

# プログラミング導入教育におけるコースウェアの違いによる 学習効果の比較

西田 知博<sup>1,a)</sup> 原田 章<sup>2</sup> 中西 通雄<sup>3</sup> 松浦 敏雄<sup>4</sup>

**概要:** 筆者らは、情報を専門としない学部における1年次配当の情報教育科目の中で、4時限を使用し、プログラミングの導入教育を行っている。この授業では短時間でプログラミングが習得できるよう、初学者向けプログラミング学習環境であるPENを用いている。2011年度からは、従来の処理結果を文字列で出力するタイプのコースウェアに加え、図形描画から入るタイプのコースウェアを開発し、クラスを2群に分けて異なるコースウェアを使って授業を行っている。2011年度から3年間の授業におけるアンケートおよび試験成績を分析した結果、2つのコースウェア間の明確な優劣は見いだせなかったが、図形描画から入るコースウェアの方が、繰り返しのようなたまづき易い学習内容でも理解度や楽しさを下げることなく学習が行えることが伺えた。

## Comparison between two types of courseware of introductory programming

**Abstract:** We have classes to teach information literacy for the freshmen of humanities majors. In the classes, we teach basic programming for 6 hours. In the classes, we use PEN: Programming Environment for Novices to teach programming in a short time. Since 2011, we provide the courseware which contains a drawing for every example and exercise in addition to traditional courseware. We made two types of classes which utilize drawing courseware and traditional courseware, and compare the learning effect between them. In the consequence, we cannot find significant difference between them. However, in classes which using drawing courseware, we can find that the students keep their motivation even when they studied complex things, such looping.

### 1. はじめに

我々はコンピュータでの処理の本質を理解するために、プログラミングを学ぶことが重要であると考え、初学者が比較的短い学習時間でこの目的のための「プログラミング」を習得できるプログラミング環境PEN (Programming Environment for Novices)[1]を開発した。また、いくつかの大学および高校において、PENを活用した授業実践を重ね、教材の開発を行ってきた[2]。

PENは、日本語ベースの分かりやすいプログラミング言語を用い、構文エラーの発生を低減するための入力支援機能を備えている。また、プログラミングの実行の様子を把握しやすくするための機能を備えており、初学者が短時間でプログラミングが学びやすい環境を提供している。また、プログラムの実行結果を視覚的に確認できるようにして学習者のモチベーションを高めるために、図形描画のための組み込み関数を提供している。

図形描画を用いることで学習者のモチベーションが上がることは経験的には感じられてはいたが、これを客観的に評価するため、2008年に図形描画を伴う例題を主としたコースと、通常のコースとで学生のモチベーションや理解度に差があるのかどうかを調査した[3]。しかしその結果は、2つのコースウェアによる学生の意識および理解度について、明確な有意差を見出すことが出来なかったというものであった。2つのコースについての有意差が見出せな

<sup>1</sup> 大阪学院大学  
Osaka Gakuin University  
<sup>2</sup> 追手門学院大学  
Otemon Gakuin University  
<sup>3</sup> 大阪工業大学  
Osaka Institute of Technology  
<sup>4</sup> 大阪市立大学  
Osaka City University  
a) nishida@ogu.ac.jp

表 1 各年度の受講者数と授業内容

年度	学部	第 1 教室	第 2 教室	第 3 教室	合計
2011	人間科学部	45 (図)	45 (図)	49 (従)	139
	文学部	59 (図)	64 (従)	59 (従)	182
2012	人間科学部	48 (従)	48 (従)	46 (図)	142
	文学部	56 (従)	69 (図)	56 (図)	181
2013	人間科学部	48 (従)	48 (図)	45 (図)	141
	文学部	60 (従)	63 (従)	59 (図)	182

かった原因として、以下のものが考えられる。

- (1) 2つのコースに難易度の差があった
- (2) 各クラスの授業の進め方に違いがあった
- (3) 各例題に費やす時間がクラスによって異なっていた

本研究ではこれらの点について配慮し、教材の構成と授業の進め方を見直し、コース間で差が出ないように注力しつつ、再調査を行った。

## 2. 授業の概要

大阪大学では、全学において1年次における情報教育科目が開講されており、文学部では1994年以降、人間科学部では1995年以降、情報教育科目「情報活用基礎」が必修科目となっている。この授業の中の調査で、両学部間では受講前でのコンピュータ習熟度やコンピュータ不安がほぼ同質と考えられることが明らかとなっている [4]。

今回はこの授業において、2011～2013年の3年間に渡り、調査を行った。

### 2.1 対象とするクラス

人間科学部および文学部の「情報関係基礎」は、1年次前期に開講され、それぞれ3つのクラスに分け(表1)、各クラスを1名の教員と2名もしくは3名のティーチングアシスタント(TA)で担当している。両学部の担当教員は互いに授業の進行状況や講義資料の情報を交換しながら、同じ計算機環境で授業をすすめている。プログラミング演習は終盤のテーマとし、両学部とも4回を割り当てた。

	図形型	従来型
第1回	逐次処理(図形)	逐次処理
第2回	条件分岐(図形)	条件分岐
第3回	繰り返し(図形)	繰り返し 実数
第4回	実数	図形描画

図 1 授業の構成

### 2.2 授業の概要

コースウェアは、例題の特徴によって2種類用意した。一つは、キーボードからの入力に対して、計算結果をコンソールに文字列として出力する形式の演習を繰り返す授業(従来型と呼ぶ)であり、もう一つは、描画用のウィンドウに図形を描画する形式の演習を繰り返すもの(図形型と呼ぶ)である。図1にそれぞれの形式での授業内容の概要を示す。また、表1に各クラスでどちらの形式の授業を行ったかを示す。なお、従来型のクラスでも4回目の授業では図形描画、図形型のクラスでも4回目の授業では実数計算を扱い、どちらの形式であっても、同質の内容が学べるようにしている。

### 2.3 授業内容

#### 2.3.1 授業内容の統一

従来型および図形型の授業は、予め配布した授業資料 [5] に沿って、例題のプログラムを教員が解説し、その後受講者が各自で練習問題のプログラムを作成するということを繰り返す。授業内容は、それ以前の数年の経験より部分的に修正を加えてきたものであり、担当教員6名の意見が反映されている。授業で取り扱っている例題および練習問題を付録に示す。

#### 2.3.2 授業時間配分の調整

過去の授業経験からそれぞれの例題、練習問題等に要する予定時間案を算出し、担当教員全員の合意によって、予定時間を定めた。各担当教員は、この予定時間にできるだけ沿って授業を進めた。それでも受講者の反応などから、個々の教員の判断で授業の進行速度を調整した。予定時間および実際にかかった時間を表2に示す。

## 3. 学習効果に関する調査

本研究の目的は、コースウェアの違いが学生のプログラミング学習にどのような影響を与えたのかを検討することである。そこで、各授業回終了後に行った調査の結果を分析し、コースウェアの違いが調査結果のどのような側面に現れているかについて検討することとした。

### 3.1 調査の概要

本研究では、プログラミング学習を行った授業回の終わりにWebを用いた調査を行った。

調査の主な内容は以下の通りである。

- (1) ログイン名
- (2) 宿題に対する達成度の自己評価\*1
- (3) 授業で扱った例題・練習問題に対する理解度や難易度の自己評価
- (4) プログラミング内容に関する理解度の自己評価

\*1 第1回目の授業にはこの内容は含まれなかった。

表 2 2013 年度の授業時間配分 (単位:分)

従来型コースウェア 第 1 回	予定	文 1	文 2	人 1
準備: プログラミングとは	25	15	25	20
例題 1.1 (入力の 3 倍)	20	15	10	20
練習 1.1 (長方形の面積)	10	10	10	10
例題 1.2 (3 の倍数の判定)	10	12	12	10
例題 1.3 (3 の倍数か否かの判定)	05	05	08	05
図形型コースウェア 第 1 回	予定	文 3	人 2	人 3
準備: プログラミングとは	25	25	25	8
例題 1.1 (緑の円)	30	30	30	30
練習 1.2 (長方形)	10	15	15	22
従来型コースウェア 第 2 回	予定	文 1	文 2	人 1
宿題の解説	10	8	5	10
練習 1.6 (合否判定・乱数)	15	9	10	15
例題 1.4 (成績判定)	10	12	13	10
練習 1.10 (うるう年)	15	15	17	15
例題 1.5 (数当てゲーム)	15	9	10	10
例題 1.6 (10 回繰り返し)	10	10	7	10
図形型コースウェア 第 2 回	予定	文 3	人 2	人 3
宿題の解説	10	8	10	11
例題 1.2 (3 の倍数の判定)	15	12	18	13
例題 1.3 (3 の倍数か否かの判定)	05	5	15	9
練習 1.5 (合否判定・乱数)	15	15	12	15
例題 1.4 (成績判定)	10	13	12	13
練習 1.9 (うるう年)	15	20	13	17
従来型コースウェア 第 3 回	予定	文 1	文 2	人 1
宿題の解説	10	10	10	11
練習 1.15 (繰返・乱数の平均)	10	15	11	15
例題 1.7 (二重ループ)	06	10	12	6
練習 1.23 (□を n 行目に n 個)	14	5	8	8
例題 1.8 (繰返 (2)-乱数)	05	5	3	7
例題 1.9 (実数-温度変換)	15	12	10	13
練習 1.29 (階乗)	15	18	20	11
図形型コースウェア 第 3 回	予定	文 3	人 2	人 3
宿題の解説	10	5	10	15
例題 1.5 (半径当て・乱数)	15	10	17	20
例題 1.6 (繰返し・乱数)	10	10	13	10
練習 1.14,15 (繰返・乱数)	15	20	15	15
例題 1.7 (円の水平配置)	10	10	10	10
従来型コースウェア 第 4 回	予定	文 1	文 2	人 1
宿題の解説	10	10	2	10
例題 1.10 (描画関数の使い方)	12	13	13	12
例題 1.11 (円の水平配置)	13	10	9	13
練習 1.33,34 (円を並べる (1,2))	17	14	19	17
例題 1.12 (円の描画・乱数)	08	6	5	8
練習 1.38,39 (円の描画 2・乱数)	10	10	14	10
図形型コースウェア 第 4 回	予定	文 3	人 2	人 3
宿題の解説	10	5	15	10
例題 1.8 (二重ループ)	08	10	10	10
練習 1.23 (n 行目に n 個並べる)	12	15	18	11
例題 1.9 (繰返 (2)-乱数)	05	3	5	7
例題 1.10 (実数-座標の平均)	15	17	27	22
練習 1.27 (階乗)	15	30	-	10

(5) 授業に対する理解度や面白さの自己評価

(6) プログラミングに対する楽しさや容易さの自己評価

ログイン名は各授業回で回答者の対応を取るために入力してもらう項目である。宿題に対する達成度の自己評価は、第 2 回目の授業回以降で設定したもので、宿題の達成にどの程度、困難さを感じたかを尋ねるような内容である。また、宿題に要した時間の長さについて尋ねるものも含めた。授業で扱った例題については、その理解の程度を尋ねた。また、練習問題についてはプログラムを作成するときのどの程度困難さを感じたかについて尋ねた。プログラミング内容に関する理解度の自己評価項目は、「変数」や「条件分岐」、「繰り返し処理」といった学習項目について自分がどの程度理解しているかを尋ねるものであった。授業に対する理解度や面白さの自己評価は、授業内容を理解できたかどうか、例題や練習問題が分からなかったときにどうしたか、全体的に授業が面白かったかどうかについて尋ねた。プログラミングに対する楽しさや容易さの自己評価では、プログラミングが楽しいかどうか、プログラミングが容易かどうか、さらに、最終回である第 4 回目を除き、次の授業が楽しみかどうかなどについて尋ねた。

なお、アンケートは実際に回答してもらったのと同じ形式で [5] から参照できるようにしている。

### 3.2 分析の目的

前節で述べたように、分析対象の授業は、文学部および人間科学部のそれぞれ 3 教室で 2 種類のコースウェアを振り分けて実施した。我々は、コースウェアの違いが、期末に行った試験成績だけではなく、授業に対する理解度やプログラミングを楽しいと感じるかどうかの程度に現れると考えた。そこで、本節の分析では、4 回の授業を通して、授業に対する理解度およびプログラミングに対する楽しさの自己評価がどのように変化したかについて、用いたコースウェア別に調べることにした。

授業に対する理解度の自己評価項目は「今日の授業は理解できましたか?」というものであった。回答用選択肢は「理解できた・だいたい理解できた・どちらかという理解できた・どちらかという理解できなかった・あまり理解できなかった・理解できなかった」の 6 段階であった。受講学生はこの中からひとつを選択した。また、プログラミングに対する楽しさの自己評価項目は、「プログラミングはあなたにとって楽しいですか?」というものであった。回答用選択肢は「とても楽しい・かなり楽しい・どちらかという楽しい・どちらでもない・どちらかという楽しくはない・あまり楽しくはない・全く楽しくはない」の 7 段階であった。受講学生はこの中からひとつを選択した。

コースウェアの違いがこうした項目の回答に影響を与えているのであれば、コースウェアが異なる群間で平均が異なってくるはずである。また、コースウェアの差が授業内

容との対応で現れるのであれば、各授業回の平均についてその変動パターンがコースウェアによって異なるはずである。このことを2011年度から2013年度の調査結果に分散分析を適用して検討することとした。各年度は前年度の反省を踏まえて授業方法や授業内容を微調整しているが、主たる内容は同じであった。そこで、3年間の結果を総合的に分析することによってコースウェアの違いを明確にできると考えた。

### 3.3 分析の方法

本節の分析において、分析対象は、プログラミングの授業全4回にすべて出席し、Web調査にすべて回答した1年生であった。表3は、各年度の分析対象者数である。

分散分析の従属変数となる変数「授業に対する理解度の自己評価」については、6段階の各段階に対して、「理解できた」を6点、「理解できなかった」を1点となるように1点刻みで数値を割り当て、この数値を得点と考え、数量化した。得点が高いほど、理解度の自己評価が高いことを表している。また、「プログラミングに対する楽しさの自己評価」の自己評価については、7段階の各段階に対して、「とても楽しい」を7点、「全く楽しくはない」を1点となるように1点刻みで数値を割り当て、この数値を得点と考え、数量化した。得点が高いほど、楽しさの自己評価が高いことを表している。

次に、分散分析の要因については以下のものを考えた。

- 授業方式 (図形型・従来型の2水準, 被験者間要因)
- 学部 (文学部・人間科学部の2水準, 被験者間要因)
- 年度 (2011年度, 2012年度, 2013年度の3水準, 被験者間要因)
- 授業回 (第1回・第2回・第3回・第4回の4水準, 被験者内要因)

以上の内容から、理解度と楽しさの自己評価それぞれを従属変数とする4要因の分散分析を行った。なお、年度によって各学部で取り扱った授業方式の教室数が異なっていたので、分散分析のモデルにはすべての交互作用を含めるのではなく、検討する意味のあるものだけを含めて分析した。

### 3.4 授業に対する理解度の分析結果

表4は、授業に対する理解度の自己評価得点について、年度ごとに各授業回のコースウェア別記述統計量(平均・標準偏差)を求めたものである。平均の値を見ると、おおむね4.5点から5.0点の値となっている。この得点は1点から6点の間に分布するので、平均的には理解度の自己評価は高かった。ただし、2011年度と2013年度の従来型第3回の平均が全体に比べてやや低い値となった。

図2は、表4の平均をグラフに表したものである。図形型では、第4回目に平均が下がる傾向が見られた。従来型

では、第3回目に平均が下がる傾向が見られた。ただし、2012年度は他の2年と比較して下がり幅が小さかった。

表5は、授業に対する理解度の自己評価について分散分析した結果をまとめた分散分析表である。学部と授業方式、年度と授業方式の交互作用、授業回の主効果、授業方式と授業回の各交互作用が有意であることが分かった。

学部と授業方式の交互作用が有意であったことから、学部によってコースウェアの違いにより生じる差が異なっていたことが分かる。図3は、理解度の平均を学部ごとにコースウェア別で示したものである。図から分かる通り、文学部では図形型の方が、人間科学部では従来型の方が高かった。ただし、その差が著しく大きいということとはなかった。

また、年度と授業方式についても交互作用が有意であったことから、年度によってコースウェアの違いにより生じる差が異なっていたことが分かる。図4は、理解度の平均を年度ごとにコースウェア別に示したものである。図から分かる通り、2011年度では図形型の方が、2012年度では従来型の方が高かった。2013年度は両者に差が見られなかった。

さらに、授業回の主効果、授業方式と授業回の交互作用が有意であったことから、4回の授業において、平均の変動パターンが授業方式によって異なることが分かる。このことは、図2から読み取ることができる。図形型は第1回の授業から第3回まであまり変動がなく第4回で下がる。従来型は第3回の授業で下がるが、第4回で戻る。授業に対する理解度が下がるということは、その回の授業内容が受講学生にとって難しかったことを表している。

以上のことから、授業内容に対する理解度の自己評価について、どちらかのコースウェアを用いた方が理解度が高くなるといった傾向は見られなかった。

### 3.5 プログラミングに対する楽しさの分析結果

表6は、プログラミングに対する楽しさについて、年度ごとに各授業回のコースウェア別記述統計量を求めたものである。この得点は1点から7点の間に分布するので、平均は4点から5点までやや高い値のものが多かった。ただし、従来型においては、2011年度の第4回で高い値、2013年度の第3回でやや低い値が見られた。図5は、平均を図示したものである。

表7は、プログラミングに対する楽しさの自己評価について分散分析した結果をまとめた分散分析表である。授業回の主効果、授業方式と授業回の交互作用が有意であった。

この結果は、図5において、従来型の第4回で平均が上がる傾向に対応していると考えられる。

表 3 各年度のコースウェア別分析対象者人数

年度	文学部			人間科学部			合計
	図形型	従来型	小計	図形型	従来型	小計	
2011 年度	28	54	82	53	34	87	169
2012 年度	41	21	62	19	48	67	129
2013 年度	28	42	70	29	19	48	118
合計	97	117	214	101	101	202	416

表 4 「授業に対する理解度の自己評価得点」の記述統計量

年度	統計量	図形型				従来型			
		第 1 回	第 2 回	第 3 回	第 4 回	第 1 回	第 2 回	第 3 回	第 4 回
2011 年度	平均	4.77	4.95	4.98	4.49	4.56	4.66	3.86	4.70
	標準偏差	1.04	0.79	0.91	1.09	1.10	0.92	1.24	1.33
2012 年度	平均	4.68	4.55	4.62	4.22	4.93	4.97	4.59	4.97
	標準偏差	1.17	1.13	1.01	1.14	1.13	1.01	1.02	1.21
2013 年度	平均	4.60	4.84	4.79	3.88	4.89	4.61	3.64	4.97
	標準偏差	1.08	0.77	1.15	1.05	1.05	1.02	1.28	0.93

### 3.6 分析結果に対する考察

#### 3.6.1 理解度の変化

授業に対する理解度の自己評価に対する分析結果から、図形型と従来型のどちらかが有意に高いという結果は得られなかった。この調査で調べた「理解度の自己評価」は「理解できたと感じた程度」である。理解できたとすることはプログラミングに対する学習意欲の維持や向上に効果があると期待できる。また、表 4 や図 2 から分かる通り、全体的に見ると図形型、従来型ともに平均が高い値になっていた。こうしたことから、本研究で行ったプログラミングの授業の内容が受講学生に対して適切であったと考えられる。一方で、図 3 や図 4 から分かる通り、コースウェアによって学部間や年度間で有意な交互作用が見られた。また、授業方式と授業回の間にも有意な交互作用が見られた。これは、図 2 から図形型の第 4 回・従来型の第 3 回で得点の平均が下がるという違いによるものではないかと考えられる。

#### 3.6.2 コースウェアと学部

コースウェアと学部の交互作用は、受講学生の傾向によって適切なコースウェアが異なる可能性を示唆している。本研究で対象とした文学部と人間科学部は、文科系の学部であるという点では同じである。また、両学部の学生に対して受講前にコンピュータ不安やコンピュータに対する態度を調べたが、両者の得点には差がないことは確認している。しかし、文学部においては図形型、人間科学部においては従来型の方が高くなった。確かに、両学部の入試科目や専門領域は異なっているので、受講学生の傾向に何らかの影響を与えている可能性がある。ただし、実質的な差はそれほど大きいものではないことから考えて、学部によって適切なコースウェアが異なると主張できるほどの明確な結果とはなっていない。

#### 3.6.3 コースウェアと年度

コースウェアと年度の交互作用は、コースウェアのどちらかが優れているわけではないことの傍証になっている。図 4 から、年度によって平均の高いコースウェアが異なったり、差がなくなったりすることが分かった。どちらかのコースウェアの方が優れていれば、年度による傾向が一貫するはずである。本研究では、授業の運営にあたって各コースウェアで授業内容が同様になるよう工夫した。しかし、授業運営には統制することのできない誤差的要因が数多くある。両コースウェアの優劣の差が大きければ、そうした誤差的要因の影響があっても一貫した差として現れると考えられるが、本研究の結果はそうならなかった。これより、図形型・従来型のどちらかが優れているわけではないと考えられる。

#### 3.6.4 コースウェアと授業回

コースウェアと授業回の交互作用は、理解度の自己評価が授業内容と対応して変動することを表している。図形型では第 4 回目で自己評価の平均が低下している。これは、第 3 回まではグラフィックスを用いた授業が行われるが、第 4 回の授業は従来型の内容となっていることが原因であると考えられる。また、従来型は第 3 回に平均が低下し、第 4 回に上昇するという傾向が現れている。これは、第 3 回目の授業が繰り返し処理に関するもので、それまで 2 回の内容よりも難しい。また、第 4 回目の授業はグラフィックスを用いた内容になっている。これらのことが、この傾向が現れる原因となっていると思われる。

このように、授業内容によって、理解度が変動するのは自然なことであるが、同じ授業内容であっても、繰り返し処理の回については、図形型と従来型で異なる傾向が見られた。どちらのコースウェアでも第 3 回は繰り返し処理を扱う回であるが、図 2 から明らかなおと、図形型は従来型のように大きく理解度が下がらなかった。このことは、図

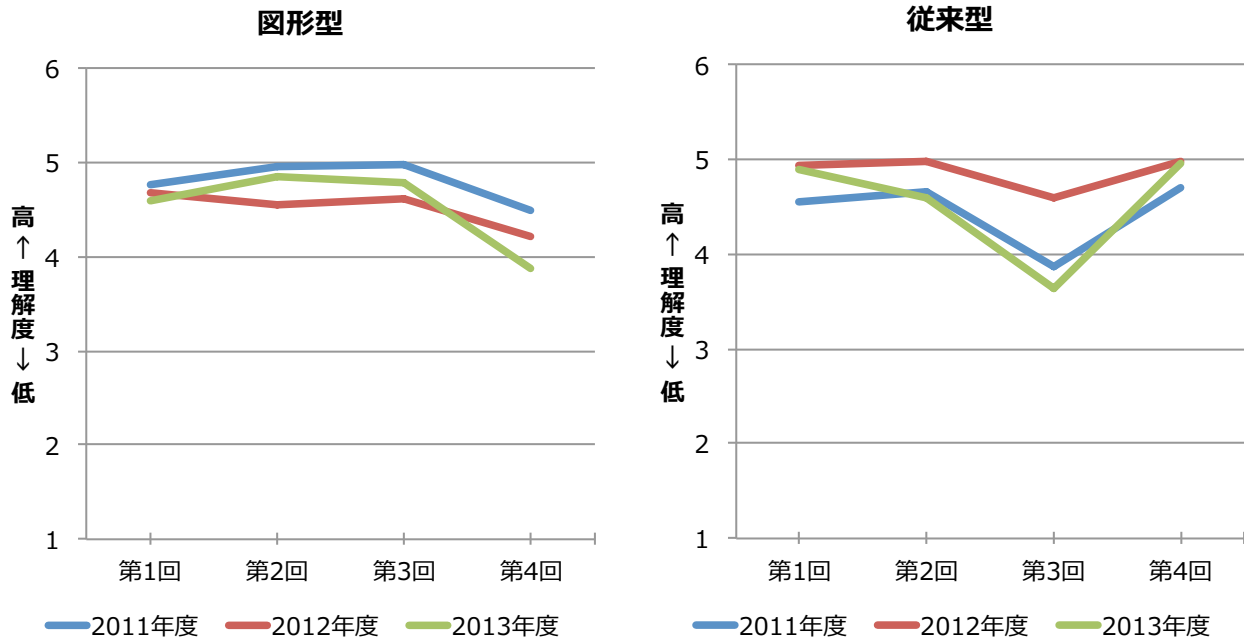


図 2 授業に対する理解度の変化

表 5 「授業に対する理解度」の分散分析表

被験者間要因	自由度	平方和	平均平方	F 値	P 値
学部	1	0.1	0.065	0.021	0.884
年度	2	8.0	4.002	1.317	0.269
授業方式	1	0.5	0.539	0.178	0.674
学部:年度	2	10.6	5.304	1.746	0.176
学部:授業方式	1	12.7	12.734	4.191	0.041 *
年度:授業方式	2	35.4	17.699	5.826	0.003 **
学部:年度:授業方式	2	1.1	0.563	0.185	0.831
誤差	404	1227.4	3.038		
被験者内要因	自由度	平方和	平均平方	F 値	P 値
授業回	3	35.2	11.72	21.55	0.000 ***
授業方式:授業回	3	104.9	34.97	64.31	0.000 ***
誤差	1242	675.4	0.54		

形型コースウェアの特徴である。従来型においても、2012年度は第3回でそれほど理解度が下がってはいなかったが、他の2年については、繰り返し処理のところで「理解できない」と感じた受講学生が増えているという結果になった。従来型の場合、授業内容の難易度がそのまま平均に現れているが、図形型の場合は、その傾向がグラフィックスを用いていることによって希薄化されたのではないかと考えられる。

このことは、プログラミングに対する楽しさの平均変動からも推測できる。楽しさについては、授業回と授業方式の交互作用が有意であった。このことから、授業方式によって授業回の変動パターンが異なっていることが分かる。図5から、従来型の第4回で2011年度と2012年度は楽しさが急に高まったことが分かる。プログラミングが楽しいということは、受講学生の学習意欲が高い、もしくは、

高まりやすい状態にあることを表している。従来型の第4回はグラフィックスを用いた回なので、それまで3回の内容と比較して視覚的に興味を惹きやすいものである。このことが、平均の上昇させたと思われる。

一方、図形型の方は4回を通じて平均がほぼ同じ値となった。はじめからグラフィックスを用いることで楽しさは維持されていると考えられる。繰り返し処理を行う第3回で、図形型は楽しさを維持できているが、従来型は2012年以外で楽しさが下がっている。この点は理解度の結果と同様である。理解度の結果では、図形型の第4回で平均が下がったが、楽しさについてはそれほど大きく下がらなかった。このことは、第3回までグラフィックスを扱うことで学習意欲が高まっていたことに因るとと思われる。

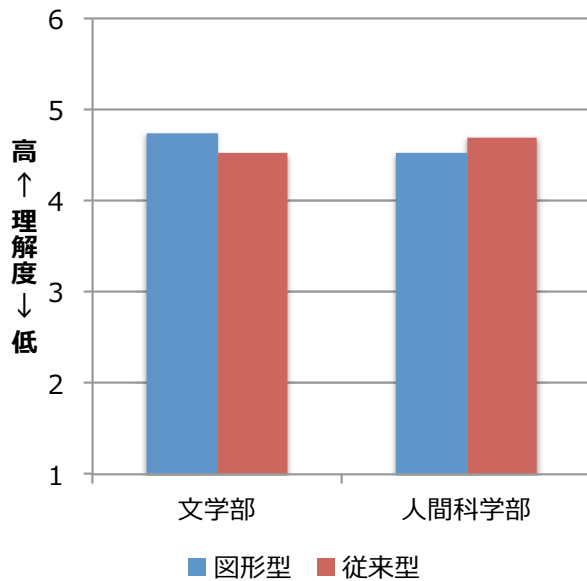


図 3 学部と授業方式の交互作用

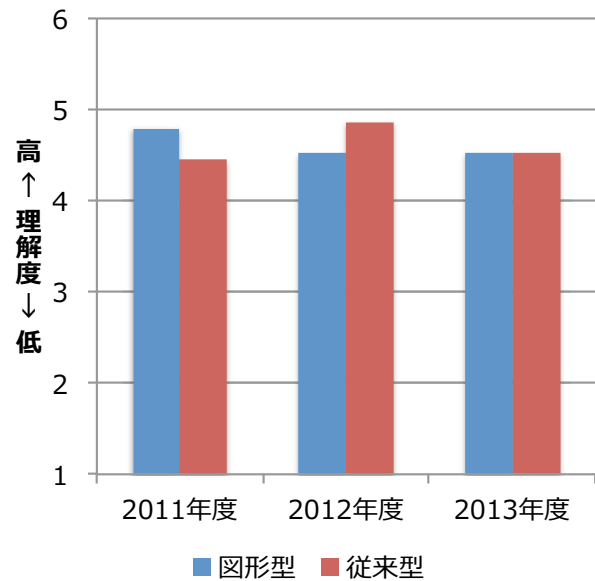


図 4 年度と授業方式の交互作用

表 6 「プログラミングに対する楽しさの自己評価得点」の記述統計量

年度	統計量	図形型				従来型			
		第 1 回	第 2 回	第 3 回	第 4 回	第 1 回	第 2 回	第 3 回	第 4 回
2011 年度	平均	4.22	4.59	4.75	4.60	4.20	4.28	3.91	6.26
	標準偏差	1.66	1.51	1.45	1.41	1.54	1.51	1.51	0.95
2012 年度	平均	4.22	4.53	4.47	4.43	4.38	4.54	4.48	4.80
	標準偏差	1.49	1.38	1.49	1.54	1.50	1.58	1.68	1.40
2013 年度	平均	4.61	4.86	5.00	4.58	4.10	4.16	3.79	4.39
	標準偏差	1.50	1.33	1.41	1.31	1.57	1.69	1.74	1.49

#### 4. まとめ

ここでは、2011～2013 年の 3 年間に渡り、図形描画を主としたコースウェアと、通常のコースウェアとで学生のモチベーションや理解度に差があるのかどうかを調査した。学部や年度の関係からは、コースウェア間の明確な優劣の違いは見つけることはできなかった。しかし、授業内容の変化との対応ではその特徴に差があることがわかり、図形型は繰り返しのようにつまづきやすい学習内容でも理解度や楽しさをほとんど下げることなく学習が行えていることが伺えた。一方で、従来型では繰り返しでは理解度や楽しさを下げる危険性が高いこと、しかし、その後でもグラフィックスのような魅力的な学習内容を提供すれば、それらを再び上げることができる可能性が伺えた。

本研究では、分析の性質上、4 回の授業すべてに出席し Web 調査に回答した受講学生が対象であったが、実際の受講学生数に比べて分析対象者数が少なかった。今後、調査の方法を改善し、分析対象者数を増やす工夫が必要である。また、期末試験の成績の分析や、どのような条件・状況のときにどちらのコースウェアを用いるのが適切かといった検討を行っていく必要があるだろう。

#### 参考文献

- [1] 中村亮太, 西田知博, 松浦敏雄: 初学者向けプログラミング学習環境 PEN Web ページ, <http://www.media.osaka-cu.ac.jp/PEN/>.
- [2] 西田知博, 原田 章, 中村亮太, 宮本友介, 松浦敏雄: 初学者用プログラミング学習環境 PEN の実装と評価, 情報処理学会論文誌, Vol. 48, No. 8, pp. 2736-2747 (2007).
- [3] 安留誠吾, 中西通雄, 景村幸弘: 図形描画によるプログラミング入門-大阪大学人間科学部・文学部での PEN を用いた実践, 情報教育研究会講演論文集, 2008 年度, pp. 531-534 (2008).
- [4] 原田 章, 中西通雄: コンピュータリテラシー教育における習熟度別クラス編成, 情報教育シンポジウム SSS2000 論文集, pp. 91-94 (2000).
- [5] 松浦 敏雄他: 情報活用基礎授業資料とアンケート, <http://lemon.media.osaka-cu.ac.jp/~matsuura/penData/>.

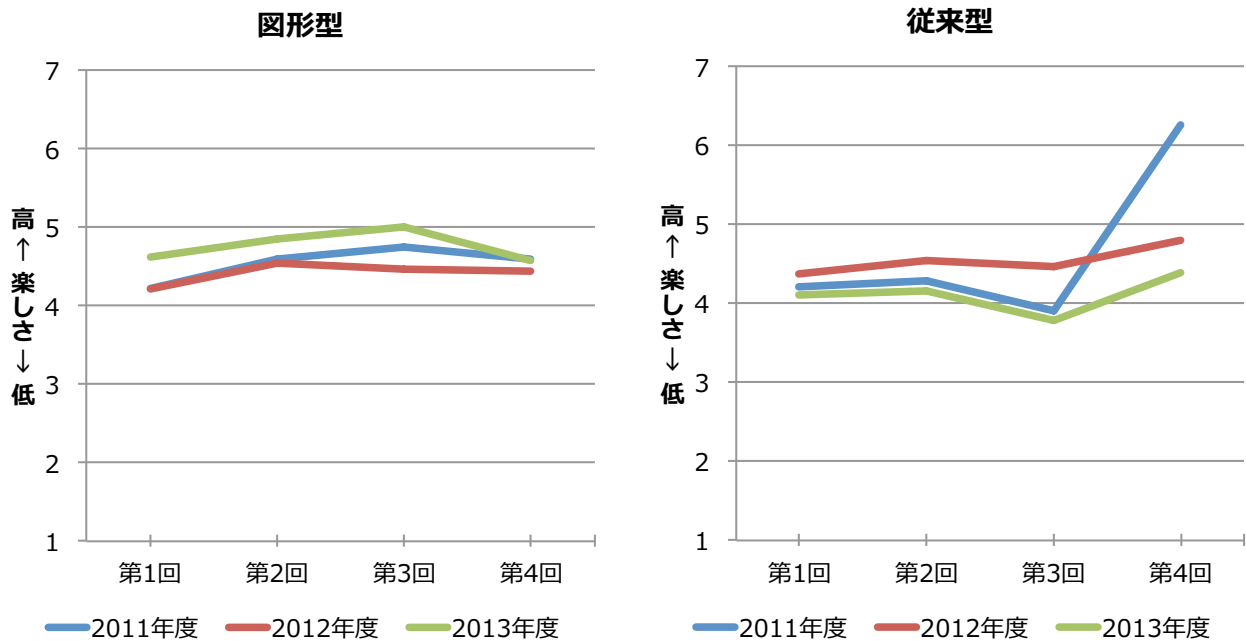


図 5 プログラミングに対する楽しさの変化

表 7 「プログラミングに対する楽しさ」の分散分析表

被験者間要因	自由度	平方和	平均平方	F 値	P 値
学部	1	0.2	0.245	0.035	0.851
年度	2	10.2	5.081	0.737	0.479
授業方式	1	3.8	3.831	0.556	0.456
学部:年度	2	40.0	20.024	2.905	0.056
学部:授業方式	1	21.6	21.621	3.137	0.077
年度:授業方式	2	41.4	20.676	3.000	0.051
学部:年度:授業方式	2	2.3	1.145	0.166	0.847
誤差	404	2784.7	6.893		
被験者内要因	自由度	平方和	平均平方	F 値	P 値
授業回	3	102.1	34.04	45.4	0.000 ***
授業方式:授業回	3	111.7	37.23	49.65	0.000 ***
誤差	1242	931.4	0.75		

## 付 録

### A.1 [1] 従来型コースウェア

#### (1) 第 1 回目の授業内容

[例題 1.1] キーボードから 1 つの整数を入力し、それを 3 倍した値を出力するプログラムを書いてみよう。

(練習 1.01) キーボードから長方形の二辺の長さを入力し、面積を表示するプログラムを書け。

[例題 1.2] キーボードから入力した数が、3 で割り切れるかどうかを調べ、割り切れるなら「3 の倍数である」と出力するプログラムを書いてみよう。

[例題 1.3] キーボードから入力した数が、3 で割り切れるかどうかを調べ、割り切れるなら「3 の倍数」と出力し、そ

うでないなら「3 の倍数でない」と出力するプログラムを書いてみよう。

#### (2) 第 2 回目の授業内容

(練習 1.6) 0~100 の整数の乱数を発生させ、その値が 60 以上の場合「合格」と表示し、そうでない場合は、「不合格」と表示するプログラムを書け。

[例題 1.4] キーボードから点数を入力し、点数が 80 以上なら A, 70 以上 80 未満なら B, 60 以上 70 未満なら C, 60 未満なら F と表示するプログラムを書いてみよう。

(練習 1.10) 西暦年号を入力し、うるう年かどうかを表示するプログラムを書け。

[例題 1.5] コンピュータが考えた 1 桁の整数を当てるゲームを作ってみよう。1 桁の整数の乱数を発生させる ( $x$  とする)。キーボードから 1 桁の整数 ( $k$ ) を入力し、正解でない ( $k \neq x$ ) なら、「違います。もう一度:」と表示して、正解



が入力されるまで繰り返す。正解が入力されたら、「正解です」と表示するプログラムを書いてみよう。

**[例題 1.6]** 0 以上 100 以下の乱数を 10 個表示するプログラムを書いてみよう。

### (3) 第 3 回目の授業内容

**(練習 1.15)** 例題 1.6 のプログラムを修正して、10 個の乱数の平均を表示せよ。小数点以下は切り捨てよ。

**[例題 1.7]** ○を横に 8 個、縦に 6 個ずつ並べて表示してみよう。

**(練習 1.23)** □を 1 行目に 1 個、2 行目に 2 個、…、9 行目に 9 個並べて表示せよ。

**[例題 1.8]** 0 以上 100 以下の乱数を 10 個表示するプログラムを書いてみよう。(for 文を使ったループ)

**[例題 1.9]** 華氏の温度は、次式で摂氏温度に変換できる。

$$\text{摂氏温度} = 5 \div 9 \times (\text{華氏温度} - 32)$$

華氏の温度を入力し、対応する摂氏の温度を表示するプログラムを作ってみよう。

**(練習 1.29)** キーボードから整数(Nとする)を入力し、N の階乗を求めるプログラムを書け。積を保持しておく変数(Fとする)を整数型で宣言した場合と、実数型で宣言した場合でどういう違いがあるかを考察せよ。

### (4) 第 4 回目の授業内容

**[例題 1.10]** 400 × 400 の描画ウィンドウを開き、丸と四角と文字列を描いてみよう。

**[例題 1.11]** 400 × 400 の描画ウィンドウに、半径 25 の 8 つの赤い円を互いに接するように水平に配置せよ。(円の中心の y 座標は 200 とする。)

**(練習 1.33)** 400 × 400 の描画ウィンドウに、半径 25 の緑色の円を中心座標が (25,25), (50,50), (75,75),... となるように対角線に並べよ。

**(練習 1.34)** 400 × 400 の描画ウィンドウに、半径 25 の緑色の円を中心座標が (375,25), (350,50), (325,75),... となるように前問とは逆の対角線上に並べよ。

**[例題 1.12]** 0 以上 400 以下の 2 つの乱数を発生させ (a, b とする), 400 × 400 の描画ウィンドウ内に、中心 (a, b), 半径 20 の緑色の円を 30 個描くプログラムを書け。

**(練習 1.38)** 0 以上 400 未満の 2 つの乱数を発生させ (a, b とする), 400 × 400 の描画ウィンドウ内に、中心 (a, b), 半径 20 の円を 30 個描くプログラムを書け。円の色は乱数で与えよ。

**(練習 1.39)** 0 以上 400 未満の 2 つの乱数を発生させ (a, b とする), 400 × 400 の描画ウィンドウ内に、中心 (a, b) の円を 30 個描くプログラムを書け。円の色は乱数で与えよ。また、円の半径も 30 以下の乱数で与えよ。

## A.2 [2] 図形型コースウェア

### (1) 第 1 回目の授業内容

**[例題 1.1]** キーボードから半径を入力し、200 × 200 の描画ウィンドウの中央にその半径で緑の円を描いてみよう。

**(練習 1.2)** キーボードから横と縦の長さを入力し、その大きさの長方形を描画ウィンドウの中央付近(長方形の左上が (100,100) の位置)に青色で描くプログラムを書け。

### (2) 第 2 回目の授業内容

**[例題 1.2]** キーボードから入力した数が、3 で割り切れるかどうかを調べ、割り切れるなら描画ウィンドウの中央に半径 50 の緑の円を描いてみよう。

**[例題 1.3]** キーボードから入力した数が、3 で割り切れるかどうかを調べ、割り切れるなら描画ウィンドウの中央に半径 50 の緑の円を描き、そうでないなら赤い円を描くプログラムを書いてみよう。

**(練習 1.5)** 0~100 の整数の乱数を発生させ、その値が 60 以上なら、画面中央に半径 50 の緑色の円を描き、60 未満の場合は、半径 50 の赤色の円を描くプログラムを書け。

**[例題 1.4]** キーボードから点数を入力し、描画ウィンドウの中央に点数が 80 以上なら半径 50 の緑の円を、70 以上なら半径 50 の青の円を、60 以上なら半径 50 の黄の円を、60 未満なら半径 50 の赤の円を描くプログラムを書いてみよう。

**(練習 1.9)** 西暦年号を入力し、うるう年かどうかを判定し、うるう年なら、画面中央に半径 50 の青い円を表示し、そうでないなら、半径 50 の赤い円を表示するプログラムを書け。

### (3) 第 3 回目の授業内容

**[例題 1.5]** コンピュータが描いた円の半径を当てるゲームを作ってみよう。10 以上 20 以下の乱数  $r$  を発生させ、400 × 400 の描画ウィンドウの中央に半径  $r$  の青い円を描き、描かれた円を見てキーボードからその半径と思われる値 ( $k$ ) を入力し、正解でない ( $k \neq r$ ) なら、「違います。もう一度:」とコンソールに表示して、正解が入力されるまで繰り返す。正解が入力されたら、「正解です」と表示するプログラムを書いてみよう。

**[例題 1.6]** 0 以上 400 以下の 2 つの乱数を発生させ (a, b とする), 400 × 400 の描画ウィンドウ内に、中心 (a, b), 半径 20 の緑色の円を 30 個描くプログラムを書け。

**(練習 1.14)** 0 以上 400 未満の 2 つの乱数を発生させ (a, b とする), 400 × 400 の描画ウィンドウ内に、中心 (a, b), 半径 20 の円を 30 個描くプログラムを書け。円の色は乱数で与えよ。

**(練習 1.15)** 0 以上 400 未満の 2 つの乱数を発生させ (a, b とする), 400 × 400 の描画ウィンドウ内に、中心 (a, b) の円を 30 個描くプログラムを書け。円の色は乱数で与えよ。また、円の半径も 30 以下の乱数で与えよ。

**[例題 1.7]** 400 × 400 の描画ウィンドウに、半径 25 の 8 つの赤い円を互いに接するように水平に配置せよ。(円の中

心の  $y$  座標は 200 とする.)

#### (4) 第 4 回目の授業内容

[例題 1.8]  $400 \times 400$  の描画ウィンドウに, 5 段に渡って, 半径 25 の 8 つの赤い円を互いに接するように水平に配置せよ.

(練習 1.23)  $400 \times 400$  の描画ウィンドウに, 1 段目に 1 つ, 2 段目に 2 つ, ..., 8 段目に 8 つ半径 25 の赤い円を配置するプログラムを書け.

[例題 1.9] 0 以上 400 以下の 2 つの乱数を発生させ ( $a, b$  とする),  $400 \times 400$  の描画ウィンドウ内に, 中心 ( $a, b$ ), 半径 20 の緑色の円を 30 個描くプログラムを書け.

[例題 1.10](実数の扱い) 0 以上 400 以下の 2 つの乱数を発生させ ( $a, b$  とする),  $400 \times 400$  の描画ウィンドウ内に, 中心 ( $a, b$ ), 半径 20 の緑色の円を  $N$  個 ( $N=100$ ) 描くプログラムを書け. また,  $N$  個の円の中心の  $x$  座標および  $y$  座標のそれぞれの平均を求めよ

(練習 1.27) キーボードから整数 ( $N$  とする) を入力し,  $N$  の階乗を求めるプログラムを書け. 積を保持しておく変数 ( $F$  とする) を整数型で宣言した場合と, 実数型で宣言した場合でどういう違いがあるかを考察せよ.