Vol.2016-CE-134 No.26
2016/3/6

グループウエアを用いたプログラミング実習教育

河村一樹[†1]

概要：2014 年度のプログラミング実習教育では，講義を一切せずに個別指導と自学自習という形で進めたが，クラス全体の進捗度合が低いという課題が残った。そこで，今年度はグループウェアを導入し，いつでも実習課題を提出できるようにした。その結果，前年度よりも進捗度合が向上した一方で，教室外での実習があまり行われていないことが明らかになった。

キーワード：プログラミング実習，JavaScript，グループウェア

Programming Training Education using Groupware

KAZUKI KAWAMURA[†1]

Abstract: *Programming Training Education at Fiscal year 2014, I tried tutoring and Self-education without lecture. The result, the overall progress degree of my lesson class was slow. So, in this fiscal year I introduced the groupware. Then the progress degree was improved more than the previous year. On the other hand, most students didn't practice outside the classroom.*

Keywords: Programming Training, JavaScript, Groupware

1. はじめに

情報システム学科（その後，情報ビジネス学科に改組）において，2001 年度からプログラミング教育に取り組んできた[1]。そのときのプログラミング教育の目的としては，基本的なアルゴリズム（整列，探索，再帰）を，プログラミング言語を使って実装できることとした。

当初は，1 年次の必修科目「演習(1)」（通年，1 コマ）において，チュートリアル形式での演習を行っていた。使用言語は C 言語としたが，実用的に使いこなせるまでに至らず，基本的なプログラミングの習得で終わっていた。この要因に，演習の時間が 1 コマしかないこと，C 言語の場合関数の扱いまで学ばないと実用的なプログラミングができないことなどが考えられた。

そこで，プログラミングよりもアルゴリズムの習得に専念できるようにと JPADet（Japanese PAD editor and interpreter）という教育支援ツールを独自に開発した[2][3]。これは，日本語と構造化チャートである PAD を用いてアルゴリズムを図化し，インタプリタ方式で実行するという仕組みを提供するツールである。これを「演習(1)」において適用し，学習効果の測定を行った[4]-[6]。この結果，アルゴリズムの習得状況やレベルは向上したが，学生の関心はアルゴリズムよりもプログラミング言語の習得にあるらしく，あまり興味を示さないことが顕著になった。

このため，2010 年度の「演習(1)」からは，プログラミング言語というよりはスクリプト言語を取り上げ，より実用的なプログラミングの演習に取り組めるように授業内容を変更した。具体的には，学生にとっても身近にアクセスできるホームページの制作に絡めて，HTML と CSS，そして JavaScript を採択した。授業の進め方としては，前期に HTML と CSS によるページデザインを主に演習し，後期に HTML に JavaScript を組み込む形でプログラミングの演習を行うこととした。これらのスクリプトの実装環境はブラウザ（Internet Explorer であればデバッグ機能あり）とテキストエディタ（Tera Pad）さえあればよく，大学だけでなく自宅等でもプログラミングができるという利便性に着目した[7]。

その後，学部改組を受けて，科目担当が一部変更になり，2013 年度から新たに「プログラミング実習」（前期，2 コマ）を受け持つことになった。これに合わせて，これまでの「演習(1)」は，全学共通の初年次教育に位置づけられアカデミックスキル（リーディング・ノートテイキング・情報検索・ライティング・プレゼンテーションの各スキル）習得の内容に変更となった。

「プログラミング実習」では，JavaScript だけに絞り，基礎（変数，基本構文，関数）から応用（整列・探索・再帰ア

†1 .東京国際大学
Tokyo International University.

ⓒ2016 Information Processing Society of Japan

ルゴリズム）そして実用（ウィンドウ・文字・画像編集，フォーム）まで含めたプログラミングの演習を行うことにした。これによって，学生も自分用の公開ホームページに動的な振舞いを取り込むことができるので，学生の興味の度合もアップすることが期待された。

2013 年度に実施した「プログラミング実習」において，新たな問題が生じた。それは，期の後半あたりから落ちこぼれてしまった学生によるプログラムのコピペが目立つようになったことである。このため，期の最後に，個人面談を行い事情聴取したところ，数名の学生によるコピペが明らかになった。コピペの理由については，授業の進捗についていけない，わからないことがわからないままになる，独力でやろうとしてもできないので他人に頼らざるを得ない，その結果，提出期限に間に合わせるためにコピペしたということであった。

プログラミング実習では，ソースコードが簡単にコピーできてしまうこと，簡単なプログラムであればほぼ同じロジックになりコピーしても見分けがつかないといったことから，コピペが横行しやすい。しかも，プログラミングの知識は積み上げ型で習得することが多いので，わからないことが生じると先に進めなくなる。このため，安易にコピペしてしまうとともに，コピペを繰り返すことになる。

そこで，2014 年度からは，それまでのような講義を一切せずに，自学自習と個別指導だけで進めるという授業パターンに改変した。講義に相当する部分は，教科書[8]の講読による自学自習とした。その際に，すべての演習問題のヒントを専用の電子掲示板にアップし，学生からの質疑応答を常時受けつけるという Q&A システムを用意した。また，演習問題の実行結果も PFD ファイルで公開することで，どういう結果が出ればいいかを確認できるようにした。個別指導では，学生一人ずつと面談しながら，わからないことやうまくうかないところなどを聞き出し懇切丁寧に指導した。それだけでなく，メンターとして，学生の動機づけを維持向上させるようなアドバイスを行った。以上の結果，コピペ問題が起きないことが明らかになった[9]

一方，個別指導としたため，学生の進捗がまちまちになるだけでなく，全体的に進捗が遅れるという傾向が表れた。教科書では基礎編，応用編，実用編と順にステップを踏んで取り上げているが，基礎編の後半で終わってしまうという状況である。これは，演習問題の提出（ソースコードリスト，実行結果の画面イメージ，実習報告書）を毎週教室で受け取ることから，早くに実習を終えても最大 7 日間待たされることになりやる気が下がるのではということである。それだけでなく，電子掲示板による学生の質問が少ないことに気になった。

そこで，2015 年度から，新たにグループウェアであるサイボウズ Live を取り入れた。これは，演習問題の提出をすべてサイボウズ Live で行うためであり，これによって，学生はいつでも提出ができ，かつ，提出時刻が記録されるので学生の進捗度合を詳細に管理できる。また，サイボウズ Live には，掲示板機能もあるので，これを Q&A システムとして利用することとした。

2. グループウェアとしてのサイボウズ Live

グループウェアは，コンピュータネットワークを利用した情報共有のためのソフトウェアであり，多くのものは，電子メール，電子掲示板，ファイル共有，スケジュール管理，ドキュメント共有，会議室予約などの機能を提供する。

グループウェア製品の中で，今回はサイボウズ社のサイボウズ Live（無料）を選んだ（図 1）。

図 1　サイボウズ Live の画面
Figure 1　The screen of Cybozu Live

グループ（300 人まで参加可能）の機能には，タイムライン，掲示板，イベント，ToDo リスト，共有フォルダ（1 グループにつき，1G バイトまで）がある。そのほかには，チャット機能，マイカレンダー機能，プロフィール機能などがある。画面の右下にあるように，Facebook や Twitter との連携機能もあり，それぞれのアカウントで簡単にユーザー登録ができる。

新着情報はすべてプッシュ型によるメール配信（メンバーとして登録したメールアドレスへ送信）となるので，見落としがなくなる。また，パソコンだけでなく，スマフォ（ガラケイ）での利用ができ，スマフォについてはアプリ登録もできるようになっている（図 2）。

2.1 グループの登録

サイボウズ Live に入ると，最初にグループを選択することになる。そこで，グループについては，全体と学生個々のグループをそれぞれ登録することにした（図 3）。

全体のグループは，”2015 年度「プログラミング実習(全体)」”とし，教員と履修した学生全員をメンバーとした。これによって，科目クラス全体で情報（連絡事項，進捗度合など）を共有することができる。

図 2　スマフォのアプリ

Figure 2　Application of smartphone

図 3　グループ登録

Figure 3　Application of smartphone

学生個々のグループは，”2015 年 P 実 1”から”2015 年 P 実 20”までの 20 個分（履修者数分）を登録した。各グループのメンバーは，教員（筆者）と学生 1 名ずつとした。これによって，各グループでは，教員とその学生だけで情報を共有することができる。そこで，このグループの共有フォルダに，出来上がった演習問題をアップロードさせることにした。その際に，アップロードした時刻が登録されることで，提出日時を正確に把握できるようになるわけである。また，他の学生はアクセス権限がないので，コピペは不可能となる。

2.2　グループ毎の機能

グループには，「イベント」「ToDo リスト」「掲示板」「共有フォルダ」「メンバー名簿」「設定」といった機能が利用できる（図 4）。

図 4　グループの機能

Figure 4　Function of group

全体のグループにおける「掲示板」は，科目クラス全体に対する連絡事項などに利用することにした。また，「共有フォルダ」には，「プログラミング実習の進め方」，「プログラムのヒントと実行結果」，「プログラミング実習報告書」，「毎週の進捗表」をアップロードした。「プログラミング実習の進め方」は，授業外で自学自習するための環境設定やエディタとブラウザの使い方について説明した文書である。「プログラムのヒントと実行結果」は，各演習問題のヒント（下部）と実行結果の画面イメージ（上部）を一緒にした PDF ファイルである（図 5）。学生が授業外で自学自習している際に，プログラミングのヒントになるような説明を記載するとともに，そのプログラムの実行結果を参照できるようにした。

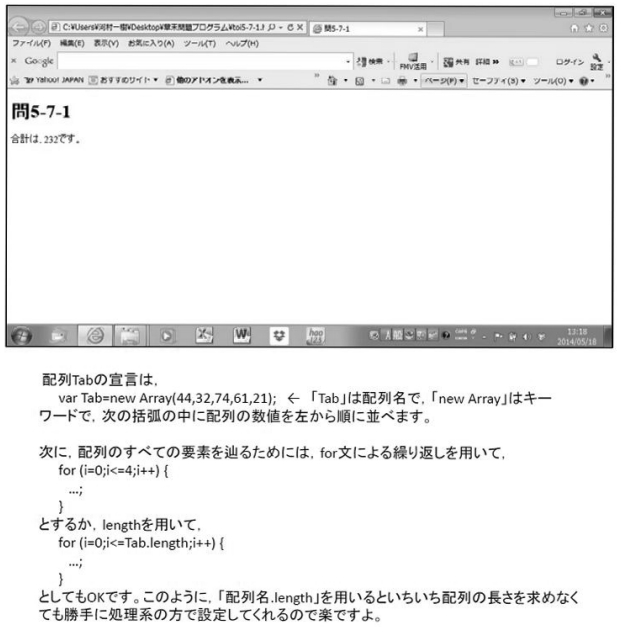

図 5　ヒントと実行結果の例

Figure 5　Example of hint and execution result

「プログラミング実習報告書」は，学生毎に，かつ，演習問題毎に，プログラミングの状況を把握するための報告書であり，Excel で入力できるようにした（図 6）。具体的には，表 1 のような事項を入力するスプレッドシートとしている。演習問題を提出する際に，この報告書も一緒にアップロードさせるように指示した。

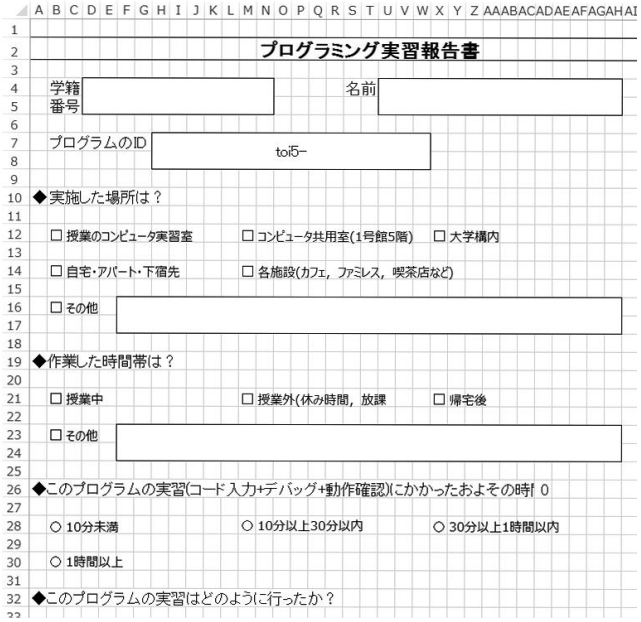

図 6　プログラミング実習報告書

Figure 5　Programming training report

表 1　プログラミング実習報告書の各項目

Table 1　Each Item of programming training report

区分	選択する項目
識別要素	学籍番号，名前，演習問題番号，提出日
実施場所	152 教室（授業で使った実習室） 大学共用室（1 号館 5 階） 大学構内（持ち込みノート PC で） 自宅・アパート・下宿先 各施設（カフェ，ファミレス，喫茶店など）
作業時間帯	授業中（月曜日の 4・5 コマ目） 授業外（休み時間，放課後）
実習時間	5 分以内 5 分以上 30 分以内 30 分以上 1 時間以内 1 時間以上
実施状況	独力でやった 先生に授業中聞いた 先輩に聞いた 友人に聞いた （正直に）コピペした 電子掲示板を参考にした

「毎週の進捗表」は，学生全員の演習問題毎の提出日時を分単位まで入力したものである（図 7）。これによって，前年度までの週単位の提出ではなく，出来た時点で即座にサイボウズ Live にアップロードできるわけで，教員により実行結果を確認後，問題がなければ，提出日時をこの進捗表に記載することにした。

進捗表の公開は，毎週の授業の最初に，全員に見せた。これによって，他の人の進捗と自分の進捗を比べてもらうとともに，遅れている学生には遅延の自覚を促すようにした。

図 7　毎週の進捗表

Figure 7　Progress table ofwWeekly

学生個々のグループについては，「掲示板」と「共有フォルダ」を用いることにした（図 8）。

図 8　個々のグループの機能

Figure 8　Function of eachgGroup

この「掲示板」では，教員と学生だけのメンバーで構成している。この中には，「緊急連絡用トピック」「挨拶・自己紹介トピック」「連絡・雑談トピック」があり，これらを使って学生とのコミュニケーションを図ることにした。とくに，自学自習中にわからないことが生じた際に，「連絡・雑談トピック」に質問を書いてアップロードするようにと指示した。その際には，ソースコードをコピーすることもできるし，画面イメージを入れることも可能であることを，あらかじめ説明しておいた。また，トピックを書き込むと，その情報はプッシュ型によるメール配信なので，メンバーである教員にも即座にトピックがアップロードされたことがわかるようになっていること，教員もできる限りメール配信をチェックしており質問に答えるようにしていることも説明した。

ⓒ2016 Information Processing Society of Japan

「共有フォルダ」には，各人が取り組んだ演習問題毎のソースコードと実行結果（画面イメージ），および，プログラミング実習報告書をそれぞれアップロードするように指示した。

3. 実証実験とその結果

サイボウズ Live を導入した上で，2015 年度に科目「プログラミング実習」を実施した。その際に，2014 年度と全く同じ形式（授業時間：前期 15 週，月曜日 4・5 時限目，授業場所：152 教室－デスクトップパソコン 25 台常設，教員卓にもデスクトップパソコンありプロジェクタにより画面投影可能）での実施となった。履修者数は，2014 年度 14 名，2015 年度 15 名であった。

3.1 演習問題の達成度

年度別に，クラス全体の演習問題の提出状況をまとめると，表 2 のようになった。参考のため 2013 年度（4 時限目は講義，5 時限目は実習という従来の授業パターン）の値も入れた。

表 2 より，年度毎に，学生一人当たりの演習問題の提出数が増えており，その分だけ多くの演習問題を解いて提出したことになる。

表 2　全体の達成度合い
Table 2　The whole achievement degree.

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

3.2 演習問題の提出率

2014 年度と 2015 年度の演習問題毎の提出率（提出者数/履修者数）は，図 9 のようになった。

これより，2014 年度は，演習問題 5-6-10（繰り返し処理まで）あたりまではほぼ全員が提出していたが，その次あたりから一気に提出数が少なくなった。ちょうどその時期は，前期の授業が終わる頃であった。また，演習問題 5-7-5 から 5-8-6 までは，一人だけ提出をした結果となっている。この学生は，高校時代からプログラミングの経験があり，授業外に自宅において，独力でプログラミングをやっていたという報告を受けた。これに対して，2015 年度は，演習問題 5-6-10 以降もそれほど提出数が下がることなく進み，演習問題 5-7-11（配列の扱い）までは 7 割の学生が終えたことがわかる。ただし，演習問題 5-8（関数）以降まで終えた学生はいなかった。

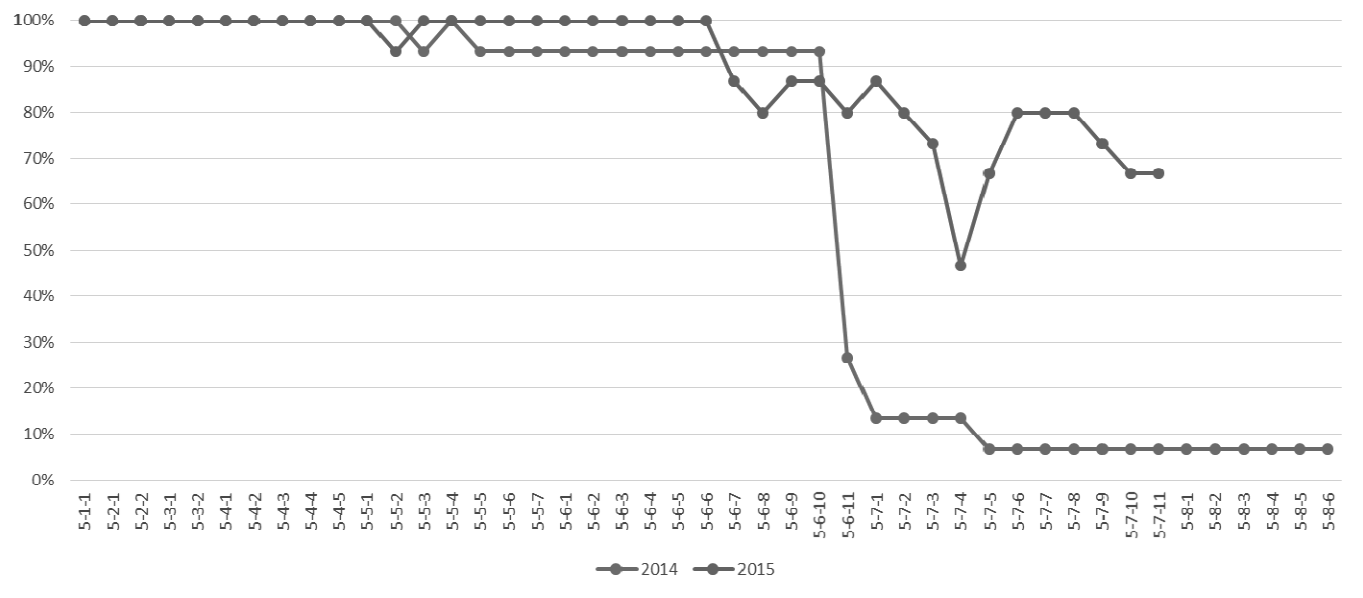

図 9　年度別演習問題の提出率
Figure 9　Presentation rate of the exercise problem by year.

以上より，2014 年度よりも 2015 年度の方が全体的にプログラミング実習の進捗率が早く，その分，演習問題の提出数も増えたことが明らかになった。これは，サイボウズ Live による提出の仕組みにより，授業中以外でも演習問題を提出できたことによるといえる。

3.3 プログラミング実習報告書の分析

2014 年度からは，演習問題の提出の際に，必ずプログラミング実習報告書を添付させている。2014 年度は手書きで

あったが，2015 年度は Excel で入力し，サイボウズ Live にアップロードできるようにした。

(1) 実施した場所と時間

自学自習を含め，プログラミング実習をどこで行ったかについて調べた（図 10）。授業内の場合は，コンピュータ実習室（152 教室）である。授業外の場合は，コンピュータ共用室（，大学構内（無線 LAN 接続可），自宅・アパート・下宿先，各施設（カフェ，ファミレス，喫茶店など）となる。この中のコンピュータ共用室には，Windows マシン百数十台と Macintosh マシン数十台が置いてあり，平日は午後 9 時まで，土曜日は午後 5 時まで開放されており，本学学生であれば誰でも自由に利用することができる。このため，自宅等にパソコンがない学生にも授業外でプログラミング実習が行えることをあらかじめ指示した。

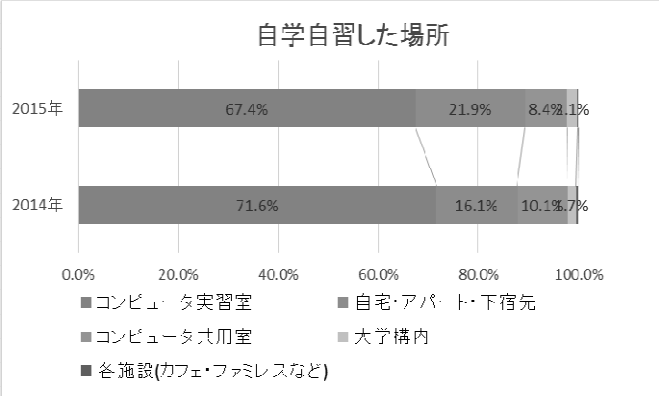

図 10　自学自習した場所

Figure 10　Place which carried out self-learning

これより，両年度ともコンピュータ実習室が 6 割以上となっており，授業中にプログラミング実習を行うことが多かった。これは，わからないことやアドバイスを教員から受けることで，中断していた，あるいは，次の新しい演習問題を続けられるからであろう。なお，2015 年度の方が教室以外でプログラミング実習を行っている割合が増えた分，コンピュータ共用室が若干減ったようである。

これに合わせて，プログラミング実習をどの時間帯で行ったかについて調べた（図 11）

これより，授業中にプログラミング実習を行う割合は，年度にほとんど変わりがなかったが，2015 年度の方が帰宅後，自宅での作業が増えた。これは，自宅等でも，演習問題ができたらすぐに提出できることから，やる気が持続できたのかもしれない。

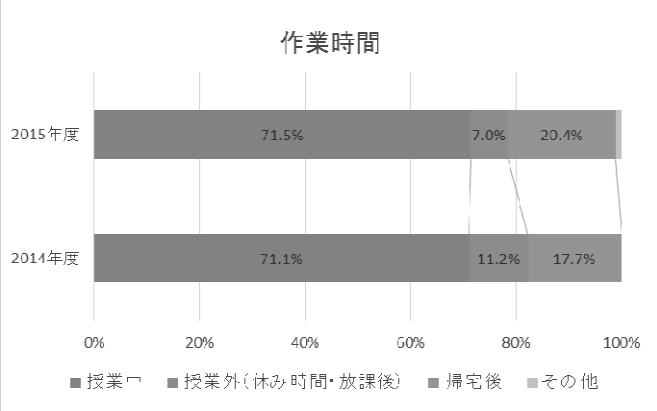

図 11　作業時間帯

Figure 10　Work hour zone

(2) 掲示板での質問

プログラミング実習を行っている際に，わからないことが生じた場合，掲示板（2014 年度は専用の BBS，2015 年度はサイボウズ Live の学生個々のグループにある BBS）で質問をしたかについて調べた（表 3）。

表 3　掲示板での質問

Table 3　Question at BBS

	2014年度	2015年度
質問をした	2.7%	2.0%
質問をしなかった	97.3%	98.0%

これより，ほとんどの学生が掲示板を利用していなかった。この理由については，「自分で解こうとした」「プログラムのヒントと実行結果を見た」「教員に聞いた」「他人に聞いた」がほとんどであった。なお，2015 年度だけ「面倒くさかった」という回答があった。

3.4 最終アンケートの分析

授業の最後に，記名式によるアンケートを実施した。

(1) 授業方法について

実習のやり方について調べた（表 4）。

表 4　実習方法

Table 4　Training method

実習方法について	2014年度	2015年度
授業中のみ	4	1
授業中および放課後（共用室）	3	3
授業中および帰宅後（自宅等）	2	9
時々授業中，おもに放課後（共用室）	2	0
時々授業中，おもに帰宅後（自宅等）	2	1
すべて放課後（共用室）	1	1
すべて帰宅後（自宅等）	0	0
その他	0	0

ⓒ2016 Information Processing Society of Japan

これより，2014 年度は回答がある程度分散したが，それでも授業中のみが多かった。これは，やはり，教員からのアドバイスや具体的な指示に依存する学生が多かったからといえる。これに対して，2015 年度は「授業中および帰宅後（自宅等)」が最も多かった。授業外でもサイボウズ Live により自宅等でも提出できたことが要因としてあげられるといえる。

これに関連して，授業パターンについて調べた（表 5)。

表 5　授業のやり方

Table 5　Method of the class

	2014年度	2015年度
講義をしなくてもよい	12	6
どちらでもない	1	3
ある程度講義をしてほしい	1	6
ちゃんと講義をしてほしい	0	0

これより，2015 年度の方では，「講義をしなくてもよい」と「ある程度講義をしてほしい」が半々となった。これについては今後考慮する必要があるのかもしれない。場合によると，授業の最初に，学生から講義してほしいことを聞き出し，少しだけ講義をするような場面を想定した方がいいのかもしれない。とくに，前期の始めの頃は，プログラミング実習のやり方（とくに授業外）がよくわからなかったりするので，授業の始めに少しだけ講義をした方がいいのかもしれない。なお，「ちゃんと講義をしてほしい」という要望は全くなかったので，講義レスでも受け入れられる授業パターンであるといえる。

(2)　サイボウズ Live について

2015 年度の学生に対しては，サイボウズ Live の必要性について調べた。

表 5　サイボウズ Live の必要性

Table 5　Necessity of Cybozu Live

	件数
必要である	6
どちらでもない	8
不要である	0
その他	1

これより，全体的な傾向としては，サイボウズ Live による提出方法は学生に受け入れられているといえる。ただし，「どちらでもない」が多いことから，もう少し意識をもたせるような工夫（提出の早さを演習の評価点に加算する）が必要かもしれない。

4.　おわりに

以上のような結果から，2015 年度に実施したグルーウウェアを用いたプログラミング実習の方が，クラス全体の進捗が早くなるとともに，その分演習問題の達成度合いも高くなることが明らかになった。

今後の課題としては，当初の教育目標であった JavaScript を実践的に使いこなせるレベルまでに至っていないことがあげられる。現状では，本教科書の「第 5 章　JavaScript の基本編」あたりで終わってしまっており，肝心の「第 7 章　JavaScript の実用編」まで実習ができていないことがあげられる。

この課題を解決するためには，学生個々の演習問題の達成度合いをもつと早める必要がある。それには，自学自習の際に，わからないことがあった場合，そこで実習を止めてしまうのではなく，学生個々のグループにある掲示板を使って質問をするような仕組みを提供することが考えられる。これによって，実習の遅れをクリアできる可能性がある。

掲示板のトピック欄では，コメントと添付ファイルを一緒にアップロードすることができる。この機能を使い，うまく動かない演習問題（構文エラー，ロジックエラー，実行時エラー）に対して，コメントには質問を，添付ファイルには JavaScript のソースコードを記入すればよい。配信メールにより教員もアップロードしたことがわかるので，それらをダウンロードして間違っている箇所を指摘することができる。これによって，多少のタイムラグ（教員の対応時間）が生じる可能性があるが，次の授業（最大 1 週間先）まで待つことなく演習問題を進めることができるはずである。来年度において試みる予定である。

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，pp.23-39，2011 年
[2]　河村一樹，斐品正照，徳岡健一：文科系向けプログラミング教育支援システム JPADet の設計と開発, 情報処理学会研究報告，2001-CE-62(2)，情報処理学会，2001 年
[3]　河村一樹，斐品正照：よくわかる日本語 PAD によるアルゴリズム演習，日刊工業新聞社，2003 年
[4]　斐品正照，松瀬賢太，河村一樹：アルゴリズム学習を支援する JPADet の評価と教材開発，情報処理学会研究報告，2003-CE-68(5)，情報処理学会，2003 年
[5]　斐品正照，河村一樹：アルゴリズム学習におけるプログラミング言語を用いた学習と構造化チャートを用いた学習の比較，教育システム情報学会第 94 回研究会報告，PP.37-44，2003 年
[6]　斐品正照，徳岡健一，河村一樹：構造化チャートを用いたアルゴリズム学習支援システム，情報処理学会論文誌，Vol.45，No.10，pp.173-180，2004 年
[7]　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)，情報処理学会，2011 年
[8]　河村一樹：JavaScript による情報教育入門, 大学教育出版, 2011 年
[9]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)，情報処理学会，2015 年
[10]　“サイボウズ Live　機能詳細” https://live.cybozu.co.jp/overview.html

ⓒ2016 Information Processing Society of Japan