

授業支援システムにおける プログラミング演習のための 学習状況把握支援機能の設計と評価

加藤利康[†] 石川孝^{††}

授業支援システムは、授業についての多様な情報を効率よく提供するとともに、教員と学生、あるいは学生同士のコミュニケーションを活性化させることを目的としている。この授業支援システムは、大学等におけるプログラミング演習などの多人数の学生を対象とした授業において有効活用され始めている。本論文は、プログラミング演習中の個々の学生が抱える多様な問題などの学習状況を巡回せずに把握し、教員への確に支援することで教員の負荷を軽減し、学生に対する指導の質を向上させることを目的とした学習状況把握支援機能を提案する。プロトタイプシステムを使って実際の授業で評価実験を行い、学習状況把握支援機能によってしか把握できない状況に対する指導内容の面から学習状況把握支援機能の有効性を検証する。

Design and Evaluation of Support Functions of Course Management Systems for Assessing Learning Conditions in Programming Practicums.

TOSHIYASU KATO[†] TAKASHI ISHIKAWA^{††}

Course Management System is to provide information efficiently with a variety of classes for students and teachers or between students have been intended to revitalize communication. This Course Management System is beginning to be used effectively in the classes for the student of a lot of people, such as a programming exercise in a university etc. This paper grasps learning situations, such as various problems which each student in a programming exercise has, with supporting exactly to a teacher, a teacher's load is reduced, and designed the learning situation grasp support function aiming at raising the quality of instruction to a student, and proposes them. Experiments in a real classes, teaching and assessment in terms of content and the number of teaching teachers, support functions to verify the validity of situational awareness learning.

1. はじめに

大学教員の教授力向上の取り組みにおいて ICT (Information and Communication Technology) を活用してどのように進めるかが世界的に考えられており [5], 教員の ICT 活用の手段として授業支援システムがある。授業支援システムは、授業についての多様な情報を効率よく提供するとともに、教員と学生、あるいは学生同士のコミュニケーションを活性化させることを目的としている。この授業支援システムは、大学等におけるプログラミング演習などの多人数の学生を対象とした授業においても有効活用され始めている。しかしプログラミング演習では、教員は限られた時間の中で多人数の学生を指導しなければならず、学生の抱える多様な問題に対して個別に対応しなければならないという現状がある。この現状に対して従来研究は、共通の問題を抱える学生の情報を教員へ提供することで、一斉指導が可能となる学習状況把握支援システムを開発している [9]。しかし、エラー要因の分析には教員の手間が増えることや、一斉指導に限定しており、個別指導のための情報を提供していないといった課題がある。そこで本研究は、エラー要因の分析をシステムが行い、一斉指導と個人情報のための情報を提供する学習状況把握支援システム [8] を開発することを目的とする。本論文は、プロトタイプシステムにおける学習状況把握支援機能の設計と実際の授業で実験を行った結果について評価し、有効性を検証する。

本論文の構成は、まず 2. で授業支援システムの現状と課題について述べ、3. でプログラミング演習のための支援機能について関連研究を調査した結果の問題点と解決手段を述べる。次に 4. で支援機能についてシナリオに基づく設計 [2] を採用して設計を行い、5. でユースケース分析による詳細設計を行う。6. にてプロトタイプシステムを実装し、学習状況把握支援機能を実際の授業による効果測定を行った結果を述べ、最後に 7. で本論文のまとめと今後の課題について述べる。

2. 授業支援システムの現状と課題

授業支援システムの現状は、大学教員に対する FD (Faculty Development) への要求や少子化による学生獲得の要求などもあり、多くの高等教育機関で導入が進んでいる。しかし、国内の利用率が米国と比べて約 40% [12] と低いのは教員の理解が不足するためである。米国では授業支援システムを活用して、教員と学生の双方向コミュニケーションを促進させて効果的な教育を実施している。利用率があがらない理由は、システムを維持する予算や、コンテンツの作成による教員の負担が増えることである。利

[†] 日本工業大学大学院 情報工学専攻
The Graduate School of Engineering, Nippon Institute of Technology

^{††} 日本工業大学 情報工学科
Nippon Institute of Technology

用率が低い理由は、システムを支える技術が未成熟であり[4]、コミュニケーションを促進する機能が不十分である。コミュニケーションの促進は、教員と学生、教員同士、学生同士の3つがある。

教員と学生のコミュニケーションの促進は、授業支援システムを活用した教員と学生のコミュニケーションについての教員の理解をさらに深める必要がある。この有効性を評価する基準として、先行研究[6] [7]において、全米の大学の教育支援組織から提供されている効果的な教育を行うための実践手法である「優れた授業実践のための7つの原則」[1]を利用した。優れた授業実践のための7つの原則とは、米国高等教育学会の研究グループを中心に開発されたものであり、学士課程教育の質的向上を効果的に促進するための方法論をまとめたものである。7つの原則は次のとおりである。

- (1) 学生と教員のコンタクトを促す
- (2) 学生間で協力する機会を増やす
- (3) 能動的に学習させる手法を使う
- (4) 素早いフィードバックを与える
- (5) 学習に要する時間の大切さを強調する
- (6) 学生に高い期待を伝える
- (7) 多様な才能と学習方法を尊重する

7つの原則は抽象的であるため、具体化した「優れた授業実践のための7つの原則とその実践手法」[10]を授業支援システムに対する要求機能とみなして、具体例としての現状の Moodle[3]で支援できない機能から、授業支援システムの課題を抽出した。Moodle とは、社会構成主義教育論[11]に基づいた CMS であり、社会構成主義教育論は、学習者がそれぞれの環境におき、相互作用することにより新しい知識を構成することである。授業支援システムの新たな要求機能は以下の2点であり、これらをまとめて学習状況把握支援機能と呼ぶ。

- 学生の学習状況を取得する：学生の教材閲覧履歴を教員に視覚的に提示する。
- 教員に学生の学習状況を提供する：学生の理解困難箇所やエラーの原因・場所を教員に提示する。

本研究の目的は、学習状況把握支援機能をプログラミング演習を対象として実現することである。授業支援システムに対するこれらの要求を具体化するには、最初から一般的な授業を対象とするのは実現が困難であるため、教員にとって学習状況の把握が特に必要とされているプログラミング演習を対象とする。研究の方法は、まず、プログラミング演習のための支援機能を関連研究を調査し、問題点を把握して解決手段を導く。支援機能はシナリオに基づく設計を採用して設計を行い、システムの要求機能を抽出する。次にユースケース分析から詳細設計を行い、プロトタイプシステムを実装する。実際の授業による効果測定を行い、有用性と有効性を考察する。

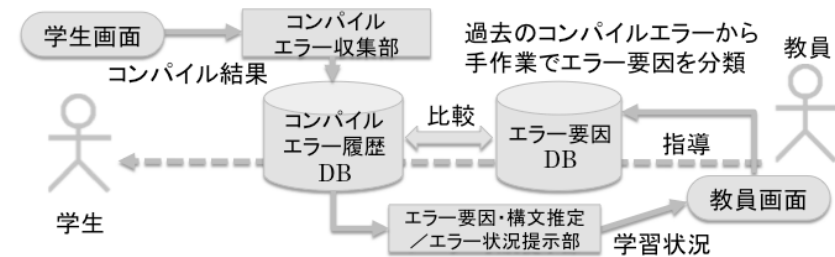


図1 一斉指導のための学習状況把握支援システム

3. プログラミング演習のための支援機能

プログラミング演習は、講義によって学習した知識を学生自身が個別的学習によって確認し定着することを目的とした授業である。しかし、教員は限られた時間の中で多人数の学生を指導しなければならない。また、学生が抱えるエラー要因に対して個別に対応しなければならない。

プログラミング演習支援と学習状況把握支援に関する関連研究として、倉澤氏らは共通の問題を抱える学生の情報を教員へ提供することで、一斉指導が可能となる学習状況把握支援システムを開発している[9]。コンパイル作業履歴を収集し、過去のコンパイルエラーの傾向分析から、エラーの原因や場所を推定・集計し、エラーメッセージとその件数およびエラー要因とエラー構文の候補を教員に提示することで、共通の問題を抱える学生を把握している(図1)。しかし、エラー要因の候補をあらかじめ分析して用意しておく必要があり、エラー要因の分析は教員が行わなければならないため、教員の手間が増える。また、一斉指導に限定しており、個別指導のための情報を提供していない。プログラミング演習は学生自身が個別的学習を行うことが基本であるため、個別指導に必要な学習の遅れや同じエラーでつまづいているなどの情報は不可欠である。

関連研究の問題点への本論文の解決手段は、模範解答からエラー要因を分析して教員へ提示する。教員はエラーの分析において一切作業する必要がない。また、一斉指導と個別指導が行えるように共通の問題を抱えた学生や行き詰まっている学生、および個別の解答履歴を把握して提示する。これらにより教員に負担をかけずに学習状況の把握を支援する。

4. 支援機能のシナリオに基づく設計

プログラミング演習の支援機能に対する設計は、具体的な要求を明確にするため、

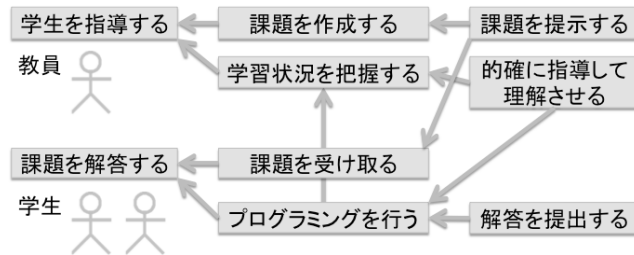


図2 アクターの目標

- 教師は課題を作成して学生へ提示し、学生は提示された課題を受け取る。学生は受け取った課題に対してプログラムの編集、一時保存、コンパイル、実行といったプログラミングを行う。システムはエラーが発生していることや、行き詰まっているといった学習状況を分析する。教師は学習状況を把握して指導を行う。学生は実行時に正解判定を確認して解答を提出し、教師は提出を確認する。

図3 プログラミング演習支援のシナリオ

「シナリオに基づく設計」[2]を用いる。シナリオに基づく設計は、分析対象のシステムに関わる以下の3つを時系列に沿って自然言語によってシナリオを記述し、どのようなシステムの機能が必要になるかを決定する。

- アクター
- アクターの目標
- 目標の実現手段

設計の手順は、次の順に行う。

- (1) アクターとアクターの目標を決めてシナリオを記述する
- (2) シナリオよりシステムの要求機能を抽出して、ユースケースシナリオを書く
- (3) システムの構成を決める

シナリオに基づく設計は、以下のようになる。

- (1) アクター、アクターの目標を決めてシナリオを記述する

プログラミング演習のアクターは、一人の教師と複数の学生である。要求分析において、ユースケースを書くときは基本的に典型例を想定して例外を扱わないことになっているが、授業支援システムでは、理解していない学生や意欲のない学生は例外ではなく、主要なアクターであるため、このような学生も想定してシナリオを書く。アクターの目標は、教師の主目標は、学生を指導することであり、学生の主目標は、課

表1 教員のユースケースシナリオ

機能名	ユースケースシナリオ
課題作成	教師は課題の内容、模範解答、出題日時を入力する。システムは入力された課題をDBに登録する。
課題提示	システムはDBに登録されている課題の出題日時が、現在の時刻より経過している課題を表示する。
学習状況分析	システムはDBに登録されているコンパイルエラーの傾向を分類してDBに登録し、模範解答に対応したエラーを分類して、共通の問題を抱えている学生のエラー内容と人数を分析する。 システムは、DBに登録されている解答開始時刻、コンパイル時刻、実行時刻、正解時刻、提出時刻を現在の時刻と統計分析して行き詰まっている学生を分析する。
学習状況提示	システムはDBに登録されている解答開始、コンパイル、実行、正解、提出したそれぞれの人数といった全体の学習進捗を表示する。 システムは学習状況分析機能で分析されたエラー分類の最も新しい時刻の結果、エラー状況を表示する。 システムは行き詰まっている学生がいる場合、表示する。 教師は全体の学習進捗とエラー状況を確認する。

題を解答することである。主目標は抽象的であるため、目標を達成するために副目標を階層的に展開して関連付ける(図2)。

- (2) シナリオよりシステムの要求機能を抽出して、ユースケースシナリオを書く

プログラミング演習支援のシナリオは、図3である。最も典型的な状況についてのシナリオを作る。シナリオとは、アクターが目標達成のために行った行為、シーンの記述である。システムの要求機能は、シナリオの動詞に着目して抽出する。シナリオから教師に対する要求機能は、課題作成、課題提示、学習状況分析、学習状況提示である。また、学生に対する要求機能は、課題受け取り、プログラムの編集、一時保存、コンパイル、実行、正解判定、解答提出である。

ユースケースシナリオは、システムの要求機能ごとに書くことでプログラミング演習に対する要求分析となる。表1は教員のユースケースシナリオである。プログラミング演習における学習状況は、演習全体の学習進捗、エラーの状況と演習の課題に対する解答プログラム、コンパイル結果、実行結果、提出状況である。演習全体の学習進捗とは、学生全員が今この課題に取り組んでいて、解答プログラムに対して、コンパイルや実行、正解、不正解、提出といった学習状況のことである。表2は学生のユースケースシナリオである。

表2 学生のユースケースシナリオ

機能名	ユースケースシナリオ
課題受け取り	学生は表示されている課題の内容を確認して、選択することで課題を受け取り、解答を開始する。 システムは解答を開始したことをDBに登録する。
プログラム編集	学生は課題に対するプログラムを入力する。
一時保存	学生は一時保存をする。 システムは入力されているプログラムをDBに登録する。
コンパイル	学生はコンパイルをする。 システムはプログラムに対してコンパイルを行い、プログラムと結果をDBに登録し、画面へ表示する。 学生はコンパイル結果を確認する。
実行	学生は実行をする。 システムはプログラムに対してコンパイルした結果の実行を行い、プログラムと結果をDBに登録し、画面へ表示する。 学生は実行結果を確認する。
正解判定	システムは実行の際に、学生の解答と模範解答を比較して正解、不正解を画面に表示する。
解答提出	学生は解答を提出する。 システムは提出した時刻をDBに登録して提出確認画面を表示する。 学生は提出状況を確認する。

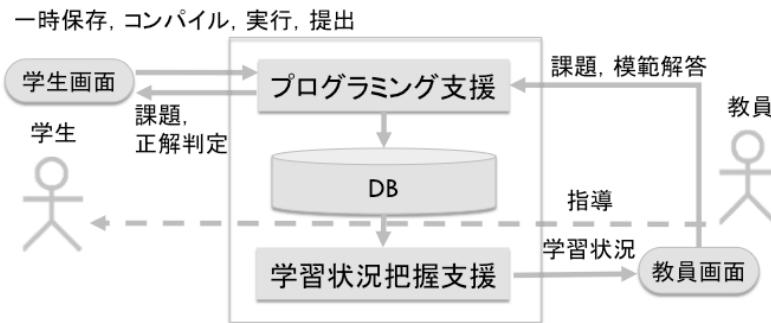


図4 プログラミング演習支援のシステム構成

(3) システムの構成を決める

プログラミング演習支援のシステム構成は、学生に対するプログラミング支援と教員に対する学習状況把握支援の2つのサブシステムで構成する(図4)。2つに分けた理由は、学習状況把握支援の有無による効果測定を行うためである。

5. 学習状況把握支援機能の設計

学習状況把握支援機能の設計は、「シナリオに基づく設計」の設計手順(2)のユースケースシナリオより詳細化する。学習状況分析機能は、ユースケースシナリオから3つのサブ機能を用いて分析する。

- (a) コンパイルエラー分類機能
- (b) エラー行分類機能
- (c) 行き詰まり検知機能

3つのサブ機能は、以下のように動作する。

- (a) コンパイルエラー分類機能は、DBよりコンパイルエラーメッセージを取得し、エラーの種類とエラー行を正規表現にて抽出し、エラー行を取得することでコンパイルエラーの傾向を分類する。
- (b) エラー行分類機能は、学生解答と模範解答の差分から対応行を同定する。「対応行」とは、エラーメッセージの先頭の学生解答のエラー行に対応する模範解答の行である。エラーメッセージで先頭に表示されたものが他のエラー要因にも繋がっているため、最初のエラーを抽出する。ここでの「エラー行」とはエラーメッセージで示された行であり、存在しないこともある。また、「対応する模範解答の行」の意味はつぎのように定義する。

- エラー行に対応する模範解答の行が存在するときはその行とする。
- エラー行に対応する模範解答の行が存在しないときは不明とする。
- エラー行が学生解答に存在しないときは、あるべき行に対応する模範解答の行とする。(ただし、あるべき行に対応する模範解答の行が存在しないときは不明とする)

対応行を同定する方法としてdiffによるテキストの差分(行の対応関係)を使う。この方法により、複数の学生が異なるエラー行でエラーが発生していても、同じエラー内容として分類し、エラー箇所と人数を把握する。

- (c) 行き詰まり検知機能は、学生のコンパイル、提出などの時刻を統計分析して、現在時刻が外れ値かどうかを統計的に判定することによって、学習の進みの遅い学生や解答に行き詰まった学生を発見する。外れ値かどうか検定するためにスミルノフ・グラブス検定を使用する。スミルノフ・グラブス検定は、標本データを入力し、検定で外れ値となった標本データが出力される。標本から外れ値となった

標本データを除外し、外れ値が検出されなくなるまで繰り返す。以下に提出時刻を例に検知方法を示す。

- 提出された学生解答の提出時刻リストを取得し、これを標本データとして外れ値検定を行う。
- このときに外れ値が検出された場合、提出時刻リストから、この提出時刻を取り除いて提出時刻リストに現在時刻を追加する。
- 再度、提出時刻リストに外れ値検定を行い、追加した現在時刻が外れ値として検出された場合は、まだ提出していない学生が行き詰まっていると把握出来る。

学習状況提示機能は、学生が解答プログラムに対して最後に行った操作の時刻や、その結果を元に分析した学習状況を教員へ提示する。教員へ提示する内容は次のとおりである。

- 全体の学習進度：課題に対する解答開始、コンパイル、実行、正解、提出の5つにおけるそれぞれの人数と、学習状況分析機能の行き詰まり検知機能で把握した行き詰まっている人数を提示する。
- エラー状況：学習状況分析機能のコンパイルエラー分類機能とエラー行分類機能で把握したエラー箇所とエラー内容、エラー人数を提示する。

なお、それぞれ人数から分類された学生の一覧を提示し、さらに個々の学生の学習進度に対する解答履歴も提示する。

6. 学習状況把握支援機能の評価

評価の方法は、実際の授業でプロトタイプシステムを使って支援機能の効果を検証する。支援機能の効果は、指導回数が増える、指導内容が変わることである。実験対象の科目は、本学の1年秋学期「プログラミング基礎・演習」の演習時間1コマ90分とする。演習の言語は、コンソールアプリケーションを中心としたJava言語である。

プロトタイプシステムは、次のアーキテクチャをもとにWebアプリケーションとして実装する。開発の使用言語は下記の通りである。

- プログラミング支援機能、学習状況提示機能:HTML, JavaScript, PHP
- 学習状況分析機能:PHP
- データベース:MySQL

プロトタイプシステムは、図5のように動作する。教員と学生の利用例は、それぞれ以下のシナリオである。

- 教員
 - (1) ログイン画面にアクセスしてID、パスワードを入力してログインする。
 - (2) 演習課題画面が表示されるので、課題作成ボタンを押す。

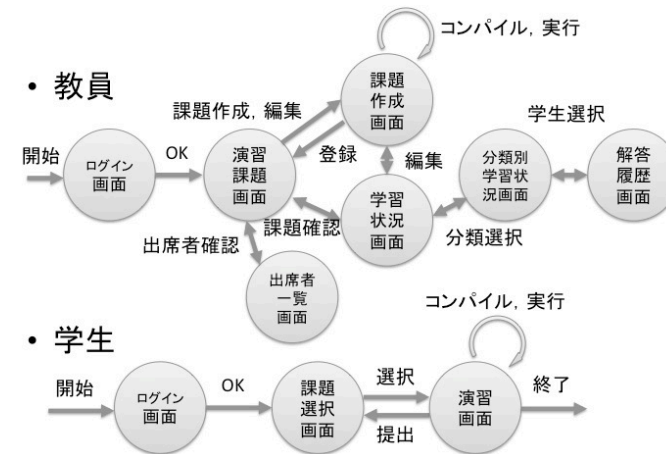


図5 プロトタイプシステムの動作

Javaプログラミング演習 演習課題 学習状況 ログアウト

課題名: デモ課題1 必須キーワード double: =: * 解答人数: 58 更新
最終更新時刻: 22:47:47

課題内容

半径5の円の面積を出力するJavaプログラムを作成しなさい。
ただし、面積の計算結果をdouble型の変数に代入してからその変数の値を出力すること。
また、円周率は3.14を使うこと。解答時間は10分です。

出題時刻 2011-10-13 21:10:00
終了時刻 2011-10-13 23:00:00

全体の学習進度

解答開始人数	コンパイル人数	実行人数	不正解人数	提出人数
58	55	49	13	46

まだ一時保存していない学生が3名います。 まだコンパイルしていない学生が9名います。 まだ実行していない学生が9名います。 まだ提出していない学生が12名います。

エラー状況

発生人数	エラー分類
1	必須キーワードがありません: double
1	必須キーワードがありません: =
1	1: <identifier> がありません。

図6 学習状況画面



図7 演習画面

- (3) 課題作成画面で課題に対する内容と模範解答プログラムを入力して、コンパイル、実行をして実行結果を確認してから、実施日、開始時刻、終了時刻を入力して登録することで出題する。
 - (4) 演習課題画面から出席人数をクリックすると出席者一覧画面が表示され出席者と座席番号、現在開始している課題内容を確認する。
 - (5) 演習課題画面から学習状況をクリックすると学習状況画面(図6)が表示される。
 - (6) 学習状況画面(図6)は全体の学習進捗とエラー状況が表示される。分類別の学習状況を確認したいときは分類の人数をクリックし、解答履歴画面を表示させる。エラー状況にて共通の問題を抱えている内容や、行き詰まっている人数の多いエラー内容を確認して一斉指導の判断材料を確認する。
 - (7) 解答履歴画面は課題に対する解答開始、一時保存、コンパイル、実行、提出といった操作の履歴を確認して個別指導を行う判断材料を確認する。
- 学生
 - (1) ログイン画面にアクセスして学籍番号、氏名、座席番号を入力してログインする。
 - (2) 課題選択画面が表示されるので課題を確認して課題名をクリックする。
 - (3) 演習画面(図7)が表示されるので課題に対する解答プログラムを入力してコンパイル、実行して実行結果を確認する。実行結果が模範解答と同じである場合、提出ボタンの横に正解と表示されるので提出する。提出すると(2)の課題選択画面に戻るので次の課題をクリックする。

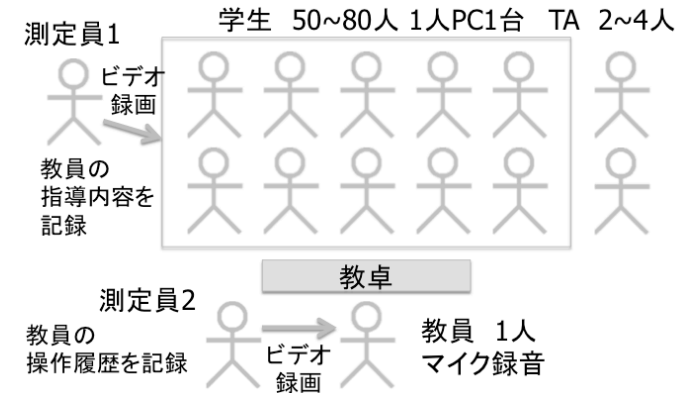


図8 評価実験の実施方法

表3 支援機能による指導回数の変化

支援機能		あり	なし
総指導回数		65	64
教員	A	38 (129)	37
	B	27 (98)	27
テーマ	算術演算子と式	16 (A)	9 (B)
	条件分岐	15 (B)	23 (A)
	論理演算子	22 (A)	18 (B)
	繰り返し for	12 (B)	14 (A)

支援機能の効果の測定方法は、指導回数が増加することに対して、教員の音声録音と授業風景をビデオ録画して、指導回数を測る。指導内容が変わることに対して、教員の音声録音と教員画面の操作履歴をビデオ録画して、指導の原因と結果の対応関係を測る。なお、指導の原因は学習状況画面、巡回、挙手のいずれかである。

評価実験の方法は、学習内容の影響と指導方法を考慮し、2人の教員で同じテーマの演習において、支援機能を使った場合と使わない場合を交互に2回(計4回)行い、一斉指導と個別指導(TAへの指導を含む)の指導回数と、指導の原因と結果の対応関係を測定する。評価実験の実施方法を図8に示す。学生が50から80人に対して教員が1人、TAが2から4人であり、測定員は2人で測定する。

実験の結果を表3、表4に示す。表3は、学習状況提示機能の有無による指導回数

表 4 指導の原因と結果の対応関係

支援機能	原因	結果		件数	合計
		指導分類	指導内容抜粋		
あり	エラー状況	一斉	文字列の連結方法	1	1
	分類別学習状況	一斉	・全体の学習進度から遅れている学生へサポートの要求を促す* (5) ・論理演算子を用いるよう指摘	9	29
		個別	・コンパイルの時間間隔が長い学生へ指摘* (2) ・不等号の使い方を指摘 (4)	20	
	巡回	一斉	switch 文について、色々なパターンを試すように指摘	1	27
		個別	プログラムの書き方を指摘 (6)	26	
	挙手	一斉	課題の内容をよく読むように指摘	2	8
		個別	for 文の中の変数の宣言方法	6	
なし	巡回	一斉	if 文について、偶数判定の方法	7	42
		個別	・偶数の判定方法 (8) ・文字列の連結方法 (5)	35	
	挙手	一斉	if, else 文を使うように指摘	4	22
		個別	・偶数の判定方法 (3) ・文字列の連結方法 (3)	18	

*支援機能によってしか把握できない状況に対する指導内容

をまとめたものである。表内の()数字は学習状況閲覧回数であり、英字は担当した教員を表す。指導回数に対する効果は、支援機能の有無によって総指導回数に差はないことがわかった。これは巡回によっても学習状況の把握が行えるためである。

表 4 は、指導の原因と結果の対応関係である。表内の()内の数値は同じ指導内容の件数を表す。指導内容は、支援機能によってしか把握できない状況に対する指導があり、変わったことがわかった。なお、支援機能によってしか把握できない状況とは、表 4 より、全体の学習進度から遅れている学生やコンパイルの時間間隔が長い学生がいることである。

支援機能に対する有用性の考察として、指導回数は教員 A が教員 B よりも学習状況を確認しているが、支援機能の有無に関係がないのは、巡回によっても学習状況を確認しているためであり、教卓に戻らなければ支援機能を利用出来ないことから、戻ら

ずに学習状況を把握して指導していた点が挙げられる。しかし、全体の指導回数のうち支援機能による情報が原因で指導にあたった割合は、65 回中 30 回の 46%となっているところから、巡回では分かりにくい指導に有益な情報が得られ、指導に結びついたと考えられる。指導内容は、学習状況提示機能を使ったときのほうが挙手の学生が、22 回から 8 回と 64%減少しているところから、助けを必要としている学生を的確に指導している。以上から、支援機能によってしか把握できない状況に対する指導ができることは定性的な有用性があると言える。また、支援機能によってしか把握できない状況に対する指導が全体の指導回数 65 回のうち 7 回の 11%あることから定量的な有効性があると言える。これらにより、本論文の提案機能は教員の負荷を軽減し、学生に対する指導の質を向上させて、教員と学生のコミュニケーションを促進させることにおいて有効であったと考えられる。

プロトタイプシステムの問題点は、学習状況の確認は教員画面でしか行えないため、教卓に戻らなければならない。また、学生全員の学習進度は課題ごとに見られるようになってきているが、全課題を一画面で見ることが出来ないことである。さらに、プログラムの実行でコマンドラインに引数を指定出来ないことである。

7. おわりに

学習状況を巡回せずに把握し、教員へ的確に支援することで教員と学生のコミュニケーションを促進させる学習状況把握支援機能の設計、実装を行い、プログラミング演習を対象として実現した。評価実験の結果から、教員は巡回では分かりにくい指導に有益な情報が得られ、助けを必要としている学生への的確な指導が行えるようになった。

今後の課題は、一般の授業を対象としたシステムを開発することである。また、授業支援システムに求められる他のコミュニケーション支援機能を開発することである。

謝辞 評価実験に協力していただいた日本工業大学情報工学科の高瀬浩史准教授、勝間田仁講師に感謝の意を表す。また、プロトタイプシステムの開発に協力してくれた、研究室の鈴木裕司氏、高石晃成氏、吉田真世登氏に感謝の意を表す。

参考文献

- 1) Chickering, A. and Gamson, Z.:Seven Principles for Good Practice in Undergraduate Education, AAHE Bulletin, March, a publication of the American Association of Higher Education (1987).
- 2) John M.Carroll, シナリオに基づく設計, 共立出版 (2003).
- 3) Moodle, <http://moodle.org/>

- 4) 役 誠雄, 西村朋子:大学教育の情報化と授業支援システム, 雑誌 FUJITSU, Vol.57 No.2 (2006-3).
- 5) 江本 理恵:ICTを活用した教育支援システムの導入とファカルティ・ディベロップメント-岩手大学の事例から-, 国立教育政策研究所紀要 第139集 (2010).
- 6) 加藤 利康,石川 孝:優れた授業実践のための7つの原則に基づく授業支援システムの要求分析, 情報処理学会コンピュータと教育研究報告, 2011.7
- 7) 加藤 利康,石川 孝:優れた授業実践のための7つの原則に基づく授業支援システム Moodleの機能拡張, 電子情報通信学会 教育工学研究報告, 2011.9
- 8) 加藤 利康, 鈴木 裕司, 高石 晃成, 吉田 真世登, 石川 孝:プログラミング演習のための学習状況把握支援システムの開発, 教育システム情報学会 第36回全国大会 (2011).
- 9) 倉澤 邦美, 鈴木 恵介, 飯島 正也, 横山 節雄, 宮寺 庸造:プログラミング演習における一斉指導のための学習状況把握支援システムの開発, 電子情報通信学会技術研究報告.ET, 教育工学 vol.104, no.703, pp.19-24 (2005).
- 10) 中井 俊樹, 中島 英博:優れた授業実践のための7つの原則とその実践手法, 名古屋高等教育研究, 第5号 (2005).
- 11) 中村 恵子:構成主義における学びの理論 心理学的構成主義と社会的構成主義, 新潟青陵大学紀要第7号, pp.167-177 (2007).
- 12) 放送大学学園:平成21年度・22年度 文部科学省先導的の大学改革推進委託事業「ICT活用教育の推進に関する調査研究」 (2011.3).