

動機づけと達成度を保証するための プログラミング入門科目の設計

佐々木 晃^{1,a)} 伊藤 克巨^{1,b)} 荒川 傑^{2,c)}

受付日 2013年3月19日, 採録日 2013年10月9日

概要: 本論文では, プログラミング入門科目の新しい構成法について述べる. プログラミングの導入科目のように, それまで経験のない事柄を新たに学生に積極的に取り組ませるには, 動機づけが欠かせない. 動機づけには, 学生が興味を持つ題材を演習に取り上げるのが効果的である. しかし, コンピュータ科学の応用範囲の拡大にともない, 学生の興味は多岐にわたっている. このような状況をふまえ, 本論文では, 仮想小クラス制 (VSC: Virtual Small Group Class) を提案する. VSC とは, 現実の様々な適用範囲に対応して, 学生が各自の興味ごとに異なるプログラミング課題を選択して挑戦できるようにするしくみである. 一方で, このように学生の選択の自由度を増すと, 選択によって学生の練度にばらつきが生じるおそれがある. そこで, 最低限の練度を担保するためのしくみとして, 認定試験 (MT: Mastery Test) を提案する. MT は期末試験とは別に行う試験である. 筆者らは, 2010 年度から所属する理系の情報系学部向けに, 提案手法に基づくプログラミング入門科目の再構成に着手した. 本論文では, VSC と MT を含むカリキュラムの構成法について述べ, さらに筆者らの取り組みと, それを通して VSC によって学生を動機づけられたこと, また MT によって最低限の学力が担保できたことを示す.

キーワード: プログラミング教育, 達成度評価, プログラミング導入科目

Design of an Introductory Programming Course Assuring Motivation and Mastery

AKIRA SASAKI^{1,a)} KATUNOBU ITOU^{1,b)} SUGURU ARAKAWA^{2,c)}

Received: March 19, 2013, Accepted: October 9, 2013

Abstract: We propose a new design of an introductory programming course such as CS1 (the first course of a computer science major, term used in the ACM Computing Curricula). For such courses, which treat entirely new topics for students, it is essential to motivate them and to retain their attention. One approach to motivating students is to select practical exercise themes which are familiar to them. However, students' interest tends to diverge because of the broader application of CS technology. With this background, we propose "virtual small group classes" (VSCs). In VSCs, students can select and then challenge practical laboratory programming work from various domain, based on their own interest. One concern about this approach is variance in their attainment level, which stems from the fact that VSC allows students some degree of freedom in selecting work. For the sake of solving this problem, we propose MT (Mastery Test) designed to assure students' acquiring fundamental knowledge and skills expected in CS1. The attainment of the mastery test is separated from their grading evaluated in the term examination. Our faculty started to reconstruct the curriculum of CS1 in 2010. Our experience of VSCs and MT indicates that VSCs efficiently motivate students and encourage their engagement in programming on one side, while ensuring the minimal levels of skills required by MT.

Keywords: computer programming education, assessment of learning attainment, introductory programming course

1. はじめに

大学の情報系の学科において、プログラミング入門科目は非常に重要である。新入生の多くは本格的なプログラミング経験がない。したがって、この科目でつまずくと、コンピュータ自体が嫌いになってしまうことすらある。また、コンピュータ科学の専門科目の多くは、プログラミング演習をとまなう。したがって、プログラミング入門科目の学習項目を習得していないと後続科目の効率的な実施が困難になる。

このような導入科目を学生に積極的に取り組ませるには、動機づけが欠かせない。動機づけには、学生が興味を持つ題材を演習に取り上げるのが効果的である。しかし、コンピュータ科学の応用範囲の拡大にとまなう、学生の興味は多岐にわたっている。

このような状況をふまえ、本論文では、仮想小クラス制 (VSC: Virtual Small Group Class) を提案する。VSCとは、現実の様々な適用範囲に対応して、学生の興味ごとに異なるプログラミング課題に挑戦できるようにするしくみである。このしくみでは、VSCごとにグラフィクスやスプレッドシートなど別々のテーマを設定し、それぞれの演習課題を用意する。しかし、基礎となるプログラミングの知識やスキルはすべてのVSCで共通とする。学生は、参加するVSCを選択し、自分の興味に合わせてプログラミングを学ぶ。

このように学生の選択の自由度を増すと、選択によって学生の練度にばらつきが生じるおそれがある。しかし、そのような状況は、プログラミング入門科目のように後続の科目の基礎となる場合、特に好ましくない。

そこで、最低限の練度を担保するためのしくみとして、認定試験 (MT: Mastery Test) を提案する。MTは期末試験とは別に行う試験である。試験は当該科目を通じて共通して確実に学んで身につけるべき項目で構成する。すべての受講生が合格できるように、簡単な問題で構成する。その代わり、合格基準を高く設定し、合格しない場合には、複数回の受験を認める。

本論文では、まず、対象となるプログラミング言語 Java による導入科目「プログラミング入門1」の授業構成について、VSC, MT 導入以前の 2009 年以前の状況と比較して述べる。次に、2010 年に導入した VSC について述べ、VSC によって学生を動機づけてきたことを示す。次に、2011 年

に導入した MT について述べ、MT によって、最低限の学力が担保できたことを示す。

2. 授業構成

本論文の対象となる科目の概要について述べる。理系の情報系学部の 1 年次の春学期の必修科目である。ほとんどの学生は、この科目で初めてプログラミングを学ぶ。1 クラス 40 名程度で 4 クラス設置されている。講師はそれぞれのクラスで異なるが、教材、試験は共通である。

90 分を 2 コマ連続して行う授業が 14 回から構成される。最初の 90 分は、プログラミングに関する知識を教える講義形式の授業である。後半の 90 分は演習であり、学生は各自課題に取り組む。TA (2 名) や教員は、質問に対応したり、課題を確認したりする。この演習時間に終わらない課題は、次週までに授業支援システム上でオンライン提出させる。

授業で使用する教材はすべてオンラインで公開している。オンライン教材は、前半の「講義」の内容に関する解説と仮想小クラス制 (VSC) のテーマに関する内容説明、VSC の演習課題から構成される。授業を効果的に進めるために、自宅学習を推進する。授業前に、オンライン教材を読んで、例題を実行させて答えさせる予習課題を義務として課する。なお本学部では、全入学生にノートパソコンが貸与される。

成績は、主に期末試験 (90 分) の結果に基づき、A+ (90 点以上)、A (80 点以上)、B (70 点以上)、C (60 点以上)、D (不合格) の 5 段階で評価する。

2011 年度の授業計画を表 1 に示す。2010 年度も MT を除けば同一である。内容は J07 [1] を意識したものであり、ごく標準的なものである。しかし、通常の文法指向の授業計画と比べると、4 回目という早い段階で手続き (メソッド) を取り上げている。これは、メソッドの練度を上げることを意識した設計である。2009 年以前は、メソッドを第 7 回目に取り上げていた。2009 年の期末試験の大問ごとの正答率を表 2 に示す。この表から分かるように、メソッドを含んだ問題の得点率が特に中低位層で低い。しかし、後続科目で、最も必要となるものの 1 つがメソッドである。また、この科目の範囲では、それほど難易度が高いとも思えない。そこで、反復して練習する回数を増やすことで、メソッドの理解度、定着度を向上させることにした。考える最も早い回でメソッドを取り上げるように設計を変更し、第 4 回目で取り上げるようになった。

2009 年度までの授業では、2010 年度以降と同様にスライドを利用した講義を前半、演習を後半に行い、演習で終了しなかった部分は課題としていた。評価は期末試験のみであった。プログラム課題は、授業ごとに全員が同一の課題に取り組み、一部の課題は力がある学生向けへの応用問題であった。授業計画は、多くのプログラミング言語の入

¹ 法政大学情報科学部
Faculty of Computer and Information Sciences, Hosei University, Koganei, Tokyo 184-8584, Japan

² 株式会社ノーテラス・テクノロジーズ
NAUTILUS Technologies, Inc., Shinagawa, Tokyo 140-0001, Japan

a) asasaki@hosei.ac.jp

b) itou@hosei.ac.jp

c) arakawa@nautilus-technologies.com

表 1 授業計画 (2010, 2011 年度)
Table 1 Curriculum of CS1.

	内容	試験(2011 年のみ)
1	導入, 入出力	
2	式と変数	
3	制御(if 文)	
4	サブルーチン (メソッド)	
5	復習(1)	
6	関数	
7	配列	
8	繰り返し	
9	複合データ (構造体)	
10	復習(2)	MT #1
11	対話型プログラミング	
12	実体と参照	MT #2
13	復習(3)	
14	復習(4)	MT #3
		評定試験

表 2 2009 年 (提案手法導入前) の期末試験の得点率

Table 2 Distribution of the score of term examination in 2009.

	正答率 (%)	正答率 (%) (中低位層)	メソッド	出題範囲
Q1	86.0	91.7	無	繰り返し
Q2	95.5	95.3	無	制御構造
Q3	76.0	80.6	無	入れ子の繰り返し
Q4	57.5	38.3	有	メソッドの書き方
Q5	50.0	43.9	無	複雑な繰り返し
Q6	54.3	48.6	有	プログラムの読解
Q7	54.7	37.2	有	メソッドの宣言
Q8	64.7	60.3	無	配列要素の抽出
Q9	45.3	33.6	無	2次元配列 (応用)

門書や解説書などの項目の順序を基本としたものである。以上のように、2009 年度以前の当該講義は、演習や実習を含む理系科目の典型的な講義方式であるとともに、プログラミングの導入科目としても一般的なものであると考えられる。

3. カリキュラム構成と仮想小クラス制

3.1 カリキュラムの構成と教材設計

カリキュラム構成は、まず講義全体をつうじて学生がどのような技能や知識を身につけるべきかの目標を、部分目標の集合として定義する。これをカリキュラムとして構築するために、1 単元を表す「レッスン」と、その内部の章立てを表す「チャック」という単位に再構成した。1 レッ

表 3 仮想小クラス制のコースと主な内容

Table 3 Courses and contents of Virtual Small Group Class.

Plain	標準的な入門書で扱われるような問題集
Spreadsheet	基礎的な統計処理, 表計算, バーチャルマシン
Web	ネットワークと Web アプリケーション
Canvas	グラフィクス, アニメーション, GUI
Sound	デジタル信号処理, 音声データ処理 (音声合成)

スは、50~70 分程度の分量であり、その中に 3~5 程度チャックを配置する。場合によって、2 以上のチャックで 1 つの部分目標を構成する場合もある。

この構成法はインストラクショナルデザインにおけるカリキュラム構成法 [2] を参考とし、学習目標を適切な粒度で明確化することに貢献している。3.2 節で述べる仮想小クラス制 (VSC) では、同一の学習項目に対して別テーマによるプログラミングの課題を設定するが、これらの並列で設定する複数課題間での目標の揺れをなくすために、本構成法は不可欠である。特に、2011 年度からは、Sound (音声を扱う) および Web アプリケーションの 2 コースがあらたに VSC のテーマとして追加されたが、レッスンおよび部分目標で設定した学習目標に従って各回の課題は設計された。以下は、第 8 回目の講義計画の構成例である。

レッスン 07 繰り返し (第 8 回目の講義分)	
部分目標 (習得すべき項目)	チャック (講義の構成)
3 種の制御構造について説明できる	7-1 (復習) 3 種類の制御構造
任意の回数だけ同じ処理を繰り返すプログラムを書ける	7-2 同じダイアログを N 回表示する
	7-3 同じダイアログを入力された回数だけ表示する
繰り返し変数を利用するプログラムを書ける	7-4 ループ変数を表示する
配列要素を反復し、集計するプログラムを書ける	7-5 配列要素を反復する
	7-6 配列要素を集計する

3.2 仮想小クラス制の設計

仮想小クラス制 (VSC: Virtual Small Group Class) とは、現実の様々な適用範囲に対応して、学生の興味ごとに異なるプログラミング課題に挑戦できるようにするしくみである。2011 年度に設定したテーマは、表 3 の 5 種類である。このしくみでは、各テーマに別々の演習課題を用意する。しかし、基礎となるプログラミングの知識やスキルはすべての VSC で共通である。学生は、参加する VSC を

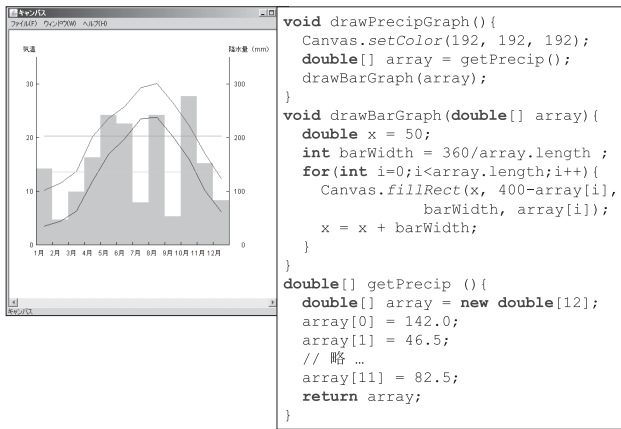


図 1 Canvas のプログラム例とスナップショット
 Fig. 1 Program and snapshot from Canvas course.

選択し、自分の興味に合わせてプログラミングを学ぶ。

VSC の特徴は、テーマに沿ったプロジェクト型の学習を、導入時教育に組み込める点である。言語機能の利用法などのスキルを獲得させるとともに、問題解決の手段である面も徐々に学ばせる必要があるが、仮想的なコースを通して後者の面を強調できる利点がある。前述のとおり VSC では、学生がコースを選択する点が 1 つの特徴であるが、1 ないし数回の講義内で選択問題に取り組みせるのではなく、ほぼ講義の全期間にわたって選択したテーマの課題を継続して取り組みせる点が重要である。また「仮想」のクラスであっても、実際には課題の一部は教室内で教員や TA とともに取り組みせる対面性を重視している。選択したテーマを継続的に取り組みせるために、小グループに分かれてプロジェクト形式の演習を行う方法があるが、基礎的な学習項目と連動させて演習を構成することは通常困難であり、初年次導入科目には向かない。本手法ではこの問題を解決している点が特色である。

VSC におけるプログラミング課題では、そのテーマの課題を実現するのに必要なライブラリがテーマごとに提供される。たとえば、Canvas (グラフィックスの課題) では、描画のためのウィンドウの生成や、その上への描画のための基本手続き群がライブラリに含まれる。各テーマ向けのライブラリでは、基本手続き群が単純な Java のクラスメソッドとして提供され、受講生はオブジェクト指向の知識を用いることなく特徴的な機能を利用できる。

VSC のオンライン教材ではまず、そのテーマで必要となる知識が「新しい内容」として提示される。グラフィックスであれば、ディスプレイでの座標系や光の 3 原色などの基礎知識などであり、アニメーションを実現するためのダブルバッファリングといった発展的な話題も含まれる。基礎知識の説明の後には、利用するライブラリの説明が例とともに示され、最後に課題が提示される。典型的な課題の個数は 3 から 4 である。

なお、音声および、Web アプリケーションのコースでは、

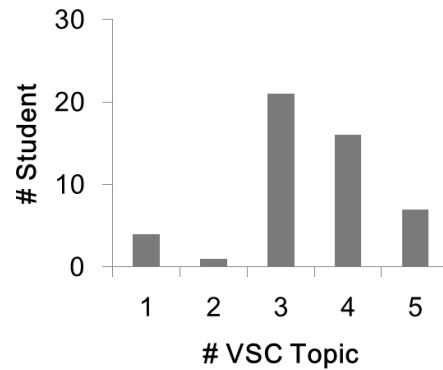


図 2 選んだ VSC のテーマ数
 Fig. 2 Number of VSC topics that student selected.

講義の前半におけるごく基礎的な部分の単元では特徴のある課題を打ち出すことは難しいため、それぞれ第 4 回、第 5 回の講義内容で初めてコースが開始される。また、第 9 回 (複合データ) の講義回のように「Plain」「Web」のみという回もある。下記は、レッスン 07 (第 8 回目) の単元における 3 コースの概略である。

Plain
[課題] 1. 配列中の特定の条件を満たす値を数えるプログラムの作成 2. 配列中の最大, 最小を探索するプログラムの作成 3. high or low トランプゲームの作成(テンプレートあり)
Canvas
[新しい内容] なし。(色の付け方などは、第 4 回で解説) [課題] 雨温図(グラフ)の描画(テンプレートあり) a. 降水量の表示 b. 最高, 最低気温の表示 c. 年間平均気温の表示
Web
[新しい内容] セッションとその仕組み, セッションの利用例 [課題] Web アプリケーションによる簡単なショッピングカート作成 a. カートに追加する商品を入力する部分の作成 b. これまでに追加された商品を表示 (配列に要素を追加するプログラムは与えられる) c. 合計金額を表示

図 1 は、第 8 回目の Canvas のプログラム例 (雨温図) の描画と実行例である。

3.3 仮想小クラス制での課題の取り組み

図 2 は、2011 年度におけるある教員のクラス (全 4 クラス中の 1 クラス) での課題の取り組み状況である。当該クラスは、学生が所属する学科の性質上、画像処理などの課題を多く扱う Canvas コースに取り組む学生が多かったが、それ以外のテーマに取り組む学生も見られ、学生の個性に合わせた対応をとることができた。音声や Web アプリケーションをテーマとする課題は、他コースよりも多くのコース独自のドメイン概念を学ぶ必要があり、難しいも

のとなっているが、これらに積極的に取り組む学生も見られた。なお、課題の達成は成績には含めないことを明示していたが、大部分の学生が少なくとも一度は複数コースの課題に取り組んだことが分かり、VSCは学生の動機づけに貢献したと考えられる。49名中ほとんどの学生が3種類以上のコースの課題に着手した（ただし1種類1回の着手も含む）。

4. 達成度保証のための認定試験と評定試験

4.1 基礎力認定試験 (MT)

本論文で紹介している導入科目では、学期末の試験とは別に行う MT (Mastery Test) という試験を 2011 年度から導入した。この試験は、受講者が当該科目を通じて共通して確実に学ぶべき項目を、すべて身につけたかどうかを確認するものである。これらの項目は、後継科目を学ぶうえで不可欠となる知識やスキルであるため、全問正答を合格基準とすべきであるが、実際には過誤を考慮した 86% を基準とした。試験の性質上、このように合格基準は高く設定されるが、1 回目で合格しない場合でも、複数回の受験を認め、すべての受講生が合格できるよう構成した。2011 年度では 3 回目でほぼ全員が合格した。この試験の実施方法は、日本の自動車運転免許における学科試験からヒントを得ている。すなわち過誤も考慮した合格基準 (90% 程度) が、路上で運転するために不可欠な技能と知識を担保するものとし、合格するまで何度も受験させるものである。下記は、配列と繰返しに関する MT の項目と試験問題のサンプルである。

[MT の項目例]

Q4 配列と繰返し
(a): 配列を生成できる
(b): 配列の各要素に値を代入できる
(c): 配列の要素を反復するための for 文を利用できる
(d): 配列の要素集計できる

[試験のサンプル]

Q4 (d) 引数 elements で受け取った配列の合計を計算し表示するプログラム断片を (d) に記載せよ。

```
void printTotal(double[] elements) {
    double total = 0.0;
    (d)
}
```

カリキュラム設計にあたっては、上述した MT の項目を設計のごく初期段階で部分目標として設定した。この部分目標は 3.1 節で述べた方法でカリキュラムに組み込まれる。この構成に従い教材および試験を設計することで、学生の

基礎的な学習目標達成を担保している。

4.2 評定試験

評定試験では、応用的な能力を問う。すなわち、表面的な知識では解くことが難しい問題を配置し、これを重点的にチェックする。これにより、本来のプログラミングの能力を測定することを狙いとする。すなわち、プログラムの経験を積んではじめて定着する、プログラムの読み書き能力を計ることができる。従来は、この手の問題を作成すると、最低限必要な知識を得たかどうかの観測が不可能であり、難しかった。逆に、最低限必要な知識と、応用的な問題の組合せは、最低限必要な知識の獲得が観測不可能であるだけでなく、応用的な知識を持っているかどうかの段階的な測定が難しく、特に中程度以上における評定が不可能であった。本アプローチでは、MT によって基礎的な学習目標の達成を担保するため、評定試験においてプログラミングに対するより深い理解や応用的事項の獲得の確認に焦点を合わせやすくなる。下記は、応用問題のサンプルである。[問題 4 (応用力を問う)]

- (a) メソッド「method」について、機能の説明を行え。
- (b) メソッド「start」においてメソッド「method」を起動せずに同じ挙動を行うよう start を書き換えよ。method の呼び出しの削除のほかには、プログラムを変更する部分は 1 行のみとする。

[プログラム]

```
void start() {
    String [] values = input.split(",");
    method(values);
    for( int i = 0; i < values.length; i++)
        (values の表示ルーチン)
}
void method(String [] array){
    (array の要素を逆順に登録するコード)
}
```

2011 年度においては、評定試験は通常のペーパー試験の形態で実施したが、試験のサンプルからも分かるように、レポート、口頭試問、発表会、実技試験など自由な形態による実施も可能と考えられる。

5. 結果

(1) 成績と課題数の相関 (2010, 2011 年度)

最終試験の得点について、2010 年度および 2011 年度の最終試験の得点と VSC の課題取り組み数を用いた回帰分析を行った (それぞれ、図 3, 図 4)。なお、両年度とも同一教員が講師を務めたクラスであり、総数はそれぞれ外れ値を除いて、N = 35, 46 である。外れ値の個数はそれぞれ 3, 1 である。x 軸は、学生が取り組んだ VSC の課題の総

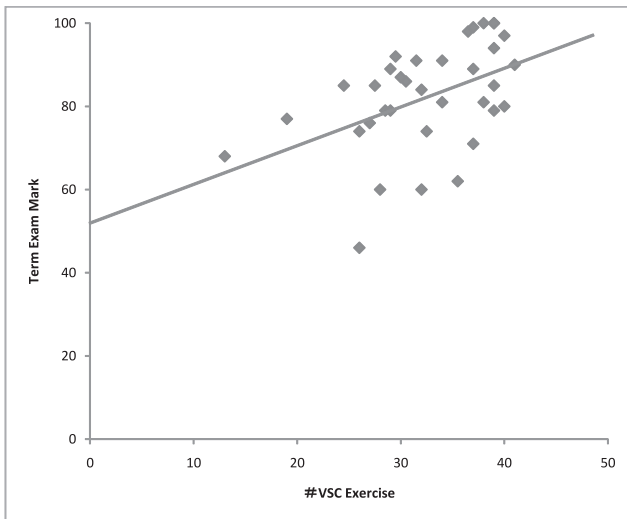


図 3 評定試験の得点と VSC の課題取り組み数による回帰分析 (2010)

Fig. 3 Linear regression for modeling of term exam. Marks by the number of VSC topics (2010).

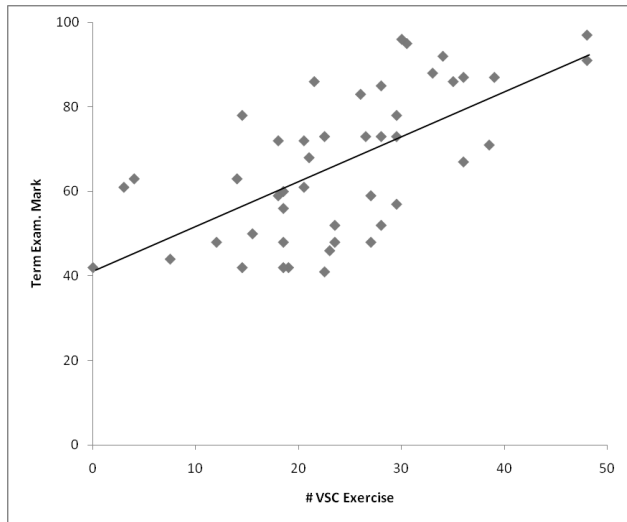


図 4 評定試験の得点と VSC の課題取り組み数による回帰分析 (2011)

Fig. 4 Linear regression for modeling of term exam. Marks by the number of VSC topics (2011).

数, y 軸は最終試験の得点である. 2011 年度の VSC は 5 コースであったが, 2010 年度では, Sound と Web のコースを除いた 3 コース体制であった. 両図において, 2010 年度の結果では, 相関係数は 0.47 であり, 回帰方程式は, $y = 0.94x + 52$ であった. 標準誤差は 11.5 である. 2011 年度の結果では, 相関係数は 0.64 であり, 回帰方程式は $y = 1.1x + 41$ であった. 標準誤差は 13.5 である.

(2) 最終試験および認定試験 (MT) の結果

以下では 2010 年度および 2011 年度の試験 (全 4 クラス) の結果を示す. 前述したとおり 2011 年度は, MT を実施している. 表 4 は, 2010 年度の最終試験の問題別の正答率を示したものである. 2010 年度の最終試験では, 基

表 4 期末試験 (2010 年度) の内容と正答率

Table 4 Distribution of the score of term examination in 2010.

	正答率 (%)	メソッド	出題範囲
Q1	93.8	無	式と変数
Q2	89.1	無	条件分岐
Q3	91.3	有	メソッド
Q4	82	有	配列と繰り返し
Q5	83	有	構造体
Q6	72.6	有	対話型プログラミング, 参照と実体
Q7	35.9	有	総合的な問題 (発展問題)

表 5 MT (2011 年度) の内容と正答率

Table 5 Distribution of the score of mastery tests in 2011.

	正答率 (%)	メソッド	出題範囲
Q1	98.8	無	式と変数
Q2	96.2	無	条件分岐
Q3	97.3	有	メソッド
Q4	90.7	有	配列と繰り返し
Q5	90.2	有	構造体

礎的な練度を問う問題群 (Q1 から Q5) と, 評定のためのより発展的な問題群 (Q6, Q7) とを分離した.

この結果では, メソッドを用いた基礎的な問題 (Q3, Q4, Q5) における正答率は, 2009 年度のメソッドを用いる問題 (Q4, Q6, Q7) より正答率が上がっていることが分かる (表 2). 一方で, 発展的な問題 (Q7) は, 得点が期待どおりには伸びていないことが分かる. すなわち, 既存の基礎的な練度を問う問題を含めた出題では, 正確な評定ができないことを示している. これは, 発展的な問題の出題数やバリエーションを増やすことができないことが主な理由である.

表 5 は, 2011 年度の認定試験 (MT) の最終的な問題別の結果である. 2011 年度の MT は, 2010 年度の基礎問題 (Q1 から Q5) に対するトピックを扱い, また難易度もほぼ同等として設計した. メソッドの知識を利用する Q3-Q5 に関して 2010 年度よりも高い正答率を達成している. 2010 年度では, 16% の学生が Q1-Q5 の基礎問題が 85% 以下の正答率にもかかわらず, 単位が認定された. 一方で, 2011 年度では, 全受験生のほとんどが 85% 以上を達成した.

(3) 評定試験の結果 (相関, 2011 年度)

表 6 は, (1) の項で示した 2011 年度の 1 クラスについて VSC の課題取り組み数 (VSC), 1 回目の認定試験の点数 (MT1), 最終的な認定試験の点数 (MTF) および評定試験 (GR) の相関を示したものである. MTF ではほぼ全

表 6 VSC の取り組み課題数, MT 得点, 評定試験の相関
 Table 6 Correlation among VSC exercises, MT scores, and the term examination scores in 2011.

	VSC	MT1	MTF
MT1	0.58		
MTF(参考)	0.23	0.19	
GR	0.64	0.67	0.06

員の学生が合格点以上の得点であるため参考値として提示している。この表では VSC と MT1 の相関および、VSC と GR の相関の高さが特徴である。VSC と GR の正の相関の高さは、VSC の枠組みがより高い学生のスキル獲得につながっていることを示唆する。VSC と MT1 の相関の高さは、VSC への取り組みに熱心でない学生は、1 回目の MT の段階で基礎スキルに関わる学習が遅れていたことを示しているが、MT を再受験するなかで最終的にはその遅れを取り戻し、最低限のスキルを得たことを示している (MTF)。また、4.2 節で述べたように、複数回行う MT の導入にともなって基礎力を担保したことにより、評定試験では基礎的な問題を大幅に減らして出題している。MTF と GR の相関の低さは、この点を反映した結果といえるが、基礎力を実際に使いこなす応用的な力を、最低限の基礎力の評価とは分離して評価できた一面を説明している。

6. 議論と関連研究

6.1 議論

2010 年度と 2011 年度において、同一教員が同一の教材に基づいて授業を行った 2 クラスにおける結果について考察する。図 3, 図 4 において 2010 年度に比べ、2011 年度における回帰方程式の切片が低いことは ($41 < 52$)、最終試験の難易度が高いことを表している。さらに、傾きがより大きい ($1.1 > 0.94$) ことから、高得点層における段階的な成績の弁別を可能としていることを表している。また、2011 年度は前年度に比べても取り組み課題数と評定試験の点数の相関係数が高めにでていることは、2011 年度においては、学生の課題に取り組む努力がより成績に結び付きやすかった可能性を示唆している。

本授業構成および仮想小クラス制による複数テーマによる課題出題は、上位層の学生にプラスに働いたと考えられる。学習の早い学生の中にはほとんどすべての問題に取り組んだ例もある。筆者らは以前、高いレベルの学生を各クラスから選抜し、通常クラスとは別のクラスに集めて講義を実施する試みを行ったが、これと同等の効果を別クラスの設置なしに得たと考えられる。一方で、テーマを多様化させ、課題のパターンを増やしたことにより、学生が解いた課題に対して TA がフィードバックをすぐに与えることが難しい場合があり、同一授業を担当する別クラスの教員、TA との情報交換などの工夫が必要である。

MT は、学生にとっては単位認定のための試験であるが、試験の各項目が講義のどの箇所で学んだかを容易に把握できるように構成されている。特に不合格となった学生は、自分の弱点を把握し、その弱点を克服するという明確なゴールを設定できるため、自主的な学習が行いやすくなったと考えられる。同様に教員も、学生の MT の答案を利用して、当該学生の理解が浅い部分を把握しやすいという利点が得られた。なお、学習事項の長期的な定着までは、一度の試験実施では保証できない点は問題である。今後、授業の終了後も MT による評価を継続することで、定着の促進とその度合いの観測を試みる予定である。

6.2 関連研究

VSC に関連した事例として、Princeton 大学の情報系学科で実施される General Computer Science^{*1} というコースがあり、これは後述の文献 [3] を教科書とした講義である。特に科学分野への応用を視点に入れた広範な分野からテーマを扱った課題が課される。このコースでの課題は、選択式ではなく各学生は毎週同じテーマに取り組む。

各週の講義において選択課題を課する事例は一般的であり、特に講義の最終課題を選択式にするという事例が報告されている (たとえば、文献 [4])。VSC の上述の 2 事例との相違点は、VSC では 1 週の講義内で複数テーマから課題を選択して取り組むことができ、かつそのテーマは講義期間中に継続しているという点にある。

継続して同一テーマに取り組ませるために、小グループごとに別々の教員がプロジェクト型の演習を担当する方法は一般的に行われている。しかし、3.2 節で述べたように、基礎力の担保が重要視される初年次導入科目の一部としてこの方法を用いることは困難である。VSC では複数のコースを持ちながら、基礎とする教授項目を揃え、共通試験の実施を可能としている。

MT に関連した事例として、岩手大学で行われている取り組みをあげる [5]。この取り組みでは、大量の問題からなるテストを Web 試験によって複数回実施し到達度を評価する。この手法では、対象とした科目が専門科目 (データベース) である点が著者らの MT と異なる。また、本研究で MT によって可能となった評定との分離の観点、必要最低限のスキルを定める講義設計については議論されていない。

プログラミングの初年次教育で利用する言語の選択に関する議論において、Java の利用に疑問を呈する記事や文献 (たとえば文献 [6]) が見られる。筆者らの取り組みにおいて Java を選択している理由は、利用している統合環境 Eclipse の利用により、教員、TA が受講生の文法エラーや実行時エラーを見つけやすい点、VSC において独自ライブ

*1 <http://www.princeton.edu/~cos126> (参照 2013-07-09)

ラリを加えやすい点、プラグインによる独自オンライン提出メカニズムの組み込みの容易さといった授業の運用の面の長所（本学部では学生へのPC貸与時にEclipseをプレインストール済みでありインストール時のトラブルは生じない）、さらに産業界からの要請などの点を総合的に考慮したことによる。

文献[7]は、オブジェクト指向言語をプログラミングの初年次教育に利用する際に、与えられたAPIなどの機能を組み合わせてコード（クライアントコード）を書く中で、制御や配列といった項目をCS1（導入科目）において学習し、クラス、メソッドを実装し提供する機能そのものを構成する（インプリメンテーションを行う）スキルは後継のCS2で学習するという、アプローチを提唱している。筆者らは、このアプローチとは反対に4週目という比較的初期に手続きの概念を学習させ、その後の演習では小さな手続きの組合せでプログラムを書くことを求めている。なお、本講義では、「9：複合データ」でクラスとインスタンスを利用するが、レコード構造、すなわちC言語における構造体としての利用であり、インスタンスメソッドなどオブジェクト指向特有の機能は利用しない。

プログラミング教育において、伝統的な標準入出力を用いるような例題や課題が学生のモチベーションを喚起しない、という問題意識から独自のライブラリを利用させるという教材が存在する[3],[8]。特に、前者は多様なテーマを扱う教科書であり、音声やWebアプリケーションなどを扱うなど共通する部分があるが、筆者らのVSCのように同じ基礎学習項目に関して複数テーマを配置していない。

Mastery Testは、受講者が基礎的な事項をすべて習得することを目的としたものである。この手法は、Bloomが提唱した完全習得学習（Mastery Learning）から一部ヒントを得たものである[2],[9]。この手法は、個々の子供が各自に適した時間を使って、最終的にすべての学習事項を習得することを目的としている。

7. おわりに

本論文では、プログラミングの入門科目における仮想小クラス制（VSC: Virtual Small Group Class）を提案した。学生の興味の多様化に合わせ、複数の課題を用意することで学習意欲を向上させた。また、Mastery Test（MT）を導入することで、単位の認定と成績の評定を明確に分離し、課題が多様化する中でも学生の達成度を広く保証できた。また、MTは、学生の達成度を自身に認識させ、学習課題を明確にした。成績の評定においては、より自由な方法で試験を実施することが可能となり、適切な評定が行えた。

今後の課題は、MTの実施や運用のコストの削減である。LMS（Learning Management System）を利用したオンラインテストの利用を検討している。また、本手法を他の教科で応用するための手法の検討も必要である。

謝辞 本研究は法政大学教育GP「高度情報処理技術者をめざす学士力の育成」による取り組みの一部である。

参考文献

- [1] 情報処理学会：情報専門学科におけるカリキュラム標準J07，入手先（<http://www.ipsj.or.jp/12kyoiku/J07/J0720090407.html>）（参照2013-03-18）。
- [2] Gagne, R.M., Wager, W.W., Golas, K.C. and Keller, J.M.: *Principles of Instructional Design (5th ed.)*, Wadsworth/Thomson Learning, Belmont, CA. 鈴木克明, 岩崎 信 (監訳): *インストラクショナルデザインの原理*, 北大路書房 (2007).
- [3] Sedgewick, R. and Wayne, K.: *Introduction to Programming in Java: An Interdisciplinary Approach*, Pearson Addison-Wesley, Boston (2008).
- [4] 横山輝明: サイバー大学における演習講義の取り組み, JAVAプログラミング演習事例報告, 入手先 (http://www.cyber-u.ac.jp/about/e-learning_research/0002/pdf/0002.08.pdf) (参照2013-07-09).
- [5] 三輪譲二: パソコンと携帯向きのWeb型試験システムを用いた専門科目の到達度評価, 日本教育工学会研究報告集, Vol.2008, No.2, pp.111-116 (2008).
- [6] Dewar, R.B.K. and Schonberg, E.: Computer Science Education: Where Are the Software Engineers of Tomorrow?, *CrossTalk, The Journal of Defense Software Engineering*, Vol.21, No.1, pp.28-30 (2008).
- [7] Roumani, H.: Practice what you preach: full separation of concerns in CS1/CS2, *Proc. 37th SIGCSE Technical Symposium on Computer Science Education (SIGCSE '06)*, pp.491-494, ACM (2006).
- [8] Felleisen, M., Findler, R.B., Flatt, M. and Krishnamurthi, S.: *How to Design Programs*, The MIT Press, England (2001).
- [9] Bloom, B.S.: The search for methods of group instruction as effective as one-to-one tutoring, *Educational Leadership*, Vol.41, No.8, pp.4-17 (1984).



佐々木 晃

1994年東京工業大学理学部情報科学科卒業。2001年同大学大学院情報理工学研究科博士後期課程単位取得退学、東京大学医科学研究所産学官連携研究員等を経て、法政大学情報科学部准教授。博士（理学）。エージェントシミュレーションとその処理系、プログラミング言語全般の研究に従事。日本ソフトウェア科学会、ACM、PAAA（Pan-Asian Association for Agent-based Approach in Social Systems Sciences）各会員。



伊藤 克巨 (正会員)

博士(工学)。1993年電子技術総合研究所入所。2003年名古屋大学大学院助教授。2006年法政大学情報科学部教授，現在に至る。音声・音楽を主とした音処理と自然言語処理全般に興味を持つ。日本音響学会会員。



荒川 傑

株式会社ノーチラス・テクノロジーズフェロー。2007年法政大学大学院情報科学研究科修士課程修了。分散システムやコンパイラ等の研究開発に従事。