

第3回 固定小数点表現、浮動小数点表現、コンピュータ上の誤差

3-1. 小数の表現—固定小数点

10進数の小数は、 $23.45_{10} = 2 \times 10^1 + 3 \times 10^0 + 4 \times 10^{-1} + 5 \times 10^{-2}$

同様に2進数の小数は、 $101.1101_2 = 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 + 1 \times 2^{-1} + 1 \times 2^{-2} + 0 \times 2^{-3} + 1 \times 2^{-4}$

a) 2進小数 0.11_2 を10進に変換せよ (上記の展開式をそのまま計算すればよい)(途中結果を残せ)

b) 2進小数 110.1011_2 を10進に変換せよ (同上)(途中結果を残せ)

逆に、10進→2進の変換

やや面倒だが、大別して2つやり方が考えられます。1つは、2進の小数の $0.1_2=0.5_{10}$ 、 $0.01_2=0.25_{10}$ 、 $0.001_2=0.125_{10}$ などを覚えていて、その組合せで10進数を表そうとする方法です。もう1つは、まともに(整数の10進数を2進数に変換するときに2で続けて割っていったように)2を続けて掛けていく方法です。(詳細はスライドを見てください。)

a) $0.3125_{10} = (\quad)_2$ (途中結果を残せ)

b) $0.4375_{10} = (\quad)_2$ (途中結果を残せ)

c) $0.1_{10} = (\quad)_2$ (途中結果を残せ)

3-2. 小数の表現—浮動小数点

有効数字とその表現(科学的記数法)

精度で見て読み取れている(または計算できている)桁数を「有効数字」と呼びます。例はスライドを見てください。

「桁の位置の都合で0を書かなければならないが、その桁数分の精度はない」ということがあります。分かりやすい例は大きな数で、地球の赤道半径は6378kmですが、メートルを単位にして表すと6378000メートルではありません。ウィキペディアによると6378136.6mだそうです。手元で6378kmしか分からないときにメートルに直すと、桁の都合で6378000mと書かざるを得ません。これをきちっと(土の精度で分かっているか)表すために、 $6378 \times 10^3\text{m}$ と書くか、 $6.378 \times 10^6\text{m}$

と書きます。これによって、分かっているのは6.378の4桁で、桁合わせの部分が 10^6 であるということが明確になります。この場合、有効数字が6378の4桁である、もしくは科学的記数法で 6.378×10^6 mと言います。

この表記法を使ったのが浮動小数点表現で、科学技術計算などによく使われています。Javaではfloat型やdouble型の変数が浮動小数点表現を使っています。(固定小数点数はint形で表し、プログラマが自分で小数点の位置を管理します。)

a) 1.101×2^4 を、10進数に変換せよ (途中経過を残せ)

.....
.....
.....

b) 1.101×2^4 を仮数部(フラクションまたはマンティッサ)と指数部(エクスポネント)で表すと、どうなるか。ただし、指数部の下駄履かせや仮数部の正規化はしなくてよい。

仮数部 = (.....) 指数部 = (.....)

c) コンピュータ上では、浮動小数点表記として、単精度浮動小数と倍精度浮動小数が使われています。

単精度浮動小数は、符号ビットが()ビット、指数部()ビット、仮数部()ビットで合計()ビット、
倍精度浮動小数は、符号ビットが()ビット、指数部()ビット、仮数部()ビットで合計()ビット、
となっています。(ビット数はスライドを見てください。この数は覚えなくて結構ですが、「見たことがある」程度にはしておいてください。)

d) (暇な人用の選択問題)

指数部のビット幅が大きいと(.....)ができるようになり、

仮数部のビット幅が大きいと(.....)ができるようになります。

(指数部の数が正の大きい数の時、負の小さい数の時、仮数部が正の大きい数の時、負の小さい数の時などに、どういう数を表していることになるのか、自分で考えてみてください。)

3-3. コンピュータ上の誤差

4種類の誤差を理解しておいて下さい。名前を覚える必要は必ずしもありませんが、「こんなことがあるのだな」の感じでは覚えておいて欲しいです。実際のプログラムで結果が疑わしいときに、思い出してほしいのです。

a) 丸め誤差の例としてスライドでは 0.1_{10} の2進表現を挙げています。 0.1_{10} の2進変換で何が起るのか、その結果どうい誤差が起きるのか、説明してください。

.....
.....
.....

b) オーバーフローの例として、ここでは整数型で階乗の計算をする例を挙げておきます。(スライドの例も見ておいてください。) 以下は $n!$ (n の階乗= $1 \times 2 \times \dots \times (n-1) \times n$)の計算結果です。案外小さな n の値でも会場は大きな値になるので、「組み合わせの数」を計算するときなどは32ビットの整数(2 の31乗-1= 2147483647 まで表現可能)もあふれてしまいます。

次の図は左側が正しい値、右側が31ビット整数を使って計算した値です。12の階乗に13を掛けたところから値がおかしくなっていることが分かります。これが初めて $2^{31}-1$ を超える場所であることを確認してください。

正しい値
1! = 1
2! = 2
3! = 6
4! = 24
5! = 120
6! = 720
7! = 5040
8! = 40320
9! = 362880
10! = 3628800
11! = 39916800
12! = 479001600
13! = 6227020800
14! = 87178291200
15! = 1307674368000
16! = 20922789888000
17! = 355687428096000
18! = 6402373705728000
19! = 121645100408832000

31 ビット整数で計算
1! = 1
2! = 2
3! = 6
4! = 24
5! = 120
6! = 720
7! = 5040
8! = 40320
9! = 362880
10! = 3628800
11! = 39916800
12! = 479001600
13! = 1932053504 ←おかしい
14! = 1278945280
15! = 2004310016
16! = 2004189184
17! = -288522240
18! = -898433024
19! = 109641728

c) 情報落ちは、浮動小数点小数で足し算(or 引き算)をするときに桁の位置が違うときに起こります。次の例で説明してみてください。

わかりやすいように、10進で考えます。浮動小数点数が、仮数部が10進5桁、指数部は問題にならないので適当な桁数であるとします。

$x = 2.3456 \times 10^4$ 、 $y = 7.8912 \times 10^{-1}$ のとき、 $x + y$ を計算するとどうなるでしょうか？

d) 桁落ちは、似たような数を引き算すると起こります。次の例で説明してみてください。

地球の赤道半径は 6378.136.6km だそうです。赤道の表面ぴったりにはロープを1周させます(鉢巻です)。そのロープを10m 延長します。少し緩むはずですが。ロープの高さを均一にしたら地面との隙間はどれだけか、計算してください。

赤道1周 = $2 \times 3.14159265 \dots \times 6378136.6\text{m} = 40075014.17\text{m}$ 伸ばしたロープ長 = 赤道1周 + 10m = 40075024.17m

すきまの高さ = $(\text{伸ばしたロープ長} / 3.14 \dots / 2) - 6378136.6 = 6378138.2 - 6378136.6 = 1.59\text{m}$

さて、もし計算機が10進6桁分しか有効数字を保持できないとすると、計算はどうなるでしょうか？

赤道1周 = $400750 \times 10^2\text{m}$ 伸ばしたロープ長 = $400750 \times 10^2\text{m}$ しか保持していないと、同じ値になってしまうので、すきまの高さは0になってしまいます。

どうしてこのようなことがおこるのでしょうか？

