharia Elétrica/Mecânica

## Métodos Numéricos Computacionais



## C++

**C++** é uma linguagem de programação compilada multi-paradigma.

Desde os anos 1990 é uma das linguagens comerciais mais populares, sendo bastante usada também na academia por seu grande desempenho.

Bjarne Stroustrup desenvolveu o C++. Serviu de inspiração para outras linguagens como Java e C#.

Versão C++20 (15 de dezembro de 2020).

<https://cplusplus.com/reference/>

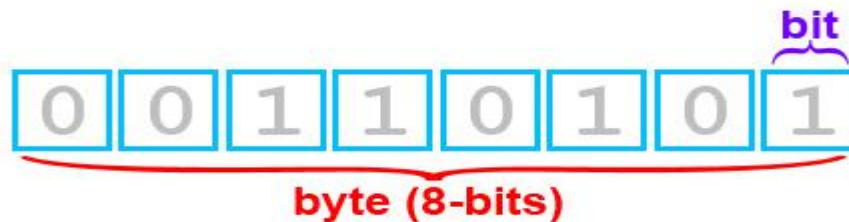
# Engenharia Elétrica/Mecânica

## Métodos Numéricos Computacionais



### Tipos de dados C++

Tipos	bytes	Intervalo
char	1	Caracteres ASCII: 'a' 'b' 'c' '0' '1' '%'...
boolean	1	true ou false
short	2	Inteiro: -32768 a 32767
int	4	Inteiro: -2147483648 a 2147483647
long	8	Inteiro: -9223372036854775808 a 9223372036854775807
float	4	Ponto flutuante: +/-3.4E-38 a +/-3.4E+38
double	8	Ponto flutuante: +/-1.7E-308 a +/-1.7E+308



# Engenharia Elétrica/Mecânica

## Métodos Numéricos Computacionais



## Tipos de dados C++

```
#include <iostream>

#include <limits>

using namespace std;

int main() {

    std::cout << "int \t" << sizeof(int) << "\t" <<
numeric_limits<int>::min() << "\t" <<
numeric_limits<int>::max();

}
```

# Engenharia Elétrica/Mecânica

## Métodos Numéricos Computacionais



## Tipos de dados C++

O que acontece??

```
int largeNum = 5000000;  
short smallNum = largeNum;
```

Confira na calculadora (Programador) do Windows!!! Veja os valores em binário.

0000 0000 0100 1100 0100 1011 0100 0000

# Engenharia Elétrica/Mecânica

## Métodos Numéricos Computacionais



## Saída de Dados em C

### Função PRINTF

A principal função de saída de dados da Linguagem C.

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

```
int main()
{
    printf ("Characters: %c %c \n", 'a', 65);
    printf ("Decimals: %d %ld\n", 1977, 650000L);
    printf ("Preceding with blanks: %10d \n", 1977);
    printf ("Preceding with zeros: %010d \n", 1977);
    printf ("Some different radices: %d %x %o %#x %#o \n", 100, 100,
100, 100, 100);
    printf ("floats: %4.2f %+.0e %E \n", 3.1416, 3.1416, 3.1416);
    printf ("Width trick: %*d \n", 5, 10);
    printf ("%s \n", "A string");
    return 0;
}
```

# Engenharia Elétrica/Mecânica

## Métodos Numéricos Computacionais



## Entrada de Dados em C

### Função SCANF

A principal função de entrada de dados da Linguagem C é **scanf**.

Para usá-la você precisa definir a **string de formato** e as variáveis onde os dados lidos serão armazenados.

A *string de formato* é o primeiro parâmetro da **scanf** e segue as mesmas regras da função **printf**. Por exemplo,

```
int idade, peso;  
float altura;
```

```
printf("Digite a idade, a altura e o peso, separados  
por um espaço em branco.\n");  
scanf("%d%f%d", &idade, &altura, &peso);
```

# Engenharia Elétrica/Mecânica

## Métodos Numéricos Computacionais



**.Operadores aritméticos** são aqueles que efetuam operações aritméticas em um ou mais tipos primitivos.

**.Operadores Binários**, atuam sobre dois operadores:

+	$a+1$	soma o valor de a com 1
-	$b-2$	subtrai 2 do valor de b
*	$c*x$	multiplica c por x
/	$d/e$	divide d por e     ( <b>cuidado</b> div por 0)
%	$f\%g$	resto da divisao de f por g     ( $f \bmod g$ )



# Engenharia Elétrica/Mecânica

## Métodos Numéricos Computacionais



Operadores aritméticos **Unários**, que atuam sobre um único identificador.

**++** Incremento

– var++ / ++var Retorna e adiciona / adiciona e retorna.

**--** Decremento

– var-- / --var Retorna e subtrai / subtrai e retorna.

**-** Negativo      -var      Inverte o sinal da variável

**+** Positivo      +var      Não tem efeito.

**~** Inversão e Incremento

**!** Negação

# Engenharia Elétrica/Mecânica

## Métodos Numéricos Computacionais



Os operadores Lógicos – em variáveis do tipo lógico

**!**      **Negação**       $!x$       Se  $x$  verdadeiro passa para falso e vice-versa.

**&&**      **E lógico**       $x \ \&\& \ y$       **E** lógico entre  $x$  e  $y$ .

**||**      **OU lógico**       $x \ || \ y$       **OU** lógico entre  $x$  e  $y$ .

```
boolean x = true;
```

```
boolean y = false;
```

```
boolean ze = x && y;
```

```
boolean zou = x || y;
```

<u><b>E</b></u>	True	False
True	True	False
False	False	False

<u><b>OU</b></u>	True	False
True	True	True
False	True	False

# Engenharia Elétrica/Mecânica

## Métodos Numéricos Computacionais



O operador de atribuição composto junta um operador aritmético com o operador de atribuição.

<code>int x = 1;</code>	<code>// Cria a variavel x e atribui 1 a x</code>	
<code>x += 5;</code>	<code>// Equivalente a x = x + 5;</code>	x é igual a 6
<code>x *= 2;</code>	<code>// Equivalente a x = x * 2;</code>	x é igual a 12
<code>x /= 2;</code>	<code>// Equivalente a x = x / 2;</code>	x é igual a 6
<code>x -= 3;</code>	<code>// Equivalente a x = x - 3;</code>	x é igual a 3
<code>x %= 5;</code>	<code>// Equivalente a x = x % 5;</code>	x é igual a 3

# Engenharia Elétrica/Mecânica

## Métodos Numéricos Computacionais



### .Operadores de igualdade

=	==	$x == y$
$\neq$	!=	$x != y$

### .Operadores relacionais

>	>	$x > y$
<	<	$x < y$
$\geq$	$\geq$	$x \geq y$
$\leq$	$\leq$	$x \leq y$

# Engenharia Elétrica/Mecânica

## Métodos Numéricos Computacionais



### .Estrutura condicional

```
if (condição)
    ...
else
    ...
```

### .Estrutura de repetição

```
for( inicialização ; condição ; incremento) {
    ...
}
```

```
while (condição) {
    ...
}
```

# Engenharia Elétrica/Mecânica

## Métodos Numéricos Computacionais



### Exercício

Escreva um programa em C++ que leia um número e calcule a raiz quadrada de um número positivo, usando o roteiro abaixo, baseado no método de aproximações sucessivas de Newton:

Seja  $Y$  o número lido:

- A primeira aproximação para a raiz quadrada de  $Y$  é:

$$X_1 = Y/2$$

- as sucessivas aproximações serão:

$$X_{n+1} = ((X_n * X_n) + Y) / (2 * X_n)$$

O programa deve calcular enquanto a diferença entre os termos seja maior que 0,0001 (tolerância).