

教養教育としての組み込み系プログラミング教材の開発と評価

岡本 雅子^{†1} 村上 正行^{†2}
吉川 直人^{†3} 喜多 一^{†4}

近年、工学系の専門教育のみならず Physical Computing としてデザインや趣味などの領域でも組み込み系への注目が高まっている。本研究では、実習用として開発されたマイコンボードを用いて、文科系学生を含む多様な受講生に対するコンピュータリテラシ教育の一環として組み込み系のプログラミングを取り上げた。本報告では、教材や実習内容について紹介する。本実践では、組み込み系の利用が学習動機を高めることやプログラミング言語の習得にも効果があることが示された。

Development and Evaluation of Embedded Programming Course Materials for Liberal Arts Education

MASAKO OKAMOTO,^{†1} MASAYUKI MURAKAMI,^{†2}
NAOTO YOSHIKAWA^{†3} and HAJIME KITA^{†4}

In recent years, attention is currently focused on embedded programming for not only professional education of engineering but also design or hobby as physical computing. We conduct embedded programming course for liberal arts students by using educational microcomputer kit as computer literacy course. This paper introduce to learning material of embedded programming and computer literacy course. The results of this study showed that the effect of enhancing learning motivation and acquisition of a programming language.

^{†1} 京都大学大学院情報学研究科

Graduate School of Informatics, Kyoto University

^{†2} 京都外国語大学 マルチメディア教育研究センター

Research Center for Multi-Media Education, Kyoto University of Foreign Studies

^{†3} 株式会社キャミー

CAMMY Co., Ltd

^{†4} 京都大学学術情報メディアセンター

1. はじめに

プログラミング教育に関する研究は、これまで、大学院、大学、専門学校、高等学校、企業などの場を対象として多くの実践が行われ、教授法や支援環境に関する研究が数多くなされてきた¹⁾。とりわけ、組み込み系システムの高機能化や複雑化にともない、大学における組み込み系プログラミング教育が注目されるようになった。組み込み系プログラミング教育は、特に技術者育成を目的とした工学系の専門教育として取り扱われることが多く、企業との産学連携により、質の高い授業が展開されている²⁾。

京都大学では、高等学校の普通科在籍時に教科「情報」の履修が義務付けられた新生を対象としたアンケート調査(平成18年度から平成21年度)を行っており、その結果をみると、大学における学習ニーズとして、特にプログラミングへの高い学習意欲が見られている³⁾。このように文系理系問わず多くの学生が興味を抱く一方で、教科「情報」の科目「情報B」においてプログラミング教育が扱われているものの、実際には学習機会が十分に与えられていないのではないかと考えられる。

京都大学の学部1・2回生を対象とした全学共通科目(平成20年度)では、前期586科目、後期387科目が開講されている⁴⁾。そのうち、情報系の科目は、基礎情報処理などの科目として主に各学部単位で授業が実施されている。その授業内容は、学部や担当教員ごとに異なるが、コンピュータの基本操作やオフィスソフトの操作方法などいわゆるコンピュータリテラシ教育に重点が置かれることが多く、プログラミングが扱われることは少ない。喫緊の課題である情報倫理やセキュリティ教育が優先され、一般教育段階でプログラミングをカリキュラムとして組み込む余裕がないためと考えられる。それに対して、理工系の学部では、学部科目でプログラミングを扱っているが、工学部情報学科を除けば数値計算に重点が置かれていることが一般的である。なお、学部によらず履修できる全学共通科目のうち、科目名に「プログラミング」が含まれているものは、前期と後期合わせて5科目が提供されている。プログラミングを半期の授業として扱うとすると、プログラミングの導入部分だけを短期間に学習したい学生にとっては、時間的・精神的負担が大きくなりすぎる可能性がある。

そこで本研究では、上述のような学生の学習ニーズを踏まえ、技術者の養成を目指すのではなく、プログラムとは何かを理解し、プログラミングの面白さを体感することを目的とした。そのための導入教育として、教養教育としての情報系科目の中でプログラミングの入門

編を扱うことを検討した。

初学者を対象としたプログラミングの授業では、入力ミスによるエラーや文法エラーを自力で解決できないことなどによって、学習者の学習意欲が低下してしまうことがよく見受けられる⁵⁾。このような場合、学習者にプログラミングへの関心はあっても、エラーへの対処ができないことが理解度や学習意欲の低下を招いていると言える。エラーを自力で解決できないひとつの理由として、プログラミングと結果との関係が直感的に理解しがたいことが挙げられる。パソコン上でコンパイル・実行するプログラミングに対して、「組み込み系」プログラミングでは、プログラミングの結果として装置が稼働の様子を見ることができる。実際、小学校などで児童を対象としたプログラミング教育では、モーターが“動く”ことやランプが“光る”ことなど、明確な反応を視覚的に観察することができるため、「組み込み系」プログラミングが取り入れられることがある^{6),7)}。また、近年、工学系の大学や工業高等専門学校では、企業との産学連携により、専門教育として「組み込み系」プログラミングを取り入れ、質の高い授業が展開されている⁸⁾。組み込み系プログラミング教育を取り入れた授業実践では、プログラミングの知識の定着や理解の向上には課題があるものの、プログラミングに対する興味の喚起や学習意欲の触発には効果があったとする報告がなされている⁹⁾。このように、「組み込み系」プログラミングは視覚的に理解しやすく、実際に動作が行われることによって、学習者の学習意欲を維持できる可能性がある。

さらに、近年ではデザインや趣味としても組み込み系を利用する Physical Computing の動きがあり、入門書なども出版され始めている^{10),11)}。このような特徴を持つ「組み込み系」プログラミングであるが、高等教育における教養教育を目的としたプログラミング学習に活用された例は少ない。

そこで本研究では、興味の喚起や学習意欲の触発といった組み込み系プログラミング教育の有効性に着目し、文科系学生を含む多様な受講生に対するコンピュータリテラシー教育の一環としてカリキュラムと教材の開発を行った上で授業実践を行った。

2. 使用教材と科目構成

2.1 使用教材

演習には、キヤミーで組み込み技術者養成用に試作したマイコンボード(図1)を教材として利用した。

このボードには、H8 マイクロプロセッサのほか、プッシュスイッチ、DIP スイッチなどが入力デバイスとして、また LED、LCD、モーター(扇風機に見立てるためのファンが付け

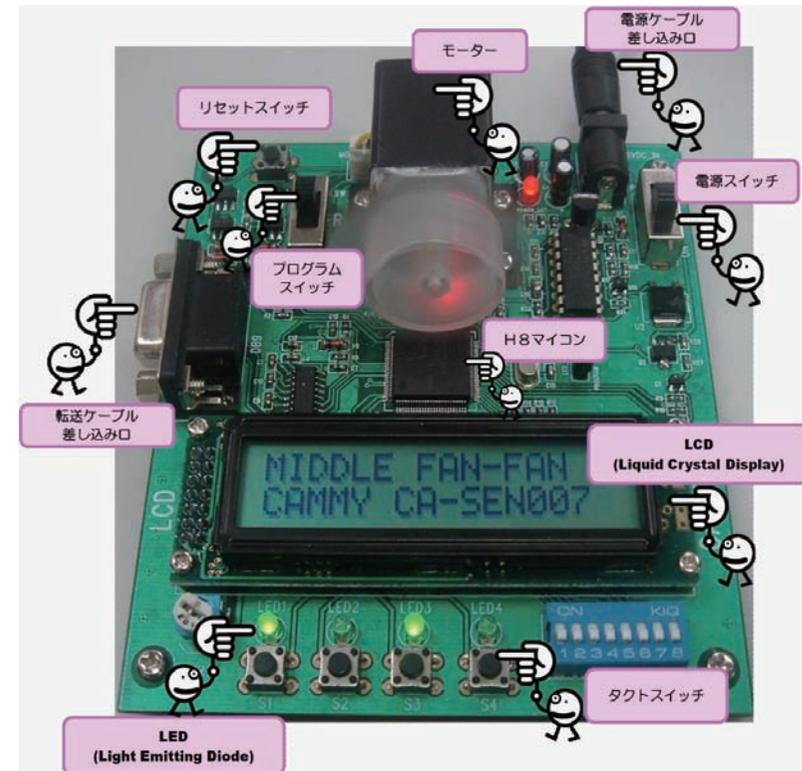


図1 授業実践に使用したマイコンボード教材

られている)などが出力デバイスとして予めボード上に実装されている。ホストコンピュータ(京都大学の教育用コンピュータ)とはシリアルラインで接続する。プログラムには、統合開発環境 GCC Developer Lite を用いて C 言語のソースコードをクロスコンパイルしたのち、マイコンボードに実行プログラムをダウンロードとして動作させる。入出力操作については、ライブラリを用意した。C 言語のテキスト教材は、これまでの教材作成のノウハウを踏まえつつ、マイコンの操作や本教材で用いるライブラリ呼び出しの事例を含めて独自に開発したものを利用した。

2.2 写経型学習

従来、プログラミングの習得法として、プログラマたちが実践してきた方法は、良いプログラムを「模倣」することを繰り返すことであった。その模倣の過程でプログラミングに必要な文法の知識や概念を理解し、その後、より複雑なプログラムを作成する中で「応用」していくという段階を経ることで技術を身につけてきた。

高本ら¹³⁾は、初心者が、プログラミング言語を理解し自分の考えに基づいて問題解決が行えるようになるまでを、規範プログラムの模倣段階、自分なりにプログラミング言語を理解する段階、問題のアルゴリズムを理解しプログラム化できる段階の3つの段階に分けて捉えることが可能であるとしている。本研究ではこの模倣段階に着目し、テキストの例題をすべて入力し、実行すること(以下、「写経型学習」と呼ぶ¹²⁾)を求めたうえで、プログラムの解説を理解するように組み立てられている。

本研究では、初学者を対象としたプログラミング教育における学習者の躓きや学習段階の調査結果を踏まえ¹²⁾、新たにテキスト教材を以下の方針を基に開発した。なお、この方針は、従来のプログラミング言語のテキストと比較して、プログラミング言語の文法面を網羅的に紹介することよりもむしろ、学習者の学習プロセスと実践的なプログラムやプログラミングプロセスにより配慮したものとなっている。

- (1) 学習の目的、学習の仕方など、学ぶ内容とともに「学び方」も明示する。
- (2) 音読への配慮：コミュニケーションの円滑化を目的に、プログラミングで使用する記号には読み方を、予約語や良く使われる表現には、読み方および適切な日本語訳を表記した。また、英語を語源とする短縮形が多用されていることから、語源とその読み方も表記した。学習者が読めない記号や単語があった場合、一覧表の中からそれらの読み方を探せるようになっている。
- (3) プログラミングの基本概念の理解への支援：本テキストでは、写経型学習での自習を前提に、サンプルプログラムについては実行可能な完全なソースコードと、その実行結果を必ず示す形で用意した。そして、サンプルプログラムを対象に、その逐行的解説とプログラム構成要素の紹介もあわせて行った。プログラミング言語の文法については、実践的に利用するポイントに焦点を当てて解説した。また、プログラムの動作の理解を助けるために図解なども重視した。
- (4) プログラミングのプロセス面での学習の支援：市販のテキストでは、あまり解説されない落とし穴や作法など、プログラミングという活動のプロセス面を意識し、「コラム」として紹介した。また、デバッグでの躓きを軽減するために、補足資料として、

表 1 授業内容

授業回数	内容
1回目	開発環境の説明, LCD と LED の制御
2回目	繰り返し処理・for 文
3回目	条件分岐・if 文
4回目	繰り返し処理・while 文
5回目	まとめ

いくつかの典型的なエラーについては、エラーメッセージとともに対処方法も紹介した。

- (5) 本テキストは、それを見ながらコンピュータに向かって実習することに配慮し、見開き2ページで完結する内容とし、1つのサンプルプログラムに複数の学習内容を詰め込み過ぎないように配慮した。

2.3 対象科目と授業の構成

2010年度京都大学全学共通科目「情報科学演習」において、文科系学生を含む多様な受講生に対するコンピュータリテラシ教育の一環として、上述の教材を用いて、組み込み系のプログラミングを取り上げた。

本授業は、前期に週一コマの科目として全15回で開講されており、そのうちの5回の授業を本実践であるプログラミングの学習にあてている。本授業では、プログラミングを教授する目的として、アルゴリズムやプログラム言語自体の習得を目的とするのではなく、プログラムにより動くこと(システムの内部構造)を認識するための最低限の制御構造として「条件分岐」、「繰り返し」などを理解させ、ものが動くことを体験させることを目指す。上述の趣旨を踏まえ、この科目では、プログラムの動作のわかりやすいC言語を選択している。学習用に開発された言語の採用なども考えられるが、授業開始時に「どのようなプログラミング言語を学びたいか」と問うたところ、「聞いたことのあるC言語を学習してみたい」との要望も多く、C言語の採用は学習の動機づけにもなると考えられる。

プログラミングの授業は、プログラミングの基本的な概念の習得を目的として計5回分実施した(表 A.1)。学習する内容は時間が限られていることから、プログラミングにおけるトピックとしては変数と代入、および制御構造(if文とfor文)を中心とし、ライブラリ等の呼び出しを除いて関数は扱っていない。教材の設計意図を踏まえ、同科目では、講義と写経型学習による自習を取り入れた授業形態を採用し、講義での解説はポイントを絞った解説のみを行い、受講生は教材に従って、各々の学習進度に応じて学習を進めた。学生が躓いた場合は、教員やTA(1名)がサポートに入り、直接指導を実施した。なお、授業は、教員1

表 2 プログラミングの必要性

項目	割合
必要	20%
ある程度必要 2	60%
人に助けをもらうつもり	3%
不要	14%
わからない	3%

名と TA1 名で行ったが、このほかマイコンボードを開発した株式会社キヤミーの社員がトラブル等の対応にあたった。また、マイコンボードの貸出しは行わなかったため、プログラミングは授業時間中のみ行える環境での授業となった。

3. 実践結果

3.1 受講生の前提知識

「情報科学演習」は、受講生が 35 名、総合人間 (17 名)、経済 (6 名)、法学 (4 名)、文学 (4 名)、教育学 (3 名)、理学 (1 名) の学生であった。

プログラミングの授業開始前に、受講生のプログラミング経験を把握するためのアンケートを実施した。このアンケートではプログラミングの経験のほか、既習のプログラミング言語や概念、実装したプログラムの規模などを回答させたが、受講生は、プログラミング経験のないものが 28 名 (以下「プログラミング初学者」と呼ぶ)、プログラミングの経験があるものが 7 名であった。

また、プログラミングの授業に対する意欲を知る目的で、「これから授業や研究などでプログラミング能力は必要だと思いますか」と問うたところ、約 8 割の学生が必要を感じる と答えた (表 2)。これは別途、実施している新入生を対象としたアンケート³⁾においてプログラミングへの学習ニーズが高いことと整合している。

3.2 授業実践の評価

3.2.1 組み込みプログラミングに対する評価

組み込みプログラミングの授業と教材、学習の難易度に関する評価を行うため、授業の最終日にアンケートを実施した。本研究では、プログラミング初学者を対象としているため、プログラミング経験のない 28 名と経験者を分けて分析を行った。その集計結果を図 2、図 3、図 4 に示す。以下、主としてプログラミング初学者に注目して考察する。

組み込みプログラミングの授業の満足度に関しては、約 9 割もの学生が肯定的な評価を行っている。このほか、全体的な理解度やプログラミングへの興味、今後の学習への意欲に

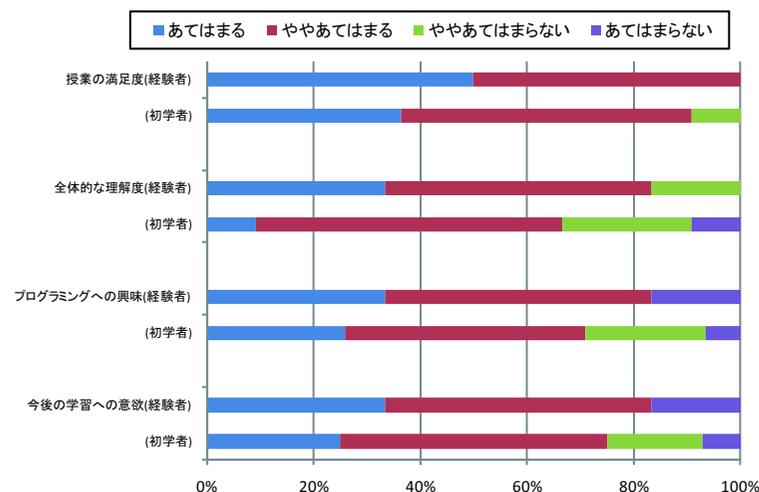


図 2 授業全体への評価

についても高い評価を得た (図 2 参照)。

また、組み込みプログラミングに関しては、約 8 割の学生が「おもしろかった」「ややおもしろかった」と肯定的な評価を示している (図 4 参照)。自由記述の中には、「視覚的にプログラムの仕組みが分かるのが良かった」「一つの製品を作る感じがおもしろかった」「入力した内容によってモーターが動いたり LED が光ったりしておもしろかった」「扇風機がこのような仕組みで動かされているというがわかって面白かったです」などの意見があった。

これらの意見は、プログラミングとその動作との関連が理解できたことで、学習意欲の維持につながったことと解釈できる。これらのことから、組み込みプログラミングは、先行研究で示されていた通り、学習の動機付けとして適していること、その理由としてプログラムの動作が具体的で可視化されることが推察できる。

一方で、「学習した内容が実生活に役立ちそうにないのが残念」「自分でいろいろと応用がきくものが良かった」「LED、LCD、ファン以外の機能が欲しかった」などの意見も見られ、教材の汎用性という面の課題も示された。

3.2.2 組み込みプログラミングによる理解

プログラミングに対する難易度に関する調査結果では、ソースコードを入力し、ボードに

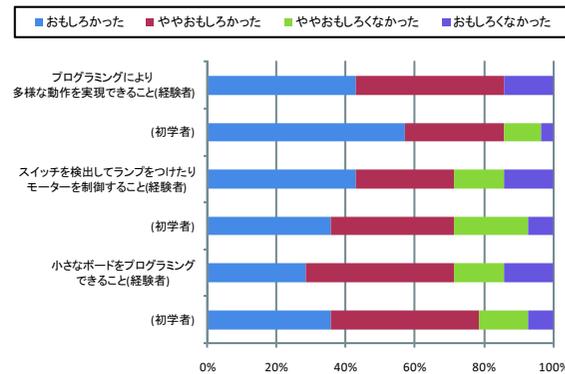


図 3 組み込みプログラミングに対する評価

転送（機器接続のための設定を含む）、コンパイル、実行することについては、「易しかった」「やや易しかった」とする回答が多く、難易度は低かったようである（図 3）。一方、エラーの発見に関しては、「難しかった」「やや難しかった」と回答する学生が多かった。自由記述から「慣れるまでエラーを探すのに時間がかかった」などの意見が見られた。プログラミングの概念として、変数や代入、繰り返し処理に関しては、「易しかった」「やや易しかった」とする回答が多かった。

最後に、受講生のプログラミングについての理解度を測るため、C 言語についてのテスト（成績評価には反映しない）ものとして（「理解度アンケート」と呼ぶ）を実施した。

同アンケートは、

- 1) 文法エラーの間違い探し (5 つ) ,
- 2) 変数, 出力に関する問題 (プログラムの記述) ,
- 3) if 文に関する問題 (プログラムの記述) ,
- 4) for 文に関する問題 (誤り箇所が発見と修正) から構成される (付録参照) .

回答は 35 点満点で計算し、プログラミング初学者の平均は 32.6 点 (全体平均 32.3 点) であった。

アンケート結果を見ると 5 回という短い回数での授業にも関わらず全般に高い点数が得られている。項目別に見ると 1) の文法エラー, 2) 変数と出力, 3) if 文に関しては、全体的に理解度が高かった。一方, 4) for 文に関しては、全員が回答していたが、2 割の学生が問題

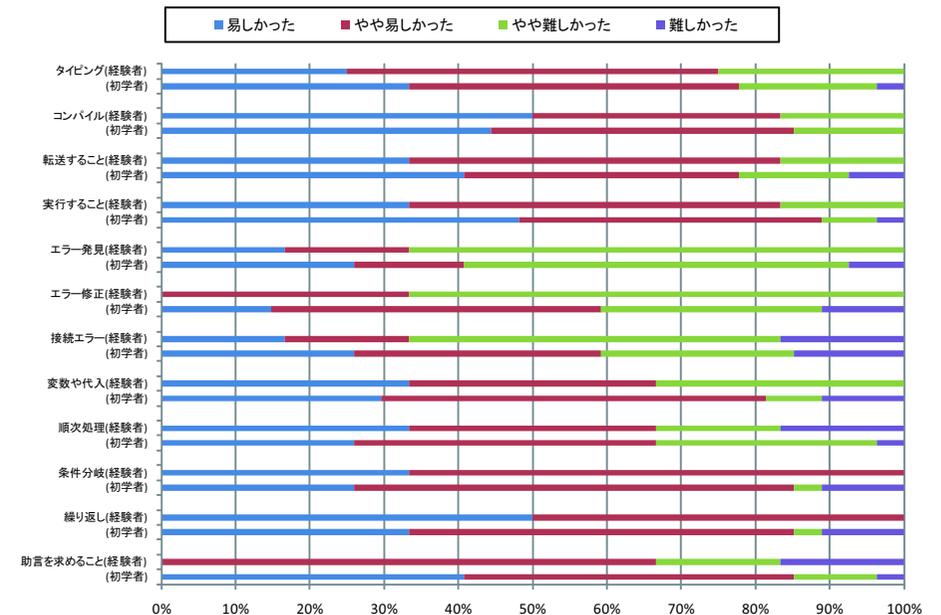


図 4 プログラミングに対する難易度

のソースコードを読み解くことができず、誤り箇所を指摘できていなかった。初学者にとって for 文の理解はプログラミング学習の一つの関門であり、アンケート結果もそれに準ずるが、正答率はかなり高い。

また、昨年度の同一科目の授業では、パソコン上でコンパイル・実行する C 言語の授業を同様の方針で作成した教材とプログラミング言語の内容で実施し、同じようなアンケートを実施した結果では、プログラミング初学者の得点は 22.3 点であった。昨年度と今年度の平均値間の差について t 検定を行ったところ、0.1%水準で有意差が見られた ($t(59) = 5.249, p < .001$)。

これらの結果も組み込み系の利用が初学者のプログラミング学習に適している可能性を示唆している。

4. おわりに

本論文では、実習用として開発されたマイコンキットを用いて、文科系学生を含む多様な

受講生に対するコンピュータリテラシ教育の一環として、組み込み系のプログラミングを取り上げた。京都大学における 2010 年度全学共通科目「情報科学演習」において同教材の評価を行った。その結果、組み込み系のプログラミング初学者に対する学習効果が見られ、学習への動機づけにも効果的であることやプログラミング言語の機能の習得についても高い効果を持つことが示唆された。

今後の課題として、プログラミングの概念などの深い理解に対する配慮や応用力を養成できる教材の開発、オープンユースの自習用端末や自宅のパソコンを利用した自習への対応が必要である。

参 考 文 献

- 1) A. Pears et al.: A survey of literature on the teaching of introductory programming, ACM SIGCSE Bulletin, Vol.39, Issue.4 (2007).
- 2) 土井滋貴ほか:「元気なら組み込みシステム技術者の養成」の取り組みについて,平成 21 年度情報教育研究会講演論文集, pp.497-498 (2009).
- 3) 森幹彦ほか: 情報教育に関する大学新入生の状況変化 京都大学新入生アンケートの結果から, 情報処理学会論文誌, Vol.51, No.10, pp.1961-1973 (2010).
- 4) 京都大学高等教育研究開発推進機構編:京都大学「全学共通科目授業内容」(平成 20 年度版), 京都大学高等教育研究開発推進機構 (2008).
- 5) 安達一寿, 中尾茂子: プログラミング学習における学生をつまづき箇所の分析, 日本教育情報学会会誌, Vol.10, No.4, pp.11-20 (1995).
- 6) 福田哲也: 大学生と中学生による小学生のためのロボット教室, 教育実践総合センター研究紀要, Vol.19, pp.129-134 (2010).
- 7) 田代久美ほか: ロボットを用いた小学校におけるプログラミング教育の研究 教育用ロボット「梵天丸」「いろは姫」の仙台市における活用事例から, 電子情報通信学会信学技報, No.30, pp.49-52 (2006).
- 8) 浅井文男: 産学連携によるマイコンプログラミング教育の試み, 平成 21 年度情報教育研究会講演論文集, pp.161-164 (2009).
- 9) 渡邊景子ほか: サッカーロボット製作による導入教育, 電子情報通信学会技術研究報告, No.106, Vol.166, pp.11-16 (2006).
- 10) 小林茂: Prototyping Lab, オライリージャパン (2010).
- 11) M. Banzi 著, 船田訳: Arduino をはじめよう, オライリージャパン (2009).
- 12) 岡本雅子ほか: 初学者を対象とした自習中心のプログラミング教育の教材開発と評価, 情報教育シンポジウム SSS2010 論文集, Vol.2010, pp.87-94 (2010).
- 13) 高本明美, 藤井美知子: 誤り分析にもとづくプログラミング学習の支援, 電子情報通信学会技術研究報告, No.94, Vol.32, pp.23-30 (1994).

付 録

A.1 理解度アンケート

(1) 次のプログラムは、「MachigaiSagashi」という文字列を LCD(液晶パネル)に表示するプログラムです。間違っている箇所が 5ヶ所存在します。それをすべて指摘しなさい。

誤っているプログラム

```
1: %include<stdio.h>
2:
3: int main(void)
4: {
5:     api_InitBoard();
6:     api_ClearLCD();
7:     api_PrintLCD(0, 0, "MachigaiSagashi");
8:
9:     return 0;
10:
11:
12:
```

(2) 次のプログラムは、LCD(液晶パネル)の下段の左から 6 番目に「*」を表示するプログラムです。空欄に相当するプログラムを記述しなさい。

プログラムリスト

```
1: #include<system.h>
2: int main(void)
3: {
4:
5:
6:     return 0;
7: }
```

(3) 次のページのプログラムは、各スイッチを押下した際にLEDランプを点灯させるプログラムである。プログラムの空欄を埋めなさい。

スイッチ	点灯させる LED ランプ
1	4のみ
2	3のみ
上述以外	1-4全部

プログラムリスト

```

1: #include<system.h>
2: int main(void)
3: {
4:     int swno ;
5:
6: api_InitBoard();
7: api_ClearLED();
8:
9: swno = api_Wait_Switch_On();
    
```

```

    api_Wait_Switch_Off(swno) ;
    
```

```

return 0 ;
}
    
```

(4) 次のプログラムは、LCD(液晶パネル)の上段に「*」のみを1秒ごとに1マスずつずらして表示するプログラムである。このプログラムについて以下の問題に答えなさい。

プログラムリスト

```

1: #include<system.h>
2: int main(void)
3: {
4:     int i;
5:
6:     api_InitBoard();
7:     api_ClearLCD();
8:
9:     for(i = 0; i < 16; i++){
10:         api_PrintfLCD(i, 9, "*");
11:         _system_Wait(1000);
12:         api_PrintLCD(i, 0, "");
13:     }
14:
15:     return 0;
16: }
    
```

(a) 上のプログラムには、文法の誤りが1ヶ所存在する。その行番号と誤りを訂正したプログラムを記述しなさい。

行番号	訂正したプログラム

(b) (a) の部分を修正すれば文法上は正しいプログラムとなるが、他に1ヶ所誤りがあるため、期待した結果(「」のみを順番に出力する)を出力することができない。誤りのある行番号と修正したプログラムを記述しなさい。

行番号	訂正したプログラム