

Introduction—約束と継承

さあ、これから数学の旅を始めよう。

この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 C++ 言語で書いてあるが、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。

旅の前に一つ言っておきたいことがある。それは、この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？と疑問に感じる人にだけ、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んだろう。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。

だったら難しい書物ではなく、分かりやすく書いてある書物の方が勉強になると思うだろうか。へっ！ そんなのは暇つぶしにはなっても、勉強にはならないよ。分かりやすい書物は著者の力量が優れていて、分かりにくい書物は著者の力量が足りないからって思うかい？ 違うね。分かりやすい書物というのは、あなたが既に理解していることが書いてあるからだ。それでは知識の確認になっても勉強にはならない。分かりにくい書物は、あなたの知らないことが書いてある。それを身につけるのが勉強である。と言っても、既に知っていることが皆無である書物では手のつけようがない。2割程度は既知である書物を頑張って読もう。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は事実だが、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。そして旅の都合もあるので、ときに変なプログラミング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。さっそく数学の旅に出ることにしよう。

TRY! C++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。



ところで、君たちが用意した C++ のソフトウェアはどんなものだろうか？ 私がここに記載したプログラムは、Macintosh の Terminal で実行できることを確認してある¹。Macintosh の Terminal にはもともと C++ が含まれている。君たちが使うソフトウェアでも間違いなく実行できると思う。C++ は十分こなれたソフトウェアなので、いまでは動作環境に依存することはあまりないはずである。

Terminal で実行する場合は、まずテキストエディタなどでコードを記述したファイルを、たとえば `sample.cpp` という名で保存しておく。その上で、保存したディレクトリに移動して `$ cpp sample.cpp` を実行すればよい²。すると、同じディレクトリに `a.out` というファイルができるので、`$./a.out` を実行すれば `sample.cpp` に記述したコードの実行結果を見ることができる。

Terminal に関する基本的な命令や知識は自分で調べてほしいが、必要なのはディレクトリを移動する `cd` とファイル一覧を表示する `ls` ぐらいのものだ。ただ、Terminal で実行するのは案外面倒なものだから、相応のアプリケーションをインストールするのが望ましいだろう。

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、気楽な一人旅に出てみよう。

¹実際の動作環境は、MacOS Big Sur (ver.11.2.1) の Terminal を使用した。C++ は version 12.0.0 である。

²このとき、C++ を実行するためのコンポーネントのダウンロードを促されるかもしれない。素直にインストールしよう。