計算機数学I (2019)
第2回

照井 章（筑波大学 数理物質系 数学域）
Akira Terui (Institute of Mathematics, University of Tsukuba)

第1回のまとめ

- コンピュータの基本構成
- コンピュータ内部の紹介
- メモリ（主記憶装置）の配置と容量
  - メモリ容量の単位
  - バイト (byte), ワード (word)

第2回の内容

- 計算機上での数値と数式の表現

2.1 数値の表現

数値の表現（テキスト p. 12）

- ビット，バイト，ワードの概念 (前回)
- 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⇄データ（符号）: 1対1対応

2.2.1 整数の表記（2の補数）

整数の表記

- 現在，ほとんどすべての計算機では2進法で数を表す
- 以下，1ワード= 32ビットとする

符号なし整数

b_{31}	…………	b_3	b_2	b_1	b_0

が表す数を v とする

$$b_j \in \{0, 1\}$$ に対し

$$v = \sum_{j=0}^{31} b_j 2^j$$ で b_j を定める

符号なし整数の例

2進数　10進数

$$0 = 0$$

$$1 = 1$$

$$10 = 2$$

$$\vdots$$

$$1 \cdots\cdots 1 = 2^{32} - 1 = 4294967295$$

符号つき整数

符号ビット

b_{31}	…………	b_3	b_2	b_1	b_0

が表す数を v とする

符号なし整数に符号 (+ / –) の情報を
加える → b_{31}

符号つき整数

1. 「符号ビット」による表現
2. 「2の補数」による表現

符号つき整数の例 (1) 符号ビット＋符号なし整数

b_{31}	…………	b_3	b_2	b_1	b_0

が表す数を v とする

$$v = (-1)^{b_{31}} \left(\sum_{j=0}^{30} b_j 2^j \right)$$

符号ビット

（0の符号は + とする）

符号つき整数の例 (1) 符号ビット＋符号なし整数

b_{31}	....	b_3	b_2	b_1	b_0

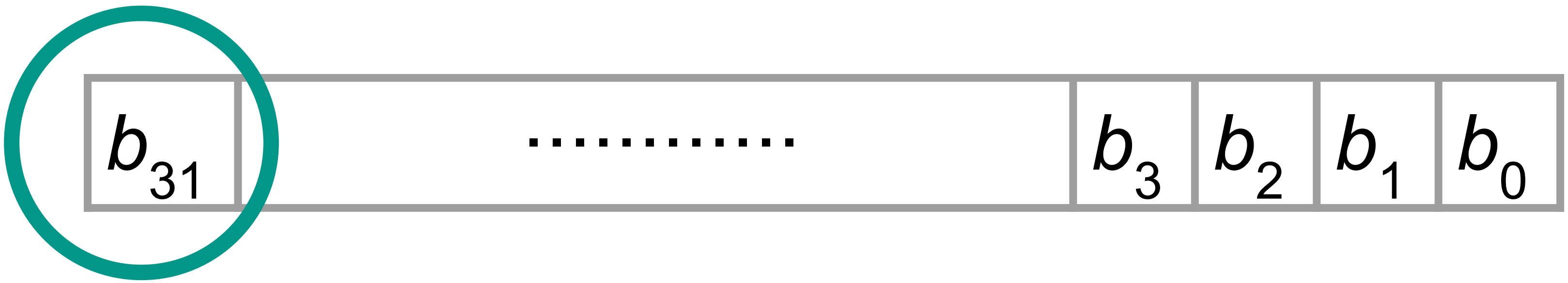
2進数	10進数

$$00\cdots000 = 0$$
$$10\cdots001 = -1$$
$$10\cdots010 = -2$$
$$10\cdots011 = -3$$
$$10\cdots100 = -4$$
$$\vdots$$

長所：符号反転が容易

短所：負の数の加算がやや面倒（減算が必要）

符号つき整数 (2) 2の補数



n に対し、v = - n を表す手順

1. n のビットをすべて反転する
2. 1を加える

$$v = (-b_{31})2^{31} + \left(\sum_{j=0}^{30} b_j 2^j \right)$$

符号つき整数 (2) 2の補数

(例) 1ワード=5ビット（数値4ビット＋符号1ビット）で –9 を表す

	符号	8	4	2	1
(1) 9の表現:	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

(4) (1) + (3) = 0 が成り立つことに注意

2.1.2 浮動小数点数
(Floating point number)

実数を有限桁の数で近似する

- 有理数(循環小数など)
- 無理数(円周率 π, 自然対数の底 e など)
- 有理数や代数的数を計算機上で厳密に表す手法もありますが、ここでは有限桁で近似する手法について議論します

実数を有限桁で近似するアイデア (1)

10桁の10進数で実数を近似する場合

符号の情報は別途もつ

$$v = \pm\ \boxed{d_0}\boxed{d_1}\boxed{d_2}\boxed{d_3}\boxed{d_4}\ .\ \boxed{d_5}\boxed{d_6}\boxed{d_7}\boxed{d_8}\boxed{d_9}$$

$$0 \le d_j \le 9$$

表現できる数値は

$$-99999.99999 \le v \le 99999.99999$$

実数を有限桁で近似するアイデア (2)

10桁の10進数で実数を近似する場合

符号の情報は別途もつ

$$v = \pm \boxed{m_0} . \boxed{m_1}\boxed{m_2}\boxed{m_3}\boxed{m_4}\boxed{m_5}\boxed{m_6}\boxed{m_7}\boxed{m_8} \times 10^{\boxed{(e_0-5)}}$$

$$0 \le m_j \le 9, \quad 0 \le e_0 \le 9 \quad (-5 \le (e_0 - 5) \le 4)$$

表現できる数値は

$$-99999.9999 \le v \le 99999.9999$$

相対誤差がほぼ一定

絶対誤差と相対誤差

- x: 真の値, x': 近似値
  - 絶対誤差: $|x' - x|$
  - 相対誤差: $|x' - x|/|x|$

実数を有限桁で近似するアイデア (2)

浮動小数点数（浮動小数）

符号の情報は別途もつ

$$v = \pm \boxed{m_0}\,.\,\boxed{m_1}\boxed{m_2}\boxed{m_3}\boxed{m_4}\boxed{m_5}\boxed{m_6}\boxed{m_7}\boxed{m_8} \times 10^{\boxed{(e_0-5)}}$$

$$0 \leq m_j \leq 9, \quad 0 \leq e_0 \leq 9 \quad (-5 \leq (e_0 - 5) \leq 4)$$

現在，実数を有限桁で近似する方法として広く用いられている

実数を有限桁で近似するアイデア (1)

固定小数点数

符号の情報は別途もつ

$$v = \pm\ \boxed{d_0}\boxed{d_1}\boxed{d_2}\boxed{d_3}\boxed{d_4}\ .\ \boxed{d_5}\boxed{d_6}\boxed{d_7}\boxed{d_8}\boxed{d_9}$$

$$0 \leq d_j \leq 9$$

現在，浮動小数に比べると
利用は限定的

浮動小数による実数の表現

$$v = (-1)^s \times M \times B^E$$

s: 符号 (sign) $\in \{0,1\}$,　M: 仮数 (mantissa),

B: 基数 (base),　E: 指数 (exponent)

浮動小数の正規化

$$\begin{aligned} v &\simeq 2.9979 \times 10^8 \\ &= 29.979 \times 10^7 \\ &= 0.29979 \times 10^9 \\ &= \cdots \end{aligned}$$

（真空中の光速 m/s）

表現を一意にするために

仮数部の整数部を1以上B未満の整数で表す

計算機上での浮動小数の例

$$v = (-1)^s \times M \times B^E$$

$s \in \{0,1\}$, M: n ビット, $1 \leqq M < 2$,

$B = 2$,　E: l ビット

$$M = m_0.m_1 m_2 \cdots m_{n-1},$$

$$m_0 = 1, \quad m_j \in \{0,1\} \quad (1 < j < n)$$

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

浮動小数演算における丸め誤差

- 丸め (rounding, round-off):
  与えられた真の値 x を有限桁の数（浮動小数）x' で近似すること
- 丸め誤差: 実数を浮動小数に丸める際に生じる際の誤差

浮動小数演算における丸め誤差

- x: 真の値, x': 近似値
  - 絶対誤差: $|x' - x|$
  - 相対誤差: $|x' - x|/|x|$

マシンイプシロン ε_M

- 非ゼロの浮動小数の丸め誤差の相対誤差の最大値
- $x \neq 0$: 真の値, $x' \neq 0$: 浮動小数に丸めた値
  F_{min}: 非ゼロの浮動小数の絶対値の最小値
  F_{max}: 非ゼロの浮動小数の絶対値の最大値

$$\varepsilon_M = \max_{F_{\mathrm{min}} \leq |x| \leq F_{\mathrm{max}}} \left(\frac{|x - x'|}{|x|} \right)$$

マシンイプシロン ε_M

- $x = 1$とおくと、$(1 + \varepsilon) \neq 1$をみたす$\varepsilon > 0$ の最小値と定めることもできる

$$\varepsilon_M = \max_{F_{\min} \leq |x| \leq F_{\max}} \left(\frac{|x - x'|}{|x|} \right)$$

IEEE 754 浮動小数規格

- 現在多くの計算機で採用されている
  浮動小数の事実上の標準規格
- 1985年に最初の規格が制定される
  現在の規格は2008年に制定
- 2進数、10進数およびいくつかの精度別に
  複数の規格から成る
- 最も広く用いられているのは、2進数の
  「倍精度」と呼ばれる規格

IEEE 754 倍精度 (64ビット)

（m_0=1 とおいた正規化数の表現）

s	e_{10}	·····	e_0	m_1	········	m_{51}	m_{52}

符号ビット

指数ビット11ビット

m_0=1

仮数ビット（正規化された暗黙の仮数ビットを含め53ビット）

$$v = (-1)^s \times M \times 2^E,$$

$$M = 1 + \sum_{i=1}^{52} \left(\frac{m_i}{2^i} \right), \quad E = -1023 + \sum_{i=1}^{52} e_i 2^i$$

(ただし -1022 ≦ E ≦ 1022)

2.1.3 自然数の表記（グレイコード）（省略）

2.1.4 整数の表記（−2進）（省略）

第2回のまとめ

- 2進数，符号つき整数の表現，2の補数
- 実数の有限桁の近似と浮動小数
- 絶対誤差，相対誤差
- 丸め誤差，マシンイプシロン
- IEEE浮動小数規格

第3回の内容

- 多精度整数の表記と加算