

## Fortran 演習課題 4: 多倍長演算

通常の浮動小数点数(実数)の精度には限りがあり, 任意精度で近似値を求めたい場合には用いることが出来ず, 多倍長演算(または任意精度演算)と呼ばれる手法を用いることになる. ここでは多倍長演算が用いられる代表例である円周率 $\pi$ の計算を考えよう.

以下では最も簡単な実装を考える. まず任意の数値 $x$ を 10 進数で

$$x = (-1)^s \sum_{j=1}^N \tilde{x}_j \times 10^{-j+e}$$

の形で表そう. ここで $\tilde{x}_j \in \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$ は各桁の数値を,  $s \in \{0, 1\}$ は符号を,  $e \in \mathbb{Z}$ は値の絶対値を保持するための整数である. 従って項数 $N$ を十分大きく取り,  $\tilde{x}_j$ を配列として保持すれば, メモリの許す限り任意の精度(桁数)で数値を表すことが出来る. 例えば $x = 1.41$ を表すには $N \geq 3$ とし,  $\tilde{x}_1 = 1, \tilde{x}_2 = 4, \tilde{x}_3 = 1$  ( $\tilde{x}_4$ 以降は全て 0),  $s = 0, e = 1$  とすれば良い. なお以下の円周率の計算では $s = 0, e = 1$  の場合が扱えれば十分であるので常にこれを仮定しても良い. すなわち, 小数点以下 $N$ 桁までの数値は長さ $N$ の整数配列で表すことが出来る. (整数部まで含めれば長さは $N + 1$ である.)

ここでは円周率の計算に, Machin の公式および $\tan^{-1}(x)$ の Taylor 展開を用いることによって得られる以下の級数展開を用いよう.

$$\pi = \sum_{k=0}^{\infty} \frac{(-1)^k}{2k+1} (a_k - b_k) = (a_0 - b_0) - \frac{a_1 - b_1}{3} + \frac{a_2 - b_2}{5} - \frac{a_3 - b_3}{7} \dots$$

ただし

$$a_k \equiv 16 \left( \frac{1}{5} \right)^{2k+1}, \quad b_k \equiv 4 \left( \frac{1}{239} \right)^{2k+1}$$

と置いた. 簡単な評価から, 小数点以下  $l$  桁目まで(ただし  $l \gg 1$ ) を正しく求めるには  $k = \lceil l/2 \log_{10}(5) \rceil$  程度までで上記の級数を打ち切って良いことが分かる.

ここで,  $a_{k+1} = a_k/5^2$ ,  $b_{k+1} = b_k/239^2$ で与えられることに注意すれば,

- 多倍長数同士の加算(足し算)
- 多倍長数同士の減算(引き算)
- 多倍長数の整数(通常の integer)での除算(割り算)

が実装出来れば円周率の計算が出来ることが分かる. 筆算の要領でこれらの演算を実装し, 実際に円周率の多倍長計算を行うプログラムを作成せよ. 以下は小数点以下 100 桁までを求めた結果である.

```
$ ./a.out 100
PI = 3.1415926535 8979323846 2643383279 5028841971 6939937510
      5820974944 5923078164 0628620899 8628034825 3421170679
```

この例ではコマンドラインで求める桁数(この場合は 100)を与えている. このようにコマンドライン引数を利用するには, サブルーチン `get_command_argument` (Fortran 2003 以降では標準)を利用すると良い. ただし, 必ずしもこの例のように実装する必要は無く, 標準入力から項数を読み込むようにしても良い.