

Understand
floating points

Know when
computers fail

Understand
floating point
operations



Floating Point

- ① Fixed point representation
- ② Floating point representation
- ③ IEEE 754 standard

Is it true for all values of x and y:

```
Int x=foo();  
Int y=bar();
```

A. $(x > 0) \vee (x - 1 < 0)$

B. $(x \& 7) \neq 7 \vee (x \ll 29 < 0)$

C. $(x * x) \geq 0$

D. $x < 0 \vee -x \leq 0$

E. $x > 0 \vee -x \geq 0$

Real Numbers

$$\begin{array}{ccccccc}
 10^m & 10^{m-1} & & 10^2 & 10^1 & 10^0 & 10^{-1} & 10^{-2} & & 10^{-n} \\
 d_m & d_{m-1} & \dots & d_2 & d_1 & d_0 & . & d_{-1} & d_{-2} & \dots & d_{-n} \\
 1 & 9 & \dots & 1 & 0 & 2 & . & 0 & 1 & \dots & 6
 \end{array}$$

$$d = \sum_{i=-n}^m 10^i \times d_i$$

Binary Representation

$$\begin{array}{cccccccccccc} 2^m & 2^{m-1} & & 2^2 & 2^1 & 2^0 & & 2^{-1} & 2^{-2} & & 2^{-n} \\ b_m & b_{m-1} & \dots & b_2 & b_1 & b_0 & . & b_{-1} & b_{-2} & \dots & b_{-n} \\ 1 & 0 & \dots & 1 & 0 & 1 & . & 0 & 1 & \dots & 1 \end{array}$$

$$b = \sum_{i=-n}^m 2^i \times b_i$$

Examples

Fractional value	Binary representation	Decimal representation
1/8	0 . 0 0 1	0.125
25/16	1 . 1 0 0 1	1.5625
43/16	1 0 . 1 0 1 1	2.6875
9/8	1 . 0 0 1	1.125
1/3	0 . (0 1)	0.(3)
1/10		0.1

$$b = \sum_{i=-n}^m 2^i \times b_i$$

Precision

8 bits 0.0001100

16 bits 0.000110011001100

24 bits 0.00011001100110011001100

32 bits 0.0001100110011001100110011001100

1/10

0.0(0011)

0.1

The Patriot Missile Failure



$$1,676\text{m/s} \times 0.34\text{s} = 569.84\text{ m}$$

0.0000000095s

0.00011001100110011001100



Fixed-point representation

n bits

0 0 0 0 1 1 0 0 1 1 0 0 1 1 0 0 1 1 0 0

more precision
less range

middle?

more range
less precision

Floating-point representation



IEEE

$$(-1)^S \times M \times 2^E$$

$$1001.1101 \rightarrow 1.0011101 \times 2^3$$

$$0.011101 \rightarrow 1.1101 \times 2^{-2}$$

$$-0.011101 \rightarrow -1 \times 1.1101 \times 2^{-2}$$

Arithmetic
formats

①

Rounding
rules

③

Exception
handling

⑤

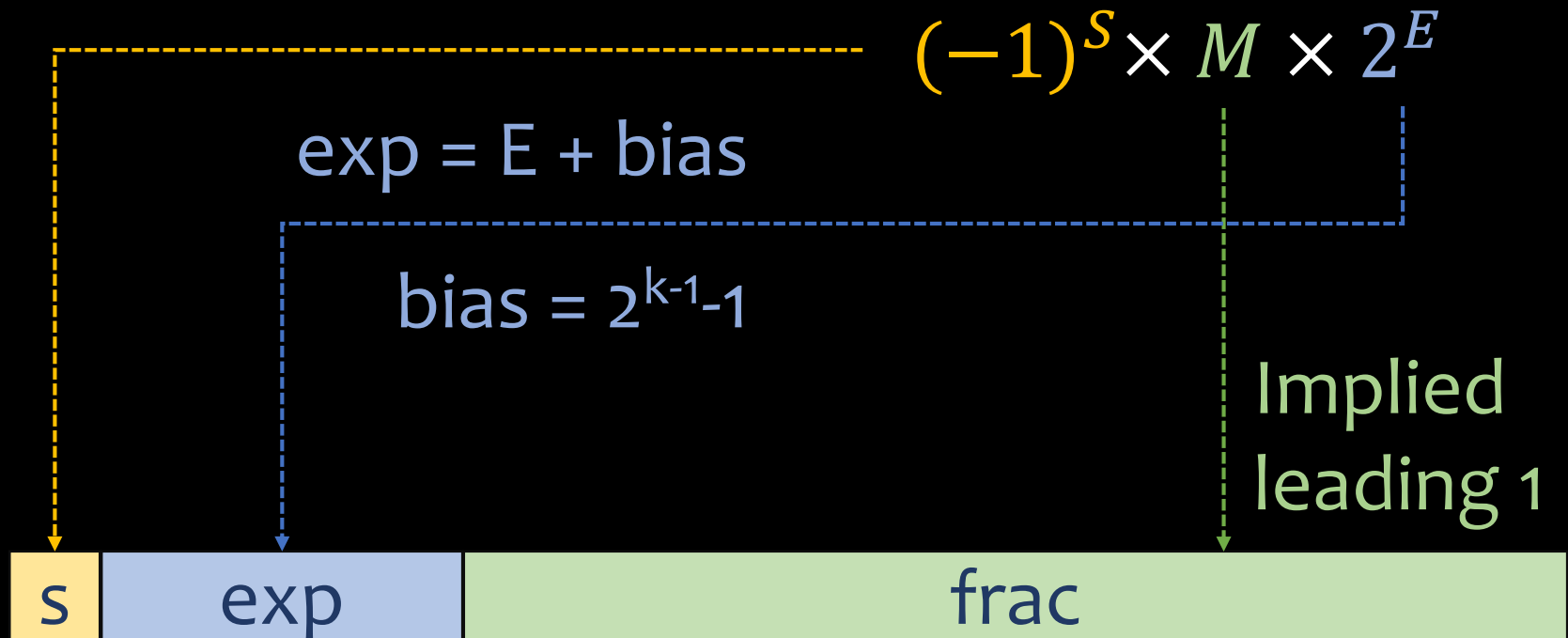
Interchange
formats

②

Operations

④

Standard floating-point format



Precision options

Single precision: 32 bits

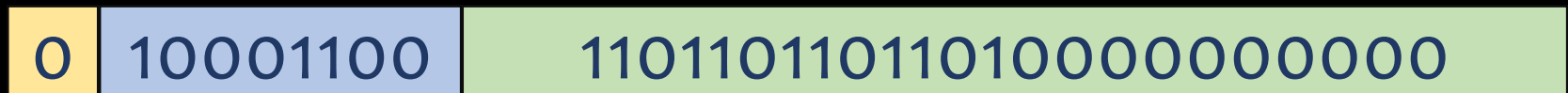


Double precision: 64 bits



Example

Single precision: 32 bits



10001100_2



$\text{exp} = 13 + 127$



$11011011011010000000000000000000_2$



1.1101101101101_2



$1.1101101101101_2 \times 2^{13}$

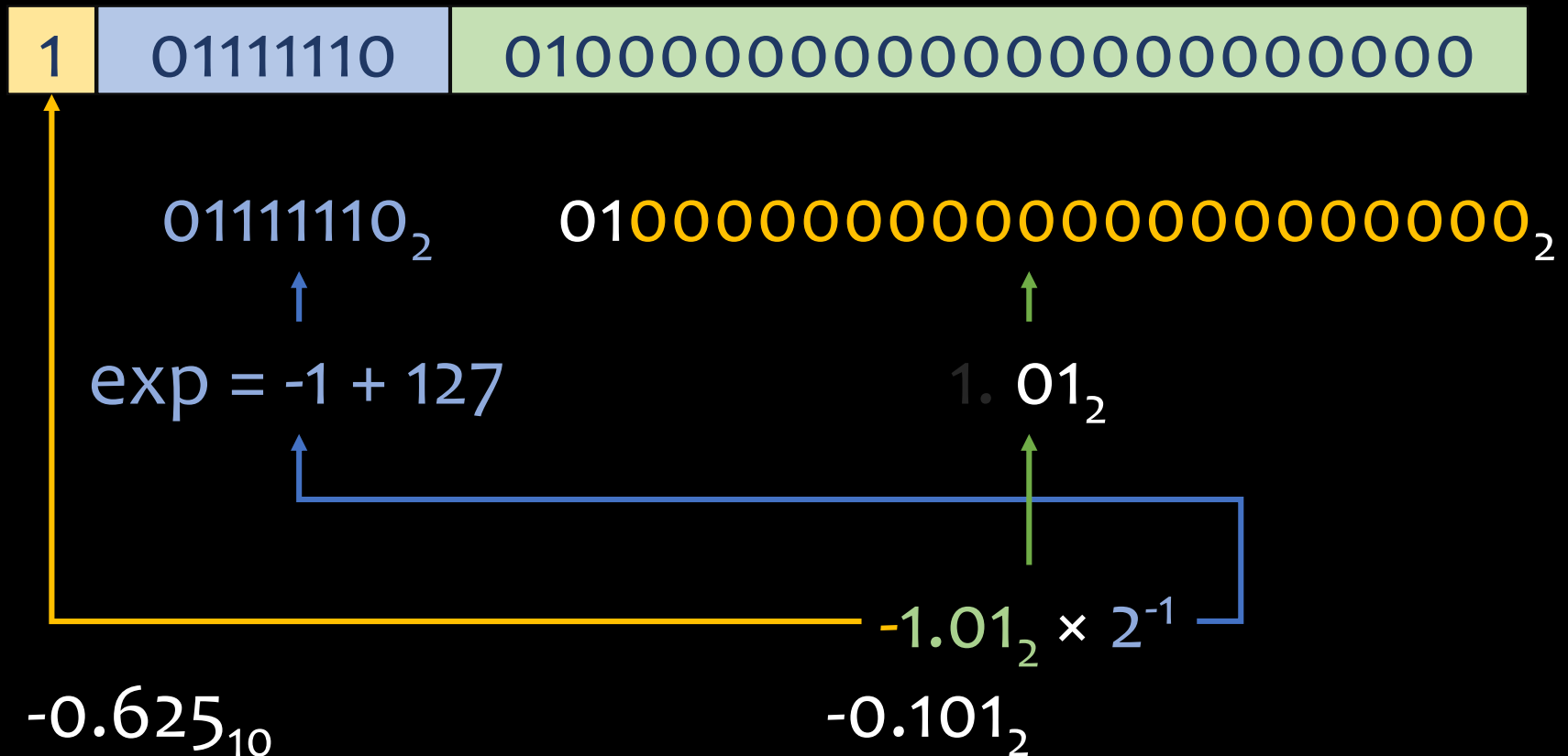
└─┘

15213.0_{10}

11101101101101_2

Example 2

Single precision: 32 bits



Summary

- Fixed-point representation
- Floating-point representation



James Gosling

Java Inventor

“95% of the folks out there are completely clueless about floating-point.”