

コンピュータの基礎知識: 二進数

情報科学の世界 2

2024 年度前期

佐賀大学理工学部 只木進一

- ① 二進数と十進数
- ② 二進数演算
- ③ 減算
- ④ 接頭辞: Prefix
- ⑤ 10 進数、2 進数、8 進数、16 進数
- ⑥ 課題

コンピュータ内でのデータの取り扱い

- コンピュータ内では2進数 (binary numbers)
- 2進数1桁 $[0, 1]$ を bit と呼ぶ
- 2進数8桁 $[0, 255]$ を Byte と呼ぶ
- 文字コード
 - ASCII コード: 7bit で数字やアルファベットを表現
 - 日本語コード: JIS、SJIS、EUC は2バイト
 - 多言語混在: UTF-8 など

十進数と桁の意味

- 十進数 (decimal numbers) では $\{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$ の 10 個の記号を使用
- k 桁目は 10^{k-1} が何個あるかを表す

$$1634 = 1 \times 10^3 + 6 \times 10^2 + 3 \times 10^1 + 4 \times 10^0$$

$$3021 = 3 \times 10^3 + 0 \times 10^2 + 2 \times 10^1 + 1 \times 10^0$$

- $10^0 = 1$
- $9 + 1 = 10$ という桁上がりの規則

十進数と二進数の相互変換

- 十進数から二進数へ
- 2 のべき乗の和で表す
- 二進数は、0b を先頭に付けて表記

$$\begin{aligned} 53 &= 32 + 16 + 4 + 1 = 2^5 + 2^4 + 2^2 + 2^0 \\ &= 0b00110101 \end{aligned}$$

$$\begin{aligned} 130 &= 128 + 2 = 2^7 + 2^1 \\ &= 0b10000010 \end{aligned}$$

$$\begin{aligned} 163 &= 128 + 32 + 2 + 1 = 2^7 + 2^5 + 2^2 + 2^0 \\ &= 0b10100011 \end{aligned}$$

$$\begin{array}{r}
 2 \overline{)53} \\
 2 \overline{)26} \\
 2 \overline{)13} \\
 2 \overline{)6} \\
 2 \overline{)3} \\
 2 \overline{)1} \\
 0
 \end{array}
 \begin{array}{r}
 1 \\
 0 \\
 1 \\
 0 \\
 1 \\
 1
 \end{array}
 \begin{array}{c}
 \uparrow
 \end{array}$$

- 2で割った商と余りを求める
- これを0になるまで繰り返す
- 余りを上から下に読む

$$53 = 0b00110101$$

なぜ、コンピュータは2進数を使うのか

- 素子の構造が簡素
 - 状態はオン (on) とオフ (off) の二つ
 - リレー (relay)、真空管 (vacuum tubes)、トランジスタ (transistors)
- 演算規則が簡素

a	b	$a + b$	$a \times b$
0	0	0	0
0	1	1	0
1	0	1	0
1	1	10	1

二進数加算乗算の例

$$\begin{array}{r} 101 \\ +) 11 \\ \hline 1000 \end{array}$$

$$\begin{array}{r} 1011 \\ +) 110 \\ \hline 10001 \end{array}$$

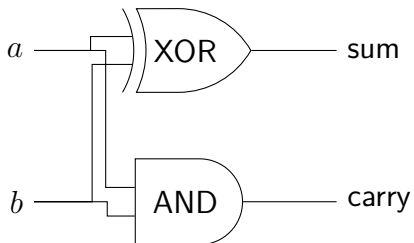
$$\begin{array}{r} 101 \\ \times) 11 \\ \hline 101 \end{array}$$

$$\begin{array}{r} 101 \\ +) 101 \\ \hline 1111 \end{array}$$

$$\begin{array}{r} 101 \\ \times) 101 \\ \hline 101 \end{array}$$

$$\begin{array}{r} 101 \\ +) 101 \\ \hline 11001 \end{array}$$

Half Adder



a	b	sum	carry
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

減算

引き算は、上の桁からの「借り」があり、足し算に比べて難しい。コンピュータは、2進数でそのように引き算をしているか。

- コンピュータが扱うのは有限桁
- 8bit と考える: 扱えるのは0から255まで

2の補数: two's complement

- 整数 n に対する 2 の補数
 - n の二進表現で 0 と 1 を反転し、1 を加える
- $n = 5$ の場合

$$\begin{aligned} 5 &= 0b00000101 \\ &\Rightarrow 0b11111010 + 0b00000001 \\ &= 0b11111011 \end{aligned}$$

2の補数を使った減算

- $9 - 5$ をそのまま実行

$$\begin{aligned} 9 - 5 &= 0b00001001 - 0b00000101 \\ &= 0b00000100 = 4 \end{aligned}$$

- 9 に 5 に対する 2 の補数を足す

$$0b00001001 + 0b11111011 = 0b100000100$$

- 二進表現が 9 桁になった。一番上の桁を削除して 4 を得る。

$$0b00000100 = 4$$

n に対する 2 の補数を使うとは

- n に対する 2 の補数とは
 - 0 と 1 を反転させる

$$0b11111111 - n$$

- 1 を足す

$$0b11111111 - n + 1 = 0b100000000 - n$$

- m から n を引く代わりに、 m に n に対する 2 の補数を足す

$$m + (0b11111111 - n + 1) = m - n + 0b100000000$$

- 後で、 $0b100000000$ 、つまり桁が溢れた部分を取り除けば良い

10進数で見てみる

9 - 5 の代わりに、9 に 5 に対する 2 の補数を足すことを 10 進で見てみる: 8bit の場合

- $256 = 0b100000000$, $255 = 0b11111111$
- 5 に対する 2 の補数

$$(256 - 1) - 5 + 1$$

- $9 + (5 \text{ に対する } 2 \text{ の補数})$

$$9 + (256 - 1) - 5 + 1 = 9 - 5 + 256$$

減算: $5 - 9$

- $9 = 0b00001001$ に対する 2 の補数

$$0b11110110 + 0b00000001 = 0b11110111$$

- 5 に 9 に対する 2 の補数を足す。9 桁目は現れない

$$0b00000101 + 0b11110111 = 0b11111100$$

- これは $4 = 0b00000100$ に対する 2 の補数

$$0b11111011 + 0b00000001 = 0b11111100$$

- 2 の補数は、対応する負の数を表している

例: $23 - 17$

- $23 = 0b00010111$
- $17 = 0b00010001$
- 17 に対する 2 の補数: $0b11101111$
- $23 + (17 \text{ に対する } 2 \text{ の補数})$

$$0b00010111 + 0b11101111 = 0b100000110 = 6 + 256$$

問題

- $n \geq 0$ の整数に対して 2 の補数をとることが、 $-n$ を表すことがわかった。
- 8bit では、256 通りの数しか表現できない。
- つまり、8bit では、0 から 255 までの整数とそのマイナス符号の整数を表すことができない。
- 負でない整数は $0 \leq n \leq 127$ までしか表現できない。

負の整数の範囲を確かめるために、 -1 と -128 の 8bit 二進表現を求めなさい。

−1

−128

接頭辞: Prefix

- 3桁毎に名前を付ける
東アジアでは4桁毎に名前を付ける
- $1\text{k} = 10^3$ 、 $1\text{M} = 10^3\text{k}$ 、 $1\text{G} = 10^3\text{M}$ 、 $1\text{T} = 10^3\text{G}$ 、 $1\text{P} = 10^3\text{T}$
- $1\text{m} = 10^{-3}$ 、 $1\mu = 10^{-3}\text{m}$ 、 $1\text{n} = 10^{-3}\mu$
- 2進の場合には、1000の代わりに $2^{10} = 1024$ を使う
- SI (System International d'Unités) 接頭辞
https://unit.aist.go.jp/nmij/library/SI_prefixes/

10 進数、2 進数、8 進数、16 進数

- n 進数: 使える記号が n 個
- 10 進数 (decimals): $\{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$
 $9 + 1 = 10$
- 2 進数 (binaries): $\{0, 1\}$
 $1 + 1 = 10$
- 8 進数 (octals): $\{0, 1, 2, 3, 4, 5, 6, 7\}$
 $7 + 1 = 10$
- 16 進数 (hexadecimals): $\{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F\}$
 $F + 1 = 10$
 - 小文字のアルファベットを使うこともある

16 進数の利用: 文字コード

- 16 進 2 桁は 8bit: 0x00 ~ 0xFF
- 16 進数は先頭に 0x を付けて表示
- ASCII コード: 英数文字を表現: 7bit
0x00 ~ 0x7F
- 通常の日本語は 16 進 4 桁 (2 Bytes)
- UNICODE
<http://www.unicode.org/charts/>

16 進数の利用: インターネット

- インターネットのアドレス標記
- 8bit 毎 (octet) に区切って記述する
- ネットマスク
- MAC (Media Access Control) アドレス

課題

インターネットの通信速度を表す場合、bps と Bps という表記が現れる。違いを調べなさい。