コンピュータ基礎 (第3回配布)

第2章データ表現

（1）数値データ（numeric data)
　　整数(integer), 浮動小数点数(floating point number)等
（2）非数値データ（nonnumeric data）
　　文字コード、論理データ、制御コードなど

担当: 福井大学 大学院工学研究科
情報・メディア工学専攻
森 眞一郎 (moris@u-fukui.ac.jp)

[高校情報 復習]データの標記法

- 2進数表現、8進数表現、16進数表現

[一般形] r進n桁表示

$$(a_{n-1}a_{n-2}..a_2a_1a_0)_{(r)} = a_{n-1}\bullet r^{n-1} + a_{n-2} * r^{n-2} + \cdot\ \cdot\ \cdot + a_2\bullet r^2 + a_1\bullet r^1 + a_0\bullet r^0$$

2進数表現
　ai のとり得る値は 0か1
8進数表現
　2進数表現したものを3桁づつまとめて表現したもの。
　aiの取り得る値は 0から7
16進数表現
　2進数表現したものを4桁づつまとめて表現したもの。
　aiの取り得る値は 0から15
　（便宜上、10～15をA～Fの1文字に対応付けて表現）
10進数表現
　aiの取り得る値は0から9

[高校復習]2進数、8進数、16進数、10進数

10進	2進	8進	16進
0	00000	0	0
1	00001	1	1
…	…	…	…
7	00111	7	7
8	01000	10	8
9	01001	11	9
10	01010	12	A
11	01011	13	B
12	01100	14	C
13	01101	15	D
14	01110	16	E
15	01111	17	F
16	10000	20	10

1. 数の表現

位取り記数法

「桁」は figure あるいは digit と表現することがある

r進数（基数 r)で 整数部 n桁 小数部 m桁の数が表す（10進数で換算した時の）値Nは

$$N = (a_{n-1}a_{n-2}\cdot\cdot\cdot a_2a_1a_0\ .\ a_{-1}a_{-2}\cdot\cdot\cdot a_{-m})_{(r)}$$
$$= a_{n-1}\bullet r^{n-1} + a_{n-2} * r^{n-2} + \cdot\ \cdot\ \cdot + a_2\bullet r^2 + a_1\bullet r^1 + a_0\bullet r^0 + a_{-1}\bullet r^{-1} + \cdot\ \cdot\ \cdot + a_{-m}\bullet r^{-m}$$

ここで
　r: 基数(radix) あるいは 底(base)　a_i: ディジット(digit)　$0 \le a_i < r$の整数

$(110)_{10}=$　　$(0.1)_{10}=$
$(110)_8=$　　$(0.1)_8=$
$(110)_2=$　　$(0.1)_2=$

1. 数の表現

r 進数

「桁」は figure あるいは digit と表現することがある

r進数（基数 r）でn桁の計算に関する用語

「桁上げ(Carry:キャリー)」 ある桁の数値が r-1 のとき 1を足すと上位桁に桁上げが発生する。
（より一般的には ある桁で加算した結果 r 以上となると上位桁に桁上げが発生）

「(桁)借り(Borrow:ボロー)」 $(10)_r$から1を引くと上位桁からの（桁）借りが発生する。
（より一般的には ある桁で減算した結果、負となる場合にrを加算して正にし、上位桁を1減算する）

「オーバーフロー」 計算の結果 n+1 桁目に桁上げが発生すること

「ビット(bit)」 2進数1桁のこと (binary digit) ..... [情報理論的には 『情報量』をあらわす基本単位]

「バイト(Byte)」 8bitを一まとめにした単位

2進数（binary number） : 基数 r=2 で表現する数 $0 \le a_i < 2$
8進数（octal number） : 基数 r=8 で表現する数 $0 \le a_i < 8$
16進数（hexadecimal number）: 基数 r=16 で表現する数 $0 \le a_i < 16$ （10以上の数はA,B,C,D,E,Fの１文字で表現）

1. 数の表現

r 進数

「桁」は figure あるいは digit と表現することがある

r進数（基数 r）でn桁の計算に関する用語

「最上位桁 と 最下位桁 」（一般系）
Most Significant Digit(MSD) Least Significant Digit(LSD)

特に２進数表現 (r=2)の場合
「最上位ビット と 最下位ビット」
Most Significant Bit (MSB) Least Significant Bit(LSB)

プログラミング言語 C や C++では
整数に関しては、
１６進数は 0x
２進数は 0b
を最上位桁の前に付けて１０進数と区別。
(滅多に使わないがMSDに0 を付けると
なんと8進数と解釈されるので要注意！）

（P.37中段） あくまでも、使用中の教科書での 統一表記ルール として
最下位桁の 後に b, o, d(あるいは省略), h をつけて、
それぞれ2進数、８進数、１０進数、１６進数を区別
（といいつつも、 文脈から明らかな場合は b,o,d,h を省略すると書かれている点に注意！！）

表記例

2進数（binary number）	:	0101**b**	111.01**b**	0.0001**b**
8進数（octal number）	:	17654321**o**	1234.5671**o**	0.0007**o**
16進数（hexadecimal number）	:	89ABCDEF**h**	1234.FEDC**h**	0.CBA98**h**

2. 基数の変換

整数部

「桁」は figure あるいは digit と表現することがある

r進数（基数 r）で 整数部I_0 (n桁) 小数部D_0 (m桁)の数が表す
（10進数で換算した時の）値Nは

$$N = (a_{n-1}a_{n-2}\cdots a_2a_1a_0 \bullet a_{-1}a_{-2}\cdots a_{-m})_{(r)} = (I_0 \quad . \quad D_0)_{(r)}$$
$$= a_{n-1}\bullet r^{n-1} + a_{n-2} * r^{n-2} + \cdots + a_2 \bullet r^2 + a_1 \bullet r^1 + a_0 \bullet r^0 + a_{-1}\bullet r^{-1} + \cdots + a_{-m}\bullet r^{-m}$$

n=4で考えると

I_0 は $I_0 = a_3r^3 + a_2r^2 + a_1r + a_0 = (a_3r^2 + a_2r^1 + a_1)r + a_0$
であるので、 I_0 を rで割った余りは a_0 となり 商をI_1 とすると

I_1 は $I_1 = a_3r^2 + a_2r + a_1 = (a_3r^1 + a_2)\ r + a_1$
であるので、 I_1 を rで割った余りは a_1 となり 商をI_2 とすると

I_2 は $I_2 = a_3r^1 + a_2 = (a_3)r + a_2$
であるので、 I_2 を rで割った余りは a_2 となり 商をI_3 とすると

I_3 は $I_3 = a_3$
であるので、 I_3 を rで割った余りは a_3 となり 商は0となる

2. 基数の変換

小数部

「桁」は figure あるいは digit と表現することがある

r進数（基数 r）で 整数部I_0 (n桁) 小数部D_0 (m桁)の数が表す
（10進数で換算した時の）値Nは

$$N = (a_{n-1}a_{n-2}\cdots a_2a_1a_0 \bullet a_{-1}a_{-2}\cdots a_{-m})_{(r)} = (I_0 \quad . \quad D_0)_{(r)}$$
$$= a_{n-1}\bullet r^{n-1} + a_{n-2} * r^{n-2} + \cdots + a_2 \bullet r^2 + a_1 \bullet r^1 + a_0 \bullet r^0 + a_{-1}\bullet r^{-1} + \cdots + a_{-m}\bullet r^{-m}$$

m=3で考えると

D_0 は $D_0 = a_{-1}r^{-1} + a_{-2}r^{-2} + a_{-3}r^{-3} = (a_{-1} + a_{-2}r^{-1} + a_{-3}r^{-2})r^{-1}$
であるので、 D_0 を r 倍すると整数部は a_{-1} となり 小数部をD_1 とすると

D_1 は $D_1 = a_{-2}r^{-1} + a_{-3}r^{-2} = (a_{-2} + a_{-3}r^{-1})r^{-1}$
であるので、 D_1 を r 倍すると整数部は a_{-2} となり 小数部をD_2 とすると

D_2 は $D_2 = a_{-3}r^{-1}$
であるので、 D_2 を r倍すると整数部は a_{-3} となり 小数部は0となる

2. 基数の変換

2進数と10進数の相互変換

「桁」は figure あるいは digit と表現することがある

r進数（基数 r）で 整数部I_0 (n桁) 小数部D_0 (m桁)の数が表す（10進数で換算した時の）値Nは

$$N=(a_{n-1}a_{n-2}\cdots a_2a_1a_0 \ . \ a_{-1}a_{-2}\cdots a_{-m})_{(r)} = (I_0 \ . \ D_0)_{(r)}$$

$$= a_{n-1}\bullet r^{n-1} + a_{n-2} * r^{n-2} + \cdots + a_2\bullet r^2 + a_1\bullet r^1 + a_0\bullet r^0 + a_{-1}\bullet r^{-1} + \cdots + a_{-m}\bullet r^{-m}$$

n=4,m=3の例

rが10なら $N = 1000a_3 + 100a_2 + 10a_1 + a_0 + \frac{1}{10}a_{-1} + \frac{1}{100}a_{-2} + \frac{1}{1000}a_{-3}$

rが2なら $N = 8a_3 + 4a_2 + 2a_1 + a_0 + \frac{1}{2}a_{-1} + \frac{1}{4}a_{-2} + \frac{1}{8}a_{-3}$

(1010.001) (r) が表す数値Nは？ 数値Nを r進数で表すと?

10進数の 0.8 0.4 0.1 は....

いろいろな求め方を使えると、あとあと便利！！

2. 基数の変換

2進数と10進数の相互変換

便利な表？

2	0. 5
4	0. 25
8	0. 125
16	0. 0625
32	0. 03175

2進数⇒10進数

1) 原理に忠実に ひたすら以下の式を計算

$$N = 2^{n-1}a_{n-1} + \cdots + 8a_3 + 4a_2 + 2a_1 + a_0 + \frac{1}{2}a_{-1} + \frac{1}{4}a_{-2} + \frac{1}{8}a_{-3} + \cdots + 2^{-m}a_{-m}$$

例題2’ 2進数 110.101b を10進数に変換せよ。

（例題2は 1101.01b なんか似ていないか？）

2） 割り算の利用 （小数部がm桁の場合）

$N' = (a_{n-1}a_{n-2}\cdots a_2a_1a_0 \ a_{-1}a_{-2}\cdots a_{-m})_{(r)}$ という整数と考えて N’を10進数で表した後に 2^mで割る。

いろいろな求め方を使えると、あとあと便利！！

2. 基数の変換

2進数と10進数の相互変換

便利な表？

0. 5	2
0. 25	4
0. 125	8
0. 0625	16
0. 03125	32
	64
	128
	256
	512
	1024

10進数⇒2進数

配布物からの訂正

1）基本形 （整数部 と 小数部を別々に計算）

例えば 26. 625d を 2進数に変換

整数部

26/2 = 13 ... 0
13/2 = 6 ... 1
6/2 = 3 ... 0
3/2 = 1 ... 1
1/2 = 0 ... 1 （最上位桁）

小数部

0.625 *2 = 1.25
0.25 *2 = 0.50
0.50 * 2 = 1.00 （最下位桁）

検算！

11010b = 16+8+2 = 26 (正解！) 0.101b = 0.5 + 0.125 = 0.625 (正解！)

2） 引き算の利用(整数部 と 小数部を別々に計算) 例えば 27. 875d を 2進数に変換

27 ー 16 = 11 (16d=10000b)
11 ー 8 = 3 (8d= 1000b)
3 ー 2 = 1 (2d= 10b)
1 ー 1 = 0 (1d= 1b)

0.875 −0.5 = 0.375 (0.5d = 0.1b)
0.375 ー0.25 = 0.125 (0.25d = 0.01b)
0.125 ー0.125 = 0 (0.125d = 0.001b)

2. 基数の変換

2進数と10進数の相互変換

2進数表現での小数表現の限界 … 有限桁の2進数で表現できない数値が存在！

0.725 *2 = 1.45
0.45 *2 = 0.9
0.9 *2 = 1.8
0.8 *2 = 1.6
0.6 *2 = 1.2
0.2 *2 = 0.4
0.4 *2 = 0.8

0.8 *2 = 1.6
0.6 *2 = 1.2
0.2 *2 = 0.4
0.4 *2 = 0.8

$0.725d = 0.101\dot{1}10\dot{0}$

循環小数表現となり、有限桁で表すことが不可能

有限ビットで表現すると、 真の値 との差が 「打ち切り誤差」 となる

少数点以下 m ビットで打ち切った場合の 打ち切り誤差 の絶対値は 2^{-m} 未満

（例えば少数点以下2ビットで打ち切ると、表現可能な小数部は 0.00 0.25 0.50 0.75 の4種類で 誤差の絶対値は 0.25 未満）

（2021.05.07追記あり　第4回配布分を参照のこと）

2. 基数の変換

2進数と16進数の相互変換

2進数　n ビット　で　$0 \sim 2^{n-1}$までの
2^n通りの数値を表現可能

2進数3桁　は　8進数1桁　に相当する表現が可能

2進数4桁　は　16進数1桁　に相当する表現が可能

10進	2進	8進	16進
0	0000	0	0
1	0001	1	1
2	0010	2	2
3	0011	3	3
4	0100	4	4
5	0101	5	5
6	0110	6	6
7	0111	7	7

10進	2進	8進	16進
8	1000	10	8
9	1001	11	9
10	1010	12	A
11	1011	13	B
12	1100	14	C
13	1101	15	D
14	1110	16	E
15	1111	17	F

3F40h　=　0011_1111_0100_0000b
(0x3F40　= 0b0011_1111_0100_0000)

3				F				4				0			
0	0	1	1	1	1	1	1	0	1	0	0	0	0	0	0

1011_1110b =　BEh
(0b1011_1110 = 0x3BE)

1	0	1	1	1	1	1	0
B				E			

1111_1111b =　FFh
(　= 15d*16d +15d = 255d)

1	1	1	1	1	1	1	1
F				F			

3. 負数の表現と加減算

符号付の2進数表現

n bitで表現された2進数の最上位ビット(Most Significant Bit:MSB)
を符号ビットとし、MSBをSとすると　S=0のとき正、S=1のとき負の数をあらわす。

下位 n-1ビットの表現法として、例えば以下の3つが考えられる

符号＋絶対値
　下位 n-1ビットで、数値の絶対値を表現する

2の補数
　符号を含めた n bitで2の補数表現とする

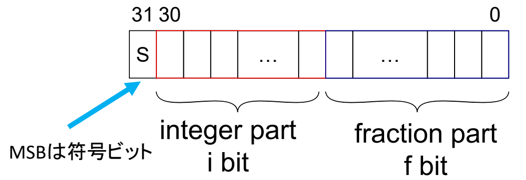
1の補数
　符号を含めた n bitで1の補数表現とする

実数の表現(2進数で符号＋絶対値表記の例)

- 固定小数点表現
  - 整数部(integer part) i桁と小数部(fraction part)j桁
  で実数を表現

10110.1　　(i=5,f=1)
-1011.01　　(i=4,f=2)
1.01101　　(i=1,f=5)

32bit固定小数点表現の例



特に
　i=31,f=0 の場合は　整数
逆に
　i=0, f=31の場合は　純小数

教科書P.40 は　4bit　でかつ
整数 or 純小数のみに言及

実数の表現(10進数)

- 固定小数点表現 （小数点の位置は絶対位置）
  - 日常的に接している実数表現
  - 整数部 (integer part) i桁と小数部(fraction part) f桁で表現

12345.6　　(i=5,f=1)
-9234.56　　(i=4,f=2)
5.23456　　(i=1,f=5)

- 浮動小数点表現 （小数点の位置は相対位置）
  $1.23456\text{x}10^4=12.3456\text{x}10^3=123.456\text{x}10^2$ $(=123456.0\text{x}10^{-1})$
  - 正規化表現　[$(-1)^S \times M \times B^E$形式] (科学記数法[有効桁数が重要])
  - 整数部1桁(0以外の数値)＋小数部k桁　(以下はk=5の例)

$1.23456\text{x}10^4$　　$-9.23456\text{x}10^3$　　$5.23456\text{x}10^{-0}$

この例では有効桁数は10進6桁