



# プログラミング実習教育のオンライン化における工夫とその学習効果および今後の活用に関する一考察

有 田 宙 史

**概要** 新型コロナウイルス（COVID-19）の蔓延により社会環境は一変した。社会全体がリモート環境への推進を余儀なくされる中、教育現場も例外ではなかった。しかしながら、業種によってもそうであるように講義内容によってオンライン化の難しいものが存在する。本稿では筆者が担当するプログラミング実習講義のオンライン化で得た知見を元にオンライン講義における課題や問題点を論じ、善後策や今後の活用法を考察する。

**キーワード** プログラミング実習教育, オンライン講義, リモート環境, BYOD

**原稿受理日** 2022年1月11日

**Abstract** The social environment has changed completely due to the spread of the novel coronavirus (COVID-19). Educational settings were no exception as society as a whole was forced to promote remote environments. However, as is the case with some industries, there are some lecture contents that are difficult to go online. In this paper, based on the knowledge gained by making the programming training lecture that I am in charge of in a remote environment, I will discuss issues and problems in online lectures, and consider the remedial measures and future utilization methods.

**Key words** programming training education, online lectures, remote environment, bring your own device

## 1. はじめに

まず、本稿では今後、永続的に社会がリモート（オンライン）環境を義務化もしくは必要とする前提で論じるわけではないことを明確にしておく。1918年から1920年までの間に猛威を振るった通称スペインかぜと呼ばれるインフルエンザ・パンデミックなど、これまで人類がいくつものパンデミックを克服してきた歴史を見ても分かる通り、いつかはリモート環境が義務化もしくは必要とされる事態は終息し、従前の社会環境に完全ではないとしても殆どが戻っていくだろう。勿論、逆に今後、同じような状況が起こるであろうが、その時、時代は変わり、技術的な環境も変容していると考えられるため、ここで論じるのはあくまで2020年～2021年を基準とした技術的環境で可能なリモート環境におけるプログラミング実習講義がどのような問題点を持ち、どのような解決策が考えられたか、そして今後どのようにそれらが活用できるであろうかという実体験を交えた考察である。

実際、ここで改めて論じるまでもなく、実習を伴うプログラミング講義においてオンライン化が難しいのは火を見るより明らかであった。なぜなら、プログラミング（実習）を行うための道具であるコンピュータは必須であり、それを持たない学生がオンライン講義で受講することは物理的に不可能だからである。このような事態は、個人で所有するには不可能に近い実験機材が必要な理系学部でより顕著であろうし、文系でも音楽であればグランドピアノのような高価で大型な楽器を使用することを想定する講義ではプログラミング講義以上の障害があることは想像に難くない。それでも、なんらかの形でオンライン化の要請に応えなければならなかった。つまり、いずれにおいても不完全ながら講義を成立させる工夫が必要になる。

リモート環境でのプログラミング実習講義においては先述の通り、コンピュータを持たない学生は物理的に受講が不可能であることは自明であるが、それ以外に存在する問題点を次の項で論じる。

## 2. オンラインでのプログラミング実習講義における問題点

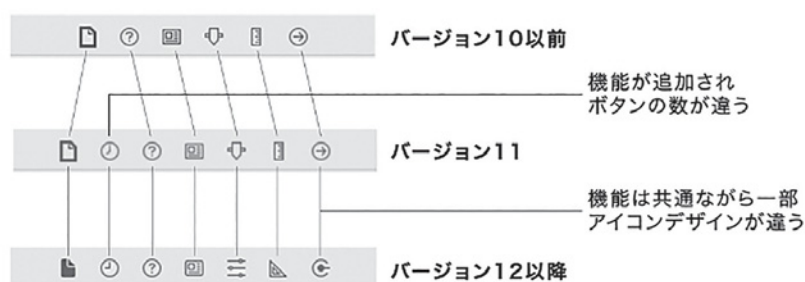
ここでは、主に実習を伴うプログラミング講義における問題点を述べるが、それ以外の一般的なオンライン講義の問題点に該当する内容も無関係ではないため含むこととす

プログラミング実習教育のオンライン化における工夫とその学習効果および今後の活用に関する一考察（有田）  
る。また、主にオンデマンドによるオンライン講義ではなく、リアルタイムオンライン講義を想定したものとする。

#### i. 各学生の準備環境の違いによる差異

実習が可能なコンピュータを学生が準備できていても、受講者数が増えるに従い OS や OS のバージョンによる開発環境アプリケーションが統一されていない可能性が高まる。開発環境アプリケーション自体もバージョンによって UI が変わっていることがあり、テキスト通りに講義をしても学生が混乱する可能性がある。図 1 は、筆者が講義で使用している iPhone アプリ開発環境の Xcode における Interface Builder 上の同一箇所の UI の変更の例を示したものである。

図 1 Xcode のバージョンによる UI の違い（一例）



バージョン12は2020年秋にリリースされたものだが、見ての通りの UI の変更がなされている。アイコンデザインが違っているのは、「こういう形のアイコンボタン」という伝え方では伝わり難い。また、機能追加によりボタンの数が違う変更もあって「いくつ目のボタン」とも伝え難い。実際はこの他にも UI は少なからず変更されているので、その箇所の操作が必要になる度、違いを意識した伝え方が必要になる。しかも、学生個々人の環境の違いはオンライン講義で目視が難しいため、混乱した学生からの自己申告があればまだいいが、教員が混乱を察知できない場合、学生が学習から取り残されることになる。さらに、図 1 に示したような環境違いに対応するために教員がそれら個別環境の差異を想定して解説しなくてはならず、環境の違いを熟知する必要があったり、その違いをフォローするための解説を必要として従来通りの講義の時間配分が難しくなるなど負担が重くなる。

ii. 学生へのフォローが難しい

学生からの質問に対して、その場で学生の状況を確認して指導できる対面講義と違い、オンラインでは状況が分かりにくい。そのため、学生側の負担が出る(後述)ものの画面共有をしてもらい問題点を確認することはできるのだが、その間、一人の学生のために講義が完全に中断してしまう。対面講義であればTSAに任せつつ講義を進行できるが、オンライン講義では「小声で」「個人的に」やりとりをすることができないため、オンラインミーティングサービスのZoomであればブレイクアウトルームを作成して、そちらで対応してもらうことになる。ところが、このブレイクアウトルームは、講義ミーティングからシャットアウトされるため、教員もTSAによるフォロー状況が掴めない。また、対面講義よりも講義中断に対して学生も質問に遠慮しがちになり、どこまで理解できているかを教員が把握しづらいため適切な指導が難しい。

iii. 学生が快適な受講環境を整えるハードルが高い

プログラミングに限らず、コンピュータを使用した実習講義では学生はコンピュータを用意するだけでなく、そのコンピュータとは別にオンライン講義の配信画面を見られるデバイス(タブレットなど)を用意しなければならないだろう。厳密にはコンピュータだけでも受講は可能だが、1つの画面内に実習に使用するアプリケーションウィンドウを開きながら配信ウィンドウも開くというのは実習作業スペースの確保や、配信画面の可視性から学習環境としても不利になる。プログラミング講義という観点からだと、ソースコードの一覧性から、できるだけ広い画面が確保できる方が望ましいので、その不利さはより顕著になる。また、こうして作業用コンピュータと配信画面用デバイスを分けた場合、学生側で実習作業に何らかの問題が起これば、解決のために画面共有をしてもらおうにも作業用コンピュータは講義ミーティングに参加していないので、ii.で述べた通り画面共有作業に対して学生に負担が出る。これを解決するにはかなり大きなモニターデュアルモニター環境で受講するなど費用的に負担が生じる。さらに、受講するにあたり配信が途切れたり遅延するのを防ぐため、FTTH回線(帯域幅に余裕を持つために、できれば1Gbps以上)で、仕様上、得てしてPing値の高くなる無線LAN環境や4Gなどの携帯電話ネットワークではなく、有線LAN環境を整える必要があるかも知れない(回線の環境について教員側の方は配信環境としてきっちり整備しておく必要がある)。

プログラミング実習教育のオンライン化における工夫とその学習効果および今後の活用に関する一考察（有田）

Zoom ではグループビデオ通話の場合の推奨帯域幅を以下の通りとしている。

*高品質ビデオの場合は 800kbps/1.0Mbps (上り / 下り)*

*ギャラリービュー / 720p HD ビデオ : 1.5Mbps/1.5Mbps (上り / 下り)*

*1080p の HD ビデオの受信の場合は 2.5Mbps (上り / 下り)*

*1080p の HD ビデオの送信の場合は 3.0Mbps (上り / 下り)*

帯域幅の要件および参加者の帯域制限について Zoom-Support より抜粋

プログラミング講義の場合、ソースコードの可視性から1080pのビデオ送受信が望ましいだろう。一般的なベストエフォート形式の回線では、同一の回線に收容される接続者が増えるほど帯域幅が減るため、安定した接続を維持するには、先述の通り、できれば1Gpbs以上の光回線が望ましい。過疎地などで光回線が通っていない地域から受講する場合、十分な帯域幅を確保できないことが考えられるが、2021年10月29日に「総務省は29日の有識者会議で、光回線などの高速インターネット網「ブロードバンド」について、過疎地でも整備・維持するよう通信事業者に義務付ける方針を示した。」と時事通信社が報道しており、今後、回線の問題は全国的に解決されていくと思われる。

#### iv. 本当に学生が受講しているか分からない

Zoomのミーティングでは参加者の一覧が出るが、これはあくまで参加しているだけで本当に受講しているかを示すものではない。ミーティングに参加しただけで、講義には参加せず遊んでいても教員側には分からないため、きちんと受講してもらうための工夫が必要となる。聴講していなければ完成できない課題を毎回出すのはそのための1つの手段である。しかしながら、特にリアルタイムオンライン講義で回線環境や、Zoomアプリケーションの不具合等によって、全編通して受講できなかった学生のために録画を残しておく必要はあり、この場合、真面目にリアルタイムで受講している学生と、回線環境等なら問題がないにも関わらず、講義中は遊んで放置し、後で録画を飛ばしながら見て課題だけ完成させる、という学生がいる可能性が生じ、真面目な学生との不公平が生じることが考えられる。対面講義でも座席に座っているだけで聴いていない可能性はあるが、少なくとも講義中に教室にいることを教員や学生自身が目視できる分、オンライン講義での不公平感は顕著になると考えられる。

v. 対面講義生との成績評価に不公平が生じる可能性

プログラミング実習講義では、学生に実習を経て成果物を作成してもらうことが基本である。実習自体は本来、講義時間内のみで行うがオンライン講義では実習の成果を講義時間内で把握することができないため、どうしても事後に成果物を提出させるしかない。オンライン講義受講生の間だけでも iv. で述べたような不公平が生じる可能性もあり、対面講義で聴講しながら、きちんと時間内に成果物を完成させるのと、講義後に完成させて評価を受けることもできるのは、対面講義でもオンライン講義でもシラバスは同じなのに実態が違う点で（些か精神論に傾くきらいはあるが）学生にとって真面目に取り組んだ自己とそうではない他者が同じ評価を得ていることを顧みて憤懣やるかたない思いを抱えることも考えられる。

これらの他、学生が自宅での受講に際し集中力が出ない（と推定される）などの問題もある。実際、提出物を見て本当に聴講していたのか疑問になるような事例もあった。提出物（プロジェクト）に含まれなければならないファイルを解説し、当初はそれなりに正しく提出していたのに突然、ファイルの足りない評価不能なものを提出してきて、それを指摘すると「今まで通りに提出しているのに今までも評価不能だったのか」と返ってくる始末で、当然、「今まで通りではないし、今までは評価できていたから指摘していない」と返答した。これは隔週で対面講義を行っていた時期で、集中力が出ないからか、本人の資質かは明確ではないが、2013年から指導してきた対面講義では見られない現象であったのは間違いない。そのため、オンライン講義を挟むことによって集中力が出なかったのだと捉えたい。

また、教員の負担増という問題もある。講義ミーティング自体の予約、Google Classroom での告知、課題提出場所の作成は勿論、オンライン講義用に改めて資料を作成しなければならない。近畿大学では学生の出欠管理について学生証を教室内でスキャンさせることで行っているが、それが不可能なため、教員側で Zoom の参加者一覧から集計するなどした出欠管理情報を大学側のシステムに登録することや、講義外の質問や受講に対する相談に応える必要が出てくるなど、対面講義ではおよそ必要のなかった負担が発生する。

### 3. 実際のオンライン講義での対応と学習効果

次に、筆者が実際にどのように出席の確認や評価を行ったのかを主題として論じたい。緊急事態宣言を受けて、唐突にオンライン化しなければならなくなった今回の事態では、準備期間が少なく、かつシラバスと乖離しない講義をする必要があったため、筆者は以下のような対策を取った。

#### i. Macを準備できる学生のみ履修可能とした

筆者の担当する講義はiPhoneアプリの開発方法をプログラミング言語Objective-Cと共に学習してもらう。iPhoneアプリの開発環境はMacのXcodeが標準なため、同環境を揃えられる学生のみとした。その際、必ずしも個人所有が必須ではなく家族や友人からの貸与が受けられる場合でも構わないこととした。また、mac OSのバージョンに依存するXcodeの導入方法を調査し、事前にGoogle Classroomに掲載した（後述）。

#### ii. 課題テキストの一部をPDF化して配布

テキストは3割程度、筆者が執筆したもので、かつ書籍として販売されているものではないため、実習で入力してもらうソースコードを再編、PDF化してGoogle Classroomで配布した。この際、出席の確認ができるように、あえて完全なものにはせず、講義を聴いていないと実習課題が完成しないようにした。

#### iii. 課題提出期限を設定して評価

Google Classroomの授業機能を使用して課題提出用のトピックを作成し、課題提出期限を講義終了後24時間とした。本来、対面講義では成果を目視できるのでこのような措置は必要ないが、実習に割り当てた時間内に完了できているかがオンラインでは把握できないので、他の講義の受講等の時間も考慮して猶予を取った。この機能では提出期間を過ぎても提出自体はできるが、提出した日時は分かるので、それによって評点を減点方式で行った。また、提出物が不完全な場合、個別に不完全であることを指摘し、修正されたものに対して評価した。ただし、学生の種々の事情を考慮して遅れて提出しても満点としたものもある。一例として、第1回講義が始まる前にMac購入の手続きをしたという学生が、まだMacが届かないため、課題を提出するのは待つて欲しいという訴えをして



きた。そもそも通常時でも1～2週間ほど納品にかかるものだが、パンデミックで生産に遅れが生じ、1ヶ月以上待たされる状態であった。どちらにせよ、急な講義形態の変更の他、履修登録の開始時期との兼ね合いもあって、調達が間に合わないのは当人の責任ではないし、わざわざ学生にとって決して安くはないMacを調達してまで受講したいという意志も尊重し、調達できた後に提出したそれまでの課題で完成しているものは満点とした。

iv. **デュアルモニタ環境で配信**

配信に使用するMacも私物なので本業用のデスクトップアイコンなど、見られでは困るようなものを事故で配信しないため講義はデュアルモニタ環境にして、デスクトップに一切アイコンを置いていないサブモニタ全体もしくは必要なウィンドウのみを共有しながら配信した。また、Zoomの共有画面には常にマウスカーソルが表示されるように設定して、学生に操作の導線が見えるように配慮した。

v. **Webカメラは共有せずに配信**

動画のストリームはフレーム(コマ)間の画の差異が大きいほど負荷が高くなる。一般的に実写は静止画よりも動く箇所が大きい(多い)ため、送出するストリームのデータ量も多くなり、またエンコードするPCもデータ量のピーク値を可能な限り平均化するために複雑な演算を必要として負荷を掛ける。そのため、画面はあくまで、Webカメラを使用せず、ほぼ静止画のデスクトップやXcodeなどのアプリケーションウィンドウを共有するに留め、配信の安定性のための負荷軽減を図った。

vi. **有線LAN環境で配信**

「2. オンラインでのプログラミング実習講義における問題点」で述べた通り、Ping値が高くなる無線LAN環境だと配信が途切れたり停止したりする可能性が高くなるため、ベストエフォートではあるが1GbpsのFTTH回線に有線LANでMacを接続して配信した。結果的に、Zoomアプリのフリーズで停止することはあったが、配信している内容が途切れるなどして聴き取れないといった事態は避けられた。

学習効果は定量化するのが難しく、筆者の講義ではMacを準備できるという学生にとってはそれなりに高いハードルがあるため、受講者も少なく、サンプル数としては参



プログラミング実習教育のオンライン化における工夫とその学習効果および今後の活用に関する一考察（有田）

考程度にしかならないが、特殊課題（便宜上こう呼称する）達成率から見ると概ね、問題はなかったように思われる。この特殊課題は通常の講義で行うソースコードを提示した実習課題ではなく、それまで講義で習得した内容を使用して自らプログラムを考えて制作するものである。2013年から2019年までの述べ受講者数271名中、課題を与えてから7週目までに完成した学生は86名であったのに対し、全てオンライン講義であった2020年は受講者数10名中、3名が完成と共に約3割でさほど差は無い。7週目までとしている根拠は、受講者の作業速度による講義の進行度次第で7回目以降の講義で課題を与えており、最遅は9回目の講義で、15回目の講義までで初週（9回目）を入れて7週となるからである。

ただし、途中で講義に来なくなる学生の割合が僅かに多い印象はある。特に、留学生は平常時、自国語を使いながら（おそらく、各々が日本語からの理解を補完していると思われる）同国出身者同士で助け合って真面目に取り組む印象だったが、オンライン講義で1人だけ受講した留学生は最初の4回でリタイアしていた。留学生が他に誰もおらず助け合いながら受講することも難しく、かつ直接指導が受けられないことから、より言葉の壁の問題もあると見られ、リタイアしたものと推測する。こうした言葉の壁によるものはオンライン講義で留学生を受け入れる上での課題であろう。途中放棄する学生については、推測の域に過ぎないので学習効果という点からは若干外れるかも知れないが、やはり対面講義よりも行き届かない故の不利はあるのだと思われる。また、「2. オンラインでのプログラミング実習講義における問題点」で述べた通り、集中力が削がれる可能性も否定できない。

学習効果について、オンライン講義が概ね問題にならない一つの傍証として、カリフォルニア大学ロサンゼルス校のディロン・H・マーフィー氏ほか4名が2021年11月14日に発表した論文(Learning in double time: The effect of lecture video speed on immediate and delayed comprehension)がある。対象がプログラミング実習講義ではないリアルタイムオンライン講義でもないが、この論文によると、231名の学生を被験者としてオンライン講義の動画を一時停止やメモを取ることを禁じ様々な再生速度で見せた結果、2倍速で見ても学習効果は下がらないことが分かったという。実験の様々な条件および結果の評価方法や論文自体の引用は割愛するが、こうした実験の一端を鑑みれば集中力の低下については（あったと仮定して）オンライン講義であるから、というよりは自宅であるという心理的あるいは物理的環境（家族からの雑音など）がもたらすのではないかと推察される。

#### 4. オンライン講義の問題点への対応策

ここでは、十分な準備時間が得られると仮定した上で、「2. オンラインでのプログラミング実習講義における問題点」で述べた問題点をできる限り払拭する対応策について一考したい。実のところ、先述した環境による集中力の低下、回線の環境やZoom等のオンラインミーティングアプリケーションのバグによるものなど物理的に対応できないものを除けば、クラウド環境を利用すれば大抵の問題には対応できる。プログラミング講義の中でも普遍的な、順次・分岐・反復を学ぶようなものでは、プログラミング言語自体は何でも構わないので、教員が任意の言語を選べる TechFUL (<https://techful-programming.com/>) を利用するといいたいだろう。Web ブラウザベースなのでタブレットでも稼働するため専用のコンピュータを用意させる必要がない。TechFULでは問題がすでに用意されているが、企業として登録していれば問題を独自で作成することができ、メンバー管理もできるので、これを利用すればクラスにのみ独自問題を出題できるし、完成した課題を提出させることも容易である。もしくは paiza.IO (<https://paiza.io/ja>) を使用する方法もある。paiza.IO は C,C++, Java, Ruby, Python, PHP, Perl など主要24言語に対応（同サイトより）しており、ブラウザ上でコードを書いて実行させることができるので、教員が問題だけ作成して実行結果をスクリーンショットさせるなどして回答を確認する。

これらは先述の通り、Web ブラウザベースであるので、プラットフォームに依存しないため、ある程度の画面サイズさえ確保できて比較的新しい Web ブラウザを搭載しているデバイスであれば利用可能であるし、Web ブラウザそのものに対するものを除いた個別環境に依存しない。Web ブラウザそのものに依存する例としては、古いバージョンの Web ブラウザや Internet Explorer のようにメーカーサポートが終了しているものなどを使用している、学生が個人的に Web ブラウザの設定をカスタマイズしていたり導入しているアドオンと競合して正常にサイトが稼働しないなどがあるが、「2. オンラインでのプログラミング実習講義における問題点」の「i. 各学生の準備環境の違いによる差異」で述べた問題点のほとんどがこれで解決はできる。

また、TechFUL を利用するメリットとして、1つの問題中でサンプルケースとテストケースを含めて作成でき、テストケースに対しての実行結果でスコア計算ができることがある。サンプルケースでは標準入力や期待出力（正しい実行結果）が公開されるが、

プログラミング実習教育のオンライン化における工夫とその学習効果および今後の活用に関する一考察（有田）

テストケースでは公開されないため、テストケースはサンプルケースの応用問題にできる。そして、テストケースの実行はブラウザ上で学生本人がするので、講義後に課題を改めて提出させる必要もないため、オンライン講義中、前半で命令等の構文とそれに関わるサンプルケースを解説し、後半にはテストケースを出題して講義時間内に提出できた学生が全編通して聴講（＝出席）していたと認定することができる可能性がある。「2. オンラインでのプログラミング実習講義における問題点」の「iv. 本当に学生が受講しているか分からない」に部分的に対応できる。勿論、出席したことそのものに対して評点を与えることはできないのが前提だが、スコア計算機能によって理解度に応じた評点を自動的に与えることができるため、学生間の不公平感はいくらか軽減されると考えられる。

逆にデメリットとしては、問題作成において、紙ベース（PDFを含む）のテキストを作成するよりも時間も手間もかかることが挙げられる。スコア計算機能は、標準入力に対して期待出力とどれだけ合致したかや、コード実行時間のタイムアウトまでに実行できたか（アルゴリズムが非効率でないか）といった基準で判定するので、単純な構文解説に留まらずに標準入力と期待出力を設定したより緻密な課題設計が必要になる。そのため、対面講義で培ってきたシラバスと乖離する可能性は高く、十分な準備時間を持ってシラバスの再編から行う必要があるかも知れない。あくまで実習環境として個別の差異問題をなくすだけが目的の場合には、paiza.IOでコードの実行環境だけクラウド化するといいたろう。これだけでもオンライン講義で把握するのが難しい学生の個別環境に配慮する必要がほぼなくなるメリットがある。

### 第3期教育振興基本計画【※計画期間：2018年度～2022年度】（平成30年6月15日閣議決定）

#### 第1部 我が国における今後の教育政策の方向性

#### IV. 今後の教育政策に関する基本的な方針

#### 5. 教育政策推進のための基盤を整備する

- 大学教育については、学生が主体的に学修するアクティブ・ラーニングへの展開を図るなど、教育の質向上の観点とともに、グローバルに進展している教育研究のオープン化に対応し、大学の知を広く国内外に発信する観点からもICTの利活用を推進することが求められる。

#### 第2部 今後5年間の教育政策の目標と施策群

#### 5. 教育政策推進のための基盤を整備する

#### 目標（17）ICT利活用のための基盤の整備

- 大学におけるICTを活用した教育の推進
  - ・ 高等教育段階において、教育の質向上や大学の知の国内外へ発信の観点から、多様なメディアを活用した遠隔教育やMOOCによる講義の発信等、ICTを活用した教育を推進する。
- ICTの活用による生涯を通じた学習の推進
  - ・ 放送大学におけるオンライン授業の充実など、ICTの活用による生涯を通じた学習機会の提供の推進を図る。

文部科学省「高等教育におけるICT活用教育について」1ページ目より

このクラウド化による対応は、以下に示すコロナ禍以前の平成30年9月に文部科学省が発行した資料である「高等教育におけるICT活用教育について」の中で大学教育において以下のような目標を掲げており、そうした国の施策と期せずして合致する基盤形成の一助となると考えられる。

ここまでWebアプリケーションでのクラウド化による対応策について述べたが、これはあくまでプログラミング言語の普遍的な構造を学ぶものについて有効な策であり、特定のプラットフォームに向けたアプリケーション開発のような閉じられた環境では、このままでは標準的な方法を学ぶには役に立たない。特定のプラットフォームに向けたアプリケーション開発については、筆者の担当するiPhoneアプリケーション開発の講義の他、Androidアプリケーション開発などが挙げられる。スマートフォン向けアプリケーション開発は厳密には国内のサービスであれば、Monaca (<https://ja.monaca.io/>)、海外のサービスなら Appery (<https://appery.io/>) などのクラウド開発環境を使えばアプリケーションの開発だけは可能であるが、使用言語的にも開発手順的にも変則手法であり、一企業内のルールとして標準化し使用する分には問題ないが、学習という点では標準手法を学ぶことが原則と考えられるため、仮にこれらを使用するとしても最終手段となると思われる。

では、閉じられた開発環境下ではどのような対応策が考えられるか、となるが、WindowsはWindows365、macOSは、Amazon EC2 Mac サービスを使用すればクラウドでOSそのものを利用することができるため標準の開発環境もクラウドで利用可能となる。予算的な問題はあるが、こういったサービスを教育機関側で導入、アカウントを学生に配布し、ソフトウェアインストールに対する一定のルールを設ければ、完全なクラウド化も可能である。ただし、“安定した環境下”で講義を行うことを考慮するなら、Microsoft自身のサービスであるWindows365はともかく、Amazon EC2 MacはAppleがいつライセンスの方針を変えてAWSでのmacOS利用を取り下げるかも不透明で、その点は注意しなければならない。そういう意味において“安定した環境下”で講義を行うならクラウドに頼らず「2. オンラインでのプログラミング実習講義における問題点」の「i. 各学生の準備環境の違いによる差異」で述べた内容について甘んじて受けるしかない。

この場合、「3. 実際のオンライン講義での対応と学習効果」の「i. Macを準備できる学生のみ履修可能とした」で述べたが、実習可能なコンピュータを用意してもらうのは勿論、実習に必要なアプリケーションのインストール方法を事前に調査、学生に案内して準備

プログラミング実習教育のオンライン化における工夫とその学習効果および今後の活用に関する一考察（有田）

してもらい必要がある。iPhone アプリケーション開発環境の Xcode に絞って述べると、同開発環境は Mac の App Store から入手可能で、AppleID を事前に所持してさえいれば、インストール自体にさほど難しいところはない。ただし、これは学生の Mac にインストールされている macOS のバージョンが最新か1つ前であることが条件で、最新の Xcode に対応していないバージョンの macOS である場合は、別の手段でインストールする必要があり、学生の負担が増えることになる。具体的には Apple Developer (<https://developer.apple.com/download/more/>) にログインして、自分の環境に合わせた古いバージョンの Xcode を探してインストールする。これらの情報を Google Classroom などで学生に知らせておいてインストールしてもらうこととなる。また、Xcode は 5GB 近い容量の大きなアプリケーションで、講義開始後にインストールを開始されると間に合わないのと、Mac 本体のストレージの空き容量に注意するといった導入に関する付随的なことも告知する必要があるだろう（筆者は実際に行った）。そして、学生が置かれている環境下を想定しながら、複数のバージョンの UI についての解説を交えて講義をするしかなく、こうした本来実習そのものの目的からは外れる予備知識を教員が持たなくてはならないのは避けられない。これは単純に教員・学生双方にとって負担ではあるが、教員側が得たノウハウそのものはオンライン講義以外でも利用できる局面がある。

## 5. BYOD と併用もしくは移行できる可能性

さて、「4. オンライン講義の問題点への対応策」で述べた内容は当然、対面講義でも利用することができる。オンラインのプログラミング実習講義では既に述べた通り、学生個人にコンピュータを用意してもらい必要がある。これは、各自がコンピュータを所有することに他ならず、教室に訪れないまでも BYOD (Bring Your Own Device) と酷似した状態となる。高等教育における BYOD の問題点は、講義内容が多岐にわたる中で個人によって環境がバラバラになることに他ならない。実習に必要なソフトウェアが導入できるかについても問題があるし、教員がソフトウェアのインストールについて指導しなければならない場合もある。マルチプラットフォームのソフトウェアであれば、OS を気にせず導入できるが、そうでない場合は、学生が所有するコンピュータによって受講できない講義が発生することになる。逆に教育機関で統一したコンピュータ環境を用意することは、学習環境に差異を持たず、どのような学生でも望む講義が(教育機関で用意した台数分の人数に限られるが)受けられるのが最大のメリットである。

令和3年の後期講義開始時点の9月は、筆者の勤務する近畿大学東大阪キャンパスでは大阪府の緊急事態宣言下の運用であり、全面的にオンライン講義で開始することとなった。また、いつの時点で緊急事態宣言が解除されるのか、あるいは再発令されるかも当時不透明だったこともあり、筆者の担当するプログラミング実習講義であるiPhoneアプリ開発入門では、標準のiPhoneアプリケーション開発環境のXcodeが稼働するMacを所有する学生に受講を限らざるを得なかった。周知の通り、10月には緊急事態宣言が全面的に解除され、以降、まん延防止等重点措置等も発令されなかったため、令和3年12月時点までは対面講義となった。開始約1ヶ月の間、オンライン講義であったため、対面講義への移行後も個人所有のMacを利用して受講希望する学生が多く、実質BYODが実現している。勿論、個人でMacを持ちながらも敢えて教室内のMacを使う学生もいる。しかしながら、そもそもWindows PCに比べてMacを所有する学生が少ない故に受講生も5～6名と少なく、これで実現していると言って差し支えないかは疑問である。ただ、すでに自宅で環境が整えられて操作に慣れた後であることもあろうが、実習中に環境別などの過度な個別対応は必要なく滞りなく講義は進んでいる。勿論、オンライン講義中は、ある程度の環境の幅を想定して、差異について解説しながら講義を行っていることもあるだろう。

教室に用意されたMacにインストールされているXcodeのバージョンは9で、「2. オンラインでのプログラミング実習講義における問題点」の「i. 各学生の準備環境の違いによる差異」で例示したように2021年10月時点の最新バージョン13とはUIも仕様も違っているし、バージョン13のリリースが2021年9月22日であったこともあり、学生はバージョン12を使用していたり13を使用していたりとばらつきがあるにも関わらず、特に混乱はない。これが数十名規模になった時、受講生が増えるほど、タイピングなどの個々人の能力の開きが大きくなり、実習に費やさなければならない時間も変動するため、TSAの手助け込みでどれほど影響があるかは不明だが、少なくとも初動で操作への慣れを促してしまえば、講義に支障が出るほどの個別対応は必要なくなることを意味している。ただし、初動がオンライン講義であったため、学生が自宅で講義時間外に自助努力で操作に習熟していった面はあると思われ、対面講義内だけで完結しようとする、初回から数回に渡り環境別の個別指導が必要になるかも知れない。

一方、クラウド環境を利用する場合には、環境別の個別指導は必要ないため、オンラインでもそのまま実行可能な講義あることを見据えたカリキュラムを用いることによって学生は教育機関側が用意した環境を利用するかBYODを選択するかの自由を享受でき



プログラミング実習教育のオンライン化における工夫とその学習効果および今後の活用に関する一考察（有田）  
るようになるであろう。

このようにオンライン講義で得たノウハウを活かすことで学生の所有プラットフォームや学習環境の差異が問題にならない程度まで矮小化できた場合、BYOD化は講義の進行に対する（少なくとも大きな）障害ではなくなる可能性があるが全面的に導入するには問題がある。

そもそも、BYOD化は学生の家計に負担をかけることになる。実際、主眼は初等中等教育であるものの文部科学省が教育においてICT環境を基盤とした先端技術・教育ビッグデータを活用することを目標に令和元年6月25日に取りまとめた資料には、以下のようにある。

*なお、BYODについては、家計に負担をかけることや使用頻度、必要な機器の保有状況等を考慮するとともに、学校段階の教育活動の実情も踏まえて検討することが必要である。*

文部科学省「新時代の学びを支える先端技術活用推進方策（最終まとめ）」29ページより

家計負担を考えなければ、Windows PCでなければならない講義、Macでなければならない講義と、実習に必要なコンピュータ（プラットフォーム）を都度、購入して貰えばいい。しかしながら、それは筆者担当講義の受講希望者数がいつも教室定員から2～3倍に上るにも関わらず、オンライン下では大きく定員割れしていることから分かるように現実的ではなく、先述の文部科学省の資料にある通り、使用頻度を考慮して教育機関が適切なコンピュータを提示する必要があるだろう。もしくは卒業までに履修可能な実習講義全てに、どのプラットフォームを使用する講義であるかを明記しておき、自ら受講希望する講義での使用頻度を考慮したコンピュータを学生に選択してもらうことも考えられる。この場合、教員だけでなく履修相談にあたる職員もまたコンピュータ、とりわけプラットフォームについての知識が必要になるであろう。現状ではMacでなければできない講義というのはかなり特殊な部類と言え、大抵はWindows PCで事足りるだろう。それでも、UNIX系専用のソフトウェアが必要な場合、macOSやLinuxなら問題なく稼働させられるが、WindowsではLinuxとのデュアルブート構成にしなければならないだろう。デュアルブート化は構成に手間がかかるため、VMwareのような仮想環境を利用することで簡単に複数のOSを導入するのも選択肢となるかも知れない。ただし、どちらにせよ、通常のコンピュータ利用では必要としない知識を学生自身が持つ



が必要になる。

このように BYOD に完全に片寄ることは学生側には学習環境の差異の問題、従来より高いコンピュータリテラシーを必要とする、家計負担が増える、教育機関側には統一環境下では必要ない知識や指導が必要になるなど全体的な負担が増える問題がある。一方、学生側にとってはコンピュータが購入できれば講義だけではなく卒業後に至るまで別の用途でも自由に使用できる可能性があるし、教育機関側としてはコンピュータ設備の定期的なリプレースや大規模なメンテナンスあるいは保守が必要なくなるメリットもある。また、BYOD では必然的に小型のノート PC やタブレットを使用するため、設備として従来のように（盗難防止の観点も含めて）大型のデスクトップ PC を必要とせず、省スペースで机を有効に活用しながら講義が受けられるメリットもある。

そして、オンライン講義実施で培った経験やノウハウは、これまで述べたようにそのまま BYOD に適用できる部分もあり、家計負担面などのデメリットも多々挙げられるが、プラットフォームもクラウドサービスで稼働させられる現在は、より BYOD の問題点を払拭しやすい状態となっていると言えるだろう。

BYOD に全面的に移行する必要があるなら、こうしたメリットとデメリットを考慮し、プログラミング実習をオンライン講義でもほぼ遜色ない講義内容へ移行させる過程で、まずは BYOD 化か否かの二元論ではなく設備投資に対するコスト面の折り合いをつけ、学生が教育機関の用意したコンピュータ設備を使用するか、BYOD で個人のコンピュータを使用するかを選べる併用状態とし、のちに全面移行する道筋を見つけることもできるのではないか。

## 6. 今後の展望と考察

教育に関して中短期的な視点で考えるのは難点がある。「1. はじめに」で述べた通り、本稿はあくまで2020年～2021年を基準とした技術的環境においての考察であるが、さらに未来を見据えたら、メタバースの進歩で仮想空間でも対面講義と同等の講義（個別に小声で指導ができるような）ができるようになるかも知れない。勿論、誰でも簡単に安価に仮想空間を利用できるようになっているという前提だが、その時代には、本稿で述べた問題点は限りなく払拭されていると言える。

今回の COVID-19 パンデミックのような事態はいつでも起こりえるかも知れないが、世界人口が18億人であった時代に4,000万人を超える死者を出したスペインかぜと呼ばれ

プログラミング実習教育のオンライン化における工夫とその学習効果および今後の活用に関する一考察（有田）

る新型インフルエンザ・パンデミックから約100年、1957年のアジアかぜ（死者200万人以上）、1968年の香港かぜ（死者100万人以上）、2009年の A/H1N1（死者18,449人）と、数度のインフルエンザ・パンデミックが起きているが、医学の進歩と共に被害も少なくなっている。COVID-19 パンデミックについても次々と変異株の登場が取り沙汰されるが、経年で2020～2021年現在のような世界規模の渡航制限が必要になることはインフルエンザ同様になくなっていこうと考えるのが自然である。そのため、いつでもオンライン講義に移行できるようにするべき、とするのは些か極端ではあるが、既述の通り対面講義を現状でそのままオンライン講義に落とし込むことが難しいのは明らかなので、オンライン講義でも可能なカリキュラムを見据えた改革をすることは、万が一の事態のために加えて、文部科学省が推奨するオンライン化やBYODの導入のためにも無駄にはならない。

対面講義であっても、OSのクラウド化やTechFulのようなWebサービスを用いることで、学習環境をすべてオンライン化してしまえば、Webブラウザが稼働するデバイスであれば、どのようなものを使用しても受講できるメリットがある。もちろん、物理キーボードを持たないタブレットであれば、画面をソフトウェアキーボードで覆われるなどして、不便を感じることもあろうが、BYODの観点からは機種を限定する必要はなくなるので、学生の家計に沿って購入するデバイスを選ぶことができる。OSのクラウド化は既述の通り、ベンダーの都合により安定的に提供されるかが鍵となるが、Webアプリケーションレベルのサービスについては、1社開発のOSと違って必要とされる限り代替のサービスや競合サービスを自由に各企業が提供できるため、それほど問題にはならないであろう。この場合、講義自体は対面で行うことを想定しているわけだが、当然、学習環境のオンライン化は教材や方法論が違うだけなので、従来の対面講義から質が低下するわけでもないし、仮に今後、オンライン講義を余儀なくされたとしても移行しやすい、かつ平常からクラウド環境を積極的に利用することでノウハウを活用できるメリットもある。

予算的な問題は、時勢や当事者間での交渉次第で状況が変わるため、あえて考慮しないところではあるが、今回のオンライン化で見たクラウド教材を積極的に活用するように移行していきながらBYODを併用することで、より学生にとっても利便性の高い学習環境を構築できるのではないだろうか。今回、オンライン講義で問題が発生した講義形態は本質的に通信教育ではないはずなので、教材がクラウド化されようと、実際に講義に出席していることが評価（加点ではなく出席が当然）の前提ではあるが、クラウド

化によりBYODを推奨することもできるし、学生としてはBYODとは言わずとも自宅のデスクトップPCなりを使用して、講義内で間に合わなかったプログラミング実習も行えるのは利便性が高くなる。そして、実習の完遂に対しても課題提出時期によって点数を設定して集計を自動化することで、よりきめ細かい評価を下すことができるだろう。

ICTを活用した教育の改革は、既に提示した文部科学省の資料からも、我が国が目指すべき目標と定めているのは明らかで、改革の速度はともかく、流れは変わらないだろう。今回のパンデミックによって、オンライン講義が初等教育から高等教育に至るまで必要とされ（付け焼き刃であったにせよ）実施されたことで、大きく前進する基礎ができたことは間違いない。

## 7. お わ り に

突発的に発生したパンデミックにより、急激に社会環境が一変し、プログラミング教育でも手探りで対応を余儀なくされた。今回のオンライン化への対応で、生活に関しては働き方を初めとして様々な事柄が見直された。今後、徐々に従前の暮らしに戻っていくとは思われるが、すでにオフィスを縮小し、コロナ禍を抜けた後もリモートワークを積極的に継続しようとする企業もある。読売新聞オンラインの2021年6月25日付の記事には以下のようにある。

ヤフーが東京都内の本社などのオフィス面積を11月までに約4割縮小する方針であることが25日、わかった。コロナ禍で在宅勤務が定着し、出社率は約1割となっている。コロナ収束後もテレワークを積極活用するため、現在のオフィス面積を維持する必要はないと判断した。

読売新聞オンライン ヤフー、都内オフィス4割縮小へ…テレワーク定着で出社率は1割より

このような流れが主流とまでは言えないが、それでも完全に従前の社会に戻るわけではないこととなる。本論でも述べた通り、ICTを活用した教育改革は国の目指す目標でもある上に、企業のこうした対応を考慮し将来を見据えた上で、クラウド教材の積極的な活用を取り入れたプログラミング実習講義への移行も必要かも知れない。実際に、プログラミング実習講義の単純なオンライン化は本論で述べた通り、様々な問題があり、

プログラミング実習教育のオンライン化における工夫とその学習効果および今後の活用に関する一考察（有田）  
その問題点を極力矮小化し、かつ対面講義でもそのまま用いることができるようにするには現状ではクラウド化による対応がベストではないか。

この度のパンデミックによるオンライン化対応で得た知見を、BYODを含めた今後のプログラミング実習教育の方法論の改革に役立て、実際に活用していくことが我が国のIT人材教育をより社会に寄りそったものにしていくと考えられる。

## 参 考 文 献

- 帯域幅の要件および参加者の帯域制限について Zoom-Support <https://zoom-support.nissho-ele.co.jp/hc/ja/articles/360004830512-%E5%B8%AF%E5%9F%9F%E5%B9%85%E3%81%AE%E8%A6%81%E4%BB%B6%E3%81%8A%E3%82%88%E3%81%B3%E5%8F%82%E5%8A%A0%E8%80%85%E3%81%AE%E5%B8%AF%E5%9F%9F%E5%88%B6%E9%99%90%E3%81%AB%E3%81%A4%E3%81%84%E3%81%A6> (accessed 16 Nov 2021)
- 光回線、過疎地に提供義務 交付金で支援—総務省：時事ドットコム(令和3年10月29日) <https://www.jiji.com/jc/article?k=2021102901160&g=eco> (accessed 16 Nov. 2021)
- 文部科学省「高等教育におけるICT活用教育について」(平成30年9月) [https://www.mext.go.jp/b\\_menu/shingi/chukyo/chukyo4/043/siryu/\\_icsFiles/afieldfile/2018/09/10/1409011\\_5.pdf](https://www.mext.go.jp/b_menu/shingi/chukyo/chukyo4/043/siryu/_icsFiles/afieldfile/2018/09/10/1409011_5.pdf) (accessed 18 Nov. 2021)
- 文部科学省「新時代の学びを支える先端技術活用推進方策（最終まとめ）」(令和元年6月25日) [https://www.mext.go.jp/b\\_menu/shingi/chukyo/chukyo3/083/siryu/\\_icsFiles/afieldfile/2019/07/08/1418542-8.1.pdf](https://www.mext.go.jp/b_menu/shingi/chukyo/chukyo3/083/siryu/_icsFiles/afieldfile/2019/07/08/1418542-8.1.pdf) (accessed 3 Dec. 2021)
- 大幸薬品 人類を脅かす感染症のパンデミック(世界的大流行) <https://www.seirogan.co.jp/fun/infection-control/infection/pandemic.html> (accessed 13 Dec. 2021)
- Apple, 統合開発環境「Xcode13」をリリース：CodeZine（コードジン）(令和3年9月22日) <https://codezine.jp/article/detail/14909> (accessed 28 Dec. 2021)
- ヤフー、都内オフィス4割縮小へ…テレワーク定着で出社率は1割：経済：ニュース：読売新聞オンライン(令和3年6月25日) <https://www.yomiuri.co.jp/economy/20210825-OYT1T50266/> (accessed 28 Dec. 2021)
- Dillon H. Murphy, Kara M. Hoover, Karina Agadzhanian, Jesse C. Kuehn, Alan D. Castel 著 [2021年11月14日] Learning in double time: The effect of lecture video speed on immediate and delayed comprehension, <Wiley Online Library> <https://onlinelibrary.wiley.com/doi/10.1002/acp.3899> (accessed 5 Jan. 2022)