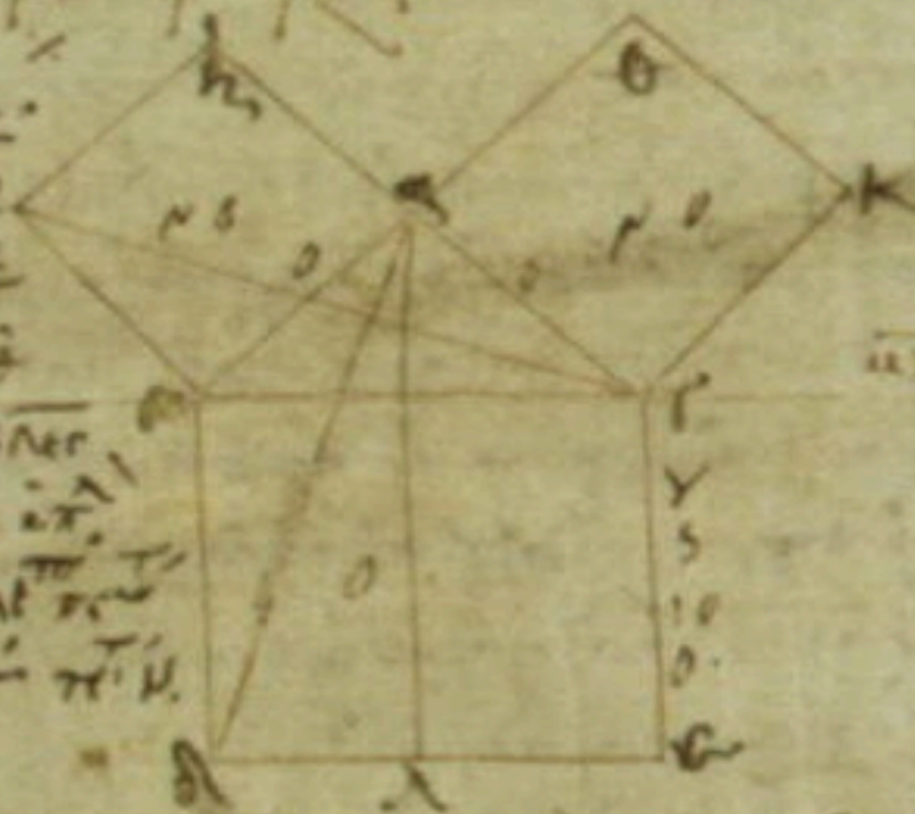
I See Your 127.32+, A Tale of Rationals

Xavier Noria
@fxn

EuRuKo 2013

```
5.00 * 4.33 = 21.65
 |      |      |
stake   |      |
        |      |
       odds    |
               |
             profit
```

```
2.25 * 4.33 =  9.74 (partial match 1)
2.75 * 4.33 = 11.90 (partial match 2)
───────────────────
5.00          21.64 (should be 21.65)
```


η
θ
κ

Natural Numbers (ℕ)

0, 1, 2, 3, ...

$$0 = \{\}$$

$$1 = 0 \cup \{0\} = \{0\}$$

$$2 = 1 \cup \{1\} = \{0, 1\}$$

$$3 = 2 \cup \{2\} = \{0, 1, 2\}$$

$$\vdots$$

$$n + 1 = n \cup \{n\} = \{0, 1, 2, \ldots, n\}$$

Generalization: ordinals and cardinals

* ——> #

* ——> #

* ——> #

* ——> #

* ——> #

Theorem (Cantor): $\mathrm{Card}(A) < \mathrm{Card}(\mathrm{P}(A))$

$n < 2^n$

$$\mathrm{Card}(\mathbb{N}) < \mathrm{Card}(P(\mathbb{N}))$$

Natural numbers map to the unsigned integer types of some languages like C

Ruby does not have unsigned integers

Integer Numbers ($\mathbb{Z}$)

$$\ldots, -3, -2, -1, 0, 1, 2, 3, \ldots$$

Integers are constructed from the naturals:

$$(a, b) \sim (c, d) \iff a + d = b + c$$

$$\mathrm{Card}(\mathbb{Z}) = \mathrm{Card}(\mathbb{N})$$

$$\begin{array}{cccccc} 0 & 2 & 4 & 6 & & 2n \\ | & | & | & | & & | \\ 0 & 1 & 2 & 3 & \cdots & n \end{array}$$

$$\begin{array}{cccccc} 1 & 3 & 5 & 7 & & 2n+1 \\ \Big| & \Big| & \Big| & \Big| & & \Big| \\ 0 & 1 & 2 & 3 & \cdots & n \end{array}$$

$$\begin{array}{ccccccccccc} 2n-1 & & 5 & 3 & 1 & 0 & 2 & 4 & 6 & & 2n \\ \vert & & \vert & \vert & \vert & \vert & \vert & \vert & \vert & & \vert \\ -n & \cdots & -3 & -2 & -1 & 0 & 1 & 2 & 3 & \cdots & n \end{array}$$

Ruby has arbitrary-precision integers

```
> (1..100).reduce(:*)
 => 9332621544394415268169923885626670049
07159682643816214685929638952175999932299
15608941463976156518286253697920827223758
251185210916864000000000000000000000000
```

Fixnums are immediate values in MRI

```
/* embeds integer in VALUE */

((VALUE)(((SIGNED_VALUE)(i))<<1 | FIXNUM_FLAG))
```

```
/* reads integer from VALUE */

(long)RSHIFT((SIGNED_VALUE)(x),1)
```

```c
/* numeric.c */

static VALUE
fix_succ(VALUE num)
{
    long i = FIX2LONG(num) + 1;
    return LONG2NUM(i);
}
```

```c
/* include/ruby/ruby.h */

static inline VALUE
rb_long2num_inline(long v)
{
    if (FIXABLE(v))
        return LONG2FIX(v);
    else
        return rb_int2big(v);
}
#define LONG2NUM(x) rb_long2num_inline(x)
```

Bignums have a mixed representation in MRI

```c
/* ruby/ruby.h */

struct RBignum {
    struct RBasic basic;
    union {
        struct {
            long len;
            BDIGIT *digits;
        } heap;
        BDIGIT ary[RBIGNUM_EMBED_LEN_MAX];
    } as;
};
```

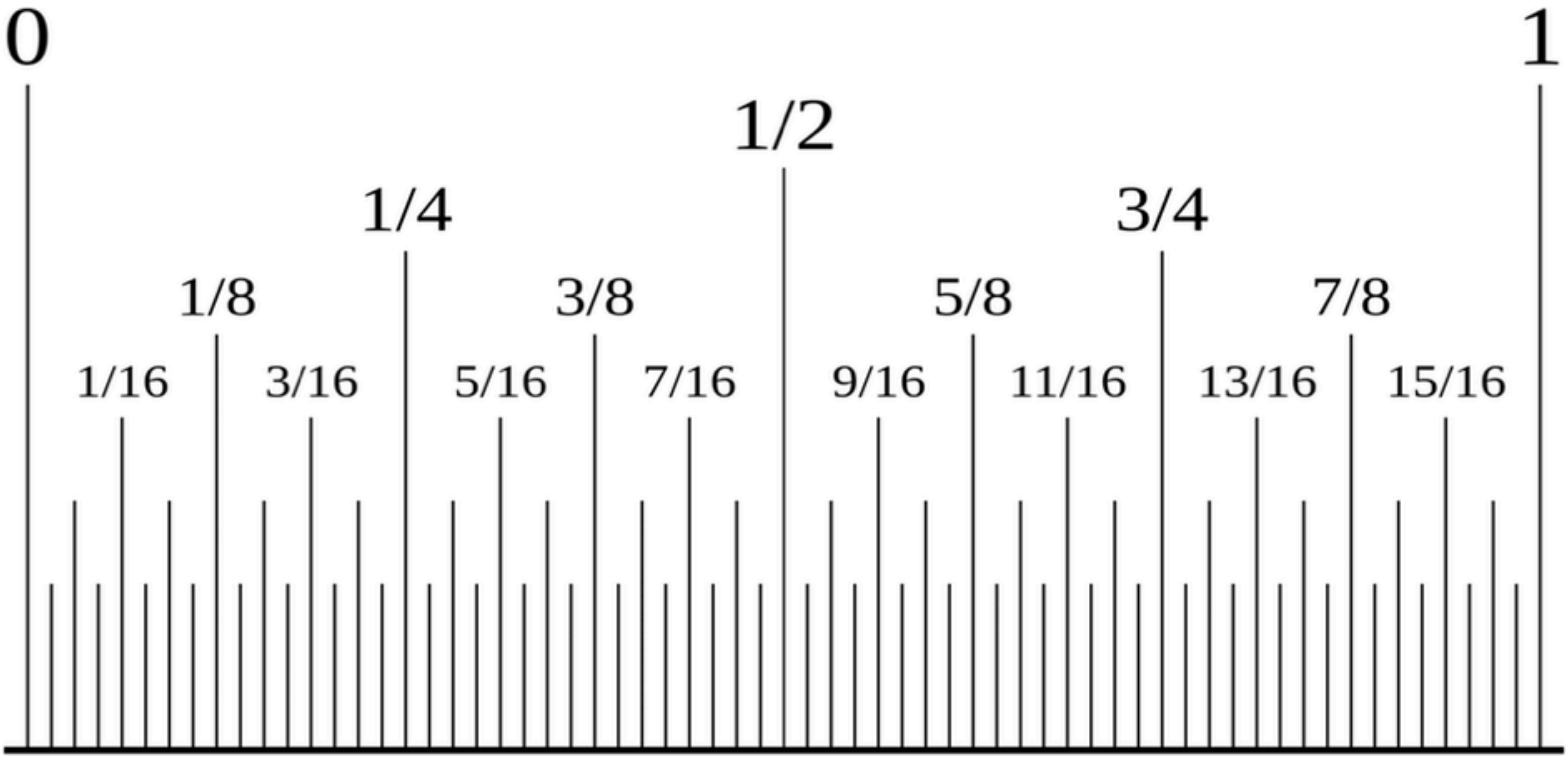
LibTomMath

Rational Numbers ($\mathbb{Q}$)

Rationals are constructed from the integers:

$$(a, b) \sim (p, q) \Leftrightarrow aq = pb$$

$$\mathrm{Card}(\mathbb{Q}) = \mathrm{Card}(\mathbb{N})$$


0
1
1/2
1/4
3/4
1/8
3/8
5/8
7/8
1/16
3/16
5/16
7/16
9/16
11/16
13/16
15/16

Ruby has rationals

```
struct RRational {
    struct RBasic basic;
    VALUE num;
    VALUE den;
};
```

* Exact arithmetic, including division
* Predictable across technologies

pgmp

Real Numbers (ℝ)

Reals are constructed from the rationals:

* Equivalence classes of Cauchy sequences of rationals

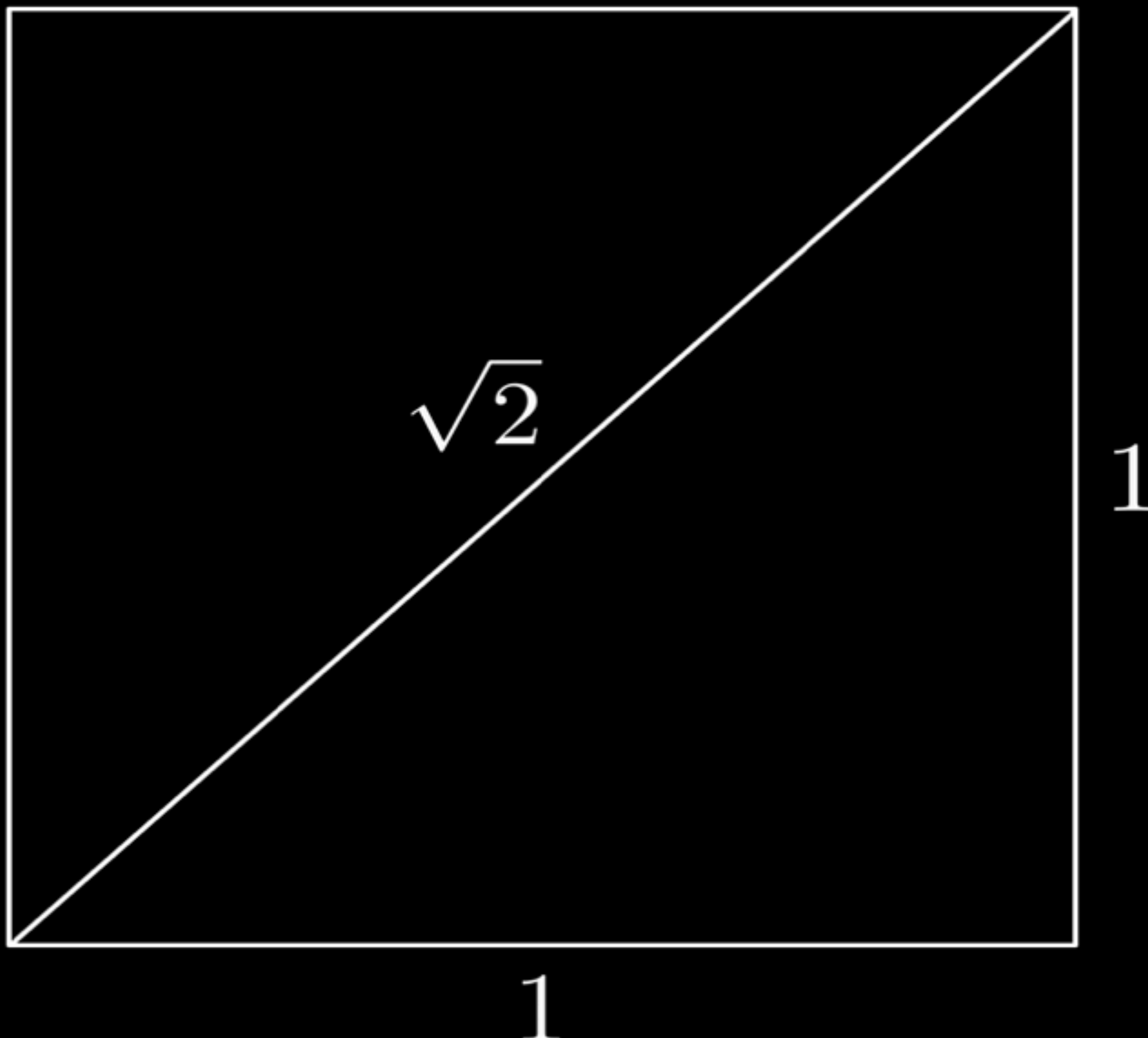
* Dedekind cuts

* Categorical characterization

* ...

$$\mathrm{Card}(\mathbb{R}) = \mathrm{Card}(P(\mathbb{N})) > \mathrm{Card}(\mathbb{N})$$

Reals that are not rationals
are called *irrationals*


$\sqrt{2}$
1
1

$$\text{sqrt(2)} = \frac{a}{b}$$

$$\mathrm{sqrt}(2) = \frac{a}{b} \quad \Rightarrow$$

$$\Rightarrow \quad 2 = \frac{a\hat{}2}{b\hat{}2}$$

$$\text{sqrt}(2) = \frac{a}{b} \quad \Rightarrow$$

$$\Rightarrow \quad 2 = \frac{a\text{^}2}{b\text{^}2} \quad \Rightarrow$$

$$\Rightarrow \quad 2b\text{^}2 = a\text{^}2$$

Math generally speaking studies numbers as

abstract entities,

irrespective of their representation

Theorem (Euclid): If p is a prime such that $p|ab$, then p divides a, or p divides b.

When is the representation of a rational in a given base finite?

1.256

$$1.256 = \frac{1256}{10\^{}3}$$

$$1.256 = \frac{1256}{10^3} = \frac{8 * 157}{8 * 125}$$

$$1.256 = \frac{1256}{10^3} = \frac{8 * 157}{8 * 125} = \frac{157}{125}$$

A fraction in lowest terms p/q has a finite representation in base b iff q divides b^n for some n.

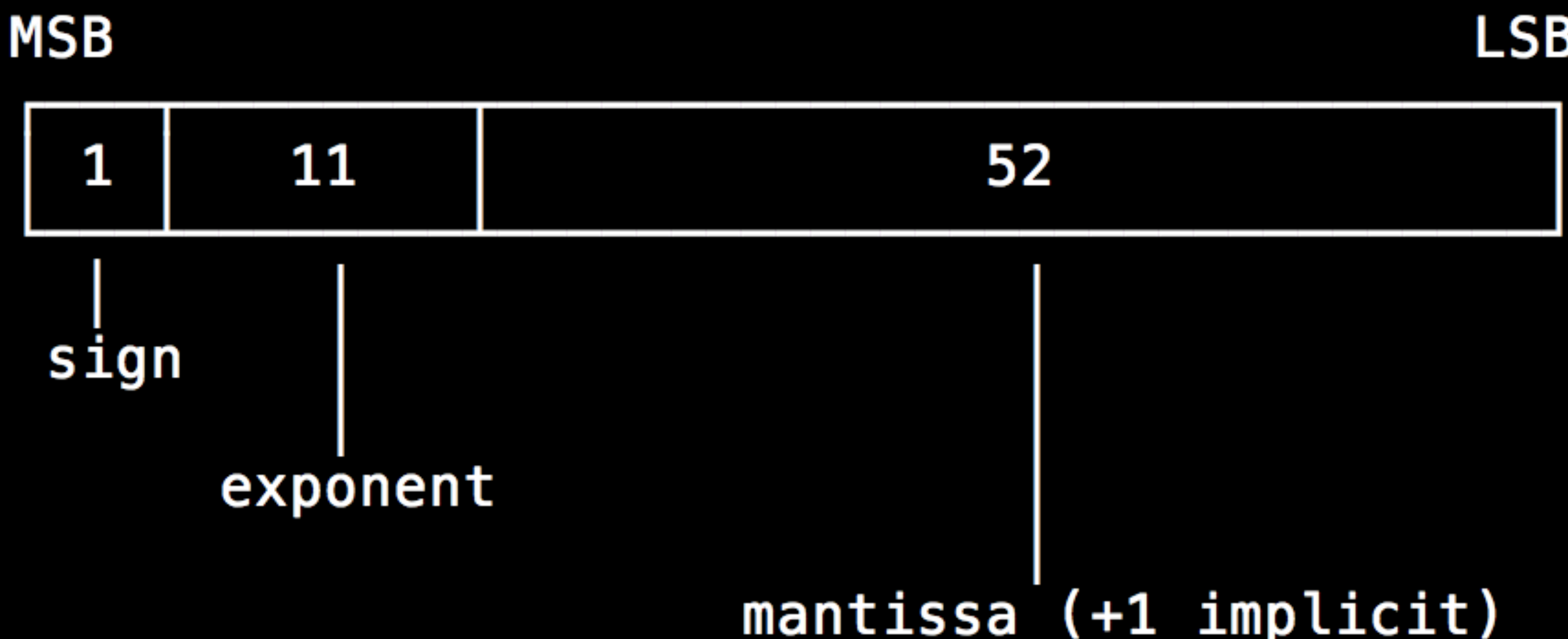
Cents < 1 with finite representation in base 2:

0.00, 0.25, 0.50, 0.75

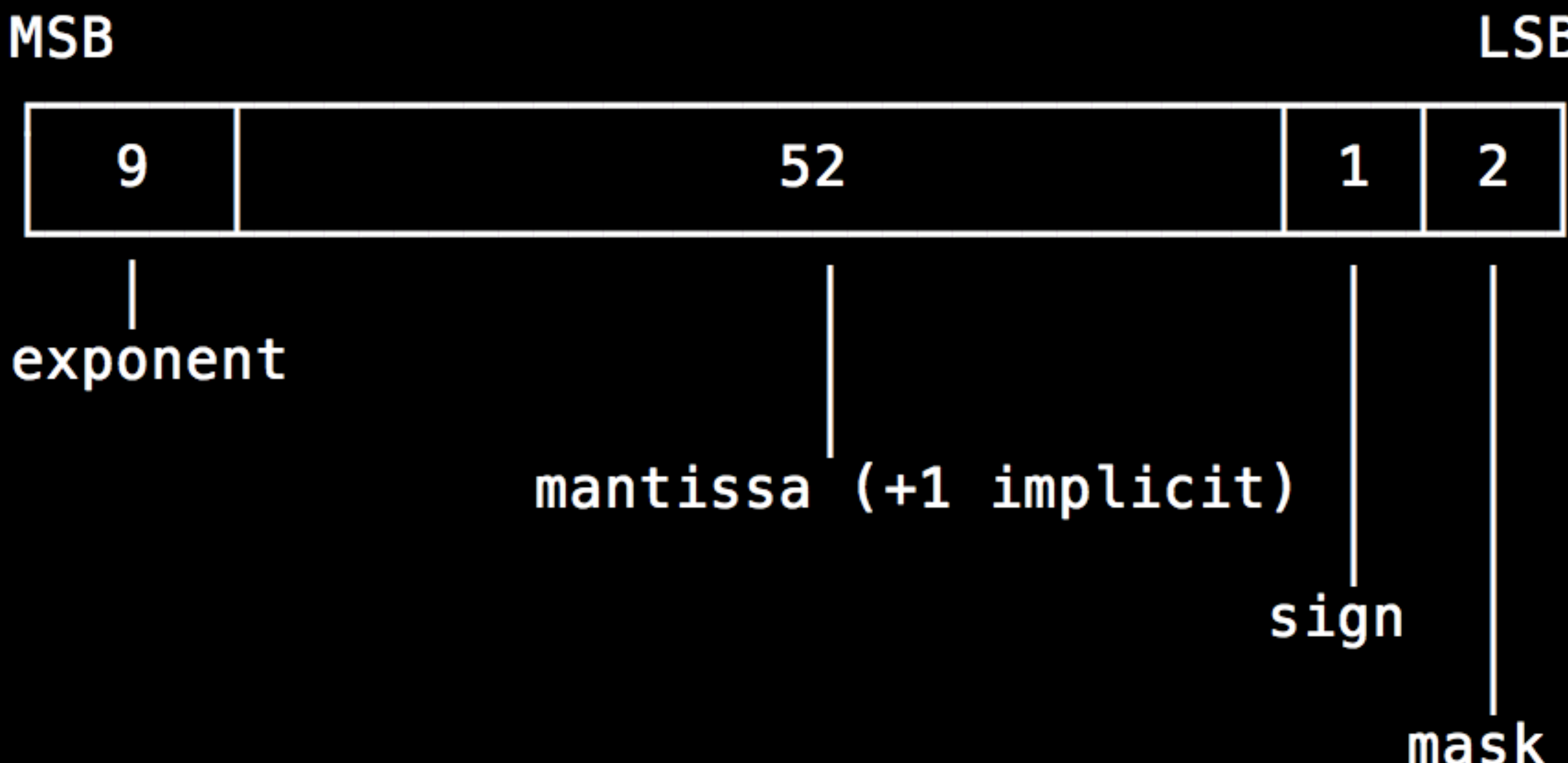
The other 96 are periodic.

Irrational numbers have an infinite
non-periodic representation in any base

IEEE 754 double precision (Wrapped by RFloat)



Immediate Flonum (Ruby 2.0, 64-bit platforms)



```
1.2 - 1.1 == 0.1 # => false
```

```ruby
require 'bigdecimal'

a = BigDecimal.new('1.2')
b = BigDecimal.new('1.1')
c = BigDecimal.new('0.1')

a - b == c # => true
```

Internal representation of a BigDecimal:

* Arbitrary-precision integer as mantissa

* Mantissa uses base 10^*N*

* Exponent

* Sign

* Flags

```c
/* ext/bigdecimal/bigdecimal.h */

typedef struct {
    VALUE  obj;
    size_t MaxPrec;
    size_t Prec;
    SIGNED_VALUE exponent;
    short  sign;
    short  flag;
    BDIGIT frac[FLEXIBLE_ARRAY_SIZE];
} Real;
```

BigDecimal caveats

```ruby
require 'bigdecimal'

one   = BigDecimal.new('1')
three = BigDecimal.new('3')

three*(one/three) == one
# => false
```

```ruby
require 'bigdecimal'
require 'bigdecimal/util'

65.1.to_d # DON'T DO THIS
```

```
require 'bigdecimal'
require 'bigdecimal/util'

'65.1'.to_d # GOOD
```

Undecidable Statements

Is there any cardinal between Card($\mathbb{N}$) and Card($P(\mathbb{N})$) = Card($\mathbb{R}$)?

Continuum Hypothesis: No

Paul Cohen proved CH to be undecidable in the 60s (provided ZF is consistent)

Thanks

Yukihiro Matsumoto
Pat Shaughnessy
Dirkjan Bussink
Charles Nutter
Koichi Sasada
Vicent Martí

Thanks for listening!