

Windows 環境で動作する BASIC 言語システム

白 石 和 夫*

A BASIC Language System which works on MS-Windows

Kazuo Shiraishi

(Faculty of Education, Bunkyo University)

要 約

数学教育で利用することを目的として、MS-Windows 上で動作する JIS Full BASIC 準拠の BASIC 処理系を作成した。Windows の GUI を活用することで操作性に優れたものとなっている。この処理系は、学習者が道具として利用することを目的としたものであり、アプリケーション作成を目的としたものではない。数値計算・グラフィックスとアルゴリズムの表現が想定される主な利用法である。JIS に従い、十進数表現、問題座標系に基づくグラフィックス、構造化プログラミングなどを実現している。問題座標系の採用で関数のグラフを描くのは容易であり、さらに、図形変換の機能の実現により、変換幾何への応用も可能になった。

なお、このシステムはボーランド社の Delphi を用いて作成したものである。

1. はじめに

前報 [1] でインタプリタ形式による BASIC 処理系の試作結果を報告した。今般、内部構造をコンパイラに改めるとともに、日本工業規格 (JIS Full BASIC [2]) との整合性を高め、さらに、MS-Windows 上で動作するものが完成したので報告する。

2. 概 要

試作した BASIC 処理系は数学教育での利用を目的としたもので、数値計算、グラフィックスの利用と、アルゴリズムの記述のために用いることを想定している。したがって、マイクロソフト社の Visual BASIC のようにアプリケーション開発を目的とした言語と

は性格が異なる。

試作した BASIC 処理系には、MS-Windows 版と、PC 9801 MS-DOS 版がある。Windows 版は、ボーランド社のプログラミング言語 Delphi 1.0 で作成されている。一方、PC 9801 版は、同じくボーランド社の turbo Pascal 6.0 で作成されたものである。

以下の記述は Windows 版に基づく。PC 9801 版では、多少、異なる部分もある。

3. 文法および機能

この BASIC は、JIS Full BASIC を目標に作成されている。以下、JIS との整合性を中心に、実現された機能を述べる。

(1) 数値演算と数値変数

数値に関する部分については、ほぼ完全に JIS の要求を満たしている。

JIS の数値の関する要求は、基本的には十

* しらいし かずお 文教大学教育学部

進演算であるが、実際には、より複雑な処理を要求される。たとえば、数値演算の結果を数値変数の精度よりもさらに高い精度で表現することが要求されている。なお、計算結果は十進数である必要はないが、十進数が正確に表現できるものでなければならない。

数値の内部表現は、少なくとも 21 桁までの十進数が正確に表現できるだけの桁数をもつ 10000 を基底とする数表現である。数値変数も同様の数表現を用いるが、計算結果を代入するとき、10 進数としてみたときの 16 桁目で四捨五入を行うことで、見かけ上、15 桁の精度をもつ十進数にしている。

JIS の数値に関する規定のうち、numeric 宣言には対応していない。それ以外の数値に関する文法には完全に対応している。ただし、arithmetic 選択子で NATIVE を選択しても DECIMAL を指定した場合と同じになる。

(2) 数値組込み関数

数値組込み関数は、JIS に規定されたものをすべて用意している。計算結果の正確さについてもほぼ JIS の規定通りである。たとえば、角の大きさの単位として度を選択したとき、 $\sin(180)$ は正確に 0 であり、 $\tan(90)$ はエラーになる。また、べき乗演算は、一般的な BASIC 処理系では底が 1 に近い数であるときに大きな誤差を持つことが多いのであるが、そのような場合にも正しい結果が得られるように考慮してある。

(3) 文字列

文字列については以下の通り。文字列の連結演算は実現している。文字列式のなかと、let 文の左辺では、文字列変数に部分文字列指定が使える。組込み関数については、DATE\$, LEN, POS, REPEAT\$, STR\$, TIME\$, USING\$, VAL を実現している。

(4) 配列

数値配列についてはほぼ JIS の規定通りの内容を実現し、行列演算が可能である。ま

た、mat-line-input 文を除き、配列入出力も可能である。しかし、文字列配列については、最小限の機能しか実現していない。

(5) 制御構造とプログラム分割

論理式の文法は JIS の規定通りに実現している。したがって、

```
IF 1<A<4 THEN
```

は文法誤りである。また、配列 M の添字の上限が 10 で、変数 A の値が 11 であるとき

```
IF A<=10 AND M(A)>0 THEN
```

を実行してもエラーにならない。

JIS Full BASIC の特徴は構造化プログラミングが可能なことであるが、この処理系では、JIS の構造化制御文をすべて実現している。実現されていないのは、構造化プログラミングによって不要となり、将来、JIS から削除することが予定されている on 文、gosub 文、return 文のみである。ただし、行番号分岐の文には飛び先について若干の制約がある。

プログラム分割の機能については、関数定義や副プログラムを、外部、内部ともに実現している。

(6) 入出力

入出力については、JIS の要求に完全には対応できていない。そのうち大きいものは、書式指定であり、今のところ、書式指定は数値式に対してのみ可能であり、しかも、書式文字として使えるのは #, %, ^, +, - のみである。

(7) ファイル

テキストファイルの操作のために必要な最小限の機能を提供している。しかし、経路を手続きの引数にすることは実現できていない。

(8) 例外状態処理およびデバッグ

例外状態処理のために、when-in 区を利用することができる。例外処理区では、例外処理区戻り文が利用できる。これによって、定義域がよくわからない関数や、桁あふれになる可能性のある関数のグラフなどが容易に

描けるようになる。

デバッグ機能については、debug 文と、break 文が利用できる。

(9) 図形 (graphics)

図形機能については、座標系の設定と、描画と座標の取得のための機能、および、絵定義と図形の変換の機能を実現している。描画領域の形状は、JIS の約束に従い、正方形である。そのため、座標系の設定を考えたときの負担が少なく、また、機種による依存が生じない。問題座標系を原則とする座標系により、関数のグラフを描く、媒介変数表示の曲線を描くなどの処理が容易にできる。また、組込み関数 $ANGLE(x,y)$ を利用することで極座標を扱うのも容易である。一方、座標値取得のための機能 (get point) があるから、描いた図形から座標値を読み取る、試行錯誤で軌跡を探る、グラフィックスのアルゴリズムを考える、などの応用が可能になる。また、図形変換の機能を実現したので、変換幾何学や自己相似図形への応用も可能である。

以下に、実現された機能について、もう少し、くわしく述べる。

「13.1 座標系」では、SET WINDOW と、ASK WINDOW を実現している。

「13.2 属性及び画面制御」では、clear 文と、次の各項目に対する set 文および ask 文を実現している：line color, line style, point color, point style, area color, text color. また、次の対象に対する set 文を実現している：text justify, color mix. その他、いくつかの ask 文が実現されているが、詳細は省く。

「13.3 図形出力」では、図形動詞として PLOT を使う前提で、主要な機能を実現されている。また、「13.4 図形入力」では、get point と、pixel size に関する ask 文を実現している。「13.5 絵及び描点出力」では、絵定義と draw 文、exit-picture 文が利用できる。draw 文では、図形の変形が可能

である。

(10) 文法の拡張

JIS との整合性を壊さない範囲で文法を拡張している。

そのひとつは、行番号の省略を許すことである。また、外部手続きに対する declare 文の省略も許している。ただし、JIS との互換性保持のために、組込み関数と同名の外部関数を定義する場合には declare 文を省略できない。

(11) 拡張機能

i) 249 桁モード

249 桁以下の十進小数が正確に表現できる、基底が 10000 の数値表現を選択できる。

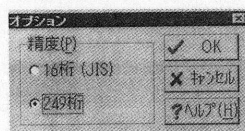


図 1

ただし、このモードでは、超越関数は使えず、べき乗演算は指数が整数でないと実行時にエラーになる。

ii) 組込み関数

組込み関数として次の拡張関数がある。

FACT(x) x の階乗

PERM(n,r) 順列の数

COMB(n,r) 二項係数 (組合せの数)

iii) 組込み絵定義

組込みの絵定義に grid と axes がある。絵定義 grid を利用した例を次に示す。

```
DEF f(x)=x^3-3*x
SET WINDOW -5,5,-5,5
DRAW grid
FOR x = -5 TO 5 STEP 0.01
    PLOT LINES: x,f(x);
NEXT x
END
```

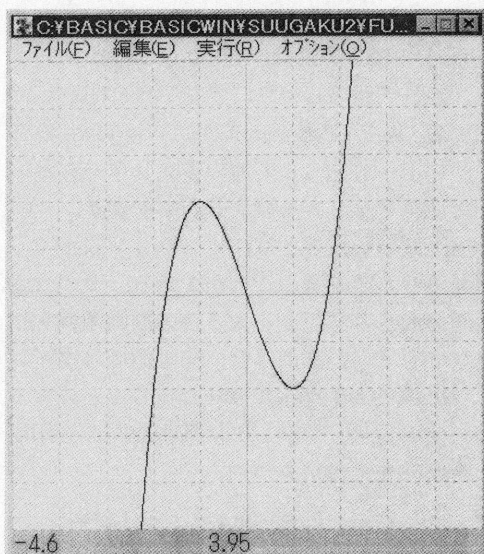


図 2

4. ユーザーインターフェース

(1) プログラムの入力

プログラムは、Windows のメモ帳と互換性のあるエディタで入力する。行番号は省略可能である。

プログラムの読み込み、保存などはメニュー選択で実行できる。保存形式は MS-DOS の標準テキストである。

エディタには、カット＆ペーストによる編集機能のほか、検索や置換などの機能もある。

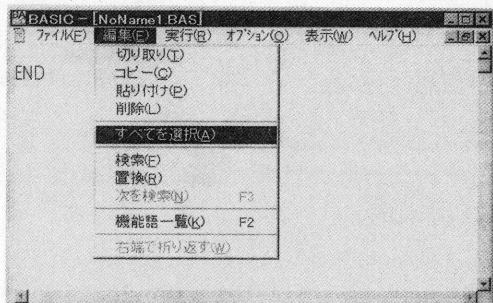


図 3

(2) プログラムの実行

プログラムの実行は、プルダウンメニューから「実行」を選択して行う。ショートカッ

トキーとして F9 を割り当ててあるので、F9 キーを押してもよい。

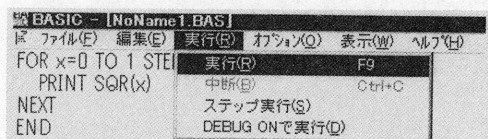


図 4

標準の状態では、プログラムを翻訳しながらプログラムのインデントを実行する。インデントなしで入力されたプログラムも、適切なインデントを行うことで、区構造の誤りを発見しやすくなる。なお、オプションとして、自動インデントしないことを選択することもできる。

プログラムに構文上の誤りがあると、誤りのある行を反転表示して誤りを報告する。単純な誤りの場合には、コンパイラに誤りを訂正させることもできる。

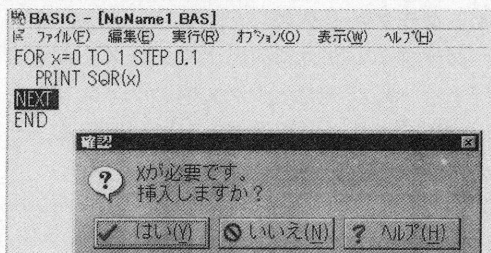


図 5

現在、誤りの自動修正の対象となっているのは、マイクロソフト BASIC の文法であり、自動修正するのは、代入文、next 文、input 文のプロンプト、print using 文、行末注釈である。

(3) print 文の出力

print 文の結果は、プログラムエディタとほぼ同等の機能をもつテキストエディタに出力される。行数の制限はないが、文字数におよそ 30 Kbyte までの制限がある。

出力結果を印刷する、ファイルに保存する、windows のカット・パッファを経由して他のアプリケーションにデータを渡すなどの操作も可能である。

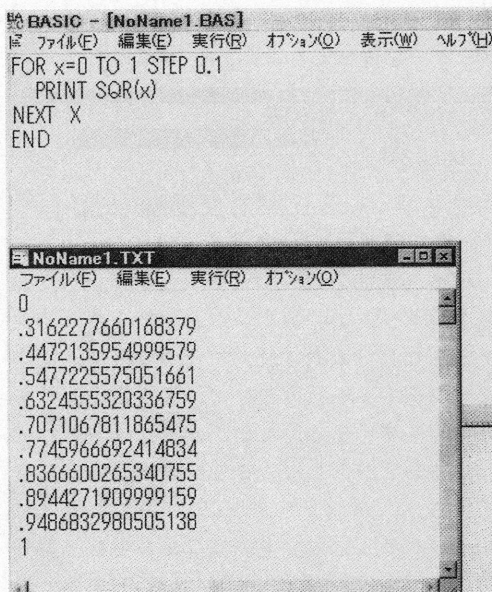


図 6

(4) グラフィックス

グラフィックスを利用するプログラムを実行すると、グラフィックス用のウィンドウを作成し、そこを描画領域として利用する。その形状は正方形である。

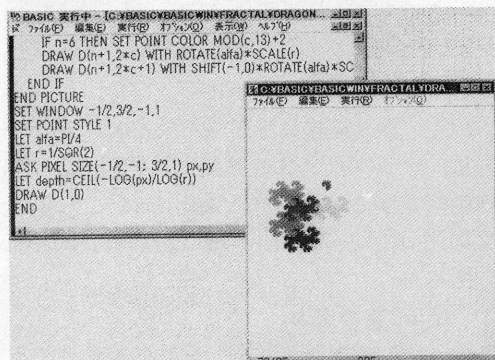


図 7

グラフィックス・ウィンドウの下部には、マウスが指す点の座標が表示される。これは、たとえば、関数のグラフを描いたようなときに頂点の座標を読み取るのに役に立つ。また、get point 文で正確な点を入力するための補

助としても機能する。

図形出力の結果は、Windows のビットマップとして保存したり、プリンタに印刷したりすることができる。また、Windows のクリップボードを介して他のアプリケーションにグラフィックスを渡すこともできる。

描画領域の大きさは、オプション設定で、異なるサイズを選ぶことができる。そのとき、画面の大きさを超える大きさの描画領域を指定してもよい。その場合、描画領域はウィンドウの中でスクロールさせて見ることになる。この使い方は、プリンタに精密な図形を印刷したいときに役に立つ。

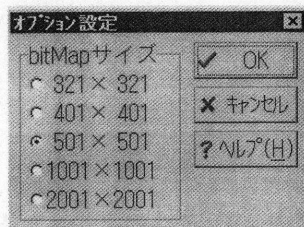


図 8

(5) 実行の中断とデバッグ

ファイルメニューから「中断」を選択すると、実行途中のプログラムの実行を中断することができる。

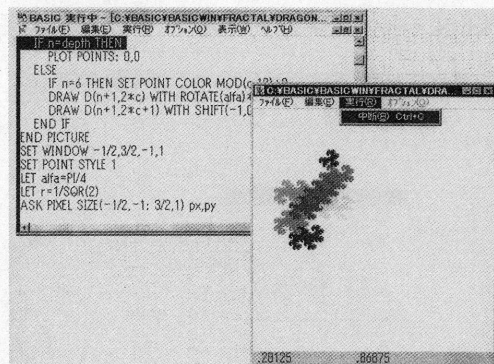


図 9

プログラムの実行が中断すると、デバッグのためのウィンドウが出現する。

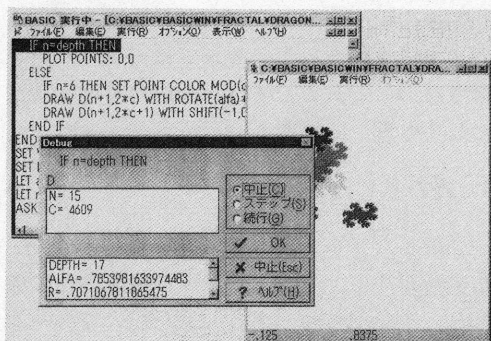


図 10

デバッグ・ウィンドウでは、次に実行される行と、現在実行中のプログラム単位と内部手続きに属するすべての変数の値が表示される。また、プログラム・ウィンドウでは、次に実行される行が反転状態で表示される。そして、実行の継続、ステップ実行、実行の中止のうちから選択することができる。

また、debug on の状態で break 文を実行しても同じ状態になる。ここで、ステップ実行を選択すれば、プログラムの流れを追跡することができる。

そして、デバッグ・ウィンドウは例外(実行時エラー)のためにプログラムが異常終了したときにも表示される。

(6) ヘルプ

この処理系には、システムと密に結合した Windows ヘルプを用意している。たとえば、意味を調べたい機能語をマウスで選択してヘルプメニューの「検索」を選ぶと、ヘルプファイルの該当ページが表示される。

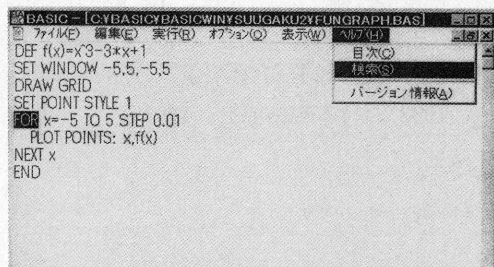


図 11

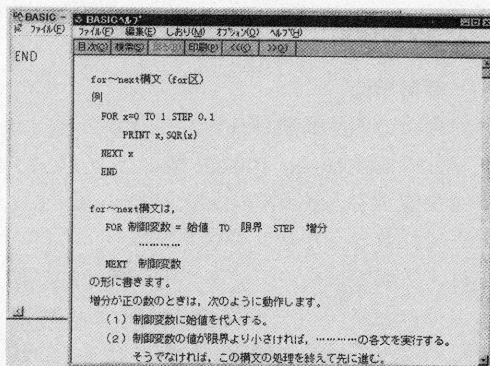


図 12

(7) 入力補助

Full BASIC の機能語は、たとえば、OPTION ANGLE DEGREES がそうであるように、長い綴りのものが多い。意味が容易に推測できる利点と裏腹に、キーボードに慣れないと入力に面倒になる欠点がある。その欠点を補うために、機能語を一覧から選択して入力する機能を用意している。

編集メニューから「機能語一覧」を選択すると、次図のようなウィンドウが表示される。

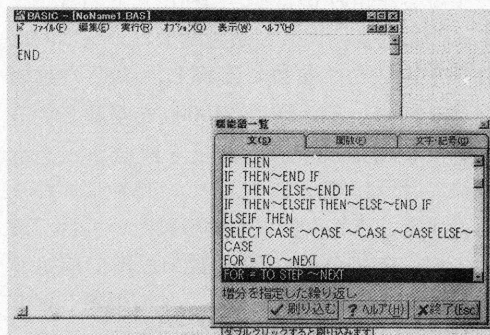


図 13

このウィンドウの下部には、選択した機能語の簡単な説明が表示される。ここで、「ヘルプ」ボタンをクリックすると、ヘルプファイルのうちの該当ページが表示される。

一方、目的の機能語のところでダブルクリックするか、「刷り込む」のボタンをクリックすれば、それがエディタに刷り込まれる。

5. 内部構造

(1) 制御構造

処理系の内部構造はコンパイラである。コンパイラは、turbo Pascal 型のオブジェクトを生成する。各オブジェクトはポインタによって結合されたリンク構造を持つ。実行時には、生成されたオブジェクトが自律的に動く。そのとき、バーチャル メソッドの呼び出しによって目的の機能が特定される。これは、機械語レベルで見ると、ポインタをたどりながら動いていく感じになる。また、一行実行するごとに中断が要求されたかどうか調べている。したがって、最適化された機械語命令を生成するコンパイラに比べると実行速度は遅い。

(2) 演算ルーチン

10000 を基底とする(十進)演算ルーチンは、8086 CPU の 16 ビット整数の加減乗除演算をもとにして組込みアセンブラで記述している。Pascal の枠組みをアセンブラで書かれた手続きを組み立てるのに利用している。

べき乗演算、および超越関数は、基本的には、十進数を 2 進数に変換し、8087 の数値演算を利用して計算し、10000 を基底とする数表現に戻した後、18 桁目を四捨五入する方法で計算している。しかし、単純にそれを実行すると正しい結果の得られない場合があるので、あらかじめ 10 進数の段階で前処理をしている場合などがある。加減乗除と平方根は、10000 を基底とする数値表現で計算している。そのため、得られる結果の精度は、十進数とみなしたとき 21 桁から 24 桁の間のいずれかになる。

(3) 組込み関数

JIS に規定される組込み関数のうち eps 関数については、JIS の規定を丁寧に読むと中

間結果の精度にもとづいて値を決定すべきものようにも思えるが、それでは実用上、用をなさないで、ここでは、変数の精度を基準にして処理している。

6. 終わりに

完成したプログラムファイルを効率よく配布するために、インターネット接続会社に加入し、インターネット上に WWW のホームページを開設した。URL は、

<http://pweb.in.aix.or.jp/~shiraishi/>
である。次に示すのは、その一部である。

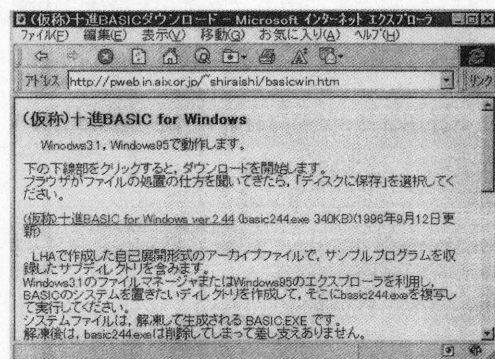


図 14

インターネット利用によって、障害が発見された場合にも迅速に対応することが可能になった。

文 献

- [1] 白石和夫：数学教育で用いるための BASIC 処理系の試作，文教大学教育学部紀要，第 27 巻，1993，pp.43-51
- [2] 日本工業標準調査会：電子計算機プログラム言語 Full BASIC (JIS X 3003-1993)，日本規格協会，1993