Numbers

GeoComput & ML

15 Apr. 2021

Number Representation

Celebrity

Ancient Babylon

- Sexagesimal

< (10) | ▽ (1)

<< < ▽▽ | 20 × 60 + 12 = 1212

Ancient Babylon

What is this number ?

▽ ▽ ▽

Ancient Egypt

computing : 5 × 6

```
o o o o o | +

o o o o o | +
o o o o o | +

o o o o o | +
o o o o o | +
o o o o o | +
o o o o o | +
```

```
          |   | 0
-------------------
 o o o o o | + | 1
 o o o o o | + |
-------------------
 o o o o o | + | 1
 o o o o o | + |
 o o o o o | + |
 o o o o o | + |
```

Ancient Egypt

Question :

how many numbers we can represent using our 10 fingers in the binary mode?

Bit and Bytes

Bit

- Binary Digit
- represented by : 0 and 1 and their combinations

Bit and Bytes

Byte

- unit of digital information
- commonly consisted of 8 bits
- historically : one word length

Raster Data Types

GDAL data type	minimum	maximum
Byte	0	255
UInt16	0	65,535
Int16, CInt16	-32,768	32,767
UInt32	0	4,294,967,295
Int32, CInt32	-2,147,483,648	2,147,483,647
Float32, CFloat32	-3.4E38	3.4E38
Float64, CFloat64	-1.79E308	1.79E308

Binary System

Conversion

- Binary to decimal

$$(1010)_2 = (10)_{10}$$

$$1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 0 \times 2^0 = 10$$

Conversion

- Decimal to binary

$(10)_{10} / (2)_{10} = 5..0$

$(5)_{10} / (2)_{10} = 2..1$

$(2)_{10} / (2)_{10} = 1..0$

$(1)_{10} / (2)_{10} = 0..1$

Floating-point

Definition

$$x = \pm \left(d_0 + \frac{d_1}{\beta^1} + \frac{d_2}{\beta^2} + \ldots + \frac{d_{p-1}}{\beta^{p-1}} \right) \beta^E$$

β : base	$0 \leq d_i \leq \beta - 1$
p : precision	$i = 0, \ldots, p-1$
$[L, U]$: exponent range	$E \in [L, U]$

Definition

- mantissa : $d_0 d_1 d_2 \ldots d_{p-1}$
- fraction : $d_1 d_2 \ldots d_{p-1}$
- sign, exponent, mantissa : stored separately

Definition

- normalisation : d_0 always non-zero unless zero
- in $\beta = 2$, $d_0 = 1$ and not stored to save space

Properties

- floating number system : finite and discrete

total number of normalized floating numbers

$$\boxed{2(\beta - 1)\beta^{p-1}(U - L + 1) + 1}$$

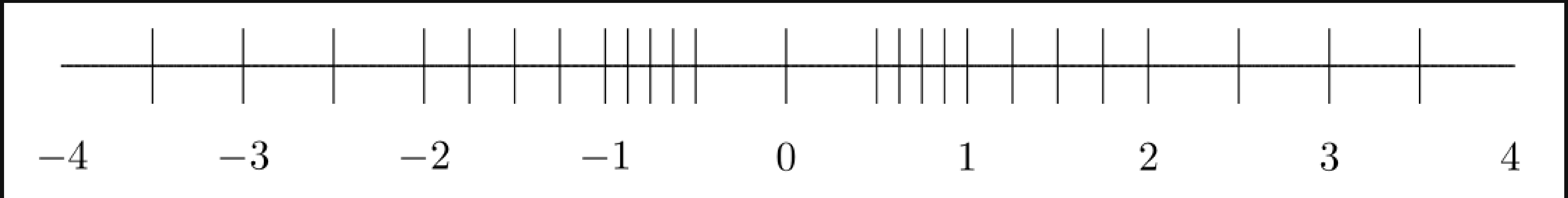
underflow level : $UFL = \beta^L$

overflow level : $OFL = \beta^{U+1}(1 - \beta^{-p})$

Properties

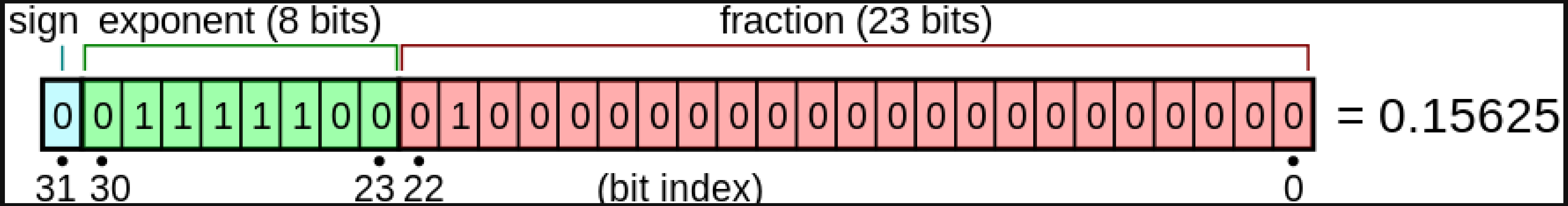
Example : toy system

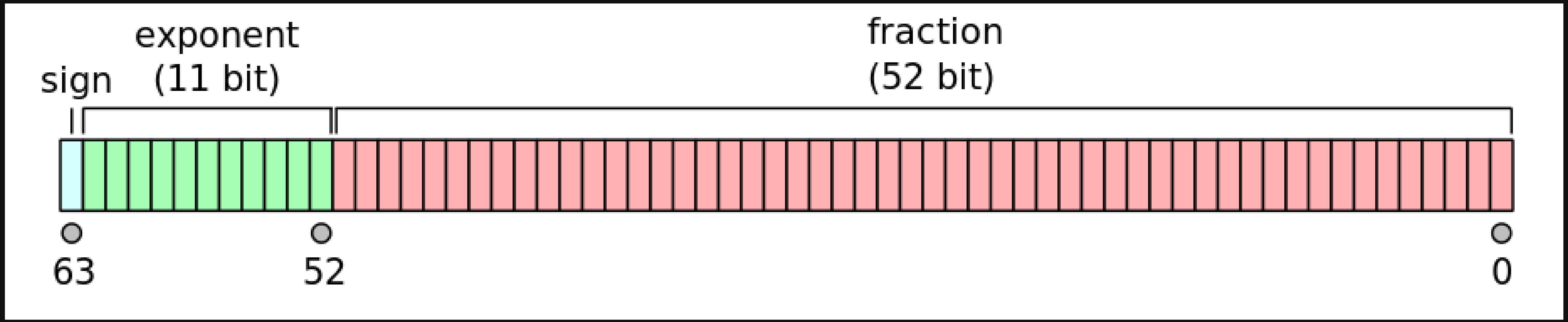
$$\beta = 2,\ p = 3,\ E \in [-1, 1]$$



IEEE 754-2008 standard

- 32-bit base-2 format (single precision)
- 64-bit base-2 format (double precision)





Approximation

machine numbers : real number exactly representable in a floating number system

- truncation : 1.751 => 1.7
- rounding : 1.751 => 1.8

Machine Precision

the accuracy of the floating point system

- truncation : $\epsilon_{mach} = \beta^{1-p}$
- rounding : $\epsilon_{mach} = \beta^{1-p}/2$

Real Cases

```
>>> import numpy as np
>>> np.arange(0,1,0.1) == 0.6


array([False, False, False, False, False, False, False, False, False,
       False])
```

Real Cases

```
main()
{
 float x = 16777216.00 ;
 float y = 1.00;
 float z = 5.00;
 printf ("%f\t%f\t%f\n",x,x+y,x+z);
}
```

```
16777216.000000    16777216.000000    16777220.000000
```

Acknowledgement

Thanks for Your Attention

There are only 10 types of people in the world. Those who understand binary and those who don't. ☻

References

- M. Holmes, Introduction to Scientific Computing and Data Analysis, 2016
- B. Gustafsson, Scientific computing from a historical perspective, 2010
- M. Heath, Scientific Computing An Introductory Survey, 2018
- https://grass.osgeo.org/grass78/manuals/r.out.gdal.html
- https://en.wikipedia.org/wiki/Babylonian_mathematics
- https://en.wikipedia.org/wiki/Ancient_Egyptian_mathematics
- https://en.wikipedia.org/wiki/Ancient_Egyptian_multiplication
- https://en.wikipedia.org/wiki/Floating-point_arithmetic
- https://en.wikipedia.org/wiki/Pythagoras
- https://global-geography.org/af/Geography/Europe/Greece/Pictures/Samos/Pythagoras_Statue_Pythagorio_

BH

A Game

10000 people stand in a queue and begin to announce their position number in sequence. Those who got odd numbers will be removed from the queue after the end of announcement. The remainders reform the queue and begin the next round announcement. Again, the odd numbers will be eliminated. The process repeats until only one person is left in the game as the winner. To be a winner, which position should you hold?