

## 例1の解法

例1の誤差は10進数  $x$  の  $\beta$  進法への変形式

$$x = f_1\beta^{-1} + f_2\beta^{-2} + f_3\beta^{-3} + \dots$$

により理解できる.

この式より32ビットの場合16進数では,

$$(0.1)_{10} = (0.199999)_{16} \times 16^0$$

となり, 16進法で表わすことができた(付録の計算1を参照).

これをまた10進法に戻すため,  $\beta$  進小数の10進変換の式

$$(0.f_1f_2\cdots f_m)_\beta = (f_1f_2\cdots f_m)_\beta \times \beta^{-m}$$

により

$$(0.199999)_{16} = (199999)_{16} \times 16^{-6}$$

計算すると0.099999964...となる.

## 例2の解法

$\sum_{n=1}^{10000} 0.01$  の計算において  $n$  項目までの部分和の大きさが  $0.01n$  であり,  $\varepsilon$  の誤差が毎回生じたとすると

$$\sum_{n=1}^{10000} 0.01n\varepsilon \cong 0.01 \times (10000^2\varepsilon)/2 = 5 \times 10^5\varepsilon$$

となってもおかしくない.

16進切り捨てのとき,  $\varepsilon = 6 \times 10^{-8} \sim 10^{-6}$  として, この値は0.03 ~ 0.5

2進0捨1入のとき,  $\varepsilon = 3 \times 10^{-8} \sim 6 \times 10^{-8}$  として, この値は0.015 ~ 0.03

となり誤差の説明ができる.

## 例3の解法

16進切り捨ての場合に  $0.1 = (0.199999)_{16} \times 16^0$  を10個足すと

$$(0.199999)_{16} \times 16^0 \times 10 = (0.FFFFA)_{16}$$

となり、1 より小さいので while の条件は満たされている。  
しかし、 $0.1 = (0.CCCCD)_{16} \times 2^{-3}$  を (毎回2進法で0捨1入しながら)10個足すと10進で約1.000000192となるのでこれは1よりわずかに大きく、while の条件は満たされなくなる。(付録の計算2を参照)

## 付録

### 計算1

まず  $0.1 = f_1 16^{-1} + f_2 16^{-2} + \dots$  この式に 16 をかけると

$$1.6 = f_1 + f_2 16^{-1} + f_3 16^{-2} + \dots$$

$f_i$  は 0 から  $\beta - 1$  までの整数, つまり  $f_i$  の最大は 15 であるので  $f_1$  より後ろの項の計は 1 以下なので  $f_1 = 1$  となり

$$f_2 16^{-1} + f_3 16^{-2} + \dots = 0.6$$

これをまた上記に従い, 16 をかけて同様の計算をすると

$$f_2 = f_3 = \dots = 9$$

したがって

$$(0.1)_{10} = (0.19999\dots)_{16}$$

2 進法も同様に計算すると

$$0.1 = (0.CCCCD)_{16} \times 2^{-3}$$

と表わすことができる.

### 計算2

$(0.1)_{10} = (0.19999\dots)_{16}$  より

$$\begin{array}{r}
 0.110011001100110011001101 \\
 +0.110011001100110011001101 \\
 \hline
 1.100110011001100110011010 \\
 +0.110011001100110011001101 \\
 \hline
 10.011001100110011001100111 \\
 +0.110011001100110011001101 \\
 \hline
 11.001100110011001100110100 \\
 +0.110011001100110011001101 \\
 \hline
 100.000000000000000000000001
 \end{array}$$

+0.110011001100110011001101  
100.110011001100110011001110  
+0.110011001100110011001101  
101.100110011001100110011011  
+0.110011001100110011001101  
110.011001100110011001101000  
+0.110011001100110011001101  
111.001100110011001100110101  
+0.110011001100110011001101  
1000.00000000000000000000010

これを 10 進に直すと, 1.000000192 となり while の条件が満たされなくなる.