Introdução a Programação – IF669
http://www.cin.ufpe.br/~if669

# Tipos primitivos de Java
# Entrada de Dados
# Formatação da Saída

## Aula 03

Ricardo Massa F. Lima
rmfl@cin.ufpe.br

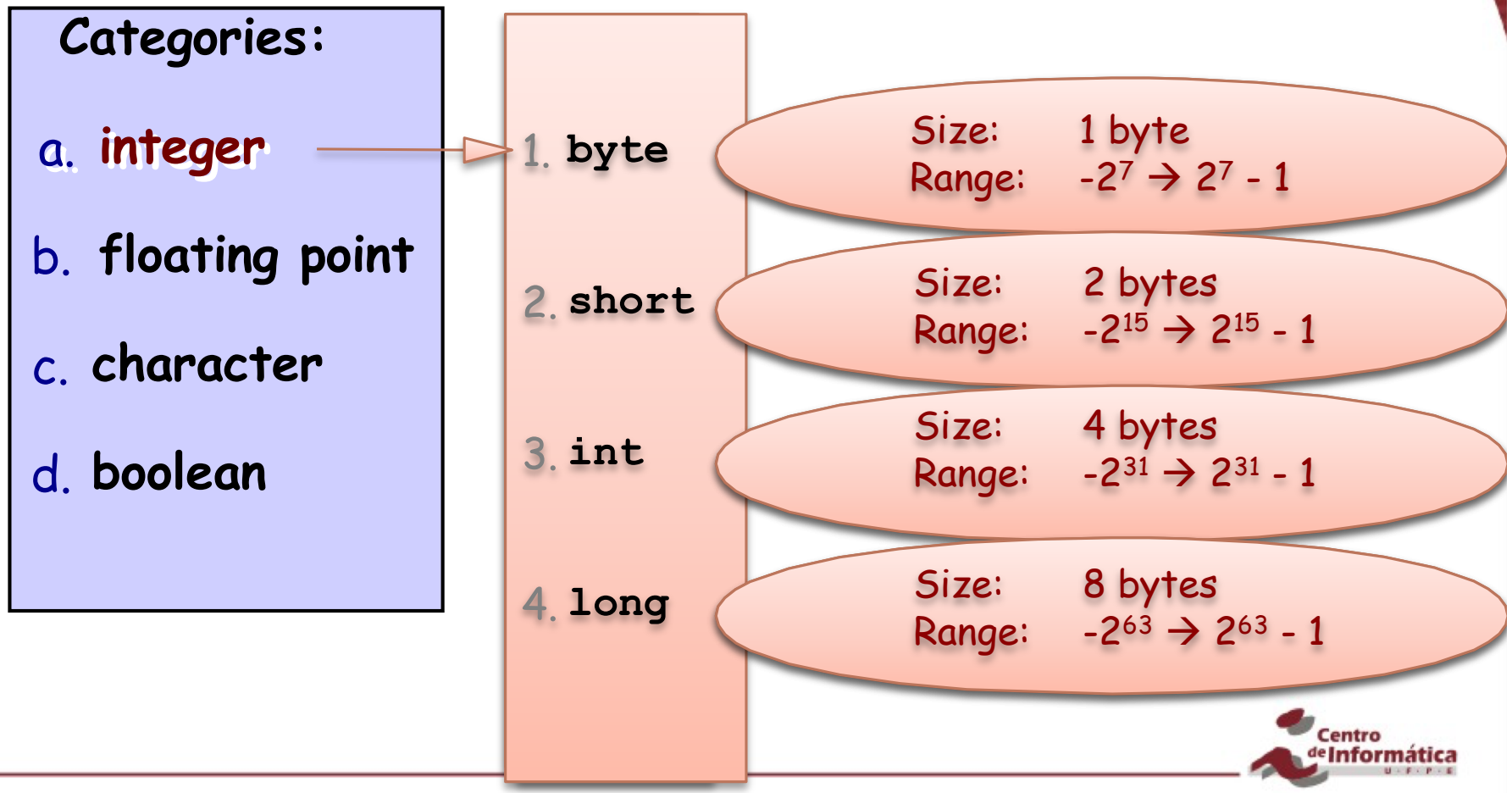
Sérgio C. B. Soares
scbs@cin.ufpe.br

# Tipos disponíveis em Java

- **Primitivos**
  - boolean
  - byte
  - short
  - int
  - long
  - char
  - float
  - double
- **Referência**
  - depois veremos isso...

# Tipos Primitivos: Inteiro

- Todos os números possuem sinal



# Exemplos de uso de valores inteiros

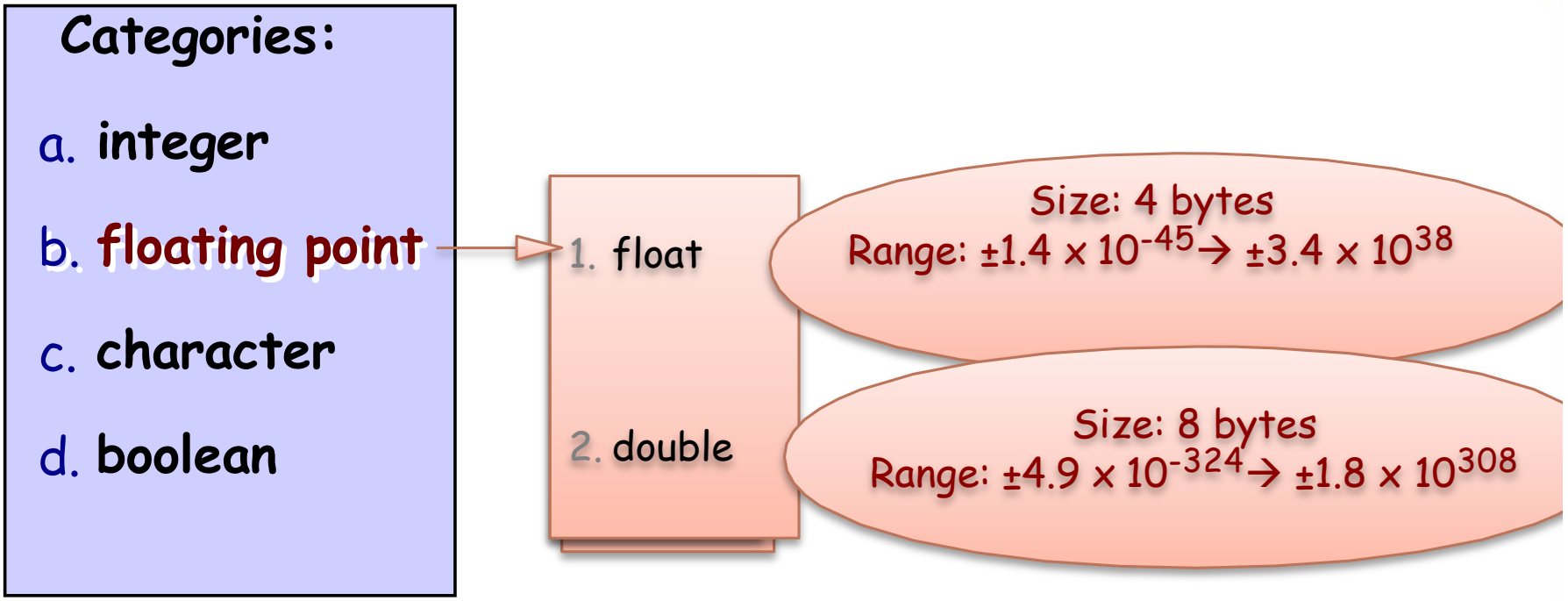
Estes são os limites positivos

```
byte  b = 127;
short s = 32767;
int   i = 2147483647;
long  l = 9223372036854775807L;
```

Valores `long` são representados com um `L` ou `l` no final (caso contrário são `int`)

# Tipos Primitivos: Real

- Números reais (possuem a parte fracionária)



# Exemplos de uso de valores reais

Valores `float` são representados com um `F` ou `f` no final (do contrário são `double`)

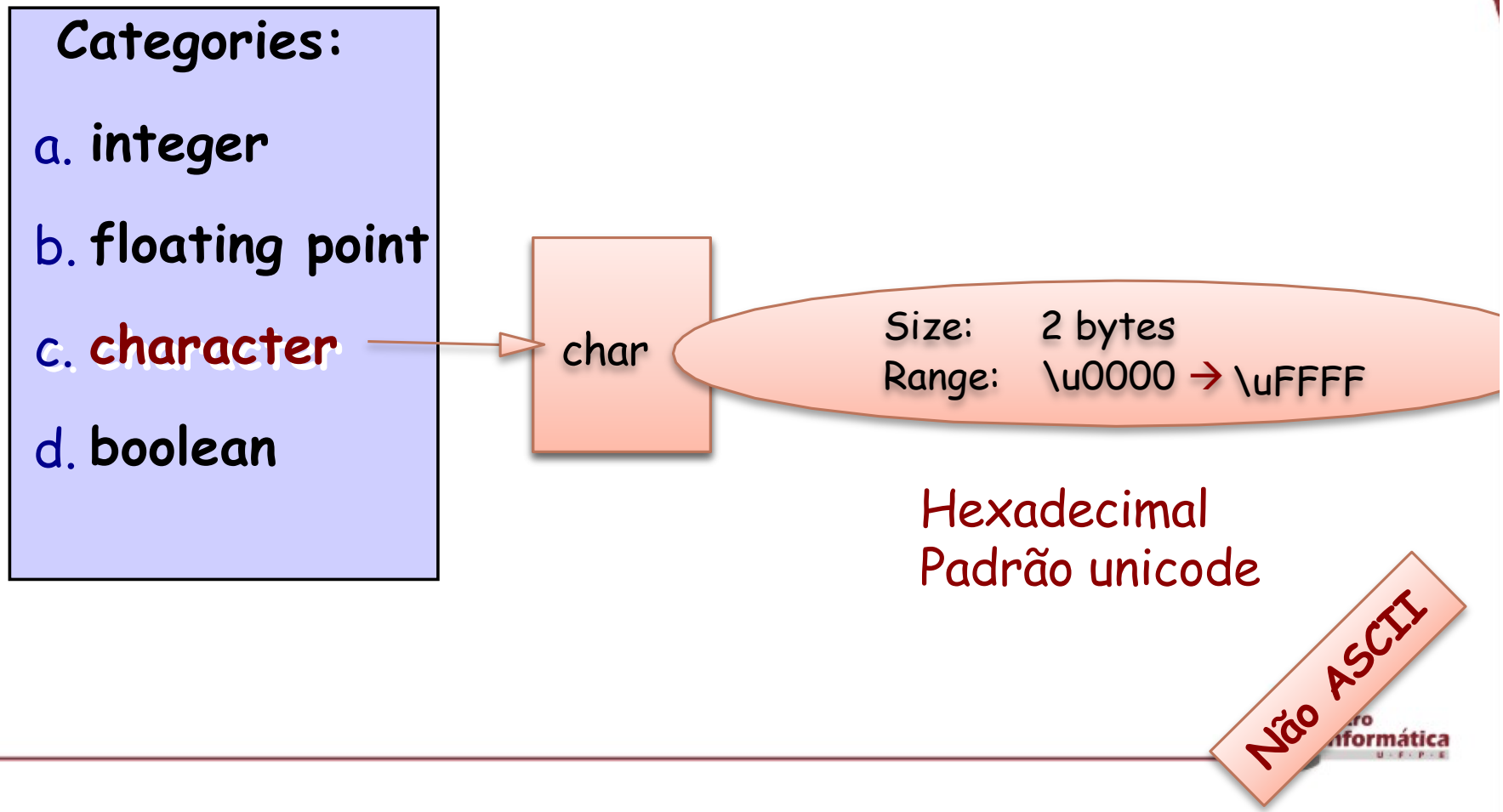
```
float f = 3.4028235E38F;
double d = 1.79769313486231570E308;
```

`7.1E2` = $7.1 \times 10^2$
`7.1e2` = $7.1 \times 10^2$

Estes são os limites positivos

# Tipos Primitivos: Caractere

- Caracteres Unicode



# Parte da tabela unicode

| | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 32 | | | 48 | 0 | | 64 | @ | | 80 | P | | 96 | ` | | 112 | p |
| 33 | ! | | 49 | 1 | | 65 | A | | 81 | Q | | 97 | a | | 113 | q |
| 34 | " | | 50 | 2 | | 66 | B | | 82 | R | | 98 | b | | 114 | r |
| 35 | # | | 51 | 3 | | 67 | C | | 83 | S | | 99 | c | | 115 | s |
| 36 | $ | | 52 | 4 | | 68 | D | | 84 | T | | 100 | d | | 116 | t |
| 37 | % | | 53 | 5 | | 69 | E | | 85 | U | | 101 | e | | 117 | u |
| 38 | & | | 54 | 6 | | 70 | F | | 86 | V | | 102 | f | | 118 | v |
| 39 | ' | | 55 | 7 | | 71 | G | | 87 | W | | 103 | g | | 119 | w |
| 40 | ( | | 56 | 8 | | 72 | H | | 88 | X | | 104 | h | | 120 | x |
| 41 | ) | | 57 | 9 | | 73 | I | | 89 | Y | | 105 | i | | 121 | y |
| 42 | * | | 58 | : | | 74 | J | | 90 | Z | | 106 | j | | 122 | z |
| 43 | + | | 59 | ; | | 75 | K | | 91 | [ | | 107 | k | | 123 | { |
| 44 | , | | 60 | < | | 76 | L | | 92 | \ | | 108 | l | | 124 | \| |
| 45 | - | | 61 | = | | 77 | M | | 93 | ] | | 109 | m | | 125 | } |
| 46 | . | | 62 | > | | 78 | N | | 94 | ^ | | 110 | n | | 126 | ~ |
| 47 | / | | 63 | ? | | 79 | O | | 95 | _ | | 111 | o | | 127 | |

# Tipos Primitivos: booleano

- **Podem apenas ser** `true` **ou** `false`

**Categories:**

a. integer

b. floating point

c. character

d. **boolean**

# Exemplos de uso de valores booleanos

```
int x = 20;
boolean b = x > 10;
if (b) {
  Util.imprima("maior");
} else {
  Util.imprima("menor");
}
```

O que será impresso?

# Mas e o tipo textual?

- O tipo textual não é um tipo primitivo

```
String nome = "Sergio";
Util.imprima(nome);
```

Por enquanto é suficiente!

# Uma outra forma de entrada de dados pelo teclado

- **Até aqui utilizávamos** `Util.leia()`
  - Mas só líamos valores inteiros
  - Como ler outros valores?

Representa o teclado

```
Scanner in = new Scanner(System.in);
String nome    = in.nextLine();
int    idade   = in.nextInt();
double salario = in.nextDouble();
```

```
import java.util.Scanner;
```

Centro de Informática UFPE

# Formatação da Saída

- Exemplo:

```
double x = 10000.0 / 3.0;
System.out.print(x);
```

- Resultado:

```
3333.3333333333335
```

- Este resultado é provavelmente indesejável devido a grande quantidade de dígitos após a vírgula
- Vamos usar o método `System.out.printf`

# Formatação da Saída

```
double x = 10000.0 / 3.0;

System.out.print(x);

System.out.printf("%,.3f", x);

System.out.printf("R$ %,.2f", x);
```

```
3333,3333333333335
```

```
3.333,333
```

```
R$ 3.333,33
```

A formatação de casas decimais irá utilizar as configurações regionais do computador

# Formatação da Saída

- O método `printf` recebe mais de 1 argumento, o primeiro sempre é o formato, seguido por diversos valores.

```
System.out.printf(formato, valor1, valor2,...);
```

- O formato é do tipo textual (String) e indica o texto que será impresso, fazendo as substituições.

# Formatação da Saída

- Dentro do formato, podem haver vários especificadores de formato, que são iniciados pelo carácter de porcentagem (%), seguidos por uma bandeira opcional e um conversor.
- Existem vários conversores e os mais utilizados são:

| | |
|---|---|
| `%d` | `int` |
| `%c` | `char` |
| `%s` | `String` |
| `%f` | `double e float` |

# Formatação da Saída

- Também é possível utilizar as bandeiras, que podem ser combinadas:

| | | |
|---|---|---|
| `printf("%.2f", 10.5);` | 10.50 | Formata com 2 casas decimais |
| `printf("%,d", 17435);` | 17,435 | Formata separando na casa dos milhares |
| `printf("%02d", 6);` | 06 | Formata com 2 dígitos, completando com zeros |
| `printf("%+f", 13.7);` | +13.700000 | Formata forçando a exibição do sinal |

# Formatação da Saída

- Também é possível utilizar as bandeiras, que podem ser combinadas:

| | | |
|---|---|---|
| `printf("Olá, %s%n", "João");` | **Olá, João** | Formata substituindo a String e pula uma linha |
| `printf("%s(%d)", "Pedro", 20);` | **Pedro(20)** | Formata substituindo ambos os especificadores |
| `printf("%c", 'A');` | **A** | Formata um caracter |

# Exercício 1 (10 minutos)

- **Escreva um programa que pede o nome e a idade (inteiro) e peso (double) de uma pessoa e imprime uma mensagem com tais informações. O peso tem que ser impresso com duas casas decimais.**
  - Ex: José, 25 anos, pesa 72,18 kg!

```
Scanner in = new Scanner(System.in);
System.out.println("Digite seu nome");
String nome = in.nextLine();
System.out.println("Digite sua idade");
int idade = in.nextInt();
System.out.println("Digite seu peso");
double peso = in.nextDouble();
System.out.printf("%s, %d anos, pesa %.2f kg!",
nome, idade, peso);
```

```
import java.util.Scanner;
```

Centro de Informática UFPE