

# Tipos básicos de dados



**Universidade Federal do Ceará - Campus Quixadá**

Roberto Cabral  
rbcabral@ufc.br

02 de Dezembro de 2020

Arquitetura e Organização de Computadores II

# Tipos básicos de dados

## Tamanho de dados da arquitetura ARM

Name	Size (bits)	Architecture (32/64 bit)
Byte	8	both
Halfword	16	both
Word	32	both
Doubleword	64	both
Quadword	128	64-bit only

# Tipos básicos de dados

Tamanhos de alguns tipos de dados C/C++ na arquitetura ARM de 32 bits.

Data type	Size	Description
int	Word	Integer
long	Word	Integer
short	Halfword	Integer
char	Byte	Byte
long long	Doubleword	Integer
float	Word	Single-precision IEEE floating-point
double	Doubleword	Double-precision IEEE floating-point
bool	Byte	Boolean
void*	Word	Address of data or code

# Tipos básicos de dados

- Em C, como seria armazenado 123?

123

# Tipos básicos de dados

- Em C, como seria armazenado 123?
- Depende!

# Tipos básicos de dados

- Em C, como seria armazenado 123?
- Depende!

- 0000007b

- 31 32 33 00



# Tipos básicos de dados

---

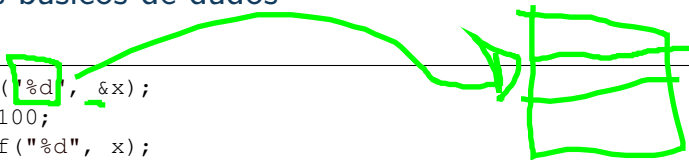
```
scanf("%d", &x);  
x += 100;  
printf("%d", x);
```

---

123

# Tipos básicos de dados

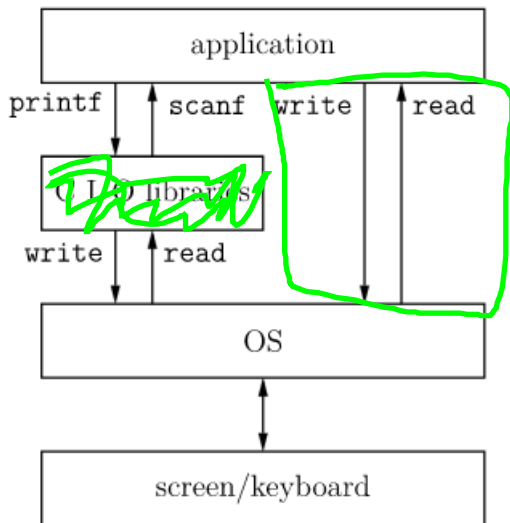
```
scanf("%d", &x);  
x += 100;  
printf("%d", x);
```



- Lê caracteres, cada um como um caractere separado do teclado e converte a sequência de caracteres no formato interno correspondente.
- Adiciona 100 à variável int.
- Converte o int resultante em uma sequência de caracteres e o exibe na tela.



# Tipos básicos de dados



# Tipos básicos de dados

- Ao usar as funções de chamada de sistema de leitura e gravação para E/S, é responsabilidade do programador fazer as conversões entre o tipo de char usado para E/S e os formatos de armazenamento usados dentro do programa.
- Em breve estaremos escrevendo nossas próprias funções em linguagem assembly para converter entre o formato de caractere usado para exibição na tela e entrada do teclado, e o formato de armazenamento interno de inteiros no sistema numérico binário.
- O objetivo de escrever nossas próprias funções é obter uma compreensão completa de como os dados são representados internamente no computador.

# Ler valor Hexadecimal do teclado



## Ler valor Hexadecimal do teclado

### Algorithm 4.2.1. Read hexadecimal value from keyboard.

*value = 0*

*While more characters*

*shift value left four bits*

*temp = new character converted to* `int`

*value = value + temp*

ler: 123valor = 0while (1) valor = 0001

# Deslocamento de bits

- Na prática, o que está acontecendo nos comandos a seguir:
  - $x = x \ll 4$
  - $x = x \gg 3$

# Deslocamento de bits

---

```
int main(void)
{
    int x;
    int leftShift, rightShift;

    printf("Enter an integer: ");
    scanf("%d", &x);

    printf("Multiply by two raised to the power: ");
    scanf("%d", &leftShift);
    printf("%d x %d = %d\n", x, 1 << leftShift, x << leftShift);

    printf("Divide by two raised to the power: ");
    scanf("%d", &rightShift);
    printf("%d / %d = %d\n", x, 1 << rightShift, x >> rightShift);

    return 0;
}
```

---

# Operações lógicas bit a bit

**Table 4.4.2.** *and*

x[i]	y[i]	x[i] & y[i]
0	0	0
0	1	0
1	0	0
1	1	1

**Table 4.4.3.** *inclusive or*

x[i]	y[i]	x[i]   y[i]
0	0	0
0	1	1
1	0	1
1	1	1

**Table 4.4.4.** *exclusive or*

x[i]	y[i]	x[i] ^ y[i]
0	0	0
0	1	1
1	0	1
1	1	0

**Table 4.4.5.** *complement*

x[i]	~x[i]
0	1
1	0

# Operações lógicas bit a bit

**Table 4.4.6. *and***

x	y	x && y
0	0	0
0	non-zero	0
non-zero	0	0
non-zero	non-zero	1

**Table 4.4.7. *or***

x	y	x    y
0	0	0
0	non-zero	1
non-zero	0	1
non-zero	non-zero	1

**Table 4.4.8. *complement***

x	!x
0	1
non-zero	0



# Converter ASCII para int

Hex character	ASCII code	Corresponding	int
0	30	00000000	
1	31	00000001	
2	32	00000002	
3	33	00000003	
4	34	00000004	
5	35	00000005	
6	36	00000006	
7	37	00000007	
8	38	00000008	
9	39	00000009	
a	61	0000000a	
b	62	0000000b	
c	63	0000000c	
d	64	0000000d	
e	65	0000000e	
f	66	0000000f	

## Converter ASCII para int

Para os caracteres de 0 a 9, a conversão do valor ASCII para um valor int pode ser feita facilmente com o código:

```
aChar = aChar & 0x0f;  
anInt = (int)aChar;
```

# Tipos básicos de dados



**Universidade Federal do Ceará - Campus Quixadá**

Roberto Cabral  
rbcabral@ufc.br

02 de Dezembro de 2020

Arquitetura e Organização de Computadores II