

# プログラミング実習のためのグラフィックを使用可能な ポータブル C 言語学習システム

本間 達\*

(東京医科歯科大学大学院保健衛生学研究科)

Portable C-language learning system with graphics library for programming exercise.

Satoru Honma\* (Tokyo Medical and Dental University, Graduate School of Health Care Sciences)

C-language learning system with graphics library is constructed on a portable medium. The system has been modified using some DOS emulators according to upgrades of OS (WINDOWS series). Students can study programming using the system on any operating system by themselves anywhere. The effectiveness of the system is evaluated by the distribution of assignments submission by year.

キーワード：臨床検査技師，医用情報学，プログラミング環境，グラフィックライブラリ

(Medical technologist, Medical information technology, Programming environment, Graphic library)

## 1. はじめに

臨床検査技師国家資格試験の受験科目の一つである医用工学概論は、臨床検査技師が取り扱う様々な機器の基礎知識を念頭に置いて出題される科目である。具体的には、医用工学・情報科学・検査機器学の3科目で構成されており、いずれも受験資格を取得するための必修科目に設定されている。

情報科学の講義では医療情報を取り扱うことを前提として、そのための道具であるコンピュータやネットワークに関する基礎知識を習得後、医療情報システムの基礎を学習する。この科目は講義のみならず実習も必修科目として設定されるが、その内容について全国共通のカリキュラムが存在しない。このため、コンピュータの利用を共通項とするものの、教員の判断で大学ごとに異なるカリキュラムを設定する。例えば、一般的なワープロや表計算ソフト、あるいは医療分野の研究でしばしば用いられる統計処理ソフトなど、アプリケーションソフトの操作を習得することを目指す場合もある。一方、一部の大学では、電気電子工学を学ぶ医用工学および検査機器学との関連から、医用機器の動作を制御するプログラムについて理解を深めるために、プログラミングの学習をおこなっている。東京医科歯科大学医学部保健衛生学科(以下、本学)では、上記の観点か

ら学科開設以来一貫して C 言語によるプログラミングの実習を行っている。C 言語は 30 コマ程度の学習時間で一通り学習可能で、機器制御に適する入門用のプログラム言語の一つであり、また C++ や Perl, PHP などの近年活用されているプログラム言語と一部の文法が共通するので、大学での実習に適している。

プログラミング初心者興味を引くために、グラフィック機能を活用するのが有効であることが示されている<sup>(1)</sup>が、C 言語は ANSI で制定されている規格にグラフィック機能が定義されていない。しかしながら、一部のコンパイラでは独自ライブラリが用意されていたり、あるいはフリーのライブラリを組み込むなどの手続きでグラフィック機能を利用できる場合もある。LSI-C コンパイラ<sup>(2)</sup>は 1991 年に一部機能を制限した試食版をフリー配布しており、これに対応したフリー版のグラフィックライブラリ<sup>(3)</sup>が一部のプログラムによって作成され、インターネット上で公開されている。これらを組み合わせると、小容量で実習用のシステムを構築できるので、本学では、近年 USB メモリ上にシステムを構築して、情報学実習に利用してきた<sup>(4)</sup>。

このシステムは、大学のみならず学生個人の PC など、任意の PC 上で利用可能である。しかしながら、LSI-C コンパイラは 16bit の OS 用に開発されているので、Windows のバージョンアップに伴い、動作環境を統一することが困難に

なってきた。具体的には以下の3点について、不都合が生じてきた。

- ①ドライブレターが一定ではない。
- ②コンパイラが動作しない。
- ③グラフィック機能が利用できない。

コンパイル時に使用するコンパイラとリンカは、定義ファイルで指定されたパスで実行するので、ドライブレターが異なると動作しない。そこで、異なるドライブレターに対応する定義ファイルを予め用意し、ドライブレターごとに定義ファイルを置換してパスを正しく指定するようなバッチファイルで対応した<sup>(4)</sup>。また、16bit アプリである LSI-C のコンパイラ自体は 64bit OS 上では動作を禁止されているが、16bit のアプリケーションを実行可能な DOS エミュレータ (MS-DOS Player)<sup>(5)</sup> を使用すればコンパイルおよび作成したプログラムの実行が可能である。そこでコンパイルの際に OS の bit 数を自動判定して、DOS エミュレータを適切に使用し、学生が特別に意識することなくプログラムをコンパイル出来るようなバッチファイルを作成した<sup>(4)</sup>。この結果、①②の問題は解決されているが、③については、OS 自体がハードウェアの直接制御を禁じているので、上述の DOS エミュレータを使用しても解決していなかった。これに対応するために、グラフ描画などの表示を実現する GNU-PLOT を導入したが、データ送信にパイプ処理が必要であることや、描画する際に数式で表記する必要があるので、操作が簡便ではなく、学生の利用は皆無であった。

一方、主として 16bit-OS 対応のアプリケーションを活用するために、グラフィック画面を描画するエミュレータ (DOSBOX)<sup>(6)</sup> が海外で利用されており、これを利用すれば③の問題が解決できる可能性が考えられた。

本研究では、この DOSBOX を導入して、64bit-OS 上で 16bit 時代のコンパイラを活用できるように C 言語プログラミング教育用システムを改良する。これを用いた実習の課題提出状況を分析して、このシステムの有効性を検討する。

## 2. ポータブル C 言語学習システムの構成

〈2-1〉 C 言語学習システムが求める要件 プログラミングの実習は、本学科の学生が共通の環境で取り組むために、情報検索室のコンピュータを使用する。しかしながらこれらのコンピュータは大学の共用設備であるので、学生が任意の時刻に使えることを保証しない。一方、実習時間外にも学生が自主的に進める場合もあるので、自宅や自習室など、任意のコンピュータで実習できることが望ましい。この観点から移動運用可能な記憶媒体上に C 言語の開発環境を用意するのが理想的であり、実習経費削減のために、無料で配布されている開発環境を、インターネット経由で入手するのが適切である。また、Windows のバージョンが異なっても動作可能であり、また PC のスペックがコンパイルなどに大幅に影響を与えないものが望ましい。

一方、C 言語のプログラムを Windows 上で作成可能で、グラフィック関数をサポートするコンパイラを Microsoft 社

や Borland 社が無料配布しており、Windows の API 関数や DirectX などを利用したプログラムも作成可能である。しかしながら、これらのコンパイラは必要な記憶容量が大きく、さらに環境定義ファイルを OS の管理するフォルダ内に配置するので、移動運用可能な記憶媒体上に設定するのが困難である。また、多くの医学系学生は本実習を除けば、基本的にプログラム開発環境を必要としないので、限定的な実習のために、大容量の開発環境を共用設備や学生個人のコンピュータに準備することは適切であるとは言い難い。

〈2-2〉 C 言語学習システムの構成 LSI ジャパン社製の LSI-C コンパイラ試食版は、インターネット経由で入手可能なフリーウェアである。300 行以内のソースファイルをコンパイル可能であり、ANSI で定義されている C 言語の関数を実装するので、入門用として適している。さらに、これと組み合わせるために一部の個人が開発した、グラフィックライブラリもフリーウェアとして提供されており、この利用により直線やだ円、長方形などの基本図形を描画する機能が利用できる。WindowsXP 以前の 32bit OS では制限なく動作するので、グラフや任意の CG を描画するなどの実習課題に対応できる。この組み合わせでは必要な記憶容量が 1MB 以下であるので、移動運用可能な記憶媒体上にシステムを容易に構築できる利点がある。実習書など既存のリソースを活用できる利点も考慮すれば、このシステムを継続利用するのが望ましい。

しかしながら 16bit の MS-DOS 用アプリケーションである LSI-C コンパイラおよびこのコンパイラで作成するプログラムは、32bit の Windows のコマンドプロンプトで動作するが 64bit の Windows では動作しないように制限されている。また、グラフィックライブラリを使用したプログラムは、PC の一部の制御コードに制限がかけられており、32bit の Windows であっても VISTA 以降の version では動作しない。

Windows Version	before XP		Compile/No Graphic	Graphic Use
			Command Prompt (Windows Accessory)	Command Prompt (Windows Accessory)
	After VISTA			
	32bit	Command Prompt (Windows Accessory)		DOSBOX (Freeware)
	64bit	MS-DOS Player (Freeware)		

図1 Windowsバージョンによるエミュレータの選択  
Fig.1 Selection of emulator by the versions of Windows.

ところで、64bit の Windows 上で 16bit のアプリケーション(プログラム)を動作するために作成された DOS エミュレ

ータの一つである MS-DOS Player もまたインターネット上でフリー配布されており、LSI-C コンパイラおよびこのコンパイラで作成するプログラムを動作することが可能である。グラフィック機能を利用することはできないが、グラフィックライブラリを利用するプログラムをコンパイルすることは可能である。また、フリー配布されている DOSBOX というエミュレータは動作するフォルダを定義ファイルで固定するので、コンパイルはできないが、事前にコンパイルされていれば、グラフィック機能を実装したプログラムを実行することが可能である。すなわち、これらのエミュレータを組み合わせると、フリーウェアの組み合わせとして 16bitOS で実現していた、グラフィック利用可能な C 言語学習システムを大幅に変更することなく 64bitOS で利用できる。すなわち、Windows のバージョンによらず、グラフィックライブラリを利用できる C 言語実習システムを、移動運用可能な記憶媒体上に構築可能である。

ところで学生がプログラムのコンパイル時に Windows のバージョンを意識し、必要に応じて MS-DOS Player を使用することは困難であるので、自動的にバージョンをチェックして、使用を自動的に選択するバッチファイルを作成した。これにより、学生はドライブレターや Windows のバージョンによらず一定の手順でコンパイルをおこなうことができる。このとき作成したプログラムも 16bit アプリケーションであるが、グラフィックライブラリを使用しない場合はコンパイラと同様に MS-DOS Player を使用すれば実行可能なので、コンパイルから連続して自動選択するように設定した。

一方、グラフィックライブラリ使用時には、学生自身がその利用を認識しているので、コンパイル後の実行ファイルを試行する際には、DOSBOX を学生自身が利用するように、手順をマニュアル化した。

**〈2・3〉 C 言語学習システムの使用手順** LSI-C コンパイラは、コンパイルする際にリンクするライブラリを相対パスではなく、絶対パスで指定しなければならない。この絶対パスは、上述した、コンパイル時に使用するバッチファイルの中に記述しなければならない。さらにコンパイラやリンクなど、使用するファイルのフォルダパスをコマンドプロンプトの環境変数で定義する必要がある。

また、LSI-C コンパイラと DOSBOX はそれぞれ動作に必要なパラメータを記述した設定ファイルを起動時に読み込んで動作する仕様であり、これらの設定ファイル内にドライブレターを指定する項目が存在する。

そこで、システムの入った記憶装置を PC に接続した際に設定されるドライブレターに対応するようなバッチファイルや設定ファイルをあらかじめ複数用意し、ドライブレターに対応して自動的にこれらを書き換えるためのバッチファイルを作成し、学生が、自分の環境に応じて決定したドライブレターに合わせたバッチファイルを起動して、その都度システムを再構成するように設定した。この手順は完

全自動化すると、ウイルス対策ソフトにより停止される可能性があるので、学生自身が準備段階として実施しなければならない項目とした。

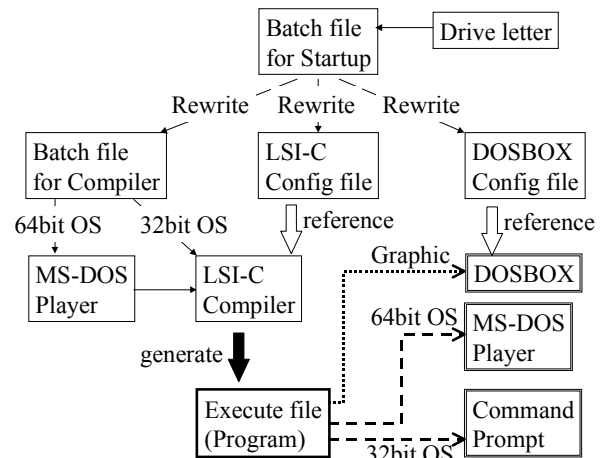


図2 C 言語学習システムの利用手順  
Fig.2 Procedure for using C-language learning system

**〈2・4〉 実習の手順** 実習の初回にあらかじめ作成したサンプルプログラムを用いて、本システムの基本的な操作を学生に指導したのち、学生自身が自分のペースでプログラミングを修得するように誘導する。具体的には C 言語の基本的な文法と例題を記載した独自作成の文法書を参照しながら、学生が基本的な関数の動作を理解し、これを組み合わせてプログラムを構築するように指導する。実習の成果を評価するために、種々のレベルの最終課題を 25 個用意しており、2~4 個の課題を学生自身が任意に選択し、作成・提出する。

実習時間だけでは課題を完成できない学生も存在するので、大学のみならず自宅などにもシステムを持ち帰り、学生自身の時間配分で学習と課題作成をおこなうことを推奨した。また文法についての不明な点は随時質問を受け付け、個別に対応している。

C 言語のプログラムはインターネット上にサンプルが豊富に存在し、あるいは上の学年の学生が同様の課題を経験しているため、一部の課題はこれらを手に入れたアレンジだけで容易に完成する。そこで最終課題の提出時には口頭試問をおこない、内容を理解していないと判断される場合には提出を認めないとした。

### 3. ポータブル C 言語学習システムの評価

**〈3・1〉 システムの動作に関する試験** 本システムを様々な Windows のバージョンごとに運用したところ、Windows 2000, XP, VISTA, 7, 8.1, 10 で動作を確認した。Windows 7 までの OS では 32bit での動作を確認しており、Windows 7 以降の OS では 64bit での動作を確認した。なお Windows 7 は 32bit, 64bit のいずれにおいても動作が確認できた。Fig.3 に示すのは実行している様子である。

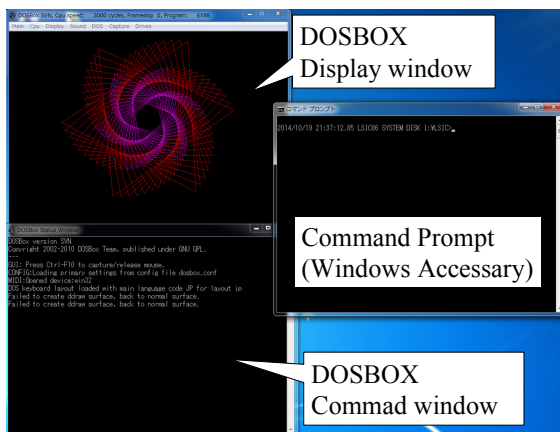


図3 システムを使用している様子

Fig.3 Overview of using the system.

〈3・2〉 学生の利用から見た評価 本システムを用いた実習で、2013年度から2016年度までの年度ごとの最終課題の提出比率を Fig.4 に示す。「タンクモデルの水位」(2-Tank model, 1-Tank model)と、「ボールの落下運動」(Bounding Ball)はグラフィック表示が必要な課題である。

DOSBOX の導入は2015年度からであるが、2014年度にボールの落下運動を、GNU-PLOT を用いてグラフ表示する提出が1件あったのみであったのに対し、2015年度以降はこれに加えてタンクモデルの課題提出があった。

グラフィック表示を必要としない課題は、年度ごとに提出割合が異なるものの、各年度で提出されている。このバラツキについて、一元配置の分散分析をおこなったが有意差は認められない。一方、グラフィック機能を用いる課題は、一元配置の分散分析で有意差が認められた。

このことから、DOSBOX を導入し、グラフィック機能を利用できるシステムにより、一部の学生はグラフィック機能を利用したプログラミングにも挑戦するようになったことが示されている。

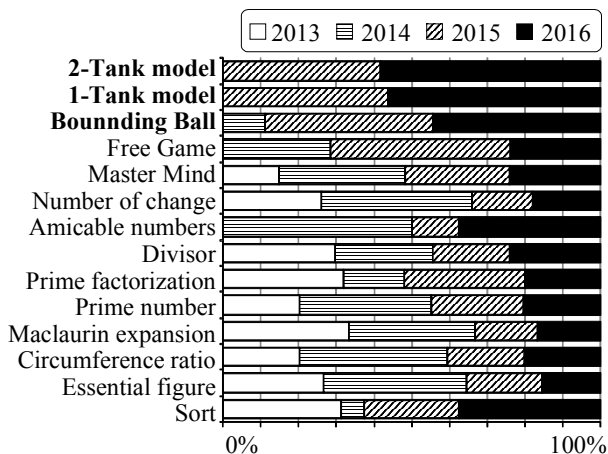


図4 年度ごとの課題提出の状況

Fig.4 Distribution of assignments submission by year.

#### 4. 考察

本研究ではインターネット経由で入手可能なフリーウェアを用いて、最新のOSにも対応可能な、グラフィックを使用可能なポータブルC言語開発環境を構築した。これを用いたプログラミング実習をおこない、必要十分な機能を有していること、および一部の学生がグラフィック機能に興味をもち、この機能を用いる課題に取り組むことを確認した。ポータブル可能な実習環境は、大学での実習時間のみならず、自宅などのPCでも自己学習が可能になるので、学生自身が自分のペースで納得するまでプログラミングの実習に取り組むことができる。

一方、基礎的なC言語を取り扱うプログラミング実習では、先輩からの示唆やインターネット経由で模範解答を入手可能なので、興味を持たない学生はこれらのコピーを提出し、プログラミングの技術を習得するに至らない場合もある。これを防ぐために一通りの口頭試問をおこなうが、医学部においては論理を構築する過程が重要であり、必ずしもプログラミング言語自体を習得する必要はないので、厳密な評価は困難である。適切な評価方法については、今後の検討課題である。

#### 5. おわりに

16bit OS で利用可能であったC言語のフリーコンパイラにエミュレータを組み合わせ、医学部学生でも容易に取り扱い可能な、プログラミング学習用のシステムを開発した。

本研究で使用したシステムは、コンパイラからエミュレータにいたるまでインターネット経由で入手可能なフリーウェアであり、利用には一切ライセンス料などが発生せず、また利用に関してユーザー登録なども必要としない。さらに最新のOSにも対応するので、長期間利用可能な実習環境であり、プログラミング実習への有効性が示された。

#### 文 献

- (1) Yukako HIRATSUKA, Tetsuo FIKUI : Effect of a Graphics Library that C Language Programming Education is Helped, Bull. Mukogawa Women's Univ., Nat. Sci.,44,75-80(1996) (in Japanese)  
平塚由花子, 福井哲夫: 「C言語プログラミング教育を支援するグラフィックライブラリーの効果」, 武庫川女子大紀要(自然科学), Vol.44, pp.75-80(1996)
- (2) エル・エス・アイ ジャパン(株): 「LSI C-86 v3.30c 試食版」, <http://www.vector.co.jp/soft/maker/lsi/se001169.html>
- (3) 小山佳孝: 「LSI-C 試食版用 簡易グラフィックライブラリ 1.093」, <http://www.vector.co.jp/soft/dos/prog/se243629.html>
- (4) Satoru HONMA, Hidetoshi WAKAMATSU : Programming exercise using the C-Language compiler on the removable storage medium, The Papers of Technical Meeting on Frontiers in Education, pp.13-16(FIE-13-003), (2013)(in Japanese)  
本間達, 若松秀俊: 「持ち運び可能なC言語学習環境を用いたプログラミング実習」, 電気学会教育フロンティア研究会資料, pp.13-16(FIE-13-003)(2013)
- (5) 武田俊也: 「MS-DOS Player for Win32-x64」, <http://homepage3.nifty.com/takeda-toshiya/msdos/>
- (6) GNU General Public License : 「DOSBOX」, <http://www.dosbox.com/>