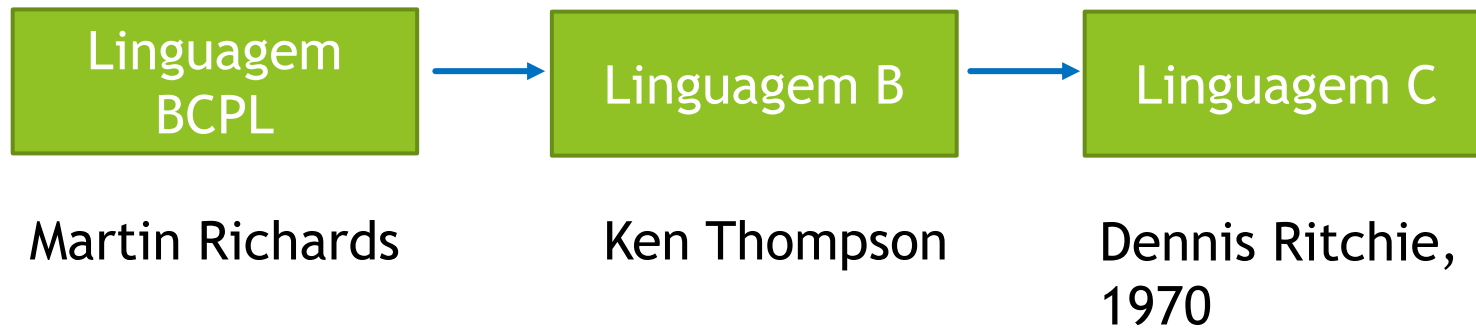


# INTRODUÇÃO À LINGUAGEM C/C++

# INTRODUÇÃO À LINGUAGEM C

## ► História



# INTRODUÇÃO À LINGUAGEM C/C++

## ► Palavras reservadas

*auto*  
*break*  
*case*  
*char*  
*const*  
*continue*  
*default*  
*do*

*double*  
*else*  
*enum*  
*extern*  
*float*  
*for*  
*goto*  
*if*

*int*  
*long*  
*register*  
*return*  
*short*  
*signed*  
*sizeof*  
*static*

*struct*  
*switch*  
*typedef*  
*union*  
*unsigned*  
*void*  
*volatile*  
*while*

# INTRODUÇÃO À LINGUAGEM C/C++

## ► Primeiro programa em C

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

← Bibliotecas (funções no arquivo .h)

```
int main()
{
    printf("\n Bom dia!");
    system("pause");
    return 0;
}
```

← Programa principal

Mas o que é o #include?

# INTRODUÇÃO À LINGUAGEM C/C++

## ► Primeiro programa em C++

```
#include<iostream>

using namespace std;

int main(){
    cout<<"Ola mundo!!!"<<endl;
}
```

← Bibliotecas (funções no arquivo .h)

← Programa principal

Mas o que é o #include?

# INTRODUÇÃO À LINGUAGEM C

## ► Primeiro programa

- **#include:** diretiva do pré-processador que permite atribuir um arquivo no código-fonte
  - ✓ Comando utilizado pelo pré-processador antes da compilação
  - ✓ A diretiva é substituída pelo conteúdo do arquivo

# INTRODUÇÃO À LINGUAGEM C

- **Comentários:** são ignorados pelo compilador

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

int main()
{
    printf("\nBom dia!");

    //comenta uma linha

    /*
    comenta
    um
    bloco
    */

    system("pause");
    return 0;
}
```

# INTRODUÇÃO À LINGUAGEM C

## Uso de variáveis em linguagens de programação

- Variável é um espaço na memória, reservado para armazenar um valor
- Assim, sua posição é referenciada por um nome; o nome da variável
- Quanto de memória é reservado para uma variável? Isso depende do tipo de dado da variável

Endereço de  
memória

1 byte é a menor quantidade de memória para armazenar algo

0	1	0	0	0	0	0	1
0	1	2	3	4	5	6	7

Dados (letra A em  
binário)

8



# INTRODUÇÃO À LINGUAGEM C

## Alguns tipos primitivos de dados na Linguagem C

- Diz respeito ao tamanho de memória que a variável vai ocupar, em Bytes, e como seu valor será interpretado e armazenado

Tipo de Dado	Formato	Bytes
Char	%c	1
int	%d ou %i	4
float	%f	4
double	%lf	8

Armazena um caractere

Armazena um número inteiro

Armazena um número decimal

Armazena um número decimal

```
printf("%d",sizeof(int));
```

- Toda variável deve ser declarada antes de usada. Declarar significa atribuir à variável um tipo de dado
- Na linguagem C++ não precisamos colocar os formatos.

# INTRODUÇÃO À LINGUAGEM C

## ► Exemplo de declaração/inicialização de variáveis

```
#include<stdio.h>

int main(){

    int valo1 = 10, valor2;
    float pi = 3.14, base;
    char sexo = 'M';

    return 0;
}
```

Em Linguagem C, as variáveis são ditas ***Case Sensitive***, isto é, letras maiúsculas e minúsculas são diferenciadas

Exemplo:

São diferentes as seguintes variáveis:

```
int a, A;
float Nota1, nota1, NOTA1, NoTa1;
```

# INTRODUÇÃO À LINGUAGEM C

- ▶ **Algumas funções da biblioteca stdio.h (standard input/output)**
- ▶ Entrada e saída padrão formatadas (formatos de dados diferentes) - printf e scanf
- ▶ `printf("");` //exibe algo no vídeo. Pode ser texto e/ou valor de variável
- ▶ `scanf("%d", &a);` //recebe dados via teclado. A variável a recebe valor informado pelo usuário
- ▶ Exemplo:
- ▶ `scanf("%d", &a);`
- ▶ `scanf("%d", &b);`
- ▶ `printf("Soma = %d", a+b);` // %d será substituído por a+b

# INTRODUÇÃO À LINGUAGEM C

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

```
int main(){
```

```
    int a;
```

```
    float p;
```

```
    char letra;
```

```
    char nome[15]; //máximo de posições na cadeia de caracteres
```

```
    printf("Ola. printf imprime dados e textos no video"); //printf mostra texto e dados
```

```
    printf("\n Informe um valor inteiro"); // \n pula uma linha
```

```
    scanf("%d", &a); //scanf recebe um dado
```

```
    printf("Voce digitou %d", a); //%d é o formato para variável do tipo int
```

```
    printf("Digite um valor decimal");
```

```
    scanf("%f", &p); //%f é o formato para variável do tipo float
```

```
    printf("Voce digitou %f", p);
```

```
    printf("Agora, informe uma letra ou caractere");
```

```
    fflush(stdin); //antes de receber um caractere, limpar o buffer do teclado
```

```
    scanf("%c", &letra); //%c é o formato para variável do tipo char
```

```
    printf("Caractere informado %c", letra);
```

```
    printf("Digite seu primeiro nome");
```

```
    fflush(stdin);
```

```
    scanf("%s", &nome); //%s é o formato para cadeia de caracteres
```

```
    printf("Seu nome eh %s", nome);
```

```
    return 0;
```

```
}
```

► Algumas funções da biblioteca stdio.h (standard input/output)

► Entrada e saída padrão formatadas (formatos de dados diferentes) - printf e scanf

# INTRODUÇÃO À LINGUAGEM C

## ► Alguns operadores aritméticos

► + - \* / %

► O operador % é chamado de módulo. Ele retorna o resto de uma divisão inteira

```
#include<stdio.h>
int main(){
    int a, b;

    printf("\n Digite um valor 1: ");
    scanf("%d", &a); //
    printf("\n Digite um valor 2: ");
    scanf("%d", &b); //

    printf("divisao = %d", a/b);
    printf("resto = %d", a%b);

    return 0;
}
```

Simulação			
a	b	a/b	a%b
1	1	1	0
4	2	2	0
9	2	4	1

Resto da divisão  
inteira

# EXERCÍCIOS - LINGUAGEM C

- ▶ A. Faça uma solução para calcular e mostrar a área de um círculo
- ▶ B. Faça uma solução para calcular e exibir a área de um triângulo
- ▶ C. Faça uma solução para ler dois números inteiros. Em seguida, mostre o resultado das seguintes operações:
  - Soma
  - Subtração
  - Divisão
  - Módulo
  - Dobro dos dois
  - Quadrado dos dois