情報処理 2 第 10 回

十進BASIC (6)

桂田 祐史 (かつらだ まさし)

2007 年 6 月 26 日

この授業用の WWW ページは http://www.math.meiji.ac.jp/~mk/syori2-2007/

「プログラミングの課題、良く分からない」という声があったので、今日は説明を補足します。結局、シラバスに載せてあったテーマの一つ数値線形代数はカットすることにします。

1 連絡事項

- なるべくじっくり取り組んで欲しいので、課題 8[1] の締切は、最終回の前日である 7 月 16 日 (月曜) にします。TEX を使ってレポートを書き、PDF ファイルを添付して送って下さい。もちろん、プログラムや実行結果 (図) を含めて下さい。

せっかく TEX を学んだのに、全然それを生かしていない、こちらの意図を汲んでいないレポートがあります。結果だけでは、一体何をやったのかすら、こちらは判断できません。例えば課題 8 では、何を選んで描いたのか (図形や写像)、またどのようにプログラムを書いたか (書き直したか)、きちんと説明を書いて下さい。

2 プログラミング自習の勧め

プログラミングについて、すべての学生は、ある程度の体験をするべきだと考えています。しかし、大人数の授業ではなかなか効果が上がりにくい (自発的に質問して学ぶ人だけ前に進める、ということになりやすい)、と痛感しています。学び始めのうちは、語学 (例えば英会話) の習得に似たところがあって、少人数の授業、集中した練習 (授業時間外の練習) が効果的です。

まずは簡単なサンプル・プログラムを自分の手で打ち込んで、走らせてみて、なぜそれでそう動くのか考え、場合によっては色々とプログラムを書き直して試すのが良いです。そもそも十進 BASIC を選択したのは、「数学向きである」という理由の他に、その気になれば君達学

[1] http://www.math.meiji.ac.jp/~mk/syori2-2007/jouhousyori2-2007-09/node12.html

生が自宅のパソコンにも簡単にインストールできる、と考えたからです (そういう意味で、もう一度『(仮称) 十進 BASIC インストール』[2] をプッシュしておきます)。

十進 BASIC の場合、ハードコピーを配布した付属のチュートリアル『(仮称) 十進 BASIC による JIS Full BASIC 入門』[3] に載っているプログラムを一つ一つ試すことから始めたらどうでしょうか。なお、プログラムを電子的に貼付けることもできますが、なるべく自分の指を使って入力することを勧めます。馬鹿馬鹿しいと思うかもしれませんが、例えば言葉を学ぶのに、何回も声に出して読むのと同じだと考えて下さい。

語学の勉強だと、個々の単語を辞書で調べるように、数学の勉強では、出て来る言葉の定義や定理を調べるように、必要に応じて、プログラミング言語の文法や命令を調べることは大事です。幸い、十進 BASIC にはオンライン・ヘルプがあって、少ない手間で調べられるようになっています。ぜひ自分で調べる習慣をつけて下さい。

3 前回の小訂正

サンプルプログラム complexmap.bas[4] で、

```
FOR t=0 to 1 STEP 0.01
  ...
NEXT t
```

としましたが、t=1 まで実行されずに[5]線がつながらないことが多いです。10 進法の 0.01 は 2 進法では無限小数になるので (2 進法で有限小数となるのは、既約分数の形に表わしたとき、分母が 2 の冪乗である有理数です)、有限桁の計算では誤差が生じてしまいます。「十進 BASIC」という名前がついていますが、OPTION ARITHMETIC COMPLEX とした場合は、実部虚部を 2 進法で計算するようです。うっかりしていました。

ここは 2 進法で切りの良いように

```
FOR t=0 to 1 STEP 1/128
  ...
NEXT t
```

のようにすべきだったかも知れません ($\frac{1}{128} = 2^{-7}$ に注意)。こうするときちんと閉じるようです。

あるいは、少し面倒になりますが、

[2] http://hp.vector.co.jp/authors/VA008683/setup.htm
[3] http://hp.vector.co.jp/authors/VA008683/tutorial/contents.htm
[4] http://www.math.meiji.ac.jp/~mk/syori2-2007/jouhousyori2-2007-09/node11.html
[5] 試しに、ループの中で PRINT t として、値を表示してみると良いでしょう。

```
N=100
dt=1/N
FOR j=0 to N
  t=j*dt
  ...
NEXT j
```

と整数の変数で FOR NEXT ループを回す手もあります (本当はこちらの方が、プログラムの書き方として「普通」です)。

4　課題 8[6] について

4.1　サンプルプログラム complexmap.bas の追加説明

complexmap.bas

```
 1  REM 関数 w=z^3 で三角形がどういう図形に写像されるか
 2  OPTION ARITHMETIC complex
 3  DECLARE EXTERNAL SUB segment
 4  LET w=3
 5  SET WINDOW -w,w,-w,w
 6  DRAW grid(0.5,0.5)
 7  PRINT "複素平面上の三角形を関数 w=z^3 で写す"
 8  PRINT "3 点の座標を入力してください。"
 9  CALL cinput(z1)
10  CALL cinput(z2)
11  CALL cinput(z3)
12  CALL segment(z1,z2,2)
13  CALL segment(z2,z3,3)
14  CALL segment(z3,z1,4)
15  END
16
17  REM 線分と線分の像
18  EXTERNAL SUB segment(z1,z2,col)
19  OPTION ARITHMETIC complex
20  LET  i=SQR(-1)
21  SET LINE COLOR col
22  REM 線分を描く
23  SET LINE width 3
24  PLOT LINES
25  PLOT LINES : re(z1),im(z1);re(z2),im(z2)
26  REM 線分上の点の写像による像を描く
27  SET LINE width 2
28  FOR t=0 TO 1 STEP 1/128
29     LET  z=(1-t)*z1+t*z2
30     CALL MAPPLOT(z)
31  NEXT t
32  END SUB
33
34  REM 写像 w=f(z) で写した点を PLOT する
35  EXTERNAL SUB mapplot(z)
36  OPTION ARITHMETIC COMPLEX
37  DEF f(z)=z^3
38  LET  w=f(z)
39  PLOT LINES: re(w),im(w);
40  END SUB
41
42  REM 複素数の入力
43  EXTERNAL SUB cinput(z)
44  OPTION ARITHMETIC COMPLEX
45  INPUT PROMPT "実部虚部を入力: ": x, y
46  LET  I=SQR(-1)
47  LET  z=x+I*y
48  END SUB
```

[6] http://www.math.meiji.ac.jp/~mk/syori2-2007/jouhousyori2-2007-09/node12.html

このプログラムについては、前回口頭で説明しましたが、繰り返しておきます。

- 全体が 4 つの部分からなる。

  1. 1 ~ 15 行の主プログラム
  2. 17 ~ 33 行のサブルーチン segment
  3. 35 ~ 41 行のサブルーチン mapplot
  4. 43 ~ 49 行のサブルーチン cinput

- サブルーチン cinput は `call cinput(z)` のように「呼び出す」。そうすると、画面に「実部虚部を入力:」と表示し、キーボードから入力された二つの実数をそれぞれ実部虚部とする複素数を作り、変数 z に代入して戻って来る。

- サブルーチン segment は `call segment(a,b,c)` のように呼び出す。ここで a, b は複素平面上の点を表わす複素数で、c は色の番号を表わす整数である。次の二つのことを行う。

  1. a, b を端点とする線分 $S = \mathrm{ab}$ を描く。
  2. S を関数 f $(f(z) = z^3)$ で写した $f(S) = \{f(z); z \in S\}$ を描く。

- S は真っ直ぐな線分なので、命令 PLOT LINES 一発で描けるが、$f(S)$ は一般には「曲がった曲線」なので、次のようにして描いている。S は、$S = \{(1-t)\mathrm{a} + t\mathrm{b}; t \in [0,1]\}$ であるから、

  $$0 = t_0 < t_1 < \cdots < t_n = 1$$

  となる数列 $\{t_j\}_{j=0}^n$ を用意すれば、$f(S)$ 上の点列 $\{f((1-t_j)\mathrm{a} + t_j\mathrm{b})\}$ が得られる。これを順に結んだ折れ線を描く。

- z が与えられたとき、$f(\mathrm{z})$ を PLOT LINES するのがサブルーチン mapplot の仕事である。

4.2 (i) について

(プログラミングが面倒？でもなるべくやって欲しい。)

サンプルプログラムは、三角形という、線分から構成されるものを扱っているため、プログラムが特に単純になっています (十分複雑だという声が聞こえてきそうだけど...)。2 点 a, b を端点とする線分は

$$\varphi(t) = (1-t)\mathrm{a} + t\mathrm{b} \quad (t \in [0,1])$$

というパラメーター表示を持ちますが、上のプログラムでは (もっと具体的に言うと、サブルーチン segment では)、これを線分を描くためには使っていません。線分ではない曲線を描く場合は、このところを「パラメーター曲線の描き方」で置き換える必要があります。

十進 BASIC の学び始めに、(パラメーター) 曲線を描くプログラムを見たことがありました。チュートリアルの例 17 あたり[7]。まず、その描き方を良く理解して下さい。

[7] http://www.math.meiji.ac.jp/~mk/syori2-2007/jouhousyori2-2007-03/node4.html

問　複素数 c, 正の実数 r に対して、c を中心とする半径 r の円は $z = c + re^{2\pi i\theta}$ $(\theta \in [0,1])$ と表わすことができる。このことを使って、円を描くプログラムを作れ。

この問については、プログラム例 (circlez1.BAS[8], circlez2.BAS[9]) を用意しておきますが、まずは自分で考えてみること。

4.3　(ii) について

(プログラミングは簡単かも。きちんとした図を出すのはそれなりに試行錯誤と頭の利用が必要？)

(ii) も事前に調査しておくのが良いですが、要点だけばらすと「円または直線は円または直線にうつる」となります。しかし、それをきちんと図にするのは、結構試行錯誤が必要だと思います (1 次分数変換や、うつす図形、描く範囲の選択)。

4.4　(iii) について

(iii) は事前に教科書や参考書を調べて見当をつけておけば、案外簡単だと思います。「平行直線群」と書きましたが、実際は「平行線分群」を描くしかありませんから、サンプル・プログラムとあまり方法が変りません。

[8] http://www.math.meiji.ac.jp/~mk/syori2-2007/circlez1.BAS
[9] http://www.math.meiji.ac.jp/~mk/syori2-2007/circlez2.BAS