


# Introdução a Linguagem C

## Programação 1

Rafael Vargas Mesquita

<http://www.ci.ifes.edu.br>  
<ftp://ftp.ci.ifes.edu.br/informatica/rafael/>



# Histórico

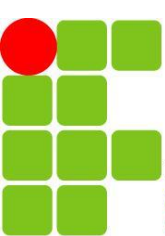
- O C é derivado de uma outra linguagem: o B, criado por Ken Thompson. O B, por sua vez, veio da linguagem BCPL, inventada por Martin Richards.
- O C nasceu na década de 70. Seu inventor, Dennis Ritchie, implementou-o pela primeira vez usando um DEC PDP-11 rodando o sistema operacional UNIX.

**PDP-11** foi uma série de minicomputadores de 16 bits fabricada pela empresa Digital Equipment Corp. nas décadas de 1970 e 1980.

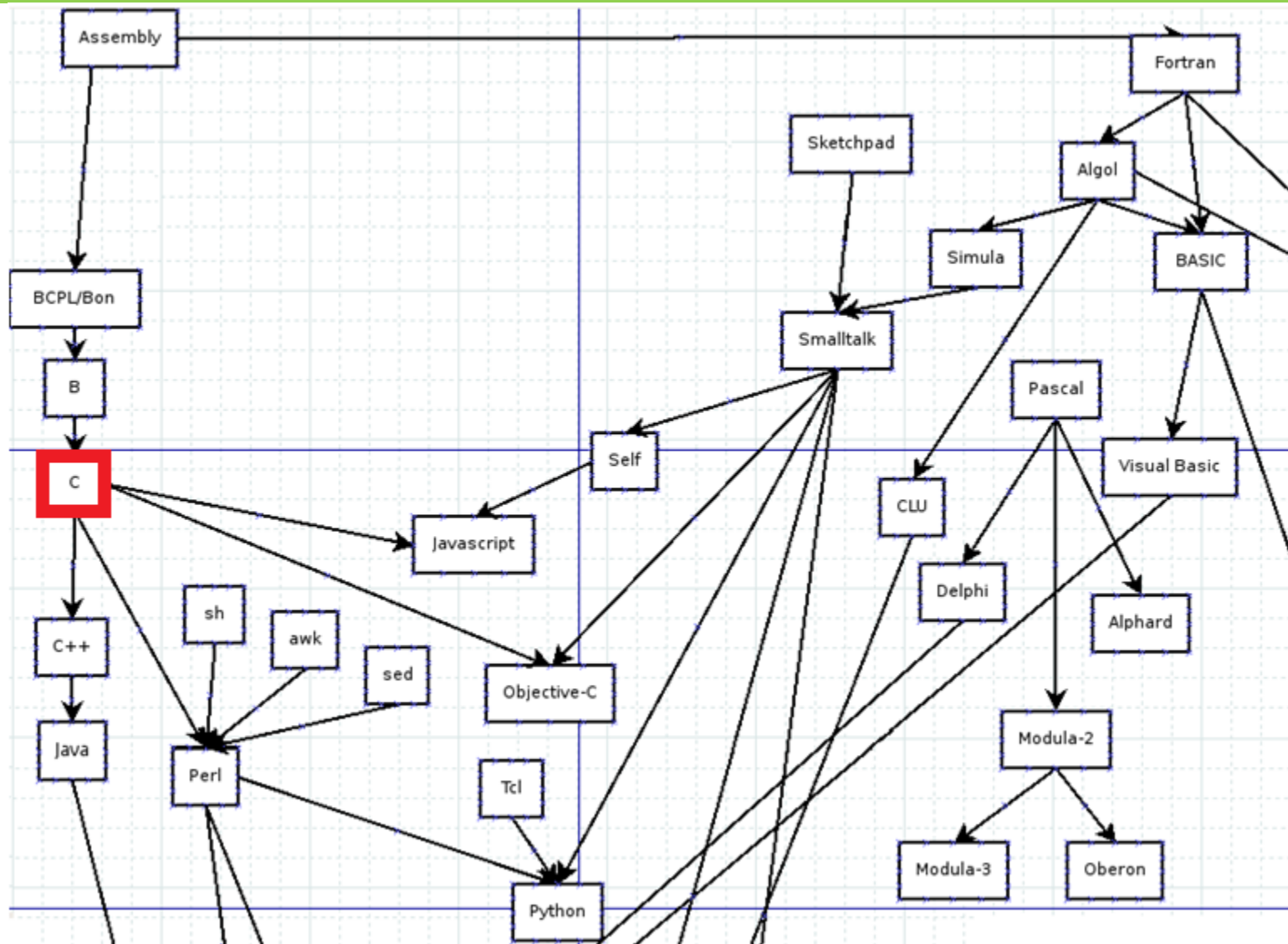


Ken Thompson e Dennis Ritchie



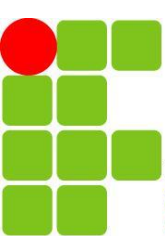


# Histórico



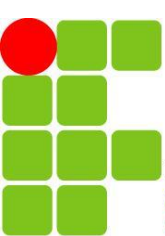
# Apresentação da Linguagem C

- O C é uma linguagem de programação genérica que é utilizada para a criação de programas diversos como:
  - Processadores de texto
  - Planilhas eletrônicas
  - Sistemas operacionais
  - Programas de comunicação
  - Programas para a automação industrial
  - Gerenciadores de bancos de dados
  - Programas de projeto assistido por computador
  - Programas para a solução de problemas da Engenharia, Física, Química e outras Ciências.



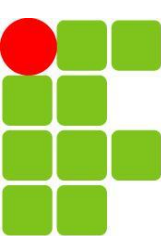
# Declaração de Variáveis

- As variáveis no C podem ter qualquer nome se cinco condições forem satisfeitas:
  - O nome deve começar com uma letra ou sublinhado ( \_ )
  - Os caracteres subsequentes devem ser letras, números ou sublinhado ( \_ )
  - O nome de uma variável não pode ser igual a uma palavra reservada
  - O nome de uma variável não pode ser igual ao nome de uma função declarada pelo programador, ou pelas bibliotecas do C
  - O nome de uma variável pode ter até 32 caracteres
- A linguagem C é “*case sensitive*”, dessa forma, ela diferencia letras maiúsculas de letras minúsculas.



# Tipos Primitivos

- Em Portugol tínhamos os tipos primitivos:
  - caracter ou cadeia
  - lógico
  - inteiro e real
- Em C podemos declarar variáveis dos tipos:
  - char: não é equivalente ao cadeia do Portugol, representa apenas uma letra.
  - int e float (ou double): são equivalentes ao inteiro e real respectivamente.
  - O tipo Lógico não existe em C, em C++ o tipo lógico é chamado bool.
  - C possui ainda o tipo void que é um tipo sem tipo, o uso do tipo void será melhor discutido posteriormente.



# Tipos Primitivos

Tipo	Num de bits	Formato para leitura com scanf	Intervalo	
			Início	Fim
char	8	%c	-128	127
unsigned char	8	%c	0	255
signed char	8	%c	-128	127
int	16	%i	-32.768	32.767
unsigned int	16	%u	0	65.535
signed int	16	%i	-32.768	32.767
short int	16	%hi	-32.768	32.767
unsigned short int	16	%hu	0	65.535
signed short int	16	%hi	-32.768	32.767
long int	32	%li	-2.147.483.648	2.147.483.647
signed long int	32	%li	-2.147.483.648	2.147.483.647
unsigned long int	32	%lu	0	4.294.967.295
float	32	%f	3,4E-38	3.4E+38
double	64	%lf	1,7E-308	1,7E+308
long double	80	%Lf	3,4E-4932	3,4E+4932

# Declaração de Variáveis

- As variáveis no C devem ser declaradas antes de serem usadas. A forma geral da declaração de variáveis é:

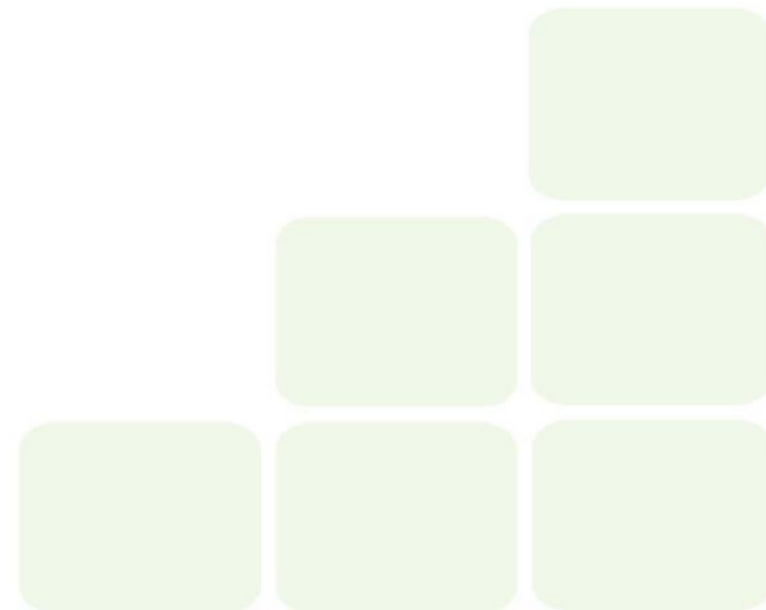
`tipo_da_variável      lista_de_variáveis;`

Exemplo:

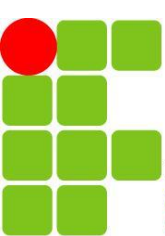
```
char ch, letra;
```

```
int count;
```

```
float pi;
```







# Definindo o Início e Fim de um Algoritmo

- No Portugol o início e o fim do algoritmo eram definidos na função início:

```
início() {  
    ... ( sequência de comandos )  
}
```

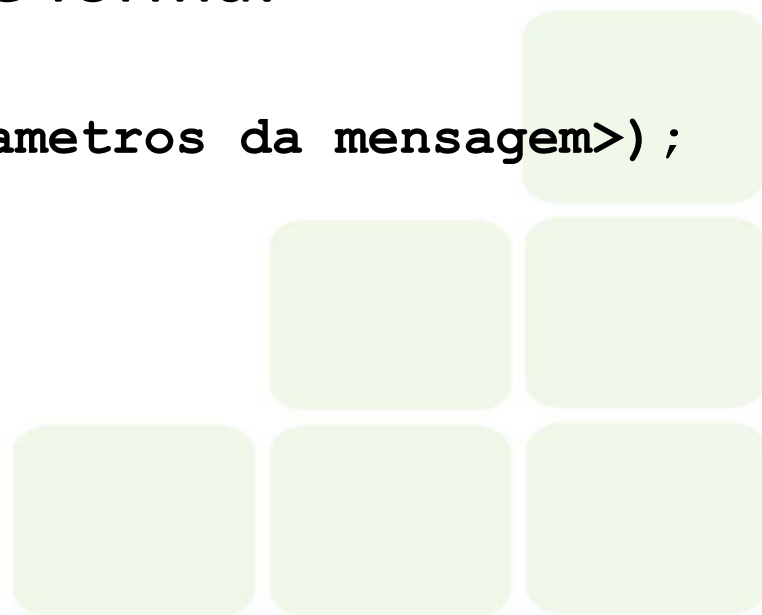
- Em C um algoritmo é definido da seguinte forma:

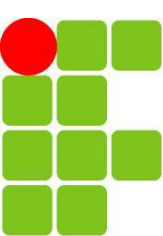
```
main() {  
    ... ( sequência de comandos )  
}
```

# Introdução a Entradas e Saídas

- No Portugol quando desejávamos escrever algo na tela utilizávamos o comando *escreva*
- Em C, o comando equivalente é o *printf* que pode ser definido da seguinte forma:

```
printf(<mensagem formatada>, <parametros da mensagem>);
```



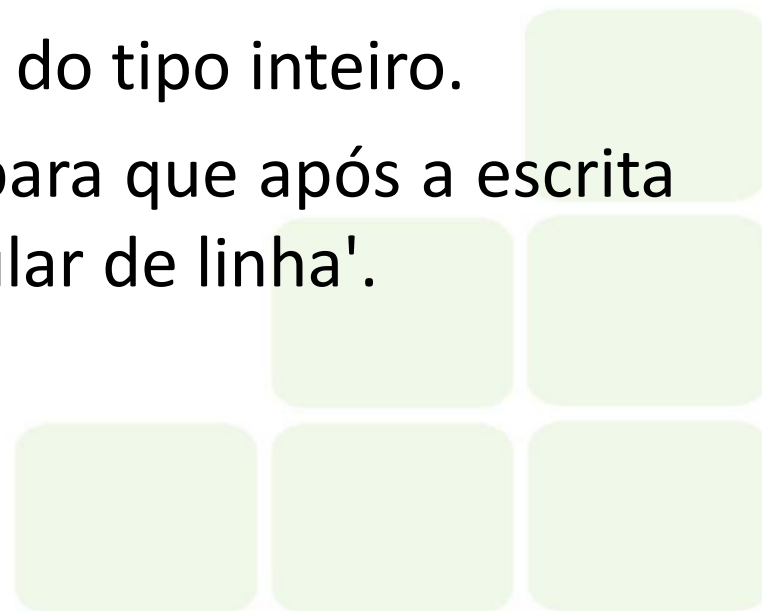


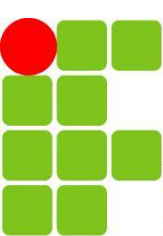
# Introdução a Entradas e Saídas

- Exemplo de uso do printf:

```
printf("O resultado é: %d \n", numero);
```

- Onde número é uma variável do tipo inteiro.
- O \n é o comando utilizado para que após a escrita da mensagem seja feito o 'pular de linha'.



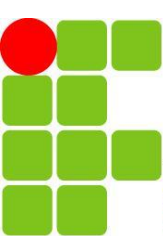


# Introdução a Entradas e Saídas

- No Portugol quando desejávamos ler algo na tela utilizávamos o comando *leia*
- Em C o comando equivalente é o *scanf* que pode ser definido da seguinte forma:

```
scanf(<identificação dos tipos das variáveis>,  
      <variáveis precedidas pelo &>)
```





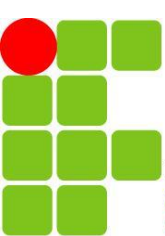
# Introdução a Entradas e Saídas

- Exemplo de uso do scanf:

```
scanf ("%d", &meses) ;
```

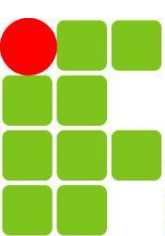
– Onde meses é uma variável do tipo inteiro.





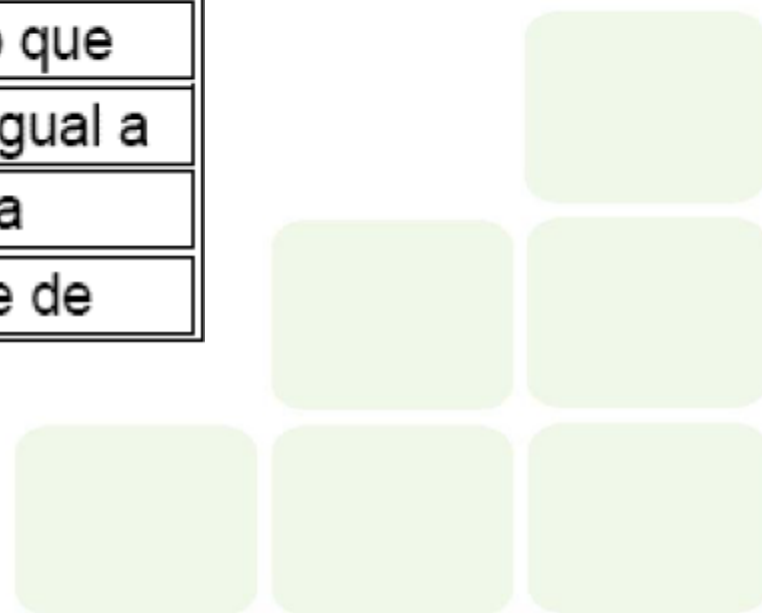
# Operadores Aritméticos

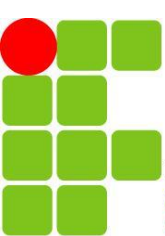
Operador	Ação
+	Soma (inteira e ponto flutuante)
-	Subtração ou Troca de sinal (inteira e ponto flutuante)
*	Multiplicação (inteira e ponto flutuante)
/	Divisão (inteira e ponto flutuante)
%	Resto de divisão (de inteiros)
++	Incremento (inteiro e ponto flutuante)
--	Decremento (inteiro e ponto flutuante)



# Operadores Relacionais

Operador	Ação
>	Maior do que
>=	Maior ou igual a
<	Menor do que
<=	Menor ou igual a
==	Igual a
!=	Diferente de



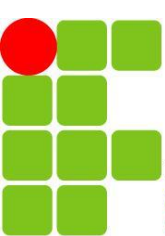


# Operadores Lógicos

Operador	Ação
&&	AND (E)
	OR (OU)
!	NOT (NÃO)

p	q	p AND q	p OR q
falso	falso	falso	falso
falso	verdadeiro	falso	verdadeiro
verdadeiro	falso	falso	verdadeiro
verdadeiro	verdadeiro	verdadeiro	verdadeiro

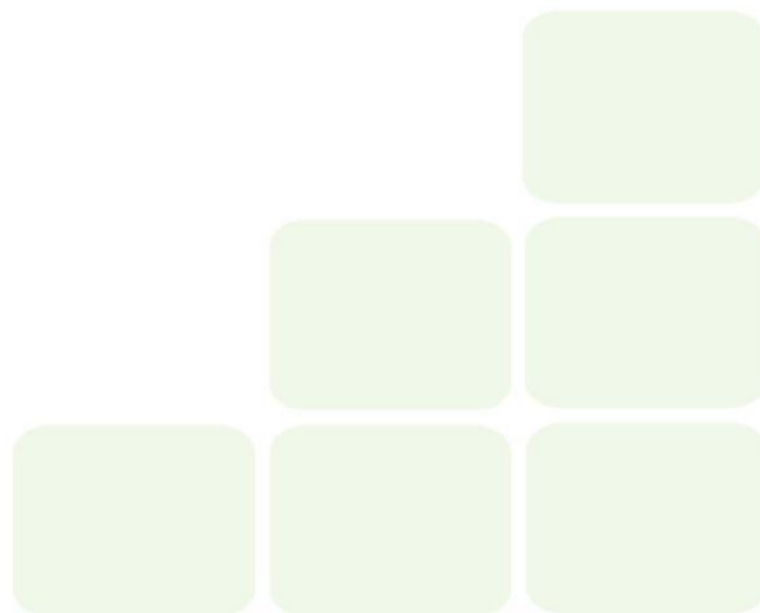


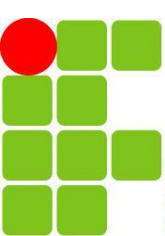


# Operador de Atribuição

- A atribuição de um valor a uma variável é algo bem simples. Basta utilizar o símbolo “=”.

```
int x;  
x = 10;
```





# Bibliografia

- SANTOS, Henrique José. Curso de Linguagem C da UFMG, apostila.
- FORBELLONE, André Luiz. Lógica de Programação – A Construção de Algoritmos e Estruturas de Dados. São Paulo: MAKRON, 1993.

