ファイル名　　MultiPrecision10.vb

モジュール名　　MultiPrecision10

（モジュール：fft4g_h と共に用いること）

説明

10^n を基数として多精度の計算を行なうモジュール。n はモジュールの Public な定数 RXdc で定義する。デフォルトは RXdc=4。基数は、同様に Public な定数 RX で定義される。デフォルトは RX=10000。このドキュメントで「桁」とは、基数 RX の「桁」と解釈すること。デフォルトでは、「1 桁」は 10 進数の 4 桁に相当する。RXdc=1、RX=10 の場合は「1 桁」と 10 進数の 1 桁が一致する。

「1 桁」を Short 型（16 bit 長）の整数に格納する。したがって、メモリーの利用効率は以下のとおりである。

RXdc=1、RX=10（3.3 bit）の場合	21%
RXdc=2、RX=100（6.6 bit）の場合	42%
RXdc=3、RX=1000（10.0 bit）の場合	62%
RXdc=4、RX=10000（13.3 bit）の場合	83%

MpMUL の乗算ルーチンで「N 桁」同士の掛け算を行なう場合、次に示す制限を目安とすること。Double 型浮動小数点を用いた高速フーリエ変換の精度に由来する。

RXdc=1、RX=10（3.3 bit）の場合　　$N < 2\times10^{12}$
（10 進数で同じ桁数）

RXdc=2、RX=100（6.6 bit）の場合　　$N < 2.5\times10^{10}$
（10 進数では 5×10^{10} 桁）

RXdc=3、RX=1000（10.0 bit）の場合　　$N < 3\times10^{8}$
（10 進数では 9×10^{8} 桁）

RXdc=4、RX=10000（13.3 bit）の場合　　$N < 4\times10^{6}$
（10 進数では 1.6×10^{7} 桁）

基数を小さくすると、演算可能な桁数は増えるが、演算速度は落ちる。デフォルトの RXdc=4、RX=10000 では、10 進数で 100 万桁の演算は可能である。1000 万桁の演算は事前に検証が必要と思われる。1 億桁の演算は危険であり、RXdc=3、RX=1000 を推奨する。

10 進数演算に基づくモジュールなので、基数変換のためのルーチンは提供されない。

メソッド名	MpADD
引数	ByVal u() As Short, ByVal pu As Integer, ByVal v() As Short, ByVal pv As Integer, ByVal m As Integer, ByVal w() As Short
説明	u(pu)から始まるm桁の符号なしの整数と、v(pv)から始まるm桁の符号なしの整数を加算し、結果をw(0)から始まるm+1桁に格納する。 u(pu)とv(pv)が最上位桁で、w(0)にはさらに桁上がりしたデータが格納される。u(pu+m-1)、v(pv+m-1)及びw(m)が最下位桁で同じ位を示す。 u()、v()、w()が同じ配列を参照してもよい。演算結果は保証される。

メソッド名	MpSUB
引数	ByVal u() As Short, ByVal pu As Integer, ByVal v() As Short, ByVal pv As Integer, ByVal m As Integer, ByVal w() As Short, ByRef isign As Integer
説明	u(pu)から始まる m 桁の符号なしの整数から、v(pv)から始まる m 桁の符号なしの整数を減算し、結果を w(0)から始まる m 桁に格納する。 u(pu)、v(pv)及び w(0)が最上位桁、u(pu+m-1)、v(pv+m-1)及び w(m-1)が最下位桁で、それぞれ同じ位を示す。 v()> u()の場合は、isign が-1 で返り、w()には 2 の補数表現の負の結果が格納される。それ以外は isign は 0 である。 u()、v()、w()が同じ配列を参照してもよい。演算結果は保証される。

メソッド名	MpNeg
引数	ByVal u() As Short, ByVal pu As Integer, ByVal m As Integer
説明	u(pu)から始まるm桁の符号なしの整数を、2の補数表現の負数に置き換える。u(pu)が最上位桁、u(pu+m-1)が最下位桁である。

メソッド名　　MpSAD

引数　　ByVal u() As Short, ByVal pu As Integer,
ByVal m As Integer,
ByVal iv As Integer,
ByVal w() As Short

説明　　u(pu)から始まる m 桁の符号なしの整数に、iv の 0 以上 RX 以下の整数を加算し、結果を w(0)から始まる m+1 桁に格納する。

u(pu)が最上位桁で、w(0)にはさらに桁上がりしたデータが格納される。
u(pu+m-1)が最下位桁で、ここに iv が最初に加算される。

u()、w()が同じ配列を参照してもよい。演算結果は保証される。

メソッド名	MpSMU
引数	ByVal u() As Short, ByVal pu As Integer, ByVal m As Integer, ByVal iv As Integer, ByVal w() As Short
説明	u(pu)から始まる m 桁の符号なしの整数に、iv の 0 以上 RX 以下の整数を乗じ、結果を w(0)から始まる m+1 桁に格納する。 u(pu)が最上位桁で、w(0)にはさらに桁上がりしたデータが格納される。 u(pu+m-1)と w(m)が最下位桁で同じ位を示す。 u()、w()が同じ配列を参照してもよい。演算結果は保証される。

メソッド名	MpSDV
引数	ByVal u() As Short, ByVal pu As Integer, ByVal m As Integer, ByVal iv As Integer, ByVal q() As Short, ByRef ir As Integer
説明	u(pu)から始まる m 桁の符号なしの整数を、iv の 0 以上 RX 以下の整数で除し、q(0)から始まる m 桁に商を、ir に剰余を格納する。 u(pu)と q(0)が最上位桁、u(pu+m-1)と q(m-1)が最下位桁で、それぞれ同じ位を示す。 u()、q()が同じ配列を参照してもよい。演算結果は保証される。

メソッド名　　MpMUL

引数
```
ByVal u() As Short, ByVal pu As Integer,
ByVal m As Integer,
ByVal v() As Short, ByVal pv As Integer,
ByVal n As Integer,
ByVal w() As Short
```

説明

u(pu)から始まる m 桁の符号なしの整数と、v(pv)から始まる n 桁の符号なしの整数を乗じ、結果を w(0)から始まる m+n 桁に格納する。

u(pu)、v(pv)、w(0)が最上位桁だが、一般に同じ位ではない。u(pu+m-1)、v(pv+n-1)と w(m+n-1)が最下位桁で同じ位を示す。

u()、v()、w()が同じ配列を参照してもよい。演算結果は保証される。

フーリエ変換、逆フーリエ変換のコンボリューション演算を利用しており、N 桁同士の乗算であっても、$O(N^2)$オーダーではなく、
$O(N \times \log N \times \log\log N)$オーダーの演算である。

メソッド名　　MpINV

引数　　ByVal u() As Short, ByVal pu As Integer,
ByVal m As Integer,
ByVal v() As Short

説明　　u(pu)から始まるm桁のデータを、最初の1桁u(pu)（ゼロでないこと）を整数部、残りのu(pu+1)からu(pu+m-1)までを小数部とみなして、逆数をv()に格納する。

v()は、v(0)を整数部、残り全てが小数部である。このメソッドを呼び出すときのv()の桁数で計算精度が決まる。u()の表すデータが厳密に1でない限り、v()は1未満の結果となり、v(0)はゼロである。

u()、v()が同じ配列を参照してもよい。演算結果は保証される。

数学　　Uに対して、漸化式

$$V_{i+1} = V_i(2 - UV_i)$$

で示されるV_iは、$i \to \infty$のとき$V_i \to 1/U$である。

メソッド名　　MpSQRT

引数　　ByVal u() As Short, ByVal pu As Integer,
ByVal m As Integer,
ByVal v() As Short, ByVal w() As Short

説明　　u(pu)から始まるm桁のデータを、最初の1桁u(pu)（ゼロでないこと）を整数部、残りのu(pu+1)からu(pu+m-1)までを小数部とみなして、平方根をv()に、平方根の逆数をw()に格納する。

v()は、v(0)を整数部、残り全てが小数部である。このメソッドを呼び出すときのv()の桁数で計算精度が決まる。w()についても同様。

u()とv()、あるいはu()とw()が同じ配列を参照してもよい。演算結果は保証される。v()とw()が同じ配列の場合は、どちらも平方が格納される。

数学　　Uに対して、漸化式

$$W_{i+1} = \frac{1}{2} W_i (3 - U W_i^2)$$

で示されるW_iは、$i \to \infty$のときUの平方根の逆数に漸近する。

$$W_i \to \frac{1}{\sqrt{U}}$$

収束した結果にUを乗じれば平方根Vを得る。

$$V = W \times U = \frac{1}{\sqrt{U}} \times U = \sqrt{U}$$

メソッド名　　MpSQRT（オーバーロード）

引数　　ByVal u() As Short, ByVal pu As Integer,
ByVal m As Integer,
ByVal v() As Short, ByVal w() As Short,
ByVal wi() As Short

説明　　前述の MpSQRT に比べて、引数に wi が追加となっている。平方根の逆数について予めある程度収束した値が用意できる場合に、漸化式の Wi の初期値として wi を用いることにより、計算が速くなる。

メソッド名	`MpDIV`
引数	`ByVal u() As Short, ByVal v() As Short,` `ByRef q() As Short, ByRef r() As Short`
説明	`u(0)`から始まる `u.length` 桁の符号なしの整数を、`v(0)`から始まる `v.length` 桁の符号なしの整数で除し、商を `q(0)`から始まる `u.length-v.length+1` 桁に、剰余を `r(0)`から始まる `v.length` 桁に格納する。 `u.length` は `v.length` 以上でなければならない。
数学	まず V の逆数 $1/V$ を `MpINV` で求め、U と $1/V$ を乗じて商 Q を求める。 剰余 R は U から VQ を差し引いて求める。