

開講コマの違いによる学習進捗の相違について — 自学実習ベースのプログラミング教育の場合 —

河村一樹^{†1}

概要: 担当科目であるプログラミング実習において、自学自習と個別指導による授業を行っている。2015年度と2016年度の授業進捗を比較すると、2016年度の方が向上したという結果を得た。両年度もプラットフォーム（グループウェアからCMSへ）の機能はほぼ同じであったが、開講コマだけが異なった。2015年度は連続コマ、2016年度はペアコマであった。これより、ペアコマの方が学習進捗が速く、学習効果が高いことが明らかになった。

キーワード: クォーター制, プログラミング教育, JavaScript, Moodle, 学習進捗

About differences in learning progress due to differences in opening courses - In case of programming education based on self-study exercise-

KAZUKI KAWAMURA^{†1}

Abstract: In programming practice which is the subject, we conduct self-study and individual instruction. Comparing the lesson progress in 2015 and 2016, we got the result that improvement in 2016 was improved. The functions of the platform (Groupware to CMS) were almost the same in both fiscal years, but only the starting commentary was different. Continuous coma in 2015 and pair coma in 2016. From this it became clear that the learning progress is faster and the learning effect is higher for the paired coma.

Keywords: Quarter system, Programming Education, JavaScript, Moodle, Learning progress

1. はじめに

筆者は、2010年から、ゼミナールあるいは実習科目において、プログラミング教育を担当している[1][2]。科目「プログラミング実習（2016年度からは「プログラミング基礎」に改名）」は、1コマ90分間で、週2コマの開講である。

本科目では、Windowsマシンに標準搭載のエディタ（メモ帳）とブラウザ（Internet Explore）さえあれば実行できること、および、より実用性のあるプログラミング（ホームページに動的振舞いの実装）を経験させたいことからHTMLにJavaScriptを組み込む形で実習を進めてきた。

2013年度は2コマ連続であったこともあり、1コマ目は講義、2コマ目は実習という伝統的な形態[3]で授業を進めた。しかし、思ったよりもクラス全体の進捗度合が低かったことだけでなく、学生同士のコピペ問題が発覚した。

このため、2014年度からは、講義を一切せずに、2コマ（連続）とも自学自習ベースの授業形態に変更した。学生による自学自習では、こちらで指定した教科書[4]を講読することになる。教科書の構成（部分的に抜粋）は、付録A.1の通りである。章・節毎に、複数のJavaScriptの例題プログラムを掲載しており、構文の文法や書き方などについて説明している。各節末には実習課題を載せており、それらを

実習課題として各人でプログラミングとデバッグを繰り返すことになる。

ただし、これだけでは自学自習がしにくいと思われ、全実習課題のヒント集と実行結果の画面を用意した。ヒント集については、専用の電子掲示板を作成し、そこで公開した。これによって、学生全員がいつでも閲覧できるだけでなく、わからないことがある場合は各ヒントのスレッドに対してReplayをすることで質問できるようにした。

また、全プログラムの実行結果については、学内の共有フォルダで公開した。これによって、実習課題の文章だけではわかりにくい場合、どういう画面表示となればよいのかについて視覚的にわかるようにした。

授業中は個別指導（教員と学生が1対1で対面し、質疑応答を行う）だけを行い、授業外は自学自習を推奨した。この狙いには、プログラミングスキルは学生個々により異なるため一斉講義では対応しにくいこと、わからないことがわかるまで徹底的に個別に指導すること、自分で理解しながらプログラミングすることによって達成感を味合わせたいことなどがあった。その結果、コピペ問題は回避することができた[5]。

一方、授業外の自学自習があまり行われていないことが

^{†1} 東京国際大学
Tokyo International University

明らかになった。この要因としては、実習課題を終えても次の授業まで最大1週分の間があり、提出できないことによるモチベーションの低下が考えられた。

そこで、2015年度からは、グループウェアである「サイボウズLive」[6]を利用することにした。サイボウズLiveでは、グループ毎に、イベント、ToDoリスト、掲示板、共有フォルダといった機能がある。この中の「共有フォルダ」に、学生個々人の実習課題（JavaScriptのソースコードと実行時の入出力データ）をアップロードさせることにした。アップロードの日時はログデータに記録されるので、これを提出日時とすることにした。これによって、学生も提出を待つことなく、できた時点でアップロードできるようになるとともに、実習課題の提出日時を評価に勘案することをあらかじめ示した。

これに合わせて、教員と全学生のグループを登録し、この中の「掲示板」においてクラス全体への連絡や全員の進捗状況を公開した。また、教員と学生個人だけのグループをそれぞれ登録し、この中の「共有フォルダ」において学生の実習課題の提出を、「掲示板」では学生との連絡や授業外での質疑応答を行った。

2016年度には、大学全体として教務制度に大幅な改変が行われた。一つはクォーター制の採用、もう一つはCMSとしてのMoodleの運用開始である。

クォーター制については、他大学と同様に、授業を短期間に集中することで教育効果を向上させること、履修を工夫することで短期留学・ボランティア活動・インターンシップ活動に参加できるようになることを目指したといえる[7]。これによって、2015年度までは2コマ連続（月曜日4・5コマ）であったが、2016年度からはペアコマ（月曜日4コマ・木曜日4コマ）の開講となった。

Moodleについては、大学全体として教材管理や授業運営のICT化を進めるという方針が打ち出されたことにより導入された。そこで、本科目においては、サイボウズLiveの代わりに、Moodleを利用することとした。サイボウズLiveの「掲示板」機能についてはMoodleの「フォーラム」機能を使うことに、サイボウズLiveの「共有フォルダ」機能についてはMoodleの「コース」内の「トピック」における「課題」を使うことにした。これによって、実習課題の提出については、2015年度と2016年度もほぼ同じように実施できるようになった。

以上のような状況のもとで、2016年度も本科目を実施したが、2015年度に比べて、クラス全体としての進捗状況が2016年度の方が向上したことが明らかになった。そこで、本稿では、2015年度と2016年度の授業内容を比較し、その差異がどのような影響を与えたのかについて考察する。

2. 年度別の比較

2015年度と2016年度の本科目の実施状況を比較したものを、表1に示す。その中の差異分について取り上げる。

表1 年度別の比較

Table 1 Comparison by year.

分類	2015年度	2016年度	差異
科目名	プログラミング実習	プログラミング基礎	無
開講コマ	連続コマ（月曜日4・5コマ）	ペアコマ（月曜日4コマ、木曜日4コマ）	有
教育目標	プログラミングの基礎と実用的な処理の習得	同左	無
授業形態	自学自習+個別指導	同左	無
教科書	自著	同左	無
実習課題	自著内で指定	同左	無
副教材（その1）	各実習課題のヒント集	同左	無
副教材（その2）	各実習課題の実行結果（画面イメージ）	同左	無
実習の言語	JavaScript	同左	無
実習の環境	編集（テキストエディタ）+実行（ブラウザ）	同左	無
実習の進め方	新たにhtmlとJavaScriptをプログラミング	あらかじめ実習課題毎にhtmlファイル（JavaScriptを埋め込む）を用意	若干有
副教材（その1+その2）の提供	サイボウズLiveの「クラス全体」の「掲示板」で一括公開	Moodleの「トピック」毎に公開	若干有
実習の提出	サイボウズLiveの「グループ」別「共有フォルダ」に、ソースコードと実行結果を1ファイルにしてアップロード	Moodleの「トピック」の「課題」に、ソースコードと実行結果を1ファイルにしてアップロード	無
実習の進捗	サイボウズLiveの「クラス全体」の「掲示板」で公開	Moodleの「ニュースフォーラム」で公開	無
実習の質問	サイボウズLiveのクラス全体用「掲示板」で公開	Moodleの「質疑応答用フォーラム」で公開	無

(1) 科目名について

名称の変更はあったが、同一科目である。

(2) 開講コマについて

まさに、ここが最も大きな差異といえる。2015年度までは、次の授業まで1週間の間があいていたわけだが、2016年度のクォーター制導入により、2~3日間と間隔が狭まったことになる。

(3) 実習の進め方について

2015年度については、実習課題毎に、新規にHTMLのソースコード中にJavaScriptを埋め込む形でプログラミングを行わせた。しかし、このやり方では、授業当初の頃（学生がプログラミングに不慣れな時期）、HTMLのソースコードのタイプミスなどによって構文エラーを引き起こし、肝心のJavaScriptのデバッグができないという事態が生じた。

そこで、2016年度では、こちらで、あらかじめ実習課題

毎に、HTML のソースコードだけを入力した html ファイルを用意した。その中のコメント行の箇所に、JavaScript を入力するように指示した。その html ファイルは次の通りである。

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"
"http://www.w3.org/TR/html4/strict.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html;
charset=utf-8">
<title>自分の名前 問題番号</title>
</head>
<body>
<h1>問題番号</h1>
<p>
<script type="text/javascript">
<!--
/* ここに JavaScript のソースコードを埋込むこと */
-->
</script>
</p>
</body>
</html>

```

(4) 副教材 (その1+その2) の提供について

自学自習をベースにするため、教科書の講読だけでは実習が進まないという懸念があったので、別途副教材を用意した。実習課題毎に、プログラミングを行うためのヒント (その1) (図1の下部) と実行結果としての画面イメージ (その2) である (図1の上部)。

2015年度については、全実習課題の副教材を、教科書の章・節毎に PDF に変換し、それらを zip ファイルにまとめた。それを、サイボウズ Live のクラス全体のグループの「共有フォルダ」にアップロードした。これによって、学生が zip ファイルをダウンロードして解凍することで、実習課題毎の副教材を参照できるようにした。

2016年度では、Moodle のクラス「プログラミング基礎」において、章・節毎に「トピック」を設定し、その中に、「リソース」として「ヒントと実行結果」(pdf ファイル) を、「活動」として「課題」((3)の html ファイル) を、それぞれアップロードした。例えば、第5章の1節「画面への出力」では、図2のようになる。

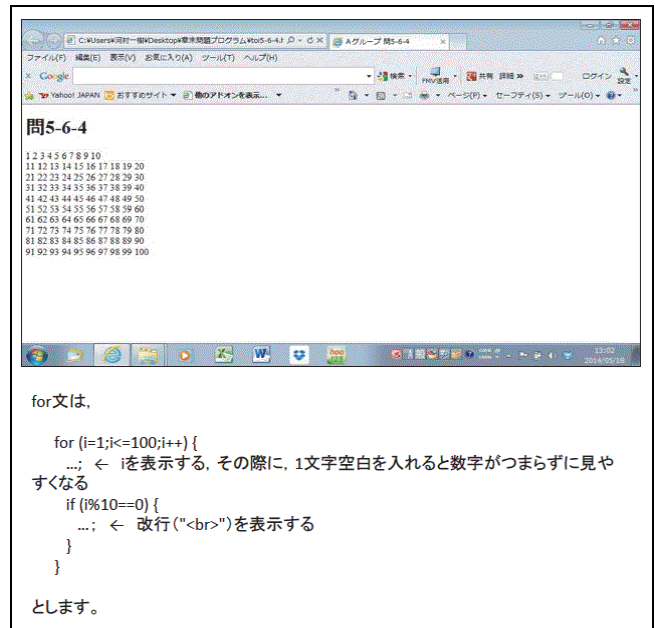


図1 副教材

Figure 1 Supplementary teaching material

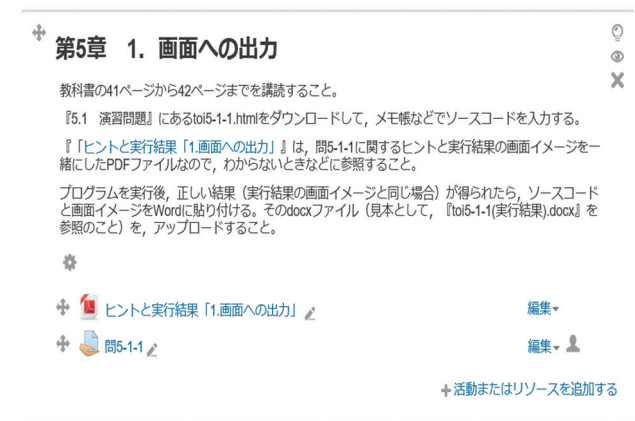


図2 Moodle による副教材の提供

Figure 2 Providing supplementary teaching materials by Moodle

3. 実施結果の比較

実習環境としてのプラットフォームは、グループウェアとしての「サイボウズ Live」から、CMS としての Moodle に変わった。しかし、表1にも示したように、基本的にはほぼ同じ機能が使えているので、2015年度と2016年度の差異はないとしてよい。最も大きな差異は、開講コマであった。このことを踏まえた上で、両年度の実習結果およびアンケートの比較を行う。

(1) クラス全体の課題達成度

年度毎のクラス全体の課題達成度を、学生一人当たりの提出数で比較した (表2)。なお、参考として、2013年度からのデータも付記した。

表 2 年度別の課題提出数

Table2 Number of task submissions by fiscal year

年度	提出数	履修者数	学生一人当たりの提出数
2013	481	19	25.3
2014	394	14	28.1
2015	527	15	35.1
2016	623	15	41.5

これより、年度毎に、学生一人当たりの実習課題の提出数

が増えていることがわかる。なお、2013-2015年度は、2コマ連続開講である。また、2013年度は1コマ講義後に1コマ実習を、2014年度からは2コマとも自学自習と個別指導とした。

(2) 課題の提出状況

2015年度と2016年度における毎週のプログラム提出累計数(図3)、および、実習課題毎の提出率(図4)についてまとめた。

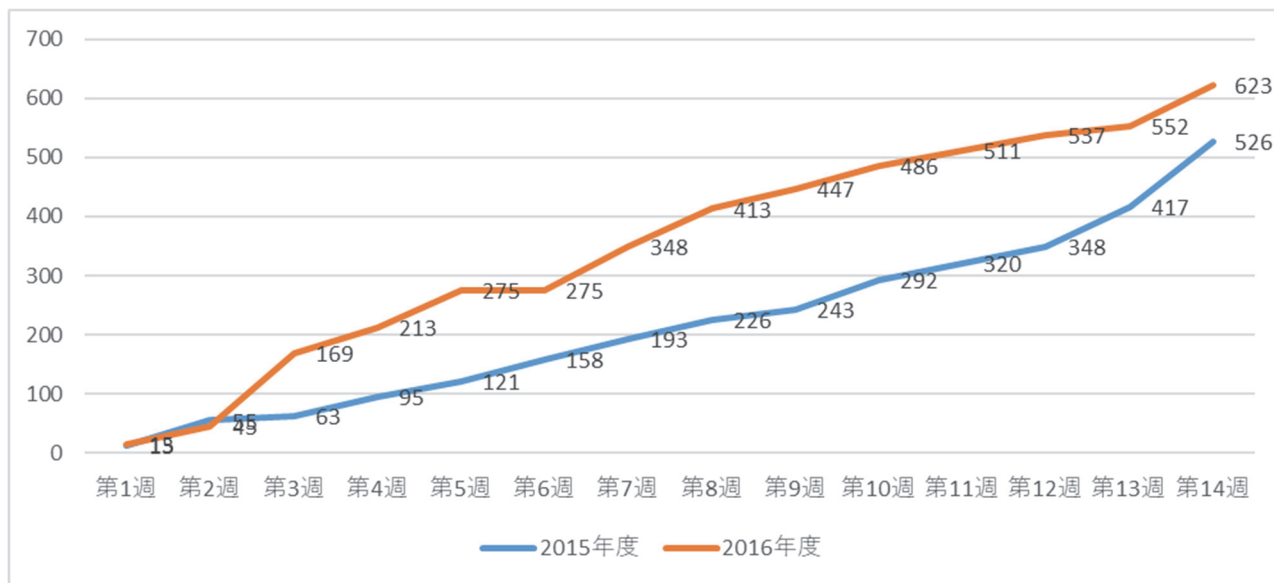


図 3 毎回の提出累計数の推移

Figure 3 Trend of cumulative number of submissions every time

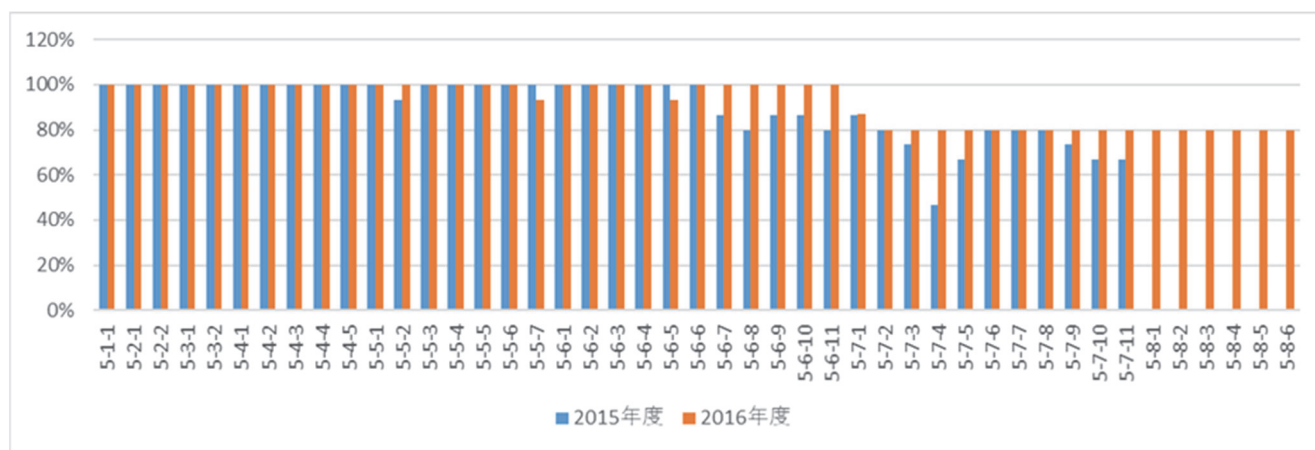


図 4 実習課題毎の提出率

Figure4 Submission rate for each practical assignment

図 3 より、第 3 週目以降から 2016 年度の方が実習課題の提出率(問 5-1-1 から問 5-8-6 まで)が向上していることが明らかになった。また、図 4 より、2015 年度は問 5-7-11 までしか到達できなかったが 2016 年度は問 5-8-6 まで 8 割の学生ができたこと、2016 年度は問 5-7-1 以降 2 割

(15 人中 3 人)の未提出だけで済んだこと、などが明らかになった。なお、問 5-8-6 を終えた学生は、さらに先の課題に取り組んだ。

表 3 は 2015 年度、表 4 は 2016 年度の学生毎の実習結果である。

表3 2015年度の実習結果

Table3 Practical training result in 2015

学籍番号	学年	学科	提出日*	総提出数
11140037	4年	情ビ	7月18日	31
12110148	4年	商	7月20日	31
13150003	3年	経営	—	23
13150013	3年	経営	—	23
13150038	3年	経営	—	28
14150014	2年	経営	7月20日	31
14150040	2年	経営	7月13日	31
14150113	2年	経営	7月13日	31
15110185	1年	商	7月13日	31
15150014	1年	経営	7月21日	31
15150020	1年	経営	7月20日	31
15150036	1年	経営	7月20日	31
15150098	1年	経営	7月20日	31
15150106	1年	経営	7月13日	31
15151009	1年	経営	—	29

*問5-7-11を提出した日付

2015年度については、問5-7-11までの提出について、73%の学生が半期終了前の第13週か14週であり、31問まで終了した。一方、4人の学生は、問5-7-11まで至らなかった。

表4 2016年度の実習結果

Table4 Practical training result in 2016

学籍番号	学年	学科	提出日*	総提出数
14110049	3年	商	7月25日	46
14110105	3年	商	7月28日	45
14110166	3年	商	7月21日	47
14110190	3年	商	7月25日	45
14111039	3年	商	7月11日	45
14151003	3年	経営	—	29
14151016	3年	経営	—	27
14151018	3年	経営	—	27
14151023	3年	経営	7月27日	45
14151027	3年	経営	7月11日	45
15110214	2年	商	7月25日	45
15115013	2年	商	6月20日	56
15150028	2年	経営	6月20日	51
15151001	2年	経営	7月25日	45
15155019	2年	経営	6月13日	56

*問5-8-6を提出した日付

2016年度については、問5-8-6までの提出について、最も早い学生(15155019)が第8週目、次の学生(15115013, 15150028)が第9週目であった。これら3人は、いずれも50問以上の実習課題を提出した。それ以外の学生は、半期終了前の第12-14週となった。一方、3人の学生は、問5-8-6まで至らなかった。

(3) アンケートの比較

授業がすべて終了した時点で、2015年度および2016年度と、同じ内容のアンケートを実施した。その結果について、以下に示す。

● 授業スタイルについて

講義を一切せずに、自学自習と個別指導だけという授業のスタイルについて、学生がどのように感じているかについて質問した(図5)。

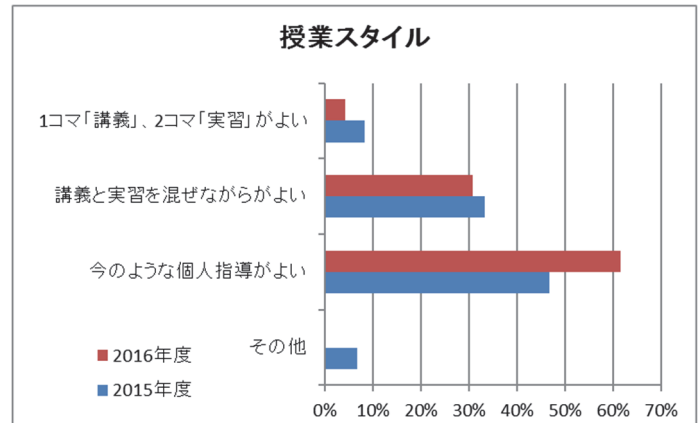


図5 授業スタイル

Figure5 Class style

これより、いずれも旧来の「講義」と「実習」という区分をつけるのではなく、自学自習と個別指導の方がいいという反応であった。また、2016年度の方が6割強の学生がこの授業スタイルを推奨していることがわかった。

● 自習方法について

自学自習をどこでいつ行っているかについて質問した(図6)。

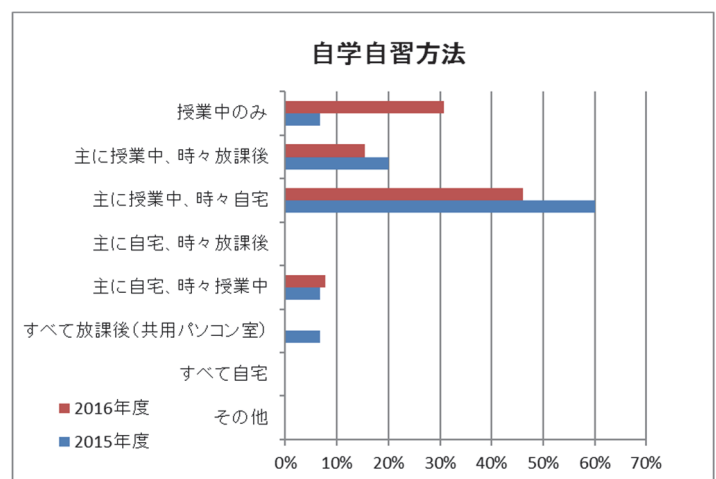


図6 自学自習方法

Figure6 Self-study method

これより、2015年度は授業中だけでなく放課後や自宅でもある程度自学自習をしていたが、2016年度は授業中だけ自学自習を行っている割合が高いことが明らかになった。

● 実習課題のヒントについて

副教材である実習課題のヒントをどの程度参照している

かについて質問した（図7）。

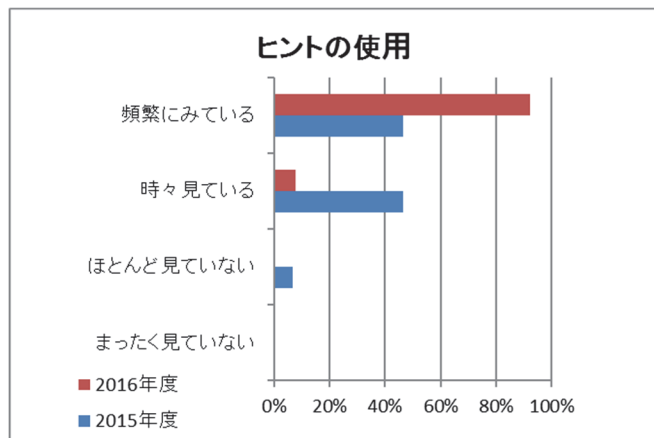


図7 ヒントの使用

Figure7 Use of hints

これより、2015年度よりも2016年度の方がヒントを頻繁に見ていることがわかる。

● プラットホームについて

2015年度はサイボウズ Live を、2016年度は Moodle を、それぞれ実習を円滑に行うためのプラットフォームとして使ったわけだが、その必要性について質問した（図8）。

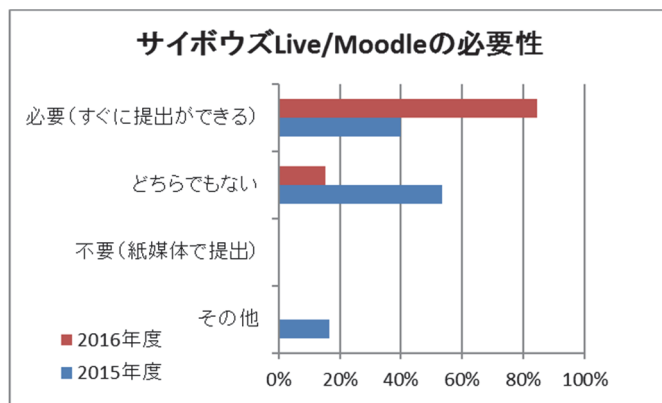


図8 プラットホームの必要性

Figure8 Necessity of platform

これより、プラットフォームの必要性については、2016年度の学生の方が賛同していることがわかった。

(4) 結果の考察

(1)から(2)までの結果から、2015年度よりも2016年度の方が実習の進捗結果が向上していることが明らかになった。その要因については、表1にある差異に着目する必要がある。

差異が「若干有」についてだが、「実習の進め方」において、2015年度はHTMLとJavaScriptを新規に入力、2016年度はJavaScriptのソースコードだけの入力とした。これによって、HTMLコードの構文エラーが無くなったことで実習がスムーズに進んだことが考えられる。しかし、2015年度においても、学生が実習に慣れてくると、自分のソース

コードをコピーして別の実習課題に利用するので、入力作業の手間はほとんど変わらないといえる。

「副教材の提供」については、プラットフォームの差はないといってよい。サイボウズ Live の「掲示板」と Moodle の「フォーラム」は、機能として同じ働きをするからである。ただし、提供の仕方が異なり、2015年度は全ファイルを一括して ZIP 形式に圧縮し、クラス全体の掲示板にアップロードした。学生は、それをダウンロードし解凍する必要があるが、教室のパソコンだけでなく他（共有パソコン室や自宅）のパソコンでも行う必要がある。また、副教材を閲覧する場合、ダウンロード先のフォルダに移行しなければならないという手間がかかる。

2016年度は、Moodle のトピックを教科書の章・節に対応させ、そこに副教材をアップロードした（図2）。このため、学生が自分の進捗に合わせて実習課題をダウンロードするとともに、対応した副教材をすぐに参照できるようにした。その結果、図7のように、2016年の方が、ヒントの使用頻度が高い結果となった。

サイボウズ Live と Moodle の必要性については、図8のような結果を得た。これより、2016年度の方が課題提出の時期について意識していたことがわかる。これは、2016年度の個別指導において、学生個人の進捗を毎回確認していたことによる。また、Moodle の必要性の方が高くなった理由としては、サイボウズ Live は本科目だけの使用であったが、Moodle は全学的に利用することが推奨されており他の科目でも Moodle を使う機会があることがあげられる。このため、学生も Moodle の操作に慣れていたことで、実習もスムーズに進んだといえる。

両年度も自学自習で進めたが、2016年度の方が授業中に行っている割合が高かった。これについては、ペアコマにより2-3日毎に授業があったため、1週間の間がある連続コマより授業外で自学自習する機会が少なかったことが考えられる。

差異が「有」についてだが、表2・図3・図4の結果を見ると、明らかに2015年度よりも2016年度の方が、学生の実習課題の達成度が高かった。上述したように、プラットフォームの差異はそれほど影響がないとすると、開講コマの差（連続コマ対ペアコマ）が最も大きな要因となる。

ペアコマの場合、授業が2-3日間毎に繰り返されるので、自学自習をしている中でわからないことがあると、次の授業における個別指導で解決できる。この結果、1週間待つよりもその分早く実習課題をこなすことができ、学生の実習進捗度合いが高まったといえる。

自学自習においてわからないことを授業外に質問できるようにと、サイボウズ Live では掲示板を、Moodle ではフォーラムを、それぞれ用意して対応するつもりだったが、実際には両年度ともあまり使われていなかった。これは、授業中に直接教員と話し合った方が理解しやすいことや電

子コミュニケーションで行うよりも手間が省けるからといえる。また、掲示板やフォーラムだと、コミュニケーションの内容が他の学生にも公開されることを、あまり好まない学生もいたのかもしれない。

以上のような自学自習と個別指導を前提としたプログラミング教育においては、連続コマよりもペアコマの方が学生の学習進捗が高いことが明らかになった。

4. おわりに

本科目において、2015年度と2016年度、ほぼ同じ状況（自学自習と個別指導、同じ機能のプラットフォームを利用）のもとで授業を行ったが、2016年度の方が学生の実習課題の達成度合いが高いことが明らかになった。その要因は、連続コマとペアコマの差異であると結論づけた。

一方、両年度とも、自学自習を、授業外よりも授業内でやることの方が多くことがわかった。今後の課題としては、授業外でも積極的に自学自習をするような仕組み作りを検討したい。

具体的には、学生のプログラミングの早さを測定することが考えられる。提出日時の早さだけでなく、一つの実習課題をどのくらいの期間で実施したかを把握することで判定できる。つまり、学生が実習課題をアップロードした日時だけでなく、ダウンロードした日時をもログで取得し、その差分を（目安としての）課題実施時間とする。この時間が短いほど、早く実習を行っているとなすことができる。そのために、Moodleの「トピック」において、「課題」ではなく「小テスト」を用いる。「小テスト」の場合、開始日時と終了日時がログとして記録されるからである。

参考文献

- [1] 河村一樹：情報ビジネス学科におけるプログラミング教育，東京国際大学論叢商学部編，第84号，pp23-39，2011年
- [2] 河村一樹：一般情報教育におけるプログラミング教育のあり方について，情報処理学会研究報告，Vol.2011-CE-108(16)，2011年
- [3] 田中宏和：アクティブラーニングによるプログラミング教育法の提案 基幹系業務システム開発演習の実践事例，日本情報経営学会誌，Vol.36，No.2，pp.63-73，2015年
- [4] 河村一樹：JavaScriptによる情報教育入門，大学教育出版，2011年
- [5] 河村一樹：講義レスによるプログラミング実習教育の試み，情報処理学会研究報告，Vol.2015-CE-128(21)，2015年
- [6] Cybozu Live ホームページ
<https://cybozu.co.jp/customer/>
- [7] 広島大学スーパーグローバル大学創成支援事業「クォーター制の導入」
http://hiroshima-u.jp/sgu/page03_01

付録

付録 A.1 教科書の構成

章・節タイトル	例題数	実習課
第5章 JavaScriptの基本編		
1. 画面への出力	1	1
2. 変数の扱い	2	2
3. 演算式の扱い	1	2
4. 画面からの入力	1	5
5. 選択文 if文, if else文, switch文	3	7
6. 繰り返し文 for文, while文, do while文, break文	4	11
7. 配列の扱い 配列, 連想配列	2	11
8. 関数の扱い 関数, 変数のスコープ	2	6
第6章 JavaScriptの応用編		
1. 整列のアルゴリズム 選択ソート, バブルソート	2	7
2. 探索のアルゴリズム 線形探索, 2分探索	3	4
3. 再帰のアルゴリズム 階乗, ハノイの塔	2	3
第7章 JavaScriptの実用編		
1. ウィンドウの扱い 確認ウィンドウの表示, ウィンドウのサイズ指定/揺れ/フェードイン・フェードアウト, 左右スクロール, 一定間隔スクロール	6	6
2. 文字の編集 文字のフェードイン/グラデーション/色変化, 1文字ずつ色変化, 文字のスクロール, 文書を1文字/1行ずつ表示	7	7
3. 画像の編集 クリックによる/一定時間毎に画像の入れ替え, 画像のフェードイン/揺れ	4	4
4. フォーム テキストボックス, チェックボックス, ラジオボタン, リストボックス	4	4