

# Fundamentos de Programação 1

---

Slides N. 2 - A

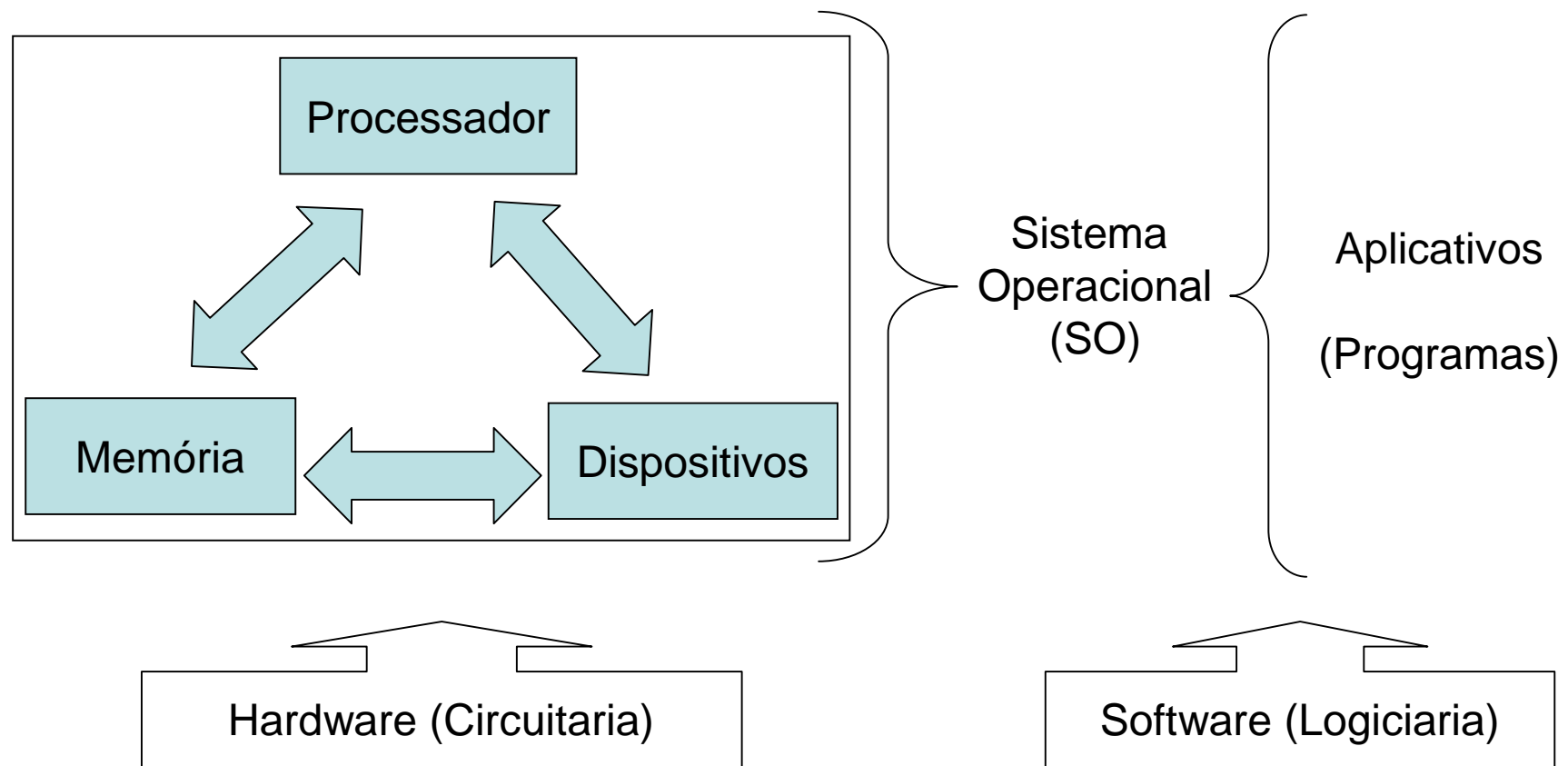
Prof. SIMÃO

Jean Marcelo SIMÃO

# Computador

---

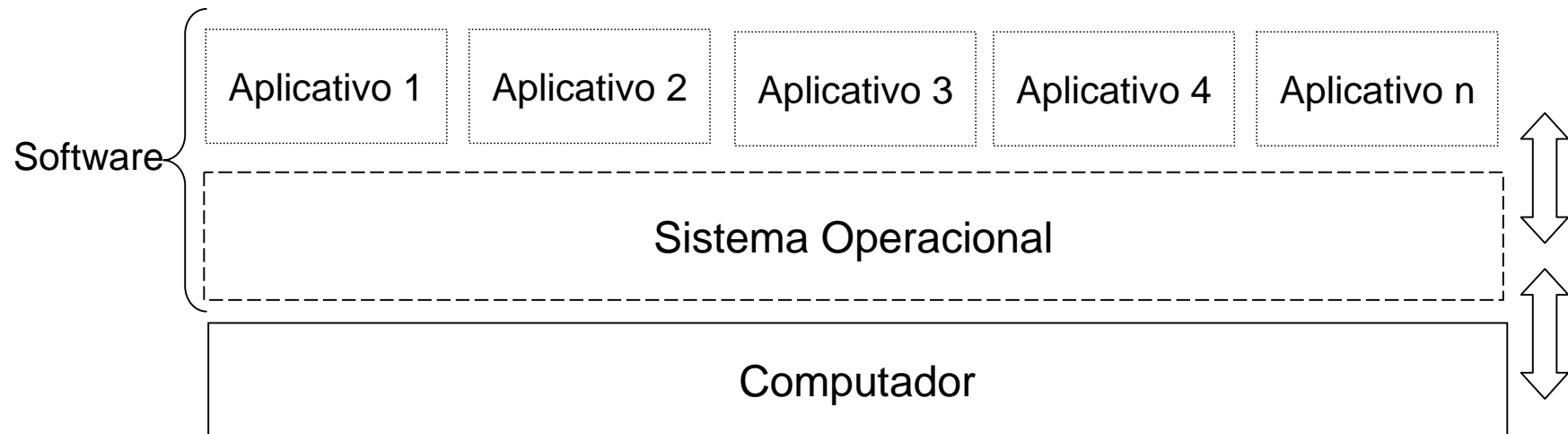
- Noção elementar de um computador.
  - Dispositivo para processar informação.



# Sistema Operacional

---

- O Sistema Operacional gerencia as relações entre as partes do computador e também os aplicativos (programas) que são executados nele.



# Software

---

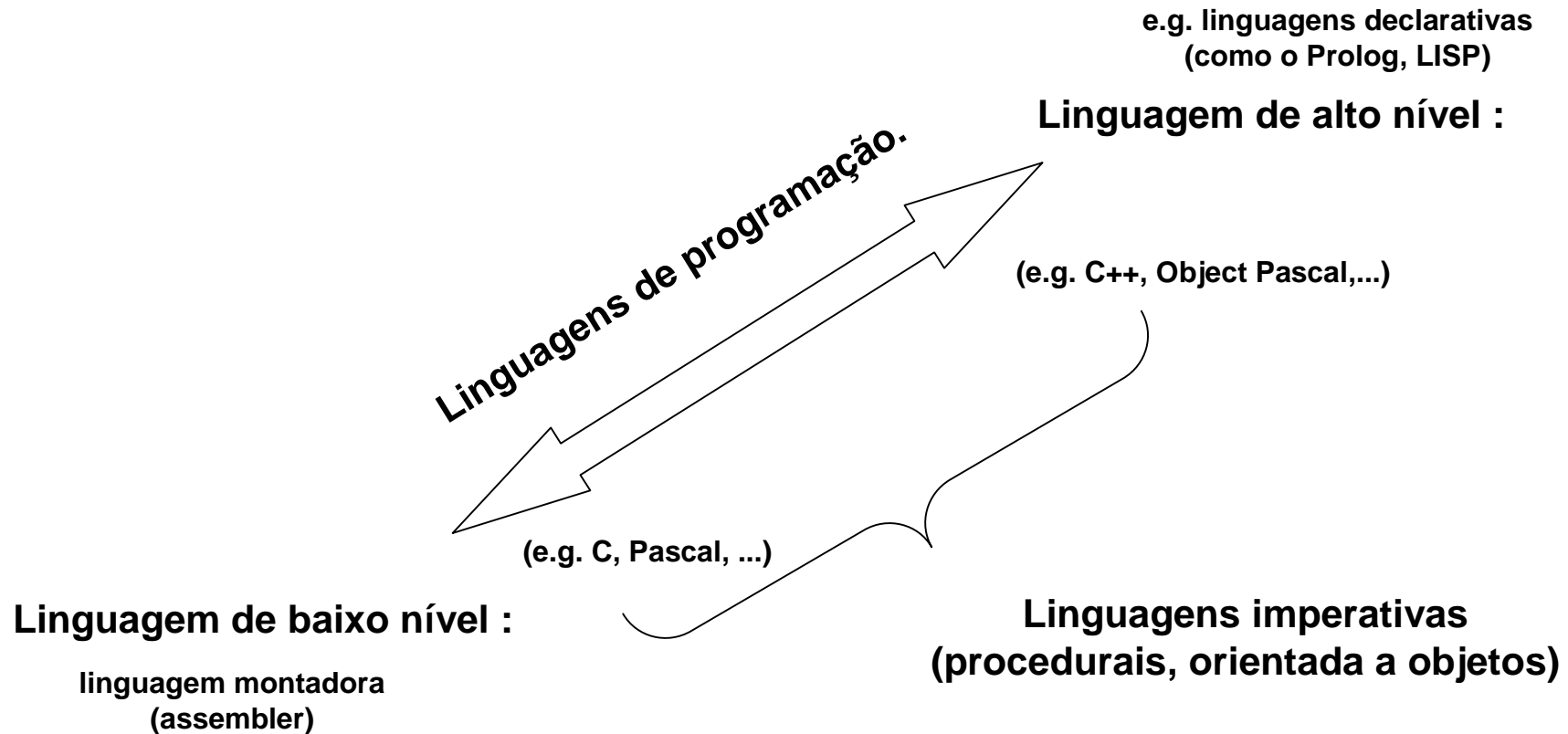
- Um software é basicamente um conjunto de instruções que dita o comportamento do computador durante um certo tempo.

Faça Operação A.  
Faça Operação B 5 vezes.  
Se resultado Operações B é 'bom'  
Então Operação D  
Senão Operação E

- O desenvolvimento de um software é feito por meio de uma linguagem de programação.

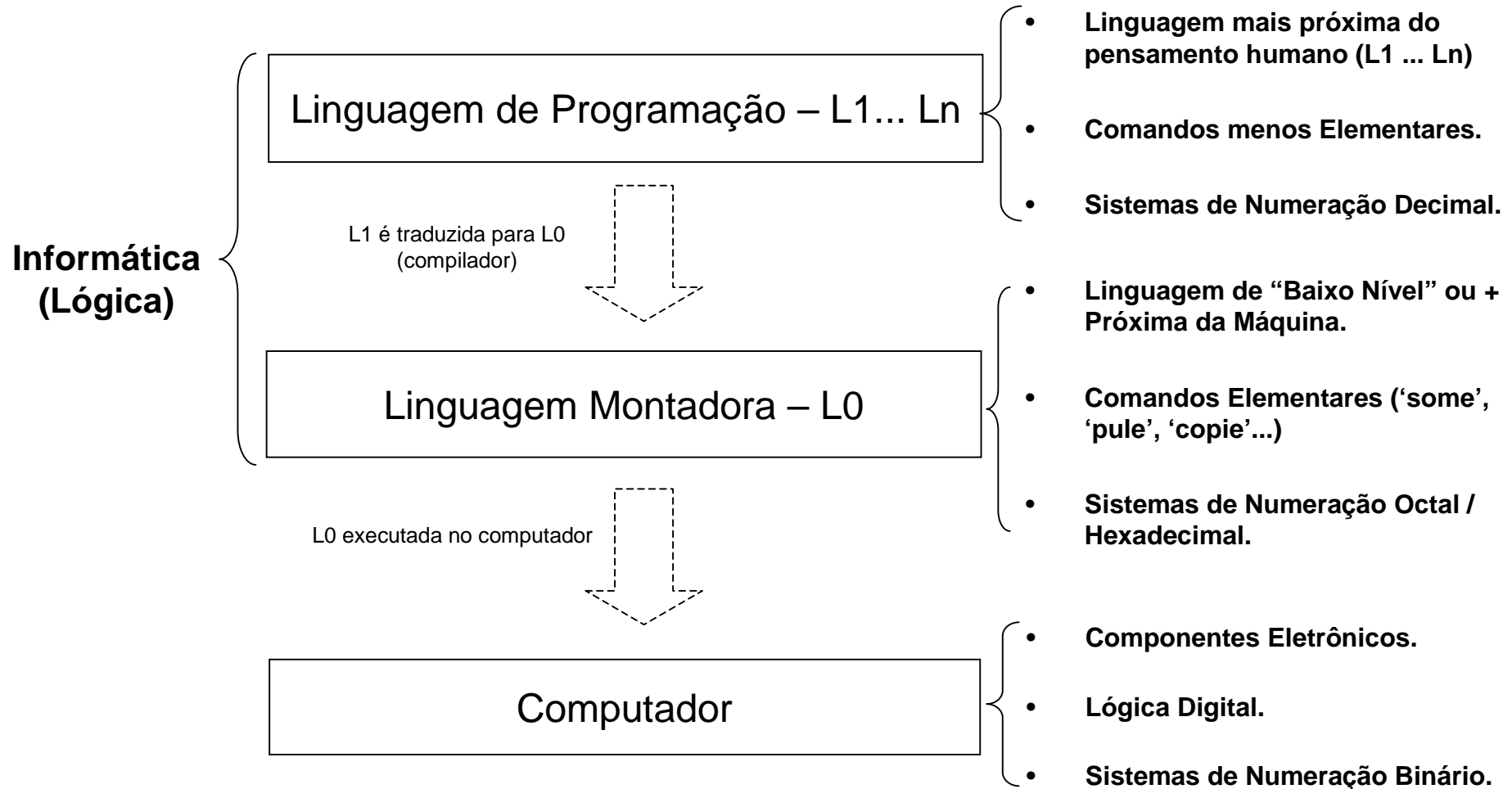
# Tipos de Linguagens de Programação

---



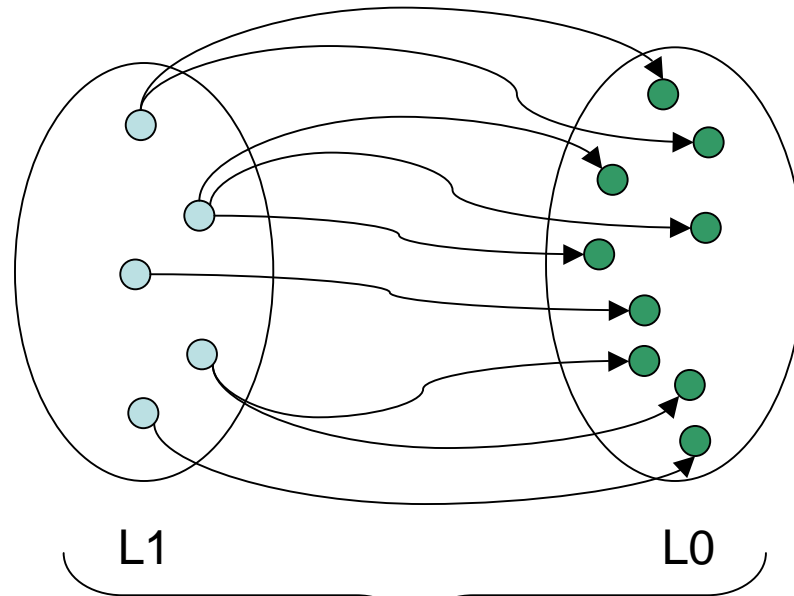
# Relação entre as Linguagens

---



# Compilador – Programa

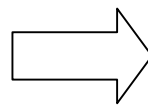
---



Um compilador traduz (mapeia)  
uma expressão em uma  
linguagem para outra linguagem.

Exemplo

Um programa em  
linguagem C



Um programa em  
linguagem de montagem

# Programa

---

- Um conjunto de expressões lógicas em uma determinada linguagem de programação.
- A essência de um programa é o seu algoritmo.
- Um algoritmo define a lógica de um programa.

**Inicio**

A = 5;  
B = 4;

**Se ( A > B ) então**

“A é maior que B”

**Senão**

**Se ( A < B ) então**

“B é maior que A”

**Senão**

“B é igual a A”

**Fim-se**

**Fim-se**

**Fim**



# Do algoritmo à linguagem de máquina

**Algoritmo:** independe do computador e da linguagem

```
Início  
// receber um valor via teclado  
leia valor  
// calcular o seu dobro  
dobro = 2 * valor  
// mostrar o resultado na tela  
imprima dobro  
Fim
```

**Programa em C:** pode apresentar certa dependência do computador

```
#include <stdio.h>  
int main()  
{  
    int valor, dobro;  
    scanf("%d",&valor);  
    dobro = 2*valor;  
    printf("%d",dobro);  
    return 0;  
}
```

**Linguagem *Assembly*:** comando corresponde a um comando em alto nível.

INICIO:	IN	A,(TECLADO)
	LD	(VALOR),A
	LD	A, 2
	MUL	A, (VALOR)
	LD	(DOBRO), A
	LD	A, (DOBRO)
	OUT	A, (VIDEO)

**Linguagem de máquina:** programa pronto para ser executado

```
00  
30  
02  
03  
01  
31
```

# Programação

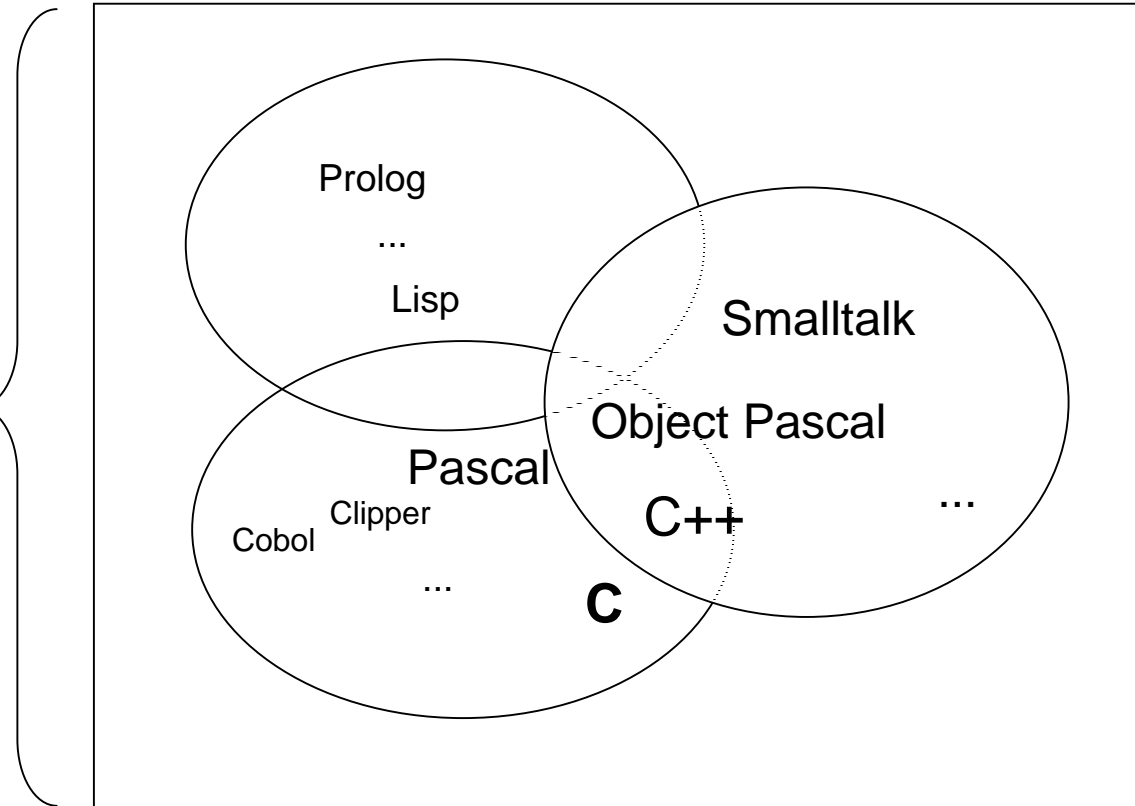
---

- **Para programar é necessário:**
  - desenvolver o raciocínio lógico.
  - ter noções de lógica.
  - saber desenvolver algoritmos.
  - conhecer uma linguagem de programação.
  - conhecer uma ou a tecnologia que suporta a linguagem de programação.
  
- **Assim sendo, trabalharemos:**
  - o aprimoramento de raciocínio lógico.
  - o elementar da lógica.
  - a capacidade de desenvolver algoritmos.
  - a linguagem de programação C.
  - um ambiente de programação em C.
  - . . .

# Linguagens de Programação

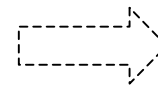
---

Uma grande diversidade de linguagens de programação



---

Por que C/C++?



# Linguagem C

[Dennis Ritchie](#) e [Ken Thompson](#), anos 70

Linguagem  
clássica

Características

- **Uma linguagem difundida:**

- Amplamente utilizada...
- Uma linguagem veterana...
- Sintaxe difundida, servindo como inspirações tecnológicas.

- **Uma linguagem multi-nível:**

- Permite compor programas com abordagens variando entre 'baixo e alto nível'

- **Organização:**

- Funções e estruturas de informação.

- **Ponteiros:**

- Permite a independência de memória pré-alocada.

# Exemplo

Início

A = 5;

B = 4;

Se (A > B) então

“A é maior que B”

Senão

Se (A < B) então

“B é maior que A”

Senão

“B é igual a A”

Fim

Cálculo do  
maior número

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

```
void main()
```

```
{
```

```
    int A = 5;
```

```
    int B = 4;
```

```
    if ( A>B )
```

```
    {
```

```
        printf(“A é maior que B!”);
```

```
    }
```

```
    else
```

```
    {
```

```
        if ( A<B )
```

```
        {
```

```
            printf(“B é maior que A!”);
```

```
        }
```

```
        }
```

```
    else
```

```
    {
```

```
        printf(“B é igual a A!”);
```

```
    }
```

```
    }
```

```
}
```

# Linguagem C

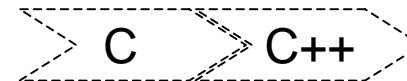
---



## Características

- Devido a sua **flexibilidade de utilização**, ela pode ser considerada como complicada:
  - Uma alternativa é o uso de outras linguagens “menos flexíveis”, i.e. especializada para a aplicação em questão.
  - Entretanto, o fato é que compreender C, com suas flexibilidades, permite compreender outras linguagens.
- Paradigma procedural:
  - O paradigma procedural é menos rico que o paradigma orientado a objetos (OO).
  - Uma alternativa a linguagem C, seria uma linguagem OO, em particular a linguagem C++.
  - Pode-se dizer que a linguagem C++ é uma evolução da linguagem C...

[Dennis Ritchie](#) e [Ken Thompson](#),  
anos 70

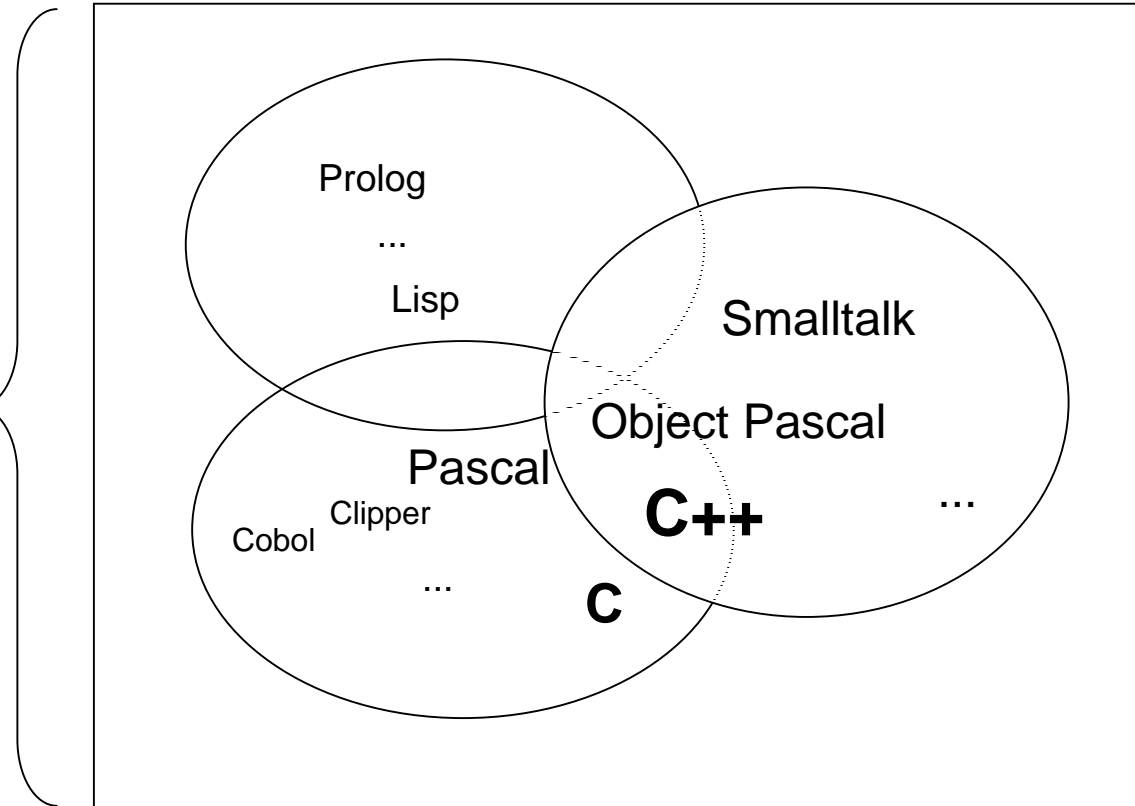


[Bjarne Stroustrup](#),  
nos anos 80

# Linguagens de Programação

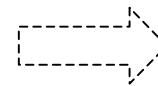
---

Uma grande diversidade de linguagens de programação



---

Por que C++?



...