

# プログラミング演習講義における声掛けのための TA 支援システムの開発と評価

横山 裕紀<sup>1</sup> 江木 啓訓<sup>1</sup>

**概要:** 多くの大学で実施されているプログラミング演習講義において, 学習をサポートする Teaching Assistant(TA) 制度が導入されている。しかし, 学生数に対して少ない人数で指導する際には, 全ての学生の学習状況を把握して対応できないという問題がある。そのため, 講義に遅れをとっている学生に対して十分な支援を行えず, 学習目標を達成させることができない。

本研究では, TA に学生の学習状況を提示する TA 支援システムを提案する。これによって, TA が講義に遅れている学生を発見しやすくなり, 学生に対して声掛けを行えるようになることを考えた。TA の行動を阻害することなく支援を行えるように, 片眼のヘッドマウントディスプレイ (HMD) を用いた。出席, 確認テスト, 課題の 3 つのデータが学習管理システム (LMS) で管理されているため, データを取得して学生の学習状況を閲覧できるようにした。

講義で使用した結果, システムを用いることによる TA の声掛け件数の増加は見られなかった。しかし, TA 支援システムを用いることによって, 教室内の巡回などの TA の行動を活発にする可能性が示唆された。また, クラスによって TA の行動に差が大きかったことから, 今後は TA の行動の変容に働きかけるようなデザインを考慮する必要があることが明らかになった。

## An Encouraging System for Teaching Assistants to Communicate with Students during Programming Exercises

YUUKI YOKOYAMA<sup>1</sup> HIRONORI EGI<sup>1</sup>

### 1. はじめに

近年, 優秀なソフトウェア開発者の育成が重要視されている。これまで主として大学教育で行われていたが, 小中学生を対象としたプログラミング教育の実践が進められている [1]。多くの大学では, プログラミング演習講義が行われている。受講者がプログラミングスキルを取得できるように, 様々な方法が実施されている。教員 1 人で受講者の学習状況を把握するのが困難である [2] ため, 講義補助者として Teaching Assistant(以下 TA とする) 制度を導入している大学も多い。しかし, 受講者に対してきめ細かい対応を行えていないのが現状である。

本研究では, プログラミング演習講義における TA を支

援するシステムを提案する。TA は主に教員を補助する役割を担い, 講義中の学生のサポートを行う。システムを用いることで, TA は講義中に教室内を巡回しながら, 学生の学習状況を把握することができる。学生の学習状況を把握できることで, TA から学生に対して, 声掛けを行いやすくなる。また, 教室の座席表と学生の学習状況を表示することで, クラスの中で講義に遅れている学生を把握することができる。これにより初学者が, つまづきやすいプログラミングの学習に貢献できると考えられる。

### 2. 関連研究

主に大学で行われるプログラミング演習講義は, プログラミングの概念や考え方を身に付けるだけでなく, 実際の研究におけるデータ処理のための技術を身に付けるためである。そのため, ツールキットなどの組み合わせでなく, 実際のコード作成を学習する必要がある。理論的な学習で

<sup>1</sup> 電気通信大学大学院情報理工学研究科  
Graduate School of Informatics and Engineering, The University of Electro-Communications

ある座学だけでなく、実際にプログラムコードを書くといった演習も不可欠である。多くの大学では、講義の時間が90分であり、この限られた時間内で座学と演習を行う。受講者の多くが初学者である場合、講義のみで取得するのは困難であり、講義時間外の自学学習も必要である。

初学者には、講義につまずいてしまうと、その後の講義に対する学習意欲が低下する傾向があるとされている [3]。また、この講義を受ける学習者には、初学者もいる一方で、個人の学習によって一定の経験を持っている学生もいる。そのため、学習者間のスキルの差が大きくなる傾向にある [4]。一般的に、数十名程度を対象としたプログラミング演習講義においては、教員1名とTA2~3名が担当する。学生数に対して少ない人数で指導するため、全ての学生に必ずしも講義時間内に対応できない。

## 2.1 コーディング過程の可視化

プログラミング演習講義では、各受講生がソースコードを書いたり、実行したりと、個別の演習を行うことが多い [3]。また、過去の初等、中等教育や、学生の興味から事前に知識をもっている学生もいる。そのため、同じ講義を受けていても、各受講生にスキルの差が生じてしまう。これは回を重ねていく毎に大きくなり、実際に演習課題などの進捗状況にも表れる。一目で学生の様子から、講義より進んでいるか、遅れているかを判断するのは困難である。

そのため、井垣らは、受講者の学習状況を把握できるC3PV(Coding Process Visualizer in Programming Practice)を設計した。受講生のソースコードの作成、編集、プログラムの実行、課題提出までをコーディング過程とし、コーディング過程を座席情報に基づいて教員とTAのパソコン画面に表示する。各受講生の座席を4分割し、その講義における全ての課題のコーディング時間、単位時間当たりのエディタ操作数、課題ごとのエラー継続時間を表示した。これを用いることで、教員やTAが自分のパソコンから閲覧し、講義に遅れている学生の早期発見を行うことができる。

## 2.2 学習者の個別支援手法

受講者の学習状況を把握するための方法として、Webベースの教育支援システムにおいて、受講者の提出した課題を分析して検知するシステムがある [4]。課題の進み具合における作業進捗集計機能と、プログラムを実行した際に発生したエラーの個数によるエラー分類集計機能を用いている。クラス全体のそれぞれの時間や個数から、平均を算出し、受講者個人と全体の遅れを比較し、相対的に遅れている受講者を検知し知らせる。このシステムは、主に演習のみの講義を対象にしており、教員やTAがPCで確認できる。

また、個々の学習者の理解状況と学習意欲に適したプロ

グラミング演習を行うシステムもある [5]。学習者の講義における過去の学習履歴に対して、協調フィルタリングを用いて、教員に表示する。それを閲覧した教員による評価と学習意欲の判定から、受講者に適した演習問題を提示する。これにより、学習者の進捗状況や学習状況に合わせた演習課題を出題することができる。

## 2.3 TA 業務支援システム

プログラミング演習講義において、学生数に対してTAの人数が少ないことが他の研究でも指摘されている [6]。講義形式によっては、学生の質疑対応や声掛けだけでなく、演習課題の提出を確認する作業もTAに含まれる。そのため、学生が挙手している際に、呼び出しの内容が質問と演習チェックであるのか区別がつかず、毎回学生に要件を聞く必要性が生じる。また、TAは学生の挙手した順番を把握できず、学生に待ち時間が生じることがある。

そのため、TAに携帯情報端末を持たせ、学生の呼び出しも質問による呼び出しボタン、課題チェックの呼び出しボタンを設置し、それぞれを使い分けることでTAの対応をスムーズに行えるようにしている研究がある。学生のボタンを押したタイミングで順番を表示することで、TAが学生の順番を把握することができ、学生側もTAが巡回するタイミングを認識できるようになっている。

TAの画面には、呼び出し学生の一覧が表示され、呼び出しの内容と学生の座席を閲覧することができる。また、演習チェック画面では各学生の演習状況を閲覧でき、学生側がチェックしてほしい課題にはアイコンが表示される。これにより、TAがスムーズに対応することができる。

このシステムは実際に情報工学系の大学の複数のプログラミング講義で利用されており、TAから良好な評価を得られている。TAは情報がない状態でその場で学生に対応するのではなく、学生の呼び出しの意図や、学生への対応の順番、学生の学習状況を把握することで、よりスムーズにTA業務を行うことができる。

## 3. プログラミング演習講義における声掛け支援

### 3.1 プログラミング演習講義におけるTAの役割

プログラミング演習講義において、TAは学生が授業に遅れをとらないように支援することが主な業務である。しかし、学生からの質問への対応がほとんどであり、TAから学生への声掛けはあまり行われていない。これはどの学生が授業に遅れをとっており、TAが声掛けすべきか判断できないためと考えられる。声掛けによる支援を受けることができないことにより、学習目標を達成することができなくなってしまう。

また、TAが質問を受けた時にその学生の過去の講義における理解度を把握できない点も問題である。特にプログ

ラミング演習講義は単元ごとの内容のつながりが強く、以前の回の理解度が質問の内容に影響している可能性がある。これに対し、本研究では過去の講義における理解度の推移をTAに提示する。これにより、TAが学生の理解度をふまえた声掛けを行えるようにする。

### 3.2 プログラミング演習講義における支援システム

プログラミング演習講義における支援システムの多くは、プログラミングをより学びやすくするなど学生を対象としたものや、教員を対象とした支援システムがほとんどである。本研究ではTAに焦点を当てる。TAが教室内を巡回し、遅れをとっていきそうな学生を発見したとき、これまでの学習内容を含めてどの部分でつまづいているかを認識することで、学生に対して声掛けの手がかりとすることができる。そのような場面において、TAが声掛けをするのに必要な情報を提示することで、TAの行動を阻害することなく支援できる。

野口らの研究[6]とは、本研究で想定する講義の形式が異なる。本研究で想定する講義では、学生の課題は自動的に評価されるため、TAが確認する作業はない。そのため、TAからより学生に対して声掛けを行うことで、講義でのつまづきを解消できると考えられる。

## 4. システム設計

### 4.1 TA支援システムの提案

TA支援システムの構成を以下に示す。TAが装着型デバイスを身につけて、教室内を巡回する。その際に講義に遅れていきそうな様子の学生を発見する。発見した学生が使用しているパソコンのIPアドレスから、サーバを介して学生の学籍番号と名前を取得する。取得した学籍番号から、学生の学習履歴のデータをサーバから読み取る。読み取ったデータをグラフ表示にして、装着型デバイスに反映させることでTAが閲覧できる。閲覧した情報を元に学生に対して声掛けを行う。

遅れている学生を発見できない場合は、学生の課題への取り組み状況を図示した教室の座席表を閲覧することで、声掛けを行う対象の学生を探せるようにした。それぞれの設計方針について説明する。

### 4.2 対象の学習データとシステム構成

システムが学生を特定するために、学生がログインしているパソコンのIPアドレスから、ログイン名と学籍番号を特定する。これは、本研究が対象とする講義では、座席の指定がなく学生は空いている席に自由に座ることができるためである。また、TAが情報にアクセスするため、着席している学生の後ろのパソコンの背面にIPアドレス情報を取得するためのQRコードを設置した。これによりTAは学生の情報を得るために教室内を巡回することができ、

また、後ろの座席に取り付けているため、学生に対しても注視することによる威圧感を与えずに行動することができる。

今回の実験で対象とする講義は、学習管理システム(LMS)であるMoodleを用いており、学生によってプログラミング演習課題の提出状況を集計し、教員が閲覧できるようになっている。学生の出席状況、各回にある確認テストの成績、毎回の課題の実施状況のデータを使用する。

次に、取得したデータをどのようにTAに表示する必要があるか検討する。TAの行動に焦点を当てると、

- 教室内を巡回し手を挙げている学生に答える
- わからない様子の学生に声をかける
- 学生と1対1でコミュニケーションする

の3つの行動である。これらを円滑に行える端末を用いる必要がある。学生の様子に応じて声掛けを行うためには、常に学生の様子が見えている必要がある。そのため、HMDを装着することで、常に学生を観察しながら学習履歴の閲覧が行える。また、Moodleで取り扱ったデータを用いることと、学生の特定にはQRコードを用いることから、ブラウザ表示ができカメラが内蔵されているHMDを用いる。使用したHMDを装着した様子を図1に示す。

HMDには、PHP言語によって作成したWebページ上で表示した。学生のデータについて、出席状況、確認テスト、課題状況のそれぞれを学籍番号別に用意した。各PCに設置したQRコードが読み込まれると、各PCのIPアドレスと学籍番号を保存したファイルをもとに、どこの席にどの学生が座っているかを特定する。その学生の学籍番号の該当する3つのファイルにアクセスし、グラフが描写されTAに表示される。図2にシステムの全体構成を、図3にQRコードを設置した教室の様子をそれぞれ示す。



図1 HMDを装着したTA

### 4.3 TAへの情報提示

学生の学習状況を取得したうえで、グラフを作成する。学習状況の点数に対応した色分けによりグラフをプロットする。TAへの表示画面を図4に示す。表示形式は、グラフを用いて横軸に各授業回をとり、縦軸は3つの項目に分

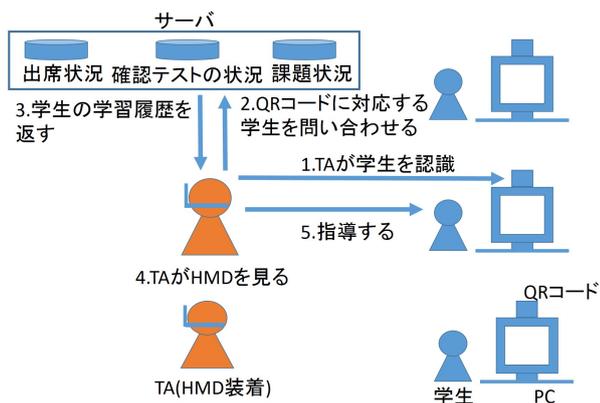


図 2 HMD への学習状況の表示の流れ



図 3 QR コードを設置した教室

ける。原点に近い順に、出席状況、確認テストの状況、課題状況のブロックに分ける。得点の高い項目は緑、低い項目は赤で図示する。各ブロック内で複数存在するものは原点に近いものから順に縦軸に並べていく。また、学生の学籍番号と名前も画面の左上端に表示する。

#### 4.4 学生の学習状況と座席表

図 5 に示すような座席表の上に、学習状況を表示できるようにした。座席表には IP アドレスと学籍番号の関係からどこに誰が座っているかを表した。一人一人の学生のデータを取得し、座っている席をデータに応じて色分け表示した。座席表は、現在行っている講義の課題の状況と、それまでの講義の課題の合計点の状況の 2 種類で作成した。各学生の点数から平均点を算出し、平均点±最大点数の 6%以内を緑色で示し、平均点+最大点数の 6%以上を青、平均点-最大点数の 6%以下を赤色で示した。また、TA の教室内の現在地を黒の丸で示した。



図 4 HMD に表示される画面の例

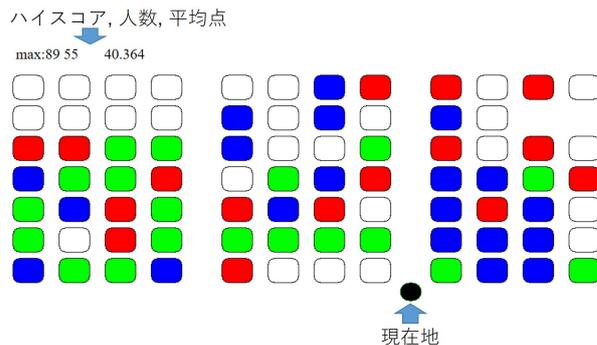


図 5 学生の学習状況と座席表の表示

## 5. 評価実験

### 5.1 実験対象の講義

理工系の大学 1 年次の C 言語を用いたプログラミング演習を実験対象とした。講義について表 1 に示す。講師 1 名に対して TA は 2 名であり、学生は 70 名程度である。この講義は 1 年生全員が受けたため、12 クラスが開講されている。教材のうち、確認テストのみ講義開始から 15 分しか受講することができない。それ以外の資料についてはいつでも閲覧可能である。講義時間は 90 分であり、座学と演習の時間配分は、各クラスの担当教員によって異なる。また、講義中の TA は主に教室内を巡回し、手を挙げた学生の質問に対応したり、学生に対して声掛けを行う。教室の巡回するタイミングは TA によって異なる。座学中は学生の邪魔にならないように巡回しない TA もいた。

2016 年 12 月から 2017 年 2 月までの、第 11 回から第 15 回の講義を実験対象とした。第 11 回は、講義の 2/3 がこれまでの復習問題であり、第 12 回から第 14 回までは座学の割合が大きく、第 15 回はすべて問題演習であった。

表 1 本研究が対象とする講義

講義名	C 言語のプログラミング演習講義
講師	1 名
TA	2 名
学生	学部 1 年生, 70 名程度
講義時間	90 分

複数のクラスに対して、システムを使用したクラス、使用していないクラスの授業の様子をビデオカメラにより記録した。クラスの条件について、表 2 に示す。システムを用いた講義を HMD、システムを用いずにビデオ撮影のみをした講義を Video、何もしていないクラスを N/A で表す。

### 5.2 2 つのクラスにおける複数講義の比較

システムを用いたクラス A と用いていないクラス B を対象に、2016 年 12 月から 2017 年 1 月までの第 11 回から第 15 回までの連続した 5 回について記録し比較した。どちらのクラスとも教室の前方からビデオカメラ 2 台により

表 2 講義の条件表

クラス	講義回				
	第 11 回	第 12 回	第 13 回	第 14 回	第 15 回
A	HMD	HMD	HMD	HMD	HMD
B	Video	Video	Video	Video	Video
C	Video	N/A	HMD	N/A	N/A
D	N/A	Video	N/A	N/A	HMD
E	N/A	Video	N/A	N/A	HMD
F	N/A	Video	N/A	N/A	Video
G	N/A	Video	N/A	N/A	Video
H	N/A	Video	N/A	N/A	Video
I	N/A	Video	N/A	N/A	Video
J	Video	N/A	N/A	N/A	N/A
K	Video	N/A	N/A	N/A	N/A

撮影した。また、TA にはボイスレコーダーを携帯させ、映像から読み取れない TA の行動を録音内容から分析した。

### 5.3 複数のクラスにおける同一講義比較

第 13 回と第 15 回の講義を対象として、複数のクラスの TA にシステムを用いることで、システムの有用性について検証した。授業日程と内容および授業進行の都合により、第 13 回は A と C のクラスにおいて、第 15 回は A と D と E のクラスにおいて使用した。第 15 回の授業は今までの復習の回であり、演習のみである。TA の声掛け行動を分析するため、実施したすべてのクラスに対してビデオカメラ 2 台による撮影とボイスレコーダによる録音を行った。比較対象としてシステムを用いないクラスに対してもビデオカメラによる記録を行った。

## 6. 結果と考察

本研究で、ビデオカメラによって撮影したクラスに関して、TA が学生から受けた質問回数と、TA から学生に対して声掛けを行った回数について、表 3 にまとめた。実験 1 において、TA がシステムを用いて学生の学習状況を閲覧した回数について表 4 にまとめた。TA がシステムを用いて学生の学習状況を閲覧した回数について、第 13 回のクラス A と C は表 5、第 15 回のクラス A と E は表 6 にそれぞれまとめた。クラス D は実験中にログを回収するサーバに不具合があったため、ログ状況を解析できなかった。

実験 1 と実験 2 の結果より、システムによって TA の声掛けはそれほど活発にならなかった。また、システムを継続的に使用したクラスと複数のクラスに使用した際から、最初はシステムに慣れるためにも多くの学生の学習状況を閲覧していた。しかし、使用し続けるうちに、座席表を閲覧して学生全体の学習状況を把握することが中心になり、各学生の学習状況の閲覧や声掛けが行われなかった。学生は TA がシステムの画面内を見ることができないため、TA

の行動が必ずしも学生の学習状況を閲覧しているとは感じていなかった可能性がある。TA が HMD を、学生の学習状況の閲覧のみに使用していることが、学習者に伝わる必要がある。

実験後の TA からのコメントより、表示した学生の学習状況の情報量は的確だったと考えられる。しかし、出席や課題、テストの状況は見えても、細かい表示やソースコードを閲覧することができていない。そのため、TA は学生の学習状況の推移しか情報を得ることができない。その学生がつかずにいる原因や遅れている具体的な理由は、学生に聞くか、PC 上で確認する必要がある。

講義を撮影したビデオより、TA によって行動のばらつきに大きな差が生じていたことがわかった。90 分の講義中、座学の間においても巡回する TA と、演習中のみ巡回する TA、自分の PC に座ったままで全く巡回しない TA に分類できた。巡回しないクラスにおいては、学生からの質問がない状況であった。また、TA の行動が決まっていないため、多くの TA が質問がない時は、手持ち無沙汰になってしまう状態が生じていた。

一方で活発な TA は一度質問を受けた学生に時間を空けた後に、声をかけにいくという行動も見られた。一般的な TA の行動原則が定まっていないため、システムを用いて学習状況を把握できても、行動に移すことに困難であることが分かった。システムで学習状況を把握するためには、実際に学生の近くまで行かない限り把握できないため、システムを用いた TA は常に教室を巡回することも分かった。常に巡回しているため、学生から巡回中に質問をされやすくなる可能性がある。

以上より、TA の行動にばらつきがあるため、声掛けの増加には至らなかった。しかし、システムによって、TA が巡回するようになったことと、提示した情報量が適切であったことが分かった。

## 7. 終わりに

本研究では、プログラミング演習講義における TA 支援システムを提案した。システムは、講義中における TA の行動を円滑に行えるように片眼の HMD を用いた。また、各パソコンの IP アドレスと、講義で使用されている学習管理システム (LMS) の Moodle を用いることで、学生を特定して、学習履歴の取得を行い、TA が閲覧できるようにシステムを設計した。

実験において、システムを使用することができ、TA が学生の学習状況を閲覧することと、遅れている学生の発見を行うことができた。しかし、使用した HMD の性能や、画面サイズと操作などの点において、TA が常時使用することに對する課題があった。

また、TA がシステムによって学習状況を閲覧できても、どのように声掛けを行う必要があるかわからないというこ

表 3 質問件数と声掛け件数

	第 11 回		第 12 回		第 13 回		第 14 回		第 15 回	
	質問 件数	声掛け 件数								
A	5	3	1	0	0	0	1	0	3	0
B	5	0	0	0	1	0	5	0	3	0
C	1	0	N/A	N/A	0	6	N/A	N/A	N/A	N/A
D	N/A	N/A	1	1	N/A	N/A	N/A	N/A	2	1
E	N/A	N/A	4	2	N/A	N/A	N/A	N/A	8	0
F	N/A	N/A	2	0	N/A	N/A	N/A	N/A	5	0
G	N/A	N/A	0	0	N/A	N/A	N/A	N/A	2	0
H	N/A	N/A	0	0	N/A	N/A	N/A	N/A	0	0
I	N/A	N/A	1	0	N/A	N/A	N/A	N/A	1	0
J	0	0	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
K	2	0	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A

表 4 閲覧状況

講義日	第 11 回	第 12 回	第 13 回	第 14 回	第 15 回
講義回閲覧した 学生数	18	1	0	0	1
1 人の学生に対する 最大閲覧回数	3	1	0	0	1
行われている講義の座席表の 閲覧回数	31	5	1	7	7
過去の学習状況に対する 座席表の閲覧回数	4	7	5	10	10

表 5 閲覧状況

講義日	A	C
講義回閲覧した学生数	0	5
1 人の学生に対する 最大閲覧回数	0	1
行われている講義の 座席表の閲覧回数	1	5
過去の学習状況に関する 座席表の閲覧回数	5	13

表 6 閲覧状況

講義日	A	E
講義回閲覧した学生数	1	8
1 人の学生に対する 最大閲覧回数	1	2
行われている講義の 座席表の閲覧回数	7	8
過去の学習状況に関する 座席表の閲覧回数	10	10

ともあった。そのため、システムによる声掛けの件数の増加は見られなかった。また、システムを使用していないクラスにおいても、記録したビデオから TA の巡回の有無、TA からの声掛けの有無がわかり、TA によって行動に大きな差があった。

以上より、システムによって支援することで TA の行動がより活発になる可能性が示唆される。また、講義における学座と演習の割合によって、TA の行動に影響を及ぼすことと、すべての TA の行動に統一性がないことが明らかになった。

今後の課題として、実際の講義においてより学生を特定

しやすい方法の検討がある。また、携帯情報端末によるシステムの実装との比較や、システムによる学習状況の閲覧後の TA の行動指針の指定、システムによる TA の行動変容に働きかける方法について、研究を進めていく。

#### 謝辞

本研究の一部は科学研究費補助金 基盤研究 (C)(一般) 15K01025 の助成を受けたものである。

#### 参考文献

- [1] 岡本 雅子: 初めてのプログラミングとつまづき, 情報処理学会論文誌, Vol.56, No.6, pp.580-583 (2015).
- [2] 西澤 泰彦: 多人数講義における問題点と教育方法, 名古屋高等教育研究, Vol.6, pp.45-57 (2006).
- [3] 井垣 宏, 斎藤 俊, 井上 亮文, 中村 亮太, 楠本 真二: プログラミング演習における進捗状況把握のためのコーディング過程可視化システム C3PV の提案, 情報処理学会論文誌, Vol.54, No.1, pp.330-339 (2013).
- [4] 加藤 利康, 石川 孝: プログラミング演習のための授業支援システムにおける学習状況把握機能の実現, 情報処理学会論文誌, Vol.55, No.8, pp.1918-1930 (2014).
- [5] 田口 浩, 糸賀 裕弥, 毛利 公一, 山本 哲男, 島川 博光: 個々の学習者の理解状況と学習意欲に合わせたプログラミング教育支援, 情報処理学会論文誌, Vol.48, No.2, pp.958-968 (2007).
- [6] 野口 剛士, 井出 敬也, 長郷 俊輔, 古賀 雅伸, 矢野 健太郎: PC を使う多人数講義における TA 業務支援システムの開発と評価, 情報処理学会研究報告, Vol.2013-CLE-11, No.31, pp.1-5 (2013).