2.1.2

◎ 浮動小数点数 (floating point number)

例) 10ケタの10進数で、実数を近似する：

① $u = \pm \boxed{d_0}\boxed{d_1}\boxed{d_2}\boxed{d_3}\boxed{d_4} . \boxed{d_5}\boxed{d_6}\boxed{d_7}\boxed{d_8}\boxed{d_9}$ $\quad 0 \le d_j \le 9$ for $j = 0, \dots, 9$

↑ 符号の情報を別途持つ。

このとき、表現できる数値の範囲は

$-99999.99999 \le u \le 99999.99999$

4/20 ↓

② 4/27

$u = \pm \boxed{m_0} . \boxed{m_1}\boxed{m_2}\boxed{m_3}\boxed{m_4}\boxed{m_5}\boxed{m_6}\boxed{m_7}\boxed{m_8} \times 10^{\pm \boxed{e_1}}$

↑ 符号の情報を別途持つ

$$\begin{cases} 0 < m_0 \le 9, \\ 0 \le m_j \le 9 \quad \text{for } j = 1, \dots, 8, \\ 0 \le e_1 \le 9. \end{cases}$$

このとき $\quad u = \pm \left(m_0 + \sum_{j=1}^{8} \frac{m_j}{10^j} \right) \times 10^{\pm e_1}$

すると、表現できる数値の範囲は

$$-9.99999999 \times 10^9 \le u \le 9.99999999 \times 10^9$$

-9999999990 $\qquad$ 9999999990

①の表現に比べて、表現できる数値の絶対値の範囲が広がる！
但し、絶対値が大きくなる程、隣どうしの数の間隔も広がる。

②のように、指数を用いた数値の表現 ⇒ 浮動小数点 (浮動小数)
「浮動小数点」の語源：表す数値の絶対値の大きさに応じて小数点の位置が移動するため。
⇔ ①は「固定小数点」。

• 浮動小数点 nまる実数の表現：

$$(-1)^S \times M \times B^E ,$$

{ S：符号，M：仮数（mantissa），B：基数（base），
E：指数（exponent）．

例）2.9979×10^8 （m/s ＝真空中での光速）
$= 29.979 \times 10^7$
$= 0.29979 \times 10^9$
⋮

のように，指数をずらすことで，1つの数値に対して
いくつもの表現がある．

そこで，仮数の整数部を 1以上 10未満にすると，
表現が一意に定まる．→ 正規化（normalization）

• 計算機上での浮動小数（＝点数）の例：実数 r の表現

符号 S： 1ビット，基数 B＝2，指数 E，
仮数 M： nビット．
仮数の正規化：$1 \le M < 2$ をみたすよう，E を定める．

（仮数の表現）$m_0 . m_1 m_2 \cdots m_{n-1}$ ，$m_0 = 1$，
$m_j \in \{0, 1\}$ for $j = 1, \ldots, n-1$.

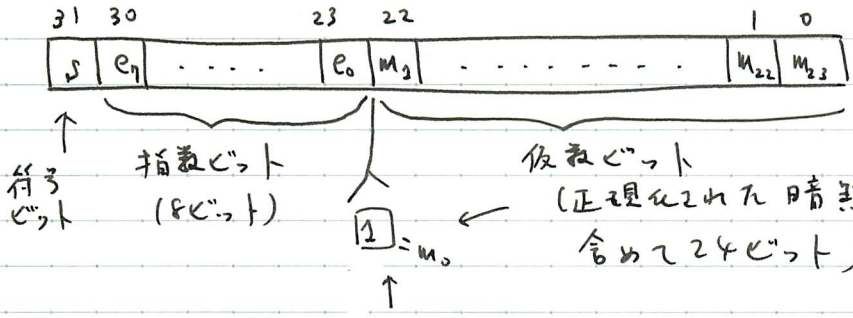
$$\text{s.t.} \quad M = \sum_{j=0}^{n-1} \left(\frac{m_i}{2^i} \right) .$$

より $r = (-1)^S \times M \times 2^E$ と表す．

アイトリプルイー

- IEEE 754 浮動小数点 規格 … 現在，幅広く用いられている．

① 単精度（32ビット）



$\begin{cases} (1)\ m_0 = 1 \text{ とおき，} & 1 \le M < 2 \text{ をみたすよう正規化する．} \\ (2)\ m_0 = 0 \text{ とおき，非正規化数でより多くの数値を表す．} \end{cases}$

(1) $\begin{cases} r = (-1)^s \times M \times 2^E, \\ M = 1 + \sum_{i=1}^{23} \left(\frac{m_i}{2^i} \right), \\ E = \left(\sum_{i=0}^{7} 2^i e_i \right) - 127. \end{cases}$

$-126 \le E \le 126$ の場合と（指数ビットはすべてが0 すべてが1 の場合を除く）

(2) $\begin{cases} r = (-1)^s \times M \times 2^E, \\ M = \sum_{i=1}^{23} \left(\frac{m_i}{2^i} \right) & \text{（非正規化数）}, \\ E = -127. & \text{（指数ビットはすべて0）} \end{cases}$

特殊な数

- ±0 : $\begin{cases} M = 0 & \text{（仮数ビットはすべて0）} \\ E = -127 & \text{（指数ビットはすべて0）} \\ s \in \{0, 1\} \Rightarrow +0 \text{ or } -0 & \text{を表す．} \end{cases}$

・ ±∞ : ビット
$$\begin{cases} M = 0 \quad (\text{仮数} \cdots \text{すべて} 0) \\ E = \cancel{127}\ 128 \quad (\text{指数ビットはすべて} 1) \\ S \in \{0,1\} \Rightarrow +\infty \text{ or } -\infty \text{ を表す}. \end{cases}$$

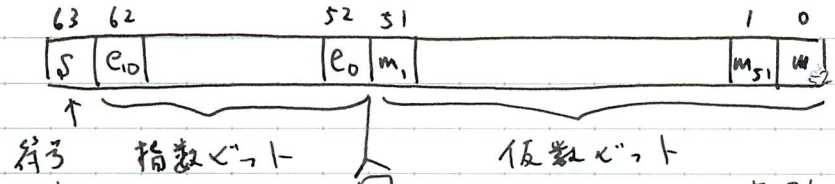
・ NaN (Not a Number / 非数). $0 \div 0$ などの値.

仮数部の
$M \neq 0$ (少なくとも1つのビットは1)

$E = \cancel{127}\ 128$ (指数ビットはすべて1)

$S \in \{0,1\}$ (符号の値は無関係)

② 倍精度 (64ビット)



(1) $m_0 = 1$ となされた正規化数

$$\begin{cases} r = (-1)^S \times M \times 2^E, \\ M = 1 + \sum_{i=1}^{52} \left(\frac{m_i}{2^i} \right), \quad E = \left(\sum_{i=0}^{10} 2^i e_i \right) - 1023. \end{cases}$$

$(-1022 \leq E \leq 1022.)$

指数ビットはすべてが0の場合とすべてが1の場合を除く.

(2) $m_0 = 0$ となされた非正規化数.

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

$$M = \sum_{i=1}^{52} \left(\frac{m_i}{2^i} \right), \quad E = -1023 \quad (\text{指数ビットはすべて} 0).$$

特殊な数　　仮数ビットがすべて0　　指数ビットがすべて0

- ±0 : $M = 0$, $E = -1023$, $S = 0$ (+0) or 1 (−0).
- ±∞ : $M = 0$, $E = 1024$, $S = 0$ (+∞) or 1 (−∞).

指数ビットがすべて1　　符号は無関係.

- NaN : $M \neq 0$, $E = 1024$, $S \in \{0, 1\}$

◎ 浮動小数点演算における誤差.

① 丸め誤差 (rounding error)

(例) π ⟶ 3.1416 | // × 10^0

仮数部：十進5桁.　　計算機内部に ガード桁 (guard digit / bit) を設けて丸めを行う.

4/27

② 桁落ち誤差 (cancellation error)

5/11

(例) 仮数部十進9桁.

$$\begin{array}{r} 1.23456789 \times 10^0 \\ -) \ 1.23450000 \times 10^0 \\ \hline 0.00006789 \times 10^0 \end{array}$$

⇓ 正規化

6.7890000×10^{-5}

有効桁数が9桁から4桁に減少.　　これらの桁は正確とは限らない.

☆ 絶対値の大きさがほぼ等しい2数の加減算で生じることがある.

③ 情報落ち.

(例) 仮数部十進3桁.

$$\begin{cases} x = 1.00 \times 10^4 \\ y = -1.00 \times 10^4 \\ z = 1.00 \times 10^0 \quad (= 1.00) \end{cases}$$

に対し, $x + y + z$ を計算する.