第6章 C言語 基礎編:初めてのプログラミング

Chapter 1C 言語プログラムの作成Chapter 2C 言語のやさしい入門

タイトル: 「新訂 新 C 言語入門 ビギナー編」 著者: 林 晴比古 出版: ソフトバンク パブリッシング株式会社

6.1 UNIX 環境の使い方

以後、

5.5 節で述べたように、C 言語は UNIX¹を記述するために開発された言語です。本テキスト では、Microsoft 社製の Windows OS² (以後、「Windows」と記す)上に UNIX 互換環境を構築 する Cygwin ソフトウェア (以後、「Cygwin」と記す)を導入し、Cygwin のソフトウェア群に含 まれるフリーで最もポピュラーな GNU³ C と呼ばれる C 処理系⁴を使って C 言語を学習します。 Cygwin の入手方法とインストール方法についてはインターネット⁵や書籍⁶などで詳しく解説さ れていますので、そちらを参照してください。なお、この授業を受講している方で、希望者には、 筆者特製の Cygwin インストールメディア (CD-ROM または DVD-ROM)を配布いたします⁷。

「CygwinのUNIX 互換環境」≒「UNIX 環境」 とし、UNIX 環境に準じた解説を行います。

* Cygwin に特有なものもあるので、注意して読んでください。

¹マルチユーザ・マルチタスク・仮想記憶方式・タイムシェアリングシステムなどのモダンな機能を備えています。 現在使用されている多くのオペレーティングシステムにも、これら UNIX のモダンな機能が取り入れられています。 ²Windows 98, Windows Me, Windows 2000, Windows XP, Windows Vista など。

³「Gnu is Not Unix」の意で、「ぐぬー」と発音します。UNIX 互換ソフトウェア群の開発プロジェクトの総称で、 フリーソフトウェアの理念に従った修正・再配布自由な UNIX 互換システムの構築を目的としています。また、GNU で開発されたソフトウェアに適用されている GPL (the gnu General Public License) は、「あらゆるソフトウェアは 自由に利用できるべき」という理念を実現したライセンスとして知られています。

 ⁴「新訂 新 C 言語入門 ビギナー編」の11ページに記述されている「C 処理系について」を参照してください。
 ⁵「Cygwin」・「入手方法」・「インストール」などをキーワードにサーチエンジンで検索するとよい。

⁶UNIX の入門書 (Cygwin のインストール CD-ROM 付) として数社から販売されています。

⁷幸山研究室のホームページ「Cygwin をインストールしよう」を参照してください (ダウンロードは学内のみ)。

http://kouyama.math.u-toyama.ac.jp/main/computer/personal/cygwin/index.htm

UNIX 環境では、キーボードから文字列のコマンド (命令)を図 6.1 のシェル⁸ (shell) に入力して コンピュータを操作する方法が一般的です。このような環境を CUI⁹ (Character User Interface) と呼び、コンピュータを熟知した人にとっては非常に有効なコンピュータの操作方法です。まず、 C言語を学習する前にシェルの使い方と基本的な命令 (コマンド)について学習しておきましょう。



図 6.1: カーネル (Kernel) とシェル (shell)

最初に、ファイル (file) とディレクトリ¹⁰ (directory) について復習しておきましょう。1.1 節で 述べたように、コンピュータに仕事をさせるにはプログラムとデータが必要で、これらはハード ディスクなどの記憶装置にファイルという形で記憶されています。また、これらのファイルを種類 や用途ごとにまとめて入れておく箱がディレクトリで、人間にも理解しやすく、効率的なファイ ルの管理(読み書き)ができる様になっています。この仕組みはオペレーティングシステムのファ イルシステム¹¹ (file system) という機能によって提供され、木構造 (tree structure; tree 構造) と 呼ばれる階層的なディレクトリ構造を持っています。UNIX も図 6.2 のように最上位ディレクト リである「/ (ルート¹²)」と呼ばれる特別なディレクトリを起点に、様々なファイルやディレク トリがこのディレクトリ構造にしたがって配置・記憶されています。なお、UNIXのディレクト リ名には古くからのしきたりがあり、「bin」はbinary、「lib」はlibrary、「tmp」はtemporary、 「var」は variety のように決まった略記が用いられます (これらのディレクトリには、用途や機

⁸オペレーティングシステムの心臓部は「カーネル (kernel)」ですが、人間と対話的にできているわけではありま せん (どちらかというとコンピュータより)。そこで人間が対話的にカーネル (コンピュータ)を管理するために作られ たソフトウェアが「シェル (shell)」になります。その語源は「カーネル」を包み込むように「シェル」が存在してい ることから「貝殻」に例えて「シェル」と名づけられました。なお、Windows にも「DOS 窓」や「コマンドプロン プト」と呼ばれる CUI が備わっています。

⁹逆に、マウスなどを使って画面に表示されている絵やアイコンなどを操作することでコンピュータを操作する環 境を GUI (Graphical User Interface) と呼び、初心者でも直感的または視覚的にコンピュータを操作することができ ます。 ¹⁰一般ユーザに親しみやすい表現として**フォルダ** (folder) という言葉も使用されています。

¹¹Windows には FAT や NTFS、UNIX には EXT2 や XFS などのファイルシステムがあります。なお、ファイス システムが提供するその他の機能としては、ファイルを作成した日時や最後に更新した日時の管理、ファイルのアク セス権限の管理などがあります。

¹²Windows では「マイ コンピュータ」が UNIX のルートに相当します。

能ごとに分類されたファイルやディレクトリがそれぞれ格納されています)。さらに、UNIX に はマルチユーザと呼ばれる複数のユーザが同じコンピュータを同時に利用できるという機能が備 わっており、通常は「home」の下位に各ユーザが自由に使用できるディレクトリが割り当てられ ます。このディレクトリをホームディレクトリと呼びます。例えば、ユーザ名が「nkouyama」の 場合、図 6.2 では「home」の下位ディレクトリである「nkouyama」がホームディレクトリとなり ます。ユーザ名は各ユーザごとに異なるので適宜読み替えてください (p.118 を参照)。



図 6.2: Cygwin (UNIX) のディレクトリ構造

なお、ディレクトリ「cygdrive」は Cygwin に特有なディレクトリで、「cygdrive」の下位ディ レクトリを通して Windows の各ドライブへアクセスできるようになっています (例えば表 6.1)。

Cygwin のディレクトリ	Windows のドライブ
a (/cygdrive/a)	A: (フロッピディスク)
c (/cygdrive/c)	C: (ハードディスク)
d (/cygdrive/d)	D: (CD-ROM)
:	
z (/cygdrive/z)	Z: (ネットワークドライブ)

表 6.1: Cygwin のディレクトリと Windows のドライブの対応

シェル (UNIX 環境) 上では、パス¹³ (path) と呼ばれる特定の場所 (ディレクトリ) を指定する 方法が非常に重要で、ユーザはディレクトリの移動や使用する命令・対象となるデータの場所な どを的確に指定しながらコンピュータに命令とデータを与え作業を進めて行く必要があります。 特に UNIX 環境では、特別なディレクトリを表 6.2 のような記号を使って表し、ディレクトリや ファイルの区切り記号にはルート「/」と同じ「/ (スラッシュ)」を使います¹⁴。なお、現在ユー ザが居る場所 (現在のディレクトリ) を**カレントディレクトリ** (current directory) または**ワーキン グディレクトリ** (working directory) と呼びます。

記号	記号の説明
/	ルートディレクトリを表す
•	カレントディレクトリを表す
••	1つ上のディレクトリを表す
~	ホームディレクトリを表す

表 6.2: 特別なディレクトリを表す記号

パスの指定方法には絶対パスと相対パスの2種類があります。絶対パスは、最上位ディレクトリ であるルートから順に下位ディレクトリをたどりながら目的のディレクトリまでを指定する方法 で、必ずルートを表す記号「/」から始まります。相対パスは、あるディレクトリ(カレントディ レクトリ)から順に上位または下位のディレクトリをたどって目的のディレクトリを指定する方法 です。特別なディレクトリを表す記号や相対パスは、最終的には絶対パスに変換されて目的のディ レクトリを示します。図 6.2 の場合のパスの書き方の例を以下に挙げておきます。ただし、相対 パスはカレントディレクトリをホームディレクトリ「/home/nkouyama」として示したものです。

絶対パスの例:

図 6.2 のディレクトリ① ・・・	/bin /bin/ /bin/.
図 6.2 のディレクトリ② ・・・	/usr /usr/ /usr/.
図 6.2 のディレクトリ③ ・・・	<pre>/cygdrive/c /cygdrive/c/ /cygdrive/c/.</pre>
図 6.2 のディレクトリ④ ・・・	/usr/doc /usr/doc/ /usr/doc/.
図 6.2 のファイル⑤ ・・・	/cygdrive/z/src/sample.c

相対パスの例: *いろいろな書き方ができます。

図 6.2 のディレクトリ① ・・・	//bin//bin///bin/. など
図 6.2 のディレクトリ② ・・・	//usr .///usr/ .///usr/. など
図 6.2 のディレクトリ③ ・・・	//cygdrive/c//cygdrive/./c など
図 6.2 のディレクトリ④ ・・・	//usr/doc//usr//usr/doc/ など
図 6.2 のファイル⑤ …	//cygdrive/z/src/sample.c など

¹³実は、Cygwinの起動時にプログラムが収められているディレクトリを登録するようになっていて、あたかもカレントディレクトリに命令が存在しているかのように振舞います。このことを「パスを通す」と呼びます。

¹⁴Windows の CUI では「¥」が使用されます。

【注意】表 6.1 の「Z: (ネットワークドライブ)」は富山大学の端末室のコンピュータに特 有のドライブで、ユーザが自由に使うことのできるドライブとなっています。このドライブ はネットワーク越しに透過的に利用できるようになっています。なお、このドライブの実体 は総合情報基盤センター内に設置されたサーバー (サービスを供給する側のコンピュータ) に接続されたハードディスクの一部です。言い換えれば、このサーバーに接続されたハード ディスクの一部 (各ユーザごとに用意されたディレクトリ)をネットワーク越しに端末室の コンピュータ (クライアント)から利用できる仕組みになっています (このような仕組みを ネットワークファイルシステムと呼ぶ)。さらに、このドライブのおかげで、どの端末室のど のコンピュータを利用しても同じ環境でコンピュータを利用できるようになっています。し たがって、端末室のコンピュータの Cygwin 環境 (UNIX 環境)でも、このネットワークドラ イブがホームディレクトリとなるように設定されています (表 6.1 で見たように、Windows 環境ではネットワークドライブとして扱われるが、Cygwin 環境ではディレクトリとして扱 われる)。

表 6.3 および表 6.4 に、富山大学の端末室のコンピュータと自宅のコンピュータの Cygwin 環境の違いについて挙げておくので、自分の環境にあわせて適宜読み替えてください。ただ し、自宅のコンピュータの Cygwin 環境については、Cygwin がデフォルトインストールさ れていることを想定しています (実は、この Cygwin 環境の方が標準的です)。また、自宅の コンピュータでは、表 6.4 のように日本語 (2 バイトコード)のユーザ名やコンピュータ名を 設定している場合があるかと思われますが、UNIX は基本的に1 バイトコードを扱うように 作られており、2 バイトコードを用いるのはあまり好ましくないことをコメントしておきま す (もし、Cygwin が正しく動作しない場合や変更しておきたい場合は幸山まで)。

	富山大学のコンピュータ	自宅のコンピュータ
ユーザ名	nkouyama	nkouyama
コンピュータ名	SCI161	fermat
ホームディレクトリ (Cygwin 環境)	/cygdrive/z	/home/nkouyama
ホームディレクトリ (Windows 環境)	Z:¥	C:¥cygwin¥home¥nkouyama

表 6.3: 富山大学のコンピュータと自宅のコンピュータ (筆者の場合)

	富山大学のコンピュータ	自宅のコンピュータ
ユーザ名	s0740058	富山太郎
コンピュータ名	SCI176	富山太郎のコンピュータ
ホームディレクトリ (Cygwin 環境)	/cygdrive/z	/home/富山太郎
ホームディレクトリ (Windows 環境)	Z:¥	C:¥cygwin¥home¥富山太郎

表 6.4: 富山大学のコンピュータと自宅のコンピュータ (学生の場合)

では、実際に Cygwin を起動し、基本的な命令を習得しましょう。最も簡単な起動方法は Windows のディストップにある図 6.3 の Cygwin のアイコンをダブルクリックします¹⁵。なお、Cygwin を 終了するにはウィンドウの「閉じる」ボタンをクリックします¹⁶。



図 6.3: Cygwin のアイコン

最初に Cygwin を起動すると図 6.5 のようなウィンドがディスクトップ上に表示されます¹⁷ (bash と呼ばれるシェルが起動します)。このとき、ホームディレクトリの作成とホームディレクトリに シェルの初期設定ファイルがコピーされます¹⁸。



図 6.4: Cygwin のウインドウ (初期起動)

¹⁵実際には、Windowsのコマンドプロンプト上でディレクトリ「C:¥cygwin」にあるバッチファイル「cygwin.bat」の記述にしたがって bash というシェルを起動しているだけです。従って、次のような方法でも起動できます。 (1) 「マイ コンピュータ」→「C:」→「cygwin」の順にディレクトリを開き、「cygwin.bat」と名前のついたアイコンをダブルクリックする。

⁽²⁾ まず、メニューバーの「スタート」→「プログラム」→「アクセサリ」→「コマンドプロンプト」を順にたどり、 プログラム「コマンドプロンプト」を起動する。次に、プロンプトに続けて「C:¥cygwin¥bin¥bash --login -i」と 入力し、[Enter] キーを押す。

¹⁶プロンプトに続けて「exit」と入力し(Enter) キーを押しても終了できます。

¹⁷このテキストは各自で印刷することを前提としているので、印刷時にインクが少なくて済むよう背景を黒から白 に、文字を白から黒に変更してあります。デフォルトは背景が黒で、文字が白となっています。

¹⁸その他に、UNIX のユーザ管理に必要なパスワードファイル (/etc/passwd) とグループファイル (/etc/group) が更新されます (Windows のAdministrator の権限が必要)。

通常は、図 6.5 のようなウインドウが表示されます (富山大学の端末室のコンピュータは、セキュ リティ上の理由により、図 6.4 のようなメッセージが毎回表示されます)。なお、図 6.5 の各部分 の解説を表 6.5 に載せておきます (こちらの内容についても、各ユーザ・各コンピュータの環境 ごとに異なりますので、118ページを参考にユーザ名・コンピュータ名・ホームディレクトリを 読み替えてください)。



図 6.5: Cygwin のウインドウ (通常起動)

記号	記号の説明	
nkouyama	ユーザ名 (各ユーザごとに異なる)	
SCI161	コンピュータ名 (理学部 (SCIence) 端末室のコンピュータ)	
Q(アットマーク)	電子メールアドレスと同じで、ユーザ名とコンピュータ名を区切る	
~ (チルダ)	通常はカレントディレクトリを表示 (現在のディレクトリがわかる)	
	図ではホームディレクトリを表す「~」が表示されている	
\$ (ドル)	プロンプト (prompt) と呼ばれ、この記号に続けて命令を入力する	
_ または	カーソル (cursor) と呼ばれ、この位置に入力した文字が現れる	

表 6.5: Cygwin のウインドウ (図 6.5) の各部分の解説

以後、Cygwinのウィンドウは下図のように角の丸い長方形で表します。なお、以下のルールに従って記述してあります。
【ルール1】 図 6.5の緑色の文字の部分とカーソルは省略し、プロンプト「\$」のみを記述します。

* UNIX 環境のデフォルトのプロンプトは「\$」です。
* ルート (root) と呼ばれるスーパーユーザのプロンプトは「#」です。

【ルール2】 入力すべきコマンド (命令) は青文字のタイプライタ体 (abcb・・・) で記述します。命令の入力が完了したら Enter キーを押して実行してください (慣れるまでは記述しますが、基本的には省略します)。

従って、以下のように作業を進めていくことになります。

\$ コマンド1 $[Enter]$
\$ コマンド2 Enter ・・・ コマンド2を実行した結果が表示される ・・・
\$ コマンド3 Enter コマンド3を実行した結果が表示される
÷

ファイルとディレクトリの操作に関するコマンドについて紹介しておきます(資料配布)。

・カレントディレクトリの表示 · · · pwd (print working directory)

- ・ディレクトリ内のファイル名の表示 ・・・ 1s (list)
- ディレクトリの移動 ··· cd (change directory)
- ディレクトリの作成 ··· mkdir (make directory)
- ・ディレクトリ名の変更およびファイル名の変更 ··· mv (move)

- ・ディレクトリの削除およびファイルの削除 ··· rm (remove)
- ・ファイルの内容の表示 (1) · · · more または cat
 - * [スペ-ス] キーまたは [f] キー¹⁹で次のページを表示。 [q] キー²⁰で終了。
- ファイルの内容の表示 (2) · · · less
 * スペース または f で次のページを表示。 b キー²¹で前のページを表示。 q キーで終了。

もし、コマンドの使い方を忘れたときはオプション「--help」を使って調べることができます。 例えば、オプション「--help」を付けてコマンド「ls」を実行すると下図のようにヘルプが表示 されます。もちろん、英語です。

By default, color is not used to distinguish types of files. That is equivalent to using --color=none. Using the --color option without the optional WHEN argument is equivalent to using --color=always. With --color=auto, color codes are output only if standard output is connected to a terminal (tty). The environment variable LS_COLORS can influence the colors, and can be set easily by the dircolors command.

Exit status is 0 if OK, 1 if minor problems, 2 if serious trouble.

Report bugs to <bug-coreutils@gnu.org>.

¹⁹forward の頭文字。

²⁰quit の頭文字。

²¹backward の頭文字。

また、多くのコマンドにはマニュアルが付属しているので、コマンド「man」を使用して調べる こともできます²²。例えば、コマンド「ls」を「man」で調べるには下図のように入力します。

```
$ man ls (Enter)
```

画面が切り替わって下記のようなマニュアルが表示されます。

```
LS(1)
                                 User Commands
                                                                         LS(1)
NAME
      ls - list directory contents
SYNOPSIS
      ls [OPTION]... [FILE]...
DESCRIPTION
       List information about the FILEs (the current directory by default).
       Sort entries alphabetically if none of -cftuSUX nor --sort.
      Mandatory arguments to long options are mandatory for short options
       too.
       -a, --all
             do not ignore entries starting with .
       -A, --almost-all
              do not list implied . and ..
       --author
              with -1, print the author of each file
```

* スペース または f で次ページを表示。 b で前ページを表示。 q で終了。

更に、シェル「bash」の便利な使い方を幾つか挙げておきます。

履歴機能 シェルにはこれまで使用したコマンドを履歴しておく機能があります。コマンドの履 歴を表示するにはコマンド「history」を利用します。更に、履歴されたコマンドを再利用する には「!」に続けて「historyで表示された番号」または「目的のコマンドの最初の頭文字²³」を 入力します。使用例を以下に挙げておきます。

²²Cygwin のマニュアルは英語ですが、日本語のマニュアルも整備されており、UNIX 環境を整備すれば利用することができます。また、インターネットや書籍で調べるのも良いでしょう。

²³頭文字が一致する履歴されたコマンドの内、一番最後に実行されたコマンドが選択され実行されます。

<pre>\$ history Enter 1 pwd 2 ls 3 history</pre>	↓ 履歴されたコマンドの一覧が表示される
\$!1 (Enter) pwd /home/nkouyama	← 履歴されたコマンドの中で最後に実行されたコマンドが表示される
\$!p Enter pwd /home/nkouyama	← 履歴されたコマンドの中で p で始まる最後に実行されたコマンドが表示される

また、 (↑) キーを繰り返し押すと、履歴されたコマンドを新しいものから順に表示します。逆に、 |↓| キーを押すと、履歴された現在のコマンドから1つ新しいコマンドを表示します。

補完機能 補完機能を使用すると、長いファイル名やディレクトリ名を全て入力する必要がなく なります。補完機能を使用するには目的のファイル名・ディレクトリ名・コマンド名の最初の数 文字を入力し、(Tab) キーを押します。使用例を以下に挙げておきます (「mple.txt」が補完さ れフィル名「sample.txt」が表示されます)。



更に、続けて2回 Tab キーを押すと、入力した最初の数文字に該当するファイル名・ディレク トリ名・コマンド名の一覧を表示します²⁴。使用例を以下に挙げておきます (1回目の Tab キー で「1e」が補完され、更に、続けて2回 Tab キーを押すことで補完された文字列「file」に該 当するファイル名やディレクトリ名の一覧を表示します)。なお、最後の入力は、「1.txt」の代 わりに「1 Tab」と入力することもできる。



²⁴検索範囲は、カレントディレクトリとパスの通っているディレクトリ内にあるディレクトリとコマンドが対象となります。

出力制御記号 通常、コマンドを実行した結果は画面に出力(表示)されますが、「I(パイプ)」 や「>(リダイレクト)」(または「<」・「>>(追記)」・「<<」)を使用すると、他の命令に出力結果を 渡したりファイルに出力結果を保存したりすることができます。使用例を以下に挙げておきます。

\$ man ls | more Enter ← 「man ls」の出力結果を「more」によって1ページずつ表示

- \$ ls --help > file.txt Enter ← 「ls --help」の出力結果をファイル「file.txt」に保存 (ファイルが存在しない場合は新規に作成される)
 - \$ ls >> file.txt [Enter] ← 「ls」の出力結果をファイル「file.txt」に追加保存 (ファイルが存在しない場合は新規に作成される)

* 単独で使用するするだけでなく、次のコマンド制御記号などと組み合わせて使用することもできます。

コマンド制御記号「; (セミコロン)」を使用すると1行に複数の命令を記述することが可能 で、先頭から順にコマンドを実行します。また、「& (アンパサンド)」を使用すると、UNIXの機 能の1つであるマルチタスク²⁵の恩恵を受けることができ、直ちに次のコマンドを入力すること ができます。使用方法を以下に挙げておきます。

\$ コマンド1; コマンド2; コマンド3 Enter	← 先頭から順にコマンドを実行

\$ コマンド & Enter	← コマンドをバックグラウンドで実行
[1] 1959	(バックグラウンド ⇔ フォアグラウンド)
\$	← 直ちに次のコマンドが入力できる状態になる (フォアグラウンド)

以上のように、この節を通してシェルの使い方やコマンドの使い方を学習しましたが、これは ほんのさわりでしかありません。シェルをもっと便利に使いこなすためには、「詳しい使い方」と か「多くのコマンドを知る」ということもありますが、その他にも「シェルの仕組み」・「シェル とカーネルの関係」・「環境変数」・「正規表現」・「ショートカットキー」など様々な事柄を知る必 要があります。従って、一度に全てを理解しようとするのではなく、必要に応じて必要な分だけ 少しずつ習得していきましょう。

自分で調べて試すことが大切!

²⁵タイムスライスという方法を用いて、複数の仕事をあたかも同時に行っているようにみせる機能です。