

# プログラミング教育における反転授業の試み

高井久美子 水谷晃三

帝京大学

## 反転授業への期待

「教える」から「学ぶ」へと教授学習パラダイムの転換の流れを受け、教育の質的転換に向けて、高等教育機関ではアクティブ・ラーニングに深く関心が寄せられている。アクティブ・ラーニングや反転授業といったキーワードがニュースで取り上げられており、世間一般でも広く注目されていると感じている。反転授業とは、授業時間内と授業時間外で行う学習活動を入れ替えた授業形態である。一般的な授業では授業を受けてから宿題に取り組む。反転授業では、授業の前に学習内容に関する知識を学び、授業中にはそれを応用して問題を解くといった活動を行う。たとえば、授業の前に講義ビデオを見ておき、授業中に課題に取り組むといった具合で、これまで宿題としてやっていた「問題を解く」ことを授業時間中にやり、授業時間中にやっていた「講義を聞く」ことを宿題としてやるのである。

反転授業は、これまでは宿題であった課題に1人で取り組む代わりに、授業時間内に教室で先生や仲間と一緒に取り組むことができるところに特徴がある。「本当に教員の存在が目の前に必要になるのは、勉強につまずいて個別の手助けをもとめているときだ。大勢に向けて内容を喋るだけなら、教室で対面する必要はない」というアイデアから始まったことであったと Aaron Sams はその著書「反転授業」<sup>1)</sup>で述べている。

本稿では、プログラミング教育において反転授業を行った事例を報告する。プログラミングの学習は知識を学ぶだけでなく、実際に知識を活かしてプロ

グラムを作ることが重要であるが、プログラミングの学習を始めたばかりのころは実際にプログラムを作ろうとしたときに小さな問題が障壁になり、先に進めなくなってしまう受講者が見受けられる。個別の手助けが必要なときにこそ教室内で対面して対応ができればと考え、筆者らはプログラミング科目で反転授業を実施することにした。すなわち、受講者は事前学習として講義ビデオを視聴して学習すべき内容を学んでから授業に臨み、授業中はプログラムを作る課題に取り組むといった形態である。本稿では授業設計や授業準備を中心に授業実践について紹介する。授業実践の詳細については、文献2)をご参照いただきたい。

## プログラミング教育への反転授業の導入事例

### □ 導入の背景

筆者らの学科では大学1年生の前期必修科目の1つとして初学者向けのプログラミング授業を設定している。表-1は本授業の授業計画であり、Javaを用いた演習を含む内容となっている。プログラミングの基本的な概念、接続、条件分岐、繰り返し処理の記述と実際のプログラムの振舞いをそれぞれ理解させることを主な学習目標としている。

プログラミングにおいては、プログラムのソースコードを作成する作業やプログラム実行時の実際の動きなど、教科書などテキスト中心の教材では十分に説明できない内容もある。これを補うため、これまでは授業時間中において15～20分程度の講義を行い、その後に課題プログラムの作成演習を行う授

業構成としていた。この講義時間をなくして演習時間を多く取るための方策が講義ビデオを用いた反転授業の導入である。

## □ 反転授業の準備

### ● 反転授業の導入の方針と講義ビデオの構成

本事例の反転授業では事前学習用の教材として講義ビデオを用意した。反転授業を初めて経験する受講者がほとんどであることと、講義ビデオの事前視聴の負担に配慮して、15回の授業のうち学習内容が新しい内容に切り替わる授業回(表-1中☆印)を反転授業とした。第5回の授業(※印)はプレ反転授

授業回	内容
1, 2	プログラミングの概要, Java の特徴, 基本的な文法
3	プログラム作成環境, 例題プログラムの作成と実行
4, 5 ※	図形描画プログラムの作成
6 ☆	変数と演算
7 ☆, 8	繰り返し処理 (While と for), 多重ループ
9	中間試験と振り返り
10 ☆, 11	条件分岐処理, 論理演算子, 繰り返し処理との組合せ
12 ☆	文字列の取り扱いと表示
13 ☆, 14	総合演習 (Calendar クラス, 型の変換, Math クラス, 複雑な図形)
15	期末試験とまとめ

※反転授業の説明と動画配信システムの操作に慣れてもらうため、第5回授業もプレ反転授業として実施した。

表-1 反転授業を導入した授業計画

業と位置付けて、反転授業の受け方や講義ビデオの視聴方法などを説明する機会を設けた。

講義ビデオの内容は、従来授業時間中に行ってきた講義をPCの画面キャプチャソフトウェア(スクリーンキャストツールなどともよばれる)を用いてビデオ化したものであり、1本あたり10~15分程度の長さである。図-1に示すように、画面上でLMSとよばれる学習支援システムで提供している教材や実際のソースコードなどを表示し、マウスポインタで示しながら音声で解説する構成とした。

### ● 講義ビデオの作成

講義ビデオの作成にあたり、以下の機材を用意した。

- 収録用のPC(授業で使用するものと同等の開発環境をインストールしたもの)
- 画面キャプチャソフトウェア
- マイク(音声収録向けの専用のもの)

本事例においてはMicrosoft WindowsがインストールされたPCを用意し、画面キャプチャソフトウェアには同社製のExpression Encoder 4(以下EE4と略す)を使用した。同種のソフトウェアは数多く存在するので使いなれたものを使用するのがよいだろう。

具体的な収録手順は次のとおりである(図-2)。まず、①EE4上で映像の入力ソースを画面とし、音声の入力ソースをマイクに設定する。次に、②出力先の設定としてファイルをアーカイブするよう

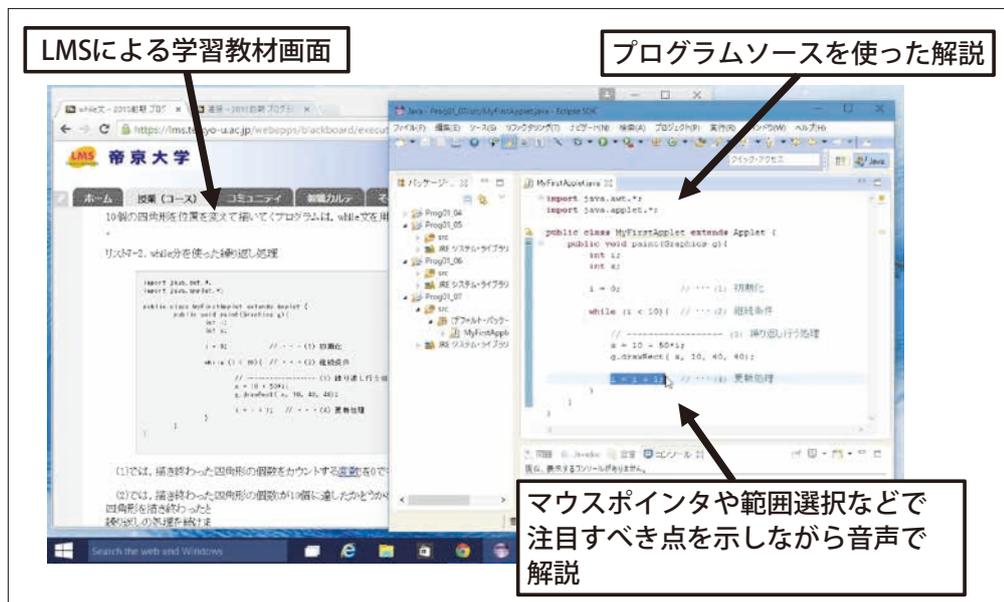


図-1 講義ビデオの画面例



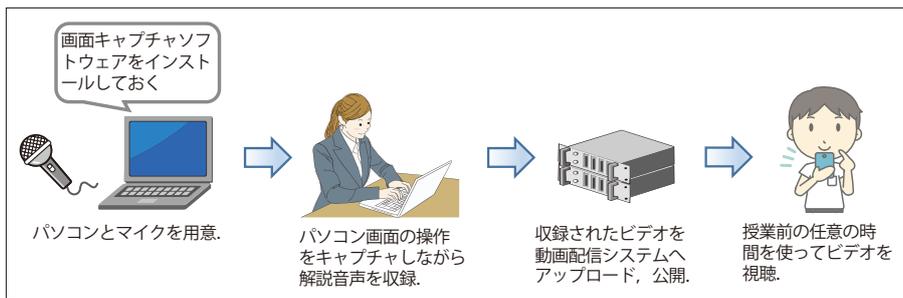


図-2 講義ビデオの準備から受講者への公開までのステップ

に指定する。以上の設定ののち、③開始ボタンを押して収録を開始する。再びEE4上の停止ボタンを押すまで、画面上の操作とマイクの音声と同時に1つのビデオファイルとして記録される。

本事例では、収録後にオーサリングソフトウェアを使用して、字幕を加えたり、不要と思われる内容をカットしたりするなどの編集を行った。完成した動画を動画配信システムにアップロードし、受講者が視聴できるようにした。

### ● 解説時の台詞について

収録後の編集作業を伴わなければ講義ビデオの作成にかかわる時間は実際の講義に要する時間と大きく変わらない。しかし、収録後の動画を確認してみると、説明の段取りが悪かったり、何度も言葉を噛んだり、同じことを繰り返し喋っていたりして自分でも気になってしまう。気にせずこのまま受講者に公開してもよいのだが、あらかじめ原稿を作成してこれを基に収録する方法を試みたこともある。

この方法では、まずナレーション原稿を作成する。15分ほどの講義ビデオに対する原稿はA4で3～4ページになった。次に、この原稿を音読して音声を収録する。それでも言葉を噛んで録り直したりすると、音声の収録は15分では終わらない。その後、収録した音声を再生しながらその内容に合わせてPCを操作して画面をキャプチャし、オーサリングソフトウェアを使って1本の講義ビデオとして仕上げる。以上の作業のため完成には相応の時間がかかる。作業負荷は小さくないがナレーション原稿が残っているため講義内容の見直しはやりやすくなる。

### ● ビデオ収録上の配慮

本事例ではPC上の画面をそのまま収録するため以下のような点を配慮した。

- 画面上のメニューやソースコードなどの文字が読みにくくならないように画面解像度やテキストのフォントサイズを調整した。具体的には、PCの画面解像度を1280×720ピクセルに設定して動画解像度もこれに合わせて収録(720p/30fps)した。
- 受講者がどのような環境で講義ビデオを視聴するのかをあらかじめ想定し、スマートフォンなど画面サイズが小さい場合でも文字や図形が小さくならないようにした。具体的には、テキストエディタのフォントサイズを大きめに設定したり、視認しやすい太めのフォントを選択したりした。

また、音声の収録にはピンマイクやいわゆるボイカル用マイクなど音声収録専用のものを用いた。PCに内蔵のものやテレビ会議向けの指向性の広いマイクは周囲の環境音が入りやすいためである。

## □ 授業の実施と結果

### ● 授業の実施と動画配信システムの視聴記録の活用

受講者には次回授業において反転授業を行う旨をあらかじめ伝えておき、授業当日は講義を一切行わずに課題プログラムの作成演習に取り掛かるという構成で反転授業を実施した。

動画配信システムには、受講者が講義ビデオを再生した時刻、受講者のID、再生時間などの視聴記録が蓄積されている。この記録を活用することで受講者の動画視聴率が推定できる。動画視聴率の低い受講者には個別にヒアリングを行い、十分な理解ができていない受講者には授業時間内にビデオを視聴させ、理解できたのちに課題プログラムの作成演習に取り掛かるように指示した。なお、授業全体の視聴率は全体で80.0%であり、視聴率の少ない受講者にはすでに理解している内容のた

め視聴しなかったという者もいた。

### ● 受講者の声

すべての授業が終了した後に反転授業に関するアンケートを行った。その結果、反転授業を「よかった」とする回答が94.6%となるなど好意的な結果が得られた。自由回答には「予習ができるのでとても良い」「理解するまで時間をかけてできる」「これからもこの反転授業を続けてほしい」などの意見があった。一方、「良くなかった」と答えた受講者の自由回答には「動画を見ても難しすぎる」「聞いていても分からない」「画質が悪い」「文字が小さい」などとあり、講義ビデオの内容や構成に改良の余地があることが分かった。

### ● 教授者側のメリット

教員側の実感として最大の効果は、演習時間中に受講者からの質問に対応できる時間が増えたことである。特にプログラミング教育においては、同じ要件に対するプログラムの書き方は千差万別であるため、受講者個々の書き方、考え方に応じた指導が欠かせない。そのための時間を多く確保できる点において、反転授業の効果はあったと考えている。

講義ビデオを使った反転授業にしたことで、ビデオを表示しながら「この部分の説明が分からない」といった具体的な質問が増えるようになった。そればかりでなく、「この説明の意味は理解できたけども、課題を解くためにどう適用するのか分からない」などのように、ビデオで得た知識を問題解決に適用する過程で生じる質問もみられるようになった。このような、受講者の能動的な学習活動における疑問に応じやすくなることも講義ビデオを用いた反転授業の1つの利点といえる。

## 今後の展望

プログラミング教育における反転授業について、準備と実施を中心に述べた。問題点としては、事前学習をしてこない受講者への対応と、事前学習として講義ビデオは視聴するものの内容を理解してきているかどうか分からないといった点が挙げ

られる。また、反転授業が主体的な学習につながっているのだろうかという疑問もある。

これらの問題に対して、現在、所属する学科のプログラミング教育に携わる教員が一丸となっており、カリキュラムや授業方法の検討に取り組んでおり、本事例後の授業においては以下のような改善案を実施している。まず、事前学習をしてこなかった受講者は、そのほかの受講者と着席エリアを別にして、授業時間中に事前学習から始めてもらうようにしている。事前学習における理解を確認するために、事前学習の教材として講義ビデオを見ながら書き込むワークシートを準備し、配付している。さらに主体的な学習となることを期待して、林らがプログラミング教育における反転授業でグループで活動を行った事例<sup>3)</sup>を参考に、授業時間中にグループワークの時間を設けて受講者同士の教え合いの機会を作るほか、プログラミングの読み書きスキルについてのルーブリックを用いて、毎授業時間の自分の到達度を受講者自身が自己評価する方法を取り入れている。また、これらの学習活動を細かくフォローできるように、SA (Student Assistant) 1名が6名から8名程度の受講者を担当する少人数担当制を取り入れている。今後も実践を積み重ね、より良い学習に向けた方策を模索していきたいと考えている。

### 参考文献

- 1) Bergmann, J. and Sams, A. (著), 山内祐平, 大浦弘樹(序文・監修), 上原裕美子(訳): 反転授業, オデッセイコミュニケーションズ(2014).
- 2) 水谷晃三, 高井久美子: プログラミング初学者を対象にした動画教材による反転授業の実践と評価, 情報処理学会研究報告, Vol.2015-CE-132, No.34, pp.1-8 (2015).
- 3) 林康弘, 深町賢一, 小松川浩: プログラミング教育における反転授業の実践と評価, 教育システム情報学会, 第40回全国大会, A2-2, pp.97-98 (2015).

(2016年5月31日受付)

高井久美子 (正会員) kumiko@ics.teikyo-u.ac.jp

宇都宮大学大学院工学研究科修士。博士(工学)。帝京大学医学図書館司書, 同理工学部情報科学技術職員, 助手, 助教を経て, 情報電子工学科およびラーニングテクノロジー開発室講師。

水谷晃三 (正会員) mizutani@ics.teikyo-u.ac.jp

帝京大学大学院理工学研究科修士。博士(工学)。帝京大学医療情報システム研究センター 助教, 講師を経て, 同理工学部情報電子工学科講師。

