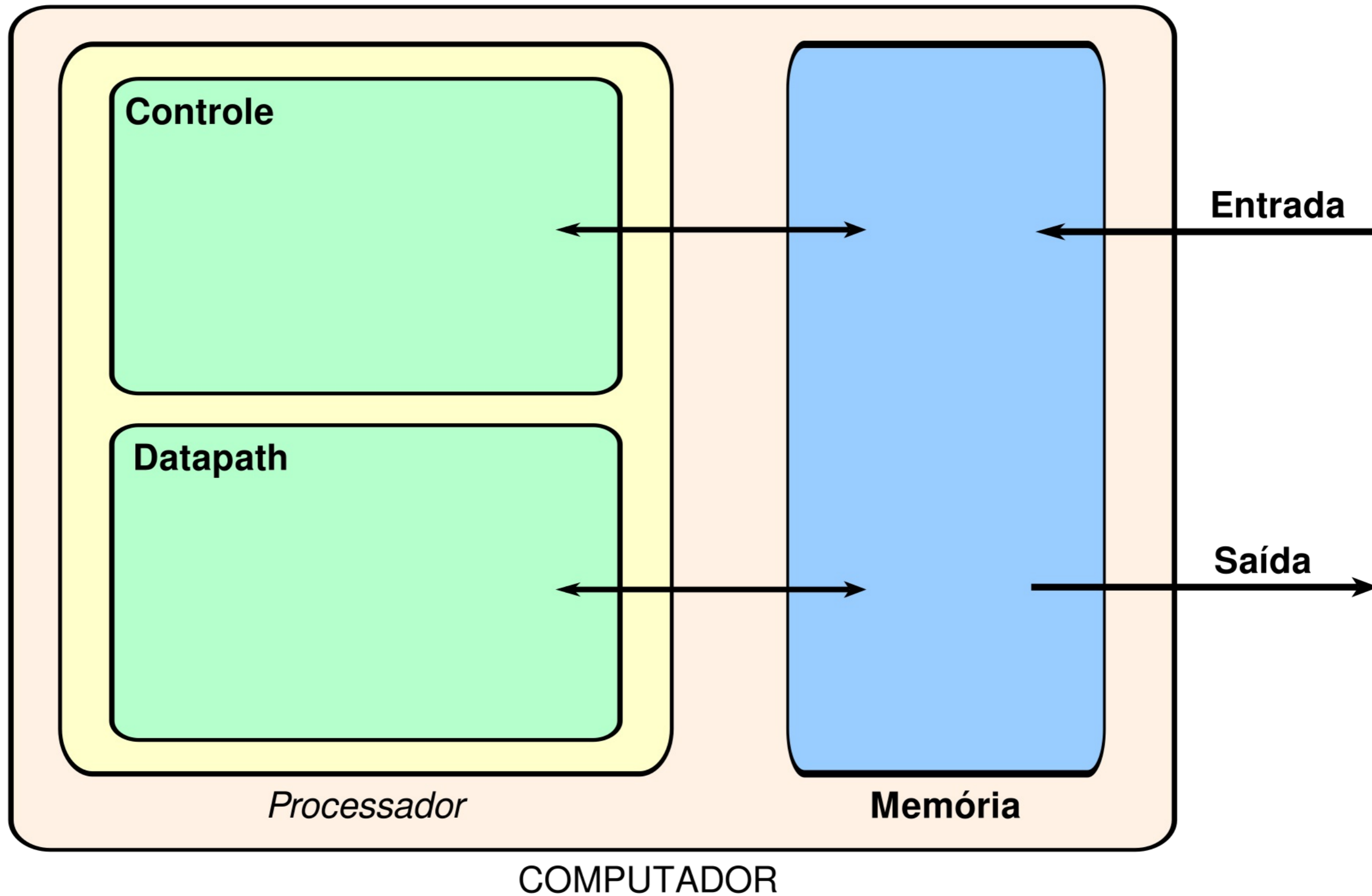


# MOTIVAÇÃO



# MOTIVAÇÃO



**int não é  $\mathbb{Z}$ ; float não é  $\mathbb{R}$**

int  $\neq$   $\mathbb{Z}$

float  $\neq$   $\mathbb{R}$

# Cálculos matemáticos no computador são exatos?



$$x = (43.1 - 43.2) + 1$$

$$x = (-0.1) + 1$$

$$x = 0.9$$



# $(\text{int})^2 \geq 0$ ?

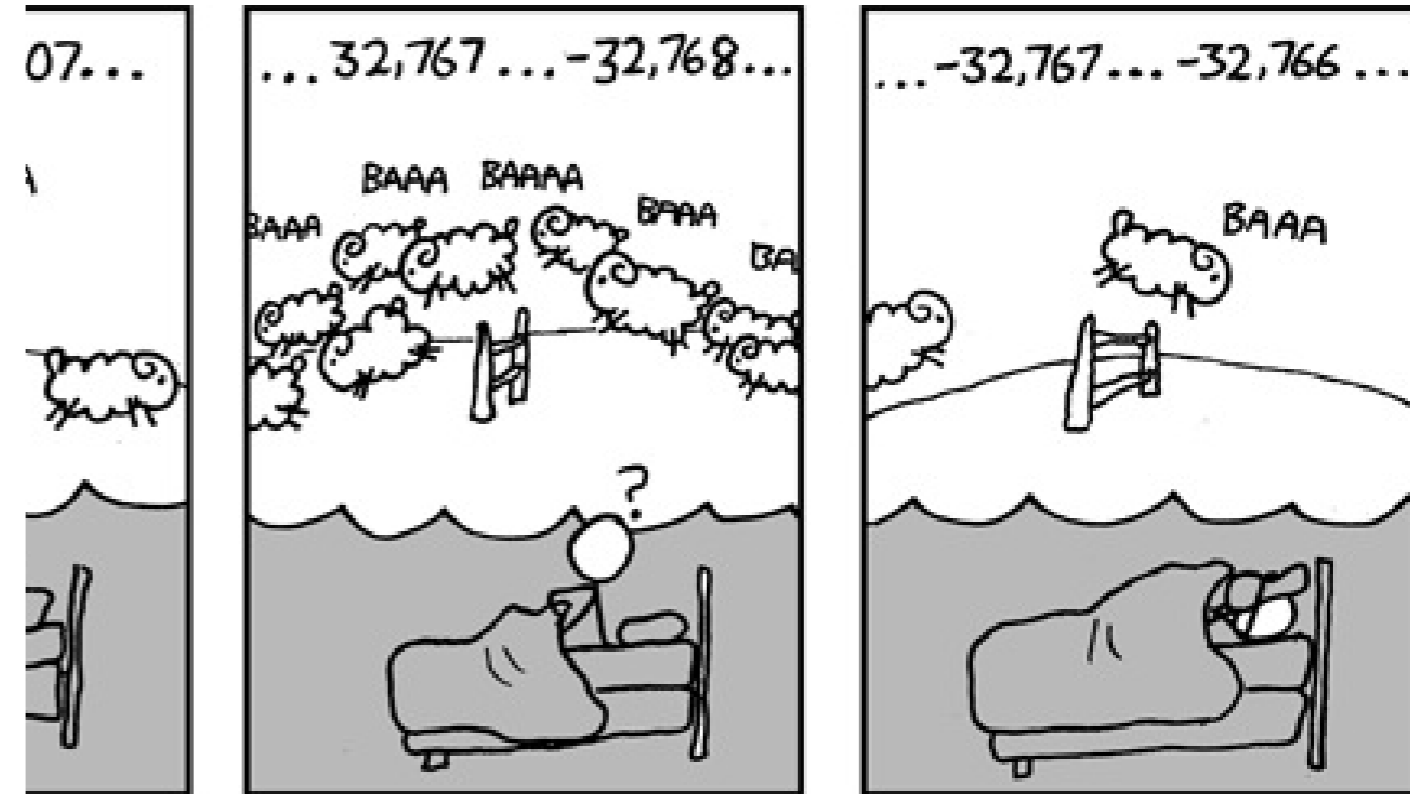
```
1  /**
2  * CR6.190A: Arquitetura de Computadores
3  * https://cursos.computacaoraiz.com.br
4  *
5  * quadrado.c
6  * 0 quadrado de um número pode ser negativo?
7  */
8
9  #include <stdio.h>
10
11 #define INICIO 0
12 #define FIM 1000000
13 #define INTERVALO 50000
14
15 int main(void)
16 {
17     for (int i = INICIO; i <= FIM; i += INTERVALO)
18         printf("%7d^2 = %11d\n", i, i * i);
19
20     return 0;
21 }
```



# $(int)^2 \geq 0?$

```
[abrantesasf@cosmos ~/cr6.190a/introducao]$ ./quadrado
```

```
0^2 = 0
50000^2 = -1794967296
100000^2 = 1410065408
150000^2 = 1025163520
200000^2 = 1345294336
250000^2 = -1924509440
300000^2 = -194313216
350000^2 = -2054051584
400000^2 = 1086210048
450000^2 = 636537088
500000^2 = 891896832
550000^2 = 1852289280
600000^2 = -777252864
650000^2 = 1593204992
700000^2 = 373728256
750000^2 = -140715776
800000^2 = 49872896
850000^2 = 945494272
900000^2 = -1748818944
950000^2 = 556867840
1000000^2 = -727379968
```



# $(\text{float} \mid \text{double})^2 \geq 0$ ?

```
1 /**
2  * CR6.190A: Arquitetura de Computadores
3  * https://cursos.computacaoraiz.com.br
4  *
5  * quadrado2.c
6  * O quadrado de um número pode ser negativo?
7  */
8
9 #include <stdio.h>
10
11 #define INICIO 0
12 #define FIM 1000000
13 #define INTERVALO 50000
14
15 int main(void)
16 {
17     for (int i = INICIO; i <= FIM; i += INTERVALO)
18         printf("%7d^2 = %13.0f\n", i, (double) i * (double) i);
19
20     return 0;
21 }
```





# (float | double)^2 >= 0?

```
[labrantesaf@cosmos ~/cr6.190a/introducao]$ ./quadrado2
```

```
0^2 = 0
50000^2 = 25000000000
100000^2 = 100000000000
150000^2 = 225000000000
200000^2 = 400000000000
250000^2 = 625000000000
300000^2 = 900000000000
350000^2 = 1225000000000
400000^2 = 1600000000000
450000^2 = 2025000000000
500000^2 = 2500000000000
550000^2 = 3025000000000
600000^2 = 3600000000000
650000^2 = 4225000000000
700000^2 = 4900000000000
750000^2 = 5625000000000
800000^2 = 6400000000000
850000^2 = 7225000000000
900000^2 = 8100000000000
950000^2 = 9025000000000
1000000^2 = 10000000000000
```



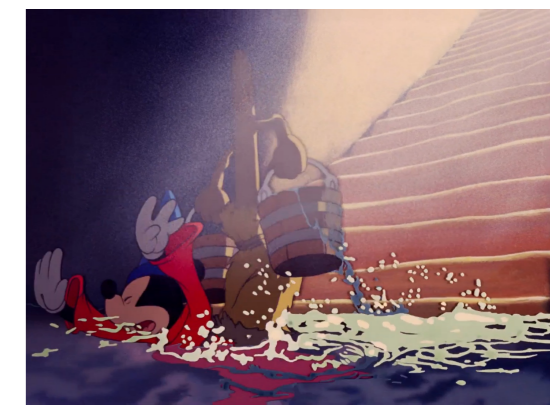
# A associatividade vale? Não para floats...



```
1 /**
2  * CR6.190A: Arquitetura de Computadores
3  * https://cursos.computacaoraiz.com.br
4  *
5  * associativa.c
6  * A matemática não é mais associativa?
7  */
8
9 #include <stdio.h>
10
11
12 int main(void)
13 {
14     printf("(-1e20 + 1e20 + 3.14) = %.2f\n\n", (-1e20 + 1e20 + 3.14));
15     printf("(-1e20 + 1e20) + 3.14 = %.2f\n\n", (-1e20 + 1e20) + 3.14);
16     printf("-1e20 + (1e20 + 3.14) = %.2f\n\n", -1e20 + (1e20 + 3.14));
17
18     return 0;
19 }
```

$$\begin{aligned} a + b + c &= (a + b + c) \\ &= (a + b) + c \\ &= a + (b + c) \end{aligned}$$

# A associatividade vale? Não para floats...



```
[abrantestasf@cosmos ~/cr6.190a/introducao]$ ./associativa
```

```
(-1e20 + 1e20 + 3.14) = 3.14
```

```
(-1e20 + 1e20) + 3.14 = 3.14
```

```
-1e20 + (1e20 + 3.14) = 0.00
```

# E a comutatividade?



```
1 /**
2  * CR6.190A: Arquitetura de Computadores
3  * https://cursos.computacaoraiz.com.br
4  *
5  * comutativa.c
6  * A propriedade comutativa vale para inteiros e floats?
7  */
8
9 #include <stdio.h>
10
11 int main(void)
12 {
13     printf("%s\n", "Comutatividade para int:");
14     printf("%d\n", 300 * 400 * 500 * 600);
15     printf("%d\n", 400 * 500 * 600 * 300);
16
17     printf("%s\n", "Comutatividade para float:");
18     printf("%.0f\n", 1e5 * 1e6 * 1e4 * 1.0);
19     printf("%.0f\n", 1e6 * 1e4 * 1.0 * 1e5);
20
21     return 0;
22 }
```

$$a \times b = b \times a$$

$$a + b = b + a$$

# E a comutatividade?



```
[abrantesasf@cosmos ~/cr6.190a/introducao]$ ./comutativa
```

```
Comutatividade para int:
```

```
1640261632
```

```
1640261632
```

```
Comutatividade para float:
```

```
10000000000000000000
```

```
10000000000000000000
```

$$a \times b = b \times a$$

$$a + b = b + a$$

# 1ª verdade nua e crua: int não é Z; float não é R

$x^2 \geq 0$ ?

int → não

float → sim

Associatividade?

int → sim

float → não

Comutatividade?

int e float → sim

Distributividade

(descubra por sim mesmo!)

# E a arquitetura dos computadores?



**Entender COMO os computadores representam e armazenam números internamente é fundamental para evitar cair em armadilhas.**

**Ao entender a arquitetura dos computadores e como funcionam, você entenderá a questão de representação e armazenamento de números inteiros e floats.**

**90-95% do tempo não há problema.**

**5-10% do tempo, tome cuidado!**

- Aplicações críticas (foguetes, projeto de pontes, usinas)**
- Vulnerabilidades**
- Computação científica**