

## Introduction—約束と継承

さあ、これから数学の旅を始めよう。それも、ちっぽけな豆旅をね。

この豆旅では二つの行程が入り混じっている。行程の一つは自分の目で読み思考すること。君たちが実際に旅をするときも、てくてく歩いたり周りを眺めたりするだろう。のんきに構えて、疲れたと思ったら気軽に休みを取りながら歩く。そんな感じで頭を働かせてほしい。もう一つの行程はコンピュータに計算させること。実際の旅でも、乗り物を利用したほうがよいときもある。乗り物は時刻が決められているものもあれば、渋滞など思わぬトラブルに遭うこともある。歩くことに較べたら融通が利かない。自分でプログラムを組んでみれば、しょっちゅうそんな目に遭ってしまう。でも、実際に体験してみることは重要だ。ここに掲げたプログラムは **Java** で書いてあるが、眺めているだけでなく実行してほしい。電車に乗っている人たちを外から眺めても、決して自分が旅をしたことにはならないのだから。

豆旅の前に一つ言っておきたいことがある。それは、この豆旅が数学やプログラミングを“習得”するためのものではない、ということだ。数学について何かを習得しようと思ったら、それなりの数学の書物をそれなりに読まなくてはならない。プログラミングの習得には、それなりの経験が必要だ。

それなり？ それってどういうこと？と疑問を感じる人にだけ、そっと教えよう。まず、それなりの数学の書物から。数学に馴染みのない人は、数式やら数表がたくさん出てくれば、それが数学の本のように感じるかもしれない。でも、それは数学の話題を提供しているのであって、それなりの数学が書かれているわけではない。私が言うそれなりの数学書とは、言葉少なに書いてあっても、多くの文章で丁寧に書いてあっても、きちんとした定義が押さえてある書物のことを指している。

では定義って何？ もっともな疑問だね。定義とは、その時点からの数学をあやふやにしないために、厳密に決める“約束”のことだ。ところが多くの場合、厳密な約束から始めて、次々と新しい拡張をしていくのは骨の折れる作業である。拡張するということは、いままでのことがらを次の段階へ“継承”していくことだ。だから継承がきちんとできてないと、実は大して難しくないことでもお手上げになって、いつのまにか数学に嫌気が差してしまうことになる。

しかし、数学にはうまい逃げ道が用意されているものだ。厳密なことはさておいて、大事な結論部分だけを使えばよろしい。公式だけ覚えて、せっせと解く作業がそうだ。これは余計な継承—本当は余計ではないのだけれど—をしないので、途中で訳が分からなくなる心配をせずに、数学に関する力を高められる利点がある。ちょうど、電子レンジの仕組みが分からなくても、肝心の使い方を覚えて調理の幅を広げられるようにである。けれど、数学をきちんと身につけるなら、少々窮屈なことを省かずに書いてある書物を読んでおくほうがよいだろう。そうすれば、電子レンジにドライアイスなんて怖くて入れられないくせに、生卵なら平気で入れて大変なことになってしまうような、そんなヘマを数学でしなくて済むのだから。

さて、それなりの数学の書物が用意できても、それなりに読まなくては意味がない。次は、それなりの読み方を。私が言うそれなりの読み方とは、まっすぐに繰り返し読むことだ。ある程度厳密に書いてある数学の書物は、やすやすと読めるものではない。読みながら理解に努めるのは当然としても、ちょっと理解に苦しむところがあったらどうしたらいいだろう。正解は、そのままもう少しだけ読み進めること。分からないまま読み進めていいのかって？ いいに決まっている。というより、むしろそうすべきだ。そして、本当に分からなくなったら戻ることだ。これは“3歩進んで2歩下がる方式”といって、なかなか根気の要る方法だが、こうすることが確固たる力を付けることにつながるのだ。また、書物によっては読んでいる途中で、練習問題に遭遇するかもしれない。そんなときは手を抜かず解いてみよう。間違いなく何かが身につくはずだ。

と、こんな風にえらそうに書き出してはいるが、私は数学の専門家ではないしプログラミングの専門家でもない。そんな者がこんな物を書くのはお門違いかもしれないが、少なくとも何も知らない人より、多少の経験はあると思う（思い過ごしかもしれないが）。しかし、専門家でないことは事実なので、随所に専門家の目に耐えかねる部分が存在するのも事実だが、あからさまな間違いを書いているつもりはないので安心してほしい。

数学については、できる限り数式や厳密な議論をしない話題にしてある。数式や厳密な議論がないので、数学という畑を眺めはするが、決して耕したりすることがない。海外旅行に出かけても、観光はするが現地の生活に溶け込むことがないのと同じだ。またプログラミングについては、基本的に一つの機能だけしか処理しないものになっている。さらに極力短くなるような書き方をしている。そして豆旅の都合もあるので、ときに変なプログラムに出会うかもしれないが、勘弁してほしい。

早い話、私は上っ面だけをなめているに過ぎないので、数学やプログラミングについて詳しく学びたいければ、それ相応の書物を手元に置いておくことを勧める。細かいことや厳密なことや正統な作法は、是非そのような書物から学んでほしい。私は単にきっかけを与えることができれば満足なのである。

おっと、前置きが長くなりすぎてしまった。さっそく数学の豆旅に出ることにしよう。

**TRY!** Java 言語に関する書籍と、プログラムが実行できるソフトウェアを用意しよう。



プログラムを実行するためには、そのためのプログラミングソフトが必要になる。Java に関するソフトウェアはインターネットのサイトから無料でダウンロードできるはずだ。(JRE ではなく) JDK と呼ばれるものを手に入れよう。もしソフトウェアが用意できないなら、プログラムリストとその後の説明を見ながら、想像を働かせてほしい。プログラミングほど想像と現実が一致しないものはないのだが、実際にソフトウェアがなければ、そうするより仕方がない。

プログラミングには関心がないという人には、豆旅の大半が意味のないものを感じるかもしれない。だが、そうではない。プログラミングというのは、ソフトウェアを開発する一部の人だけに必要な素養ではないのだ。もちろん、誰もが大規模なソフトウェアの開発などしない。しかし、コンピュータソフトは誰もが使うはずだ。とくに、表計算ソフトやデータベースソフトでは、たとえプログラミングをしなくとも、プログラミング的思考ができるのとそうでないのとでは、できる仕事や作業に雲泥の差がつくのだ。電子レンジを使うだけでも食事はできるが、フライパンや鍋を自在に使えてこそ質のよい食事ができるのと似ている。電子レンジを使うだけのレベルがソフトウェアを使うだけのレベル、プロの料理人のレベルがソフトウェア開発を職業とするレベルとするなら、家庭で料理を作るレベルがプログラミングができるレベルと言えるだろう。

家庭料理を人任せにしているのでは一人前とは言えない。プログラミングの理解を深めてこそ一人前なのだ。簡単な料理が自分で作れる程度に、簡単なプログラムが組める程度になろうじゃないか。



ところで、君たちが用意した Java のソフトウェアはどんなものだろうか？ 統合開発環境 (IDE) のものなら快適に作業が進むが、私がここに記載したプログラムは、Windows ならコマンドプロンプト、Macintosh ならターミナルという環境で実行している<sup>1)</sup>。だから、君たちが使うソフトウェアで実行する場合と少し勝手が違うかもしれない。また、私が古臭い記述をしていることもあるだろう。バージョンによってそれぞれの仕様があるから警告を受けることもあるだろう。私としてはあまり細かい点に注意を払いたくないので、不具合が生じたときは大変申し訳ないが、君たちが使っているソフトウェアの仕様に合うよう、プログラムに修正を加えてほしい。旅に出れば各地方で方言に出会うこともあるだろう。細かい仕様はそんなものだと思って、いらいらせずに豆旅を楽しんでもらいたいものだ。

また、プログラムで使われる数値や入力される数値は限定的である。だから何も知らずにプログラムの入

---

1) 実際の動作環境は、MacOS Big Sur (ver.11.2.1) の Terminal で Oracle 社の JDK ver.15.0.2 を使用した。

力要求に対して、勝手な数を入力すると、簡単にエラーを引き起こしてしまう。ここがバスの切符売り場と知らずに、電車の切符を買い求めるようなものだ。係員が親切なら君たちに正しい指示をしてくれる。残念ながら私が用意したプログラムは不親切である。間違った入力に対し親切な応答をさせるには、さらにたくさんコードを書く必要があるが、そうはしていない。だからプログラムは自分一人で楽しんでもらいたい。つまりは“一人旅”ということだ。

さあ、気楽な豆旅に出てみよう。