

情報処理2 第4回

# 十進BASIC (1) 入門

かつらだ まさし  
桂田 祐史

2013年5月8日

この授業用の WWW ページは <http://www.math.meiji.ac.jp/~mk/syori2-2013/>

## 1 連絡事項

- 本日、「(仮称)十進 BASIC による JIS Full BASIC 入門」という冊子を配布します。紙に氏名を記入して一部ずつ受け取って下さい。

## 2 (仮称)十進 BASIC 入門

今日は十進 BASIC をとりあえず体験してみる (なるべくたくさんサンプル・プログラムを実行する) のが目標です。

諸君はプログラミングが初めてではないはずですから、文法を一から細かく解説するつもりはありません (例えば変数がどういうものであるかなどは「分かっている」と仮定します)。プログラム例から、この命令は何をするものかと理解して (英単語の意味を想起することを勧めます)、軽く記憶して下さい (例えば「PRINT で文字を表示する、キーボードからの入力 INPUT」位)。

自分でプログラムを書く場合は、漠然とした記憶だけでは足りませんが、次の3つのことを心掛けて下さい。

1. 気軽に試す精神を持つ (「試す」行為は頭が良く働きます)。
2. サンプル・プログラムの真似をする (最初はそのまま試す、で構わない)。
3. オンライン・ヘルプ (後で紹介します) を調べる。

(プログラミング言語は人工言語で、自然言語とは大きな違いがありますが、例えば英会話の学び方には大いに参考になるところがある、と私は思います。ある程度のマメさが必要です。)

## 2.1 紹介 (十進 BASIC のすすめ)

「(仮称)十進 BASIC」が正式な名称ですが、少々長いので、この講義では「十進 BASIC」と呼ぶことにします。

十進 BASIC は、[文教大学の白石和夫氏](#)によって作成されたフリー・ソフトウェアで、数学教育での利用を目的として、JIS Full BASIC を Windows 環境で実現することを目標に作られたものだそうです (現在では、Mac 用、Linux 用も配布されています)。数の精度を 10 進 1000 桁に設定する **1000 桁演算モード**や、**有理数演算モード**、**複素数演算モード**が用意されているため、確かに、数学の計算に便利に使えます<sup>1</sup>。また**入手・インストールも大変簡単**であるだけでなく、**親切なオンライン・ヘルプ**を備えていて、パソコンやプログラミングの初心者にも安心して勧められるものです<sup>2</sup>。

### 十進 BASIC の勧め

- (普通の人に) 最近の Windows パソコンや Mac は使いやすくなりましたが、何か計算をしたい場合にはある意味で不便になってしまっています (プログラミングの敷居が高い)。電卓がわりにパソコンに十進 BASIC をインストールすると便利です。
- (数学村の人に) 特に C 言語等の現在「ふつう」のプログラミング言語では、高精度計算、有理数計算などが簡単にはできません。数学の世界にいる者には、既に他のプログラミング言語に慣れていても、十進 BASIC を習得する価値は大いにあります。
- (数学の教師になろうという人に) 中学高校の教室で生徒にプログラミング体験をさせるときに、十進 BASIC は有力な選択肢になると思われます。
- (情報処理 2 履修者に) Windows の動くパソコンには簡単にインストールできるので、自宅に Windows の動くパソコンがあれば、この課題に関しては自宅でほとんどのことを済ませられるでしょう。お家でプログラミングがマスターできる？

### 大昔話: BASIC とは…

BASIC とは、元々は大型電子計算機 (昔コンピューターのことを電子計算機と呼びました) 向けに、**プログラミングの教育目的**で作られたプログラミング言語ですが、パソコン普及期から 1990 年代半ばくらいまでの多くのパソコンには、BASIC インタープリターが搭載され (そのほとんどが Microsoft BASIC で、これが Microsoft の最初の主力製品でした)、色々な意味で最も普及しているプログラミング言語でした。例えば高等学校の数学の教科書にも BASIC のプログラムが掲載されていました (今でも掲載されている?)。今では BASIC の処理系が搭載されているパソコンを探す方が難しくなっているのは、筆者には隔世の感があります。

<sup>1</sup>通常のプログラミング言語では、10 進 16 桁程度の精度の演算しかできない場合がほとんどで、不便なことが案外と多いのです。

<sup>2</sup>念のため十進 BASIC の欠点も述べておくと、インタープリター形式のため、コンパイラ形式のプログラミング言語 (処理系) と比べると、実行速度が遅く、例えば大規模な反復が必要な計算には向かないことや、システム・プログラミングには不向き (というか、ほぼ不可能) なことがあげられます (ですから、例えば C 言語の代りとして使うことはできません)。

昔話: 卒研にて… (時間に余裕があれば)

十進 BASIC を利用するにあたって必要な情報のほとんどは、作者の WWW ページ「(仮称) 十進 BASIC ホームページ」<sup>3</sup> から得られます。またこのページから十進 BASIC の配布ファイルを手に入れることも出来ます。この WWW ページは、例えば Google のような検索エンジンで「十進 BASIC」を検索すればヒットするでしょう。

十進 BASIC の使い方は、WWW ページの「(仮称) 十進 BASIC による JIS Full BASIC 入門」<sup>4</sup> 等でも解説されていますが、Windows 版の配布ファイル中に PDF 形式のファイル ([http://www.geocities.jp/thinking\\_math\\_education/tutorial.pdf](http://www.geocities.jp/thinking_math_education/tutorial.pdf) — あれ?今読めないですね。)があるので、通常は自分で印刷して持っているといいでしょう。本日はそれを配布することにしました。次回以降の授業には持ってきて下さい。

## 2.2 インストールについて

通常のインストールは、「(仮称) 十進 BASIC のホームページ」<sup>5</sup> にある

「Windows95/98/Me/NT4.0/2000/XP/Vista 新規インストール」<sup>6</sup>

からインストーラー BASIC762setup.exe (数字の部分はバージョンなので頻繁に変る) を入手して、実行して、インストーラーの指示に従うだけです。

情報処理教室の Windows 環境では、既にインストール済みです。

## 2.3 使い方超入門

最初は百聞は一見に如かずで、まずは私 (桂田) のしていることを真似してください。

---

<sup>3</sup><http://hp.vector.co.jp/authors/VA008683/>

<sup>4</sup><http://www.koshigaya.bunkyo.ac.jp/shiraish/basic/tutorial/contents.htm>

<sup>5</sup><http://hp.vector.co.jp/authors/VA008683/>

<sup>6</sup><http://hp.vector.co.jp/authors/VA008683/setup.htm>

真似して下さい

1. スタートメニューの [すべてのプログラム] の「その他のソフト (プログラミング系)」から、[十進 BASIC] の [BASIC] を選んで起動する。
2. プログラムを入力するウィンドウに PRINT 1/2+1/3 と入力する。

```
PRINT 1/2+1/3  
END
```

3. **F9** キーを押して実行する。
4. ボタンを押すことで演算モードを変えてから、**F9** を押して再実行する。
5. [ファイル (F)] メニューから、「名前をつけて保存 (A)」を選び、syori2 フォルダに first.BAS という名前で保存する。
6. 一度十進 BASIC を終了してから再起動し、そのウィンドウに、first.BAS をドラッグ・アンド・ドロップしてみる。( .BAS が十進 BASIC に関連付けられていれば、BASIC のプログラムをダブルクリックして十進 BASIC を起動することも出来ます。)

プログラミングについてまったくの初心者でない限り (C 言語を学んだはずですね?)、敷居は低いと思いますが、事前に少し注意しておきます (主に C 言語との違いの部分)。

- 変数宣言は必要ない。
- 名前を構成するローマ字は大文字と小文字の区別がされない。例えば hensu と HENSU は同じとみなされる。
- 変数への代入は LET 変数名=式 とする (入力は、LET を省略して、変数名=式 だけで良い)。
- C 言語の printf() に相当するのは PRINT 命令、scanf() のようなキーボードからの値の入力には INPUT 命令を用いる。
- オンライン・ヘルプ (**ヘルプ (H)**) がとても親切に出来ているので、使い慣れることがお勧めです。例えば PRINT, INPUT という単語だけ覚えておけば、どう使うか、すぐに調べられるはず。
- もともと BASIC ではすべての行に行番号が必須でしたが、十進 BASIC では省略可能です。必要がない限り行番号は書かないことにしましょう。

### 3 レポート課題 4

配布したチュートリアル『(仮称) 十進 BASIC による JIS Full BASIC 入門』の 4 ページから載っているサンプル・プログラムを実際に入力・実行・保存し、**本日一番最後 (授業終了 10 分前以降が目安) に実行したプログラム (何とか.BAS) を Oh-o! Meiji で送って下さい。**

プログラムの先頭付近の注釈 (REM 文) に学年・組・番号・氏名を書くこと。

例えばこんなふうにして下さい (プログラムはもっと凝ったのにして下さい)

```
REM kadai4.BAS
REM 情報処理2 課題4
REM 2-16-99 数学 学
FOR k=1 TO 10
PRINT RND
NEXT k
END
```

締切は一応5月8日(水)18:00とします(介護実習などで休んだ人が後から提出する場合はその旨書いて下さい)。

次回は FOR~NEXT 命令を使うので(それで課題が出ます)、なるべくチュートリアルの 2.3 まで進んでおく和良好的です。

グラフィックスまで習得しておく色々楽しいことができます。

必ずしもチュートリアルの端から順に全部試す必要はありません。自分で面白そうと思ったものを適当に選んでもらって構いません。

## 4 余裕があればトライ: T<sub>E</sub>X でレポートを書く

プログラミングのレポートには、本来は、何をするか(解くべき問題は何か)、プログラムそのものと説明、実行結果とその分析などを書くべきです。

前節で示したように、注釈として色々な説明を書くことが出来ますが、あまり長いもの(例えば実行結果)を入れるのは変ですね。

次回からは T<sub>E</sub>X でレポートを書いてもらいますが、やり方を簡単に書いておきますので、余裕のある人はトライしてみてください(kadai4.BAS のかわりに、kadai4.pdf を提出してくれても良いです)。

```

\documentclass[12pt]{jarticle}
\usepackage[a4paper]{geometry}% 好みの問題
\usepackage{amsmath,amssymb}% 今回は不要かも
\usepackage{moreverb}% 今回これが必要
\begin{document}
\title{情報処理2 課題4 レポート}
\author{2年16組99番 数学 学}
\date{2013年5月8日}
\maketitle

\section{プログラム}

次は、チュートリアルの p.~11 の例13のプログラムです。
10個の疑似乱数を発生させて表示できます。

\verbatiminput{kadai4.BAS}% これで kadai4.BAS を取り込みます。

\section{プログラムの実行結果}

kadai4.BAS の実行結果は次のようになる。

\verbatiminput{kadai4.TXT}% これで kadai4.TXT を取り込みます。

\section{結果の分析}
(以下略)

\end{document}

```

(念のため) コマンドプロンプトでこんなふうに処理

```

Z:\¥.windows2000¥syori2>pllatex kadai4.tex
Z:\¥.windows2000¥syori2>dviout kadai4.dvi
Z:\¥.windows2000¥syori2>dvipdfmx kadai4.dvi
こうすると kadai4.pdf が出来上がる。

```

`\verbatiminput{kadai4.BAS}` のかわりに `\listinginput{1}{kadai4.BAS}` とすると、行番号をつけて表示します。

**メモ** 以前は `verbatimfiles.sty` を使っていましたが、今年度から `moreverb.sty` に乗り換えようかと考えています(タブで字下げしてくれるなど便利そうなので)。念のため `verbatimfiles.sty` を使うやり方を説明しておきます。

- `\usepackage{moreverb}` の代わりに `\usepackage{verbatimfiles}`

- `\verbatiminput{ファイル名}` の代わりに `\verbatimfile{ファイル名}`
- `\listinginput{1}{ファイル名}` の代わりに `\verbatimlisting{ファイル名}`