

2. 数式と数値の表現 . 基型アルゴリズム

2.1 数値の表現 (テキスト p.11)

- ・ ビット, バイト, ワードの概念 (→ ノート p.2)
- ・ 以後, 1ワード = n ビットとする.
- ・ 加減乗除などの基本演算を ワード単位 で考える.
- ・ 数値 $\longleftrightarrow$ データ (符号)
1234

2.1.1 整数の表記 (2の補数)

- ・ 現在, ほとんどのコンピュータで数値を2進法で扱う.
- ・ 以下 1ワード = 32 ビット.

① 符号なし 整数



$$b_j \in \{0, 1\}$$

が表す整数 u のとき,
$$u = \sum_{j=0}^{31} b_j 2^j$$

で b_j を定める.

例	0 - - - -	0 = 0
	0 - - -	0 1 = 1
	0 - -	1 0 = 2
	⋮	
	1 - - -	-- 1 1 = $2^{32}-1 = 4294967295$

② 符号付き 整数



↑
 u の符号

が表す数値を u とする.

・ 負の数と「2の補数 (complement)」で表す。

・ n に対し、 $-n$ を表す手順：

① n のビットをすべて反転。

② 1 を加える。

$$\left. \begin{array}{l} \text{①} \\ \text{②} \end{array} \right\} \begin{array}{l} -n \text{ は} \\ n + (-n) = 0 \\ \text{となる。} \end{array}$$

形式的には、
$$u = \sum_{j=0}^{30} b_j 2^j - b_{31} \cdot 2^{31} \quad \text{で } b_j \text{ を定める。}$$

例) 1ワード = 5ビット (数値 4ビット + 符号ビット) で -9 を表す場合。

① $9 =$

0	1	0	0	1
---	---	---	---	---

② ビット反転

1	0	1	1	0
---	---	---	---	---

③ 1 を加える。

1	0	1	1	1
---	---	---	---	---

 $= -9$

↑
「16の補数」とみられる。
(-2^4)

2.1.2 浮動小数点数 (floating point number)

例) 1079 の 10 進数に 実数に近似する。

方法②
$$u = \pm \underbrace{d_1 d_2 d_3 d_4}_{\substack{\uparrow \\ \text{符号の情報も別途持つ。}}} . \underbrace{d_5 d_6 d_7 d_8 d_9}_{\substack{0 \leq d_j \leq 9 \\ j=0, \dots, 9}}$$

2.9.2.2. 表現できる数値の範囲は

$$-99999.99999 \leq u \leq 99999.99999$$

方法② $u = \pm \boxed{m_0} . \boxed{m_1} \boxed{m_2} \boxed{m_3} \boxed{m_4} \boxed{m_5} \boxed{m_6} \boxed{m_7} \boxed{m_8} \times 10^{\pm \boxed{e_1}}$

↑
符号の情報を別途持つ。

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

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

すると、表現できる数値の範囲は

$$- \underset{\text{11}}{9.99999999} \times 10^9 \leq u \leq \underset{\text{11}}{9.99999999} \times 10^9$$

$$- \underset{\text{11}}{9999999990} \quad \quad \quad \underset{\text{11}}{9999999990}$$

方法①の表現に比べて、同じ桁数で表現できる絶対値の範囲が広がる！

但し、 u の絶対値が大きくなるほど、隣どうしの数の間隔も広がる。

方法③のように、指数を用いた数値の表現 $\Rightarrow$ 浮動小数点 (浮動小数)

「浮動小数点」の設想: u の絶対値が大きいほど、小数点の位置が移動する。

$\Rightarrow$ ① $\rightarrow$ 「固定小数点」。

③ 浮動小数点による実数の表現。

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

$\begin{cases} S: \text{符号 (sign)}, & M: \text{仮数 (mantissa)} \\ B: \text{基数 (base)}, & E: \text{指数 (exponent)} \end{cases}$

例) $u = 2.9979 \times 10^8$ ($\% \approx$ 真空の光速)

$$= 29.979 \times 10^7$$

$$= 0.29979 \times 10^9$$

のため、指数 (小数点の位置) を与えることで、1つの数値に対していくつかの表現が有り得る。

②

そこで、仮数の整数部と1以上10未満の3つのことで

表現を一意的に定める $\rightarrow$ 正規化 (normalization)

③ 計算機上の浮動小数点の例：実数 r の表現

$S \in \{0, 1\}$, $B = 2$, E : 整数。

M : m ビット。

M の正規化: $1 \leq M < 2$ とする。 E を定める。

$$M = m_0 . m_1 m_2 \dots m_{n-1}, \quad \begin{cases} m_0 = 1. \\ m_j \in \{0, 1\} \\ \text{for } j = 1, \dots, n-1 \end{cases}$$

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

$$\text{つまり } u = (-1)^S \times M \times 2^E \quad \text{と表す。}$$

④ 浮動小数点演算における誤差 $\rightarrow$ 本授業では省略
(昨年(2015年)の授業で少し程度扱った。)

2.1.3. フレイトッド $\Rightarrow$ 省略。

2.1.4. -2進表記 $\Rightarrow$ 省略。