

まえがき

プログラミングの写経型学習とは

本書の背景となったのは、企業や学校での教育実践です。一つは著者の一人、吉川がキヤミーという会社の中でプログラマを育成してきた方法です。同社では、プログラミングの初学者の研修として、まず中身の理解は横へ置いておいて、教科書の例題をすべて入力し、実行することでプログラムに慣れるという、言わば「習うより慣れろ」式の学習方法を採用してきました。それが済むと今度は実際に課題を与えられて、自分で考えながらプログラムを構成していくことで実践的な能力を養成します。

もう一つは、プログラミングによる問題解決の過程を、具体例をなぞりながら体験的に学習してもらうというものです。これは、著者の一人、藤岡が京都市立堀川高校で問題解決型の科目の実践に採用してきたアプローチです。

本書は、著者の喜多や岡本が、吉川や藤岡らとの共同研究を通じ、これらの実践で得られた知見を生かし、なおかつ初学者ができるかぎり自習で学習を進められるように構成したC言語の学習書です。

例題をまずはどんどん実行してC言語に慣れ、そして問題解決に挑戦する、こういう形でのプログラミングの学習方法をここでは写経に例えて写経型学習と呼んでいます。

本書の構成

本書は、大きく2部構成を取っています。前半は、C言語の主な文法・機能を自習中心で学んでいくためのワークブックです。

最初に、C言語に現れる文字や言葉について解説していきます。ソースコードを的確に入力できること、音読できることや英語の語源を知っていることは、プログラムについて学習者が指導者や先輩などと円滑にコミュニケーションしながら学習を進める上で重要なことだと著者らは考えているからです。

その次からのC言語の文法にかかわる事項の学習では、例題はすべて完全なプログラムとして書かれています。それ単独で入力すれば、コンパイルし実行することができます。実行例も提示してあるので、学習者は実際にプログラムを入力してみて、コンパイル、実行の結果を確認しながら学習を進めてください。

C 言語の文法の解説は、その詳細を網羅的に行うものではなく、実際によく使われ、初学者にとって重要なものに限定しています。また、例題は単に文法を示すためのものではなく、できるだけ実際にプログラムでよく使われるパターンを示すようにしています。

プログラムで繰り返し処理を行う for 文は、初学者にとってのプログラミング学習の一つのヤマです。そして、実際のプログラムでも他の命令と組み合わせてさまざまな使い方をします。本書では、for 文に慣れるために、for 文を使った例題をかなり多くドリルとして掲載しています。できるだけ、実際に入力し実行してみてください。また、「こういうことをするには、こうすればいいのだ」という事例集としても使えると思います。

本書の後半は、実際に問題解決のために C 言語でプログラミングする手順を示しています。取り上げた例題は 3 目並べという簡単なゲームです。実際のプログラムも 200 行程度ですが、初学者にとっては、この程度の量のプログラムでもとても難しいと思います。問題の処理手順を普通の言葉（日本語）で考え、それを分析してプログラムとしてどのように構成していけばよいのかを学びます。そして、それを C 言語の関数などとしてコーディングとテストを繰り返してプログラムを仕上げていく手順を示しています。

前半と同じように、プログラムを入力し、コンパイル、実行しながら読み進めてください。それができたら、今度は自分でさまざまなプログラムの作成に挑戦していくことでプログラミングの能力を身につけることができていくと思います。

本書における C 言語の捉え方

C 言語は、大学でのプログラミング教育で最も多く取り扱われているプログラミング言語の一つです。一方、C 言語にはデータの格納場所を示す「ポインタ」など初学者にとって習得の難しい概念も少なくありません。

また現代の実用的なプログラミングでは、C 言語が使われることが減ってきており、C 言語の文法などは継承しつつ、高度なプログラミングを安全に行える C# や Java などの言語が使われることが多くなってきています。

本書では、これらのより新しい言語へのステップアップを想定しつつ、プログラミングの基本である変数や代入、制御構造、関数呼び出しなどを学ぶための言語として C 言語を捉えて紹介しています。したがって、初学者にとって難しいもの、より新しい言語での扱いが異なるものについては学習対象から外してあります。具体的には以下のようなものです。

- ポインタは、関数の引数の受け渡しに限定して紹介しています。ポインタのインクリメント演算や動的なメモリ管理は省きました。
- 文字型の配列としての文字列の扱いは紹介していません。
- 複数のソースコードにわたるプログラムの記述方法は省略しました。
- 入出力は主に printf, scanf を用いた標準入出力のみを扱っています。ファイルの取り扱いについては、巻末の付録で簡単なものを紹介しています。

したがって、C 言語でより本格的なプログラムを作成するには、本書の学習後、さらに範囲を広げて学習を進めていただく必要があります。本書は、機能的には基本的な事項だけを取り上げながら、より規模の大きなプログラムを書く際に必要になる考え方はしっかり身につけられるように配慮したつもりです。

プログラミング環境について

本書の草稿の一部を用いて、京都大学で行った授業では、プログラミング環境として MinGW(およびフリーのテキストエディタ)を用いました。コンパイラやプログラムの実行をコマンドプロンプトで行わなければならないことや文字列としての漢字の扱いに若干、注意が必要であることなどの難点はあるのですが、必要なものを USB フラッシュメモリなどに格納しておけば、大学の演習室に設置している教育用の Windows PC でも学生が自宅で使う PC でもインストール作業なしに使えることが MinGW を選んだ理由です。

本書を書くにあたって、共同研究にご協力いただいた株式会社キヤミーのみなさん、京都大学での授業実践にご協力をいただいた京都外国語大学准教授、村上正行先生、そして共立出版編集部の方の石井徹也さんのご支援を頂きました。ここに御礼申し上げます。本書がプログラミングを学ぼうとする方の手助けに少しでもなれば幸いです。

2012 年 1 月

著者を代表して 喜多 一