

CBT を活用した反転型授業の Java プログラミング授業での実践

山川広人^{*1}, 加藤巽^{*2}, 上野春毅^{*2}, 小松川浩^{*1}

^{*1} 千歳科学技術大学 情報システム工学科

^{*2} 千歳科学技術大学 大学院 光科学研究科

Trial of Flipped Classroom Using Computer-Based Testing in Java Programming Basics Course

Hiroto Yamakawa. ^{*1}, Sig Committee^{*1}

^{*1} Information Systems Engineering, Chitose Institute of Science and Technology

^{*1} Graduate School of Photonics Science, Chitose Institute of Science and Technology

本稿では、①CBT を使った知識習得レベル診断 ②個人や診断レベルに応じたグループでの実習準備 ③診断レベルや実習の到達度合いを用いた授業進度の動的な調整 の3点の工夫を施した、CBT を活用した反転型のプログラミング授業の実践について報告する。情報系学科の Java プログラミング科目を対象に、15回の授業の中で、上記の工夫を用いて動的に授業進度を調整した事例を示す。導入前との定期テストの結果の比較で得点分布が高得点側に推移している事を示し、工夫の効果を検討する。

キーワード: 研究会報告, 書式, 執筆要領

1. はじめに

プログラミング科目は知識・スキルの定着と積み上げが重要であるが、通常の講義・実習を一度に行う授業スタイルでは、理解につまずいた学生の把握・支援や、学生自身の挽回が難しい課題がある。こうした中で本研究は、基礎的な知識の取得を予習部分とし、予習を前提に知識・スキルの定着のための時間（実習やトレーニング）を授業内で確保する反転授業⁽¹⁾に着目した。しかしながら、反転授業は、単に予習を課せば効果が得られるものではなく、反転的な授業サイクルを軌道に乗せ維持する工夫が重要となる。

本研究グループでは、これまで主体的な学びにむけた適応型の e ラーニングの実現を目指す中で、項目反応理論 (Item Response Theory, IRT) を用いた形成的評価を狙った Computer-Based Testing (CBT) システムを開発してきた⁽²⁻⁴⁾。さらに CBT を活用した反

転的なプログラミング授業のモデルの提案⁽⁵⁻⁷⁾を行い、モデルを試行した科目において、定期試験結果の得点分布が高得点側に推移する効果などを示した。

本稿は提案モデルの汎用化を目指す中で行った、Java プログラミング授業での実践結果を報告するものである。2章では本研究で用いる CBT について述べる。3章では、提案モデルと、実践を行った Java プログラミング授業への導入方法を述べる。4章では実践結果を述べ、5章で提案モデルの効果を検討する。

2. 形成的評価を狙った CBT システム

本研究で利用する CBT は、学生の形成的評価を可能とすることを狙い、IRT を用いて、ある單元における学生の知識習得レベルを7段階で診断できるようにしている⁽²⁻⁴⁾。CBT の利用イメージを図1に示す。予習時など、学生の知識習得にむけた学習は、解説教材

や演習機能での演習問題を通じて行われる。演習問題には難易度（後述する診断時にむけたレベル 1～7）が付与されており、学生は単元の中で難易度を自由に選んで演習を進められる。知識習得レベルの診断時など、学生の形成的評価は CBT 機能によるテスト問題を通じて行われる。テスト問題は、演習機能の演習問題と全く同一のものが利用される。学生が出題されたテスト問題を解くたびにその正誤データが蓄積され、IRT による難易度判定モジュールが新たに問題すべき難易度を調整した上で、次の問題を出題する。本研究では、テスト終了時に診断された学生の実力（到達した難易度）を、知識習得レベルとして見なしている。

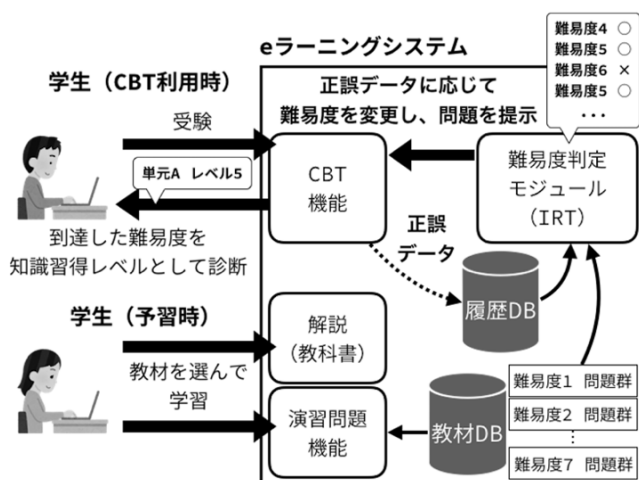


図 1 CBT の利用イメージ

3. 反転型のプログラミング授業モデル

前述の CBT を用いることで、例えば反転授業では、予習の段階で解説と演習を学生に指示し、授業では CBT を用いて予習の成果を診断するといった工夫を狙えるようになる。また診断結果を授業内でさらに活用し、学生の状況に適応した学習を図る工夫も狙えるようになる。この章では授業モデルの概要を紹介し、実践環境となる Java プログラミング授業と導入方法について述べる。

3.1 CBT による反転型プログラミング授業のモデル

本研究グループでは、①CBT を使った知識習得レベル診断 ②個人や診断レベルに応じたグループでの実習準備 ③診断レベルや実習の到達度合いを用いた授業進度の動的な調整 の 3 点を工夫した反転型のプログラミング授業をモデル提案してきた⁽⁶⁾⁷⁾。このモデル

の毎回の授業シナリオは次の(1)～(6)の流れで進む。

- (1) 学生は e ラーニングでの予習を前提に出席する
 - (2) 授業開始時に、学生は予習内容と紐付いた CBT で知識習得レベルの診断を行う
 - (3) 学生は個人ワークで、その回の実習に求められる具体的な知識や文法を確認し補習する
 - (4) CBT の結果を反映したグループの中で、学生は個人ワークの結果を確認し、時には教え合う
 - (5) 学生は個人でプログラミング演習を行い、より深い知識の理解や、知識の利用方法を訓練する
 - (6) 演習を終えた学生は、その日の自分の理解度を振り返り、カードにコメントとして記入・提出する
- 前述の工夫①は(1),(2)、工夫②は(3),(4)に関わるもので、工夫③は(2)～(6)の状況に応じて行う。詳細を以下に述べる。

3.1.1 提案モデルの工夫とその狙い

工夫①では、CBT を使うことで学生が予習・既習している知識を習得レベルとして診断する。反転授業サイクルを軌道に乗せるためには、その日の授業や実習のスタートラインに立つために、学生が予習や前段階の授業を通じた前提知識を備えておく必要がある。工夫①を取り入れることで、学生自身はもちろん、教員は授業の履修者全体に渡る既習状況を、IRT の基準に基づいた知識習得レベルの数値で確認できることを狙っている。

工夫②では、学生は配布されたワークシート（例：知識を問う穴埋め問題やプログラミングの文法問題）を用いて、授業内の実習やトレーニングにむけた知識の補習や応用方法の確認を行う。学生の予習が行われていたとしても、全員が同じ知識習得レベルに到達できるとは限らない。工夫②を取り入れることで、学生がその日の実習や知識のトレーニングにむけ、自身に不足している知識の補習や準備を行えることを狙っている。ワークシート上の不明点には、予習用の解説教材などを見て取り組むことや、教員・ティーチングアシスタントに個別にアドバイスを求めながら進めることを想定している。グループワークでは工夫①の成果も利用し、知識習得レベルの偏りを避けた学生のグループングを行う。この中でグループ全員がワークシートの内容を正答・説明できることを目標に据えることで、レベルの高い学生が他の学生を教えるといった学

生間の相互作用も狙っている。

工夫③では、授業の中で得られる(2)の診断状況、(3)、(4)の個人・グループワークの進捗状況、(5)のプログラミング実習の進捗状況、(6)の振り返りコメントなどを判断材料として、教員が授業の中でつまづいている学生の存在を俯瞰的に確認し、授業進度を調整する。実際の授業では、授業スケジュールの進行速度や単元の難易度の上昇に応じて、一部の学生が授業スケジュールの半ばから取り残されてしまう例も快々にして起こりうる。工夫③を取り入れることで、知識習得レベルや、知識の補習・実習の進捗を測定し、学生が授業の中で置かれている状況に素早く適応し、補習の機会を設けるといった対策を可能にすることを狙っている。具体的な手段として、次回の授業内容へのフィードバックや、授業スケジュール自体の切り替え（補習内容を取り扱う授業の追加）ができる。これは、反転授業全体のサイクルの維持に重要と考えている。

3.1.2 提案モデルの適用方法

これらの工夫を授業に適用するには、予習/CBT の教材内容と授業内容が密接につながっている必要がある。また工夫③の授業進度の調整は、毎回の授業を区切りとして行うことを想定している。このため提案モデルの導入時には、知識の積み上げが段階的に行われるよう、ひとつの単元の内容を予め初級編、中級編、上級編といった階段状の段階に分け、これを各回の授業に順に割り当てる形で授業スケジュールの初期設計を行う。さらに単元の中で、初級編・中級編・上級編の授業内容と予習/CBT の演習問題の難易度を対応させる。これにより、各回の授業の受講に必要な知識習得レベル(学生にとっては各回の予習の最低到達目標)が明確になり、「ある単元の中級編の授業で学生をつまづきが見られた場合には、上級編の前に、中級の予習・実習が行える授業を追加する」といった進度調整が行いやすくなる。

3.2 Java プログラミング授業への適用

3.1 の授業モデルを実践環境となる Java プログラミングの授業に盛り込むには、単元の中での段階分け、授業日への割り振り、対応する予習/CBT 用の教材準備を行い初期スケジュールとして定める必要がある。単元・段階・教材の難易度の対応を表 1 に示す。実践

表 1 授業の初期スケジュールと診断レベルの対応

#	単元	授業内容	授業内容と予習/CBT の対応
1	基本文法と 実行方法	初級編	レベル 3 程度
2		中級編	レベル 5 程度
3		上級編	レベル 7 程度
4	クラス構造	初級編	レベル 3 程度
5		中級編	レベル 5 程度
6		上級編	レベル 7 程度
7	クラス設計	初級編	レベル 3 程度
8		中級編	レベル 5 程度
9		上級編	レベル 7 程度
10	ライブラリと 例外処理	初級編	レベル 3 程度
11		中級編	レベル 5 程度
12		上級編	レベル 7 程度
13	調整用の予備・中間テストなど		
14			
15			

は、とある大学の情報系学科の Java プログラミングの基礎科目（学部 2 年・秋学期開講）を用いる。

教材について、解説（教科書）となる部分は、教員が単元の内容を説明する動画を作成した。従前の授業で、対面での解説に用いたスライドに教員の音声吹き込み、動画として再生できるようにしたものである。動画は 1 つあたり 3～5 分の内容で、単元ごとに 10～15 個程度準備した。音声は、従来の対面型の知識の解説が動画で代替できるものになるよう意識した。動画は一般的な動画サイトで学生に公開した。演習問題（CBT のテスト問題を兼ねる）部分は、レベルごとに 8 問～10 問を目安として、単元ごとに約 60 問～80 問を整備した。これは、先に述べた CBT 機能の難易度判定モジュールの仕様により、レベルごとに最低 8 問以上の問題セットが必要であるためである。

提案モデルに沿った授業を進めるには、学生がきちんと予習を行う体制が前提となる。今回の実践では、履修前の学生に表 1 の初期スケジュールを予め提示した上で、「単元の割り当てられている授業の中で CBT による知識習得レベルが 7 と診断されることを目指し

て予習する」こと、「診断されたレベルは最終的な成績の配点に組み込む」ことを伝え、学生にとっての予習の意義を持たせている。

4. 実践結果

3.2 で述べた適用準備を行った上で、平成 29 年度秋学期（履修者 75 名）の授業で実践を行った。工夫①，②，③に関連する実践結果を以下に示す。

4.1.1 工夫①に関する結果

工夫①では、学生の予習・既習の知識習得レベルを診断できることを狙っている。2 つの単元の診断人数の推移を図 2・図 3 に例示する。両方の単元で、初回は多くの学生がレベル 1～2 にとどまっている点が共通している。図 2 の単元では、3 回目の授業でレベル 5,6,7 に人数が集中している一方、図 3 の単元では、3 回目の授業でもレベル 1 や 4 にとどまっている学生がいる。「クラス構造」の単元に入り難易度が上がったことにより、授業でワークやプログラミングに取り組むための知識が不足している学生が増加している可能性や、予習/診断のための教材が学生にとって十分に学習しやすいものではない可能性が考えられる。こうした可視化は、工夫③でも重要な判断基準となる。

4.1.2 工夫②に関する結果

工夫②では、個人ワークやグループワークを通じた、実習や知識のトレーニングにむけた学生の知識の補習や準備を狙っている。実際の授業では、学生は、わからない点は予習用に準備されている動画や、ティーチングアシスタント・教員・グループワークのメンバーからのアドバイスを参考に解決するルールとして、教員が改めて講義を行うことはしなかった。表 2 は、図 2・図 3 で例示した授業での、プログラミング演習の達成率を示している。例として、図 2・図 3 の 1 回目にあたる授業では、レベル 3 程度の内容が必要とされる授業内容が展開されている。一方で図 2・3 を見ると、ほとんどの学生がレベル 1 に偏っている。2 回目・3 回目でも、授業内容に必要なレベルを下回っている学生は多い。しかし表 2 の実績によれば、こうした状況でも 9 割以上の学生がそれぞれの難易度の課題を達成できている。個人ワークやグループワークで、予習やこれまでの積み上げの中で不足していた知識の補習が進

図 2 「基本文法と実行方法」のレベル診断人数

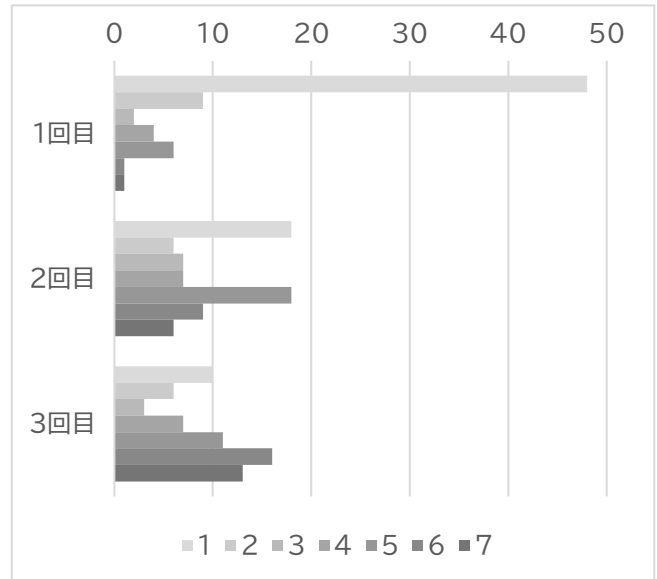


図 3 「クラス構造」のレベル診断人数

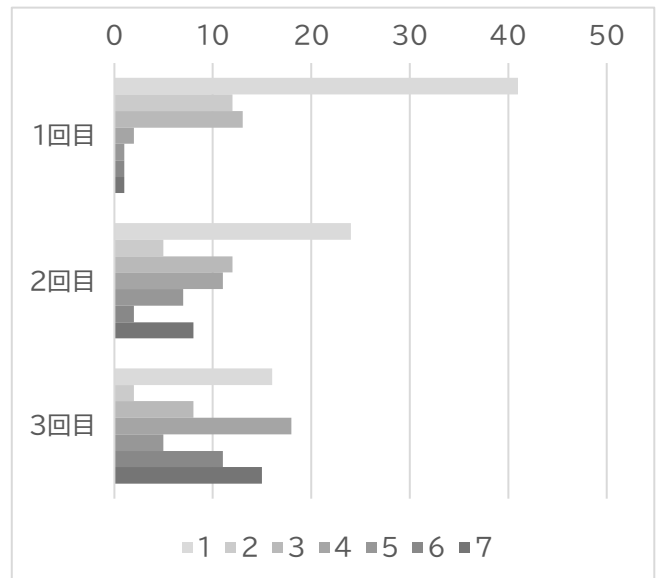


表 2 各回の演習達成率（演習完了者÷出席者）

#	単元	授業内容と予習/CBT の対応	達成率
1	基本文法と実行方法	レベル 3 程度	99%
2		レベル 5 程度	93%
3		レベル 7 程度	97%
4	クラス構造	レベル 3 程度	93%
5		レベル 5 程度	99%
6		レベル 7 程度	85%

んでいる効果の現れと考えている。

4.1.3 工夫③に関する結果

工夫③では、図2・図3や表2の情報に現れてくる情報に基づいて、つまづいている学生の有無を俯瞰的に確認し、授業進度を動的に変更できることを狙っている。本稿の実践では「クラス構造」の3回目の状況に着目した。図3を見ると、「クラス構造」の3回目では、大半の学生の知識習得レベルが4以下であると診断されている。また、同じく演習達成率もそれまでに比べて極端に低い。実際の授業では、学生がグループワーク・プログラミング演習に多くの時間をとられており、授業時間が終了してから1時間程度残って演習を完了した例も多数見られた。この回の学生の振り返りコメントでは、「難しい」「理解できない」「死にそうです」といったネガティブな趣旨が目立った。これらの点から、「クラス構造」の3回目は学生にとって難易度がこれまで以上に上昇しており、プログラミング演習を達成できない、まさに授業につまづいてしまった学生が発生している可能性があるかと判断した。そこで実践では、予備回から1回分を「クラス構造」の割り当てに充当し、補習回として4回目を実施するように授業スケジュールを変更した。また「クラス構造」の中級編まではレベル診断の人数も課題達成率も順調であったことから、4回目ではワークシートもレベル4から7（つまり、中級から上級編）の中で段階的な確認内容や類題を用意し、知識定着の重点化を意識した。4回目のレベル診断人数を図4に、演習達成率を表3に示す。これによりレベル4以上と診断された学生も増え、また課題達成率も96%に引き戻すことができた。学生の振り返りも「前回よりも理解できた」「流れがわかった」「インスタンス化（クラス構造の知識の語句）が理解できた」といった前向きな趣旨のコメントが増えた。これらを確認した上で、次の「クラス設計」の学習に移ることができた。その後、「クラス設計」の授業でも同様に動的な変更を行った。

表4は、前年度に行われた中間テスト・期末テストとの得点分布の比較である。期末テストは、全く同じ問題で実施している。中間テストは都合上、実施時点が異なってしまったため、既習範囲からの出題になるように10%程度の問題を入れ替えている。したがって中間テストはそのまま比較することが正確とはいえないが、どちらも得点分布が高得点側に推移している。

図4 「クラス構造」4回目のレベル診断人数

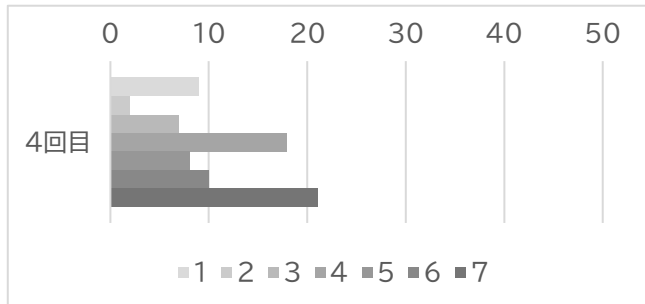


表3 「クラス構造」4回目の演習達成率

#	単元	授業内容と 予習/CBTの対応	達成率
7	クラス構造	レベル7程度	96%

表4 定期テストの得点分布比較（人数）

	～60	～70	～80	～90	～100
H28 中間	26	16	12	10	3
H29 中間	2	19	14	16	16
H28 期末	2	19	14	16	16
H29 期末	1	2	20	20	29

注) 受験者数：H28年度67名、H29年度72名

5. 効果の考察

4.1.3で示したように、本稿の実践では「クラス構造」の3回目（上級編）の授業で、レベル診断結果や演習達成率、グループワークや演習、振り返りでの学生の様子をアラート情報として捉え、動的に次週の授業を、内容を絞った4回目（補習）を行うよう、授業スケジュールを切り替えた。図4、表3は、3回目につまづいた学生達を4回目で挽回させ、次の単元に移る体制を整えることができたことを示唆している。実践では授業スケジュール自体を切り替える形となったが、授業の難易度（例として、初級・中級・上級や、対応する知識レベル）を調整するといったゆるやかな反映も可能であろう。

学生の定期テストの結果では、本手法を実践した平成29年度の方が、学生の得点分布が高得点側に推移している。これは提案手法を用いることで、授業の中で定量的・定性的な視点で学生の状況を判断し、授業進度を調整できるようにしたことで、昨年度は学生が

つまづいていた単元(進度調整を行った「クラス構造」「クラス設計」)で、学生が挽回できる機会を実現できたことの現れと考えている。さらにこうした授業進度や内容の調整の検討は、従来は教員が講義に利用していた時間を、個別のアドバイスや、授業内でリアルタイムに学生の理解度や進捗を確認・判断し動的な授業計画の調整を図る時間に転換して行える。この時間の利用方法の転換を可能にすることが、提案モデルの導入の大きな利点と考えている。

6. おわりに

本稿では、①CBT を使った知識習得レベル診断 ②個人や診断レベルに応じたグループでの実習準備 ③診断レベルや実習の到達度合いを用いた授業進度の動的な調整 の3点の工夫を施した、CBT を活用した反転型のプログラミング授業を、情報系学科の Java プログラミング科目を対象に導入し実践した。実践結果では、教員が授業の中でつまづき始めた学生の存在を俯瞰的に判断し、動的に補習を実施して挽回を図る事例を示した。しかしながら提案モデル中の3つの工夫の貢献度など、細部にわたる検証は行えておらず、今後の課題である。提案モデルの導入には事前準備のコストも大きく、引き続き汎用性の追求が課題となる。

謝辞

本研究は JSPS 科研費基盤研究(C) 17K00492 の助成を受けたものである。

参 考 文 献

- (1) 重田勝介:"反転授業 ICT による教育改革の進展", 情報管理, Vol.56, No.10, pp.677-684 (2013)
- (2) 平澤梓, 光永悠彦, 小松川浩:"項目反応理論を用いた適応型 e ラーニングシステムに関する研究", 教育システム情報学会研究報告, Vol.29, No.2, pp.49-50 (2014)
- (3) 平澤梓, 光永悠彦, 小松川浩:"項目反応理論を用いた適応型テストの能力値推定精度に関する研究", 教育システム情報学会研究報告, Vol.30, No.2, pp.33-94 (2015)
- (4) 吉田史也, 光永悠彦, 山川広人, 小松川浩: IRT を用いた反復学習を目的とした e ラーニングシステムの試行, 教育システム情報学会研究報告, Vol.31, No.2, pp.23-26

(2016)

- (5) 加藤巽, 上野春毅, 吉田史也, 立野仁, 山川広人, 小松川浩:"適応型学習支援システムの反転授業への導入と評価", 教育システム情報学会研究報告 Vol32, No.2, pp.79-82 (2017)
- (6) 山川広人, 上野春毅, 立野仁, 深町賢一, 小松川浩:"CBT を中心とした反転型プログラミング実習の実践", 私立大学情報教育協会 平成 29 年度教育改革 ICT 戦略大会資料, pp.202-203 (2017)
- (7) Ueno, H., Kato, T., Yoshida, F., Tsukada, N., Tateno, H., Fukamachi, K., Yamakawa, H. and Komatsugawa, H. : A Model of Flipped Classroom Using an Adaptive Learning System", 25th International Conference on Computers in Education: Work in Progress Posters Proceedings, pp.1-3 (2017)