

# コロナ禍の計算機演習： 数学科における数式処理を用いた計算機実習の 遠隔授業による取り組み

照井 章\*

筑波大学 数理物質系

(Received June 19, 2021    Revised December 9, 2021    Accepted January 24, 2022)

## 概 要

Due to the worldwide spread of a new coronavirus disease (COVID-19), many university courses have shifted to distance learning, including courses with computer exercises. In this paper, we present a case study of our efforts to maintain the same level of teaching effectiveness when we conducted the transition from an in-person course with computer exercises to an online course in the college of mathematics.

## 1 はじめに

2019年に発生が報告 [18] され、2020年初めから世界に広がった新型コロナウイルス感染症 (COVID-19) は、教育面でも世界的に大きな影響を及ぼした。日本国内の多くの大学では、2020年度の新学期から、開講される授業の多くが遠隔授業<sup>1)</sup>に移行し、その一部は本稿執筆時の2021年前半も継続している。遠隔授業への移行により、それまで面接授業の中で行われていたさまざまな授業方法やコミュニケーションの手段、形態が再構築されている。

面接授業から遠隔授業への移行が授業方法に大きく影響する授業の一つに、計算機を用いた実習が挙げられるだろう。大学における計算機を用いた実習の形態は、従来より行われてきた端末室での一斉授業 ([11], [43]) の形式に加え、近年では、反転授業の導入により、履修者はあらかじめインターネット等の手段で提供される教材で授業内容を学習し、面接授業では個別指導を中心に行うことで、学生の習熟度の差に応じた教育効果を挙げるといった試みも行われている ([16], [29])。いずれの場合にも、コロナ禍より前は、通信教育を前提とするような場合を除き、一斉授業もしくは個別指導は面接授業の形態で行われることが (参考文献中の事例も含め) 多かったも

\*terui@math.tsukuba.ac.jp

<sup>1)</sup>本稿における「面接授業」、「遠隔授業」等の用語は、文部科学省が定める大学設置基準、大学通信教育設置基準、平成13年文部科学省告示第51号(いわゆる「メディア授業告示」)等における用例に従う。

のと思われるが、コロナ禍に伴い、面接授業の形態で行われていた各種指導を遠隔授業（もしくは面接授業と遠隔授業のハイブリッド型）に移行し、それまでの面接授業と同等の学習効果を挙げるための取り組みが進められている ([9], [20], [44]) .

筆者は、筑波大学（以下「本学」とも称する）理工学群数学類が開設する授業科目「計算機演習」（以下「本授業科目」とも称する）を担当している．本授業科目は、数式処理を用いた計算機実習を含む授業科目であり、学内の端末室で面接授業の形式で実施していたが、コロナ禍に伴い、遠隔授業に移行した．本稿では、情報学を専門としない理学系学部の学生を対象にした計算機の実習を伴う授業、特に、数式処理や  $\text{T}_\text{E}_\text{X}$  といった、数式を計算機上で直接的に取り扱う授業において、面接授業から遠隔授業への移行および遠隔授業の実施の際、それまでの面接授業における学習効果を維持することを目的に行った対応を報告する．この目的を達成する上での主な課題は以下の通りであった．

1. 数式処理や  $\text{T}_\text{E}_\text{X}$  といった、数式を計算機上で直接的に取り扱う授業を、面接授業から遠隔授業に円滑に移行すること．
2. それまでデスクトップアプリケーションを用いていた面接授業から、ウェブアプリケーションを用いた遠隔授業への切り替え（教材の書き換えや、学生のレポート回収、採点、返却に至るワークフローの切り替え）を短期間で実現させること．
3. 計算機の操作方法やプログラミングを含めた指導（個別指導を含む）に関し、遠隔授業においても、面接授業と同等の指導を実現させること．

以下では次の内容を述べる．第 2 節では、本稿の対象である授業科目「計算機演習」の概要を述べる．第 3 節では、遠隔授業への移行と準備の経過について述べる．第 4 節では、遠隔授業の授業環境について述べる．第 5 節では、遠隔授業に移行した際の授業内容について述べる．第 6 節では、遠隔授業の実施手順や取り組み内容について述べる．第 7 節では、遠隔授業の実施結果や授業に用いたウェブアプリケーションのトラブルへの対応などについて述べる．

## 2 授業科目「計算機演習」の概要

「計算機演習」は、筑波大学理工学群数学類開設の授業科目である．本授業科目の制度上の概要は以下の通りである．

授業方法 演習．

授業時間数 半期、週 1 時限（75 分間） $\times$  15 週 = 全 15 回（初回ガイダンスを含む）．

単位数 1.5 単位<sup>2)</sup>．

標準履修年次 2 年次．

授業概要 計算機による数値/数式計算の技術や、数学における計算機の利用方法の習得を目的とし、数式処理システムやプログラミング言語を用いた演習を行う．

<sup>2)</sup>本授業科目は、本学が定める授業科目の区分のうち、数学類において卒業に必要な履修科目の「専門科目」、および数学類以外のいくつかの学類において卒業に必要な履修科目の「専門基礎科目」に位置づけられているが、本学学群学則 [41, 第 30 条および別表第 2] では、専門科目および専門基礎科目の「演習」科目の 1 単位あたりの授業時間は 15 時間と定められている．本学では、1 時限（75 分間）を 1.5 時間として単位を計算しており [42]、専門科目および専門基礎科目の「演習」科目の単位数は、毎週 1 時限の授業を実施する場合、授業期間が 10 週間で 1 単位、15 週間で 1.5 単位と定められている．

教職科目 数学類では、中学校教諭一種免許状（数学）、高等学校教諭一種免許状（数学）、同（情報）を取得可能である。本授業科目は、数学類において高等学校教諭一種免許状（情報）を取得するために教職課程として履修すべき科目のうち、「教科及び教科の指導法に関する科目」区分「教科に関する専門的事項」の「コンピュータ及び情報処理（実習を含む）」事項の必修科目に定められている [42]。

その他の本授業科目の特徴は以下の通りである。

履修者の内訳 数学類生のはほぼ全員<sup>3)</sup>が2年次に履修する。数学類以外の学生も履修することがある。（数学類以外でよく見かける履修者の所属は、理工学群物理学類、化学類、および生命環境学群地球学類<sup>4)</sup>で、それ以外では情報学群の各学類の学生が履修することがある。）

数学類の計算機関連の授業科目における位置付け 数学類開設授業科目の中で、計算機に関連する授業科目は、本授業科目の他に、標準履修年次が3年次の「計算機数学Ⅰ」（春学期開設）および「計算機数学Ⅱ」（秋学期開設）がある。「計算機数学Ⅰ」は筆者が担当し、拡張 Euclid 互除法等、計算機代数の入門に当たる内容を講義する一方、「計算機数学Ⅱ」は数値解析が専門の教員が担当し、数値計算の入門に当たる内容を講義している。本授業科目では、3年次の両授業科目に接続する入門的な位置付けとして授業を行っている（授業内容に関しては第 2.1 項を参照）。

数学類における教員養成を背景とした位置付け 数学類には中学校または高等学校の教員免許取得を目指す学生も一定数存在する<sup>5)</sup>一方で、本授業科目履修前、あるいは履修当初の数学類の学生からは「パソコンは苦手」という声をしばしば聞く。本授業科目は教員免許状（数学）取得のための授業科目には位置付けられておらず、中学校・高等学校の数学の授業で直ちに活用可能な ICT を扱っているとは限らないが、数学類生の多数が履修すること、新学習指導要領（中学校学習指導要領（平成 29 年 3 月告示）[26]、高等学校学習指導要領（平成 30 年 3 月告示）[25]）により、今後、中学校・高等学校の数学の授業においても ICT のより一層の活用の推進 [24] が予想されることから、本授業科目では、教員への採用等、学生が卒業した後の ICT の活用も踏まえ、学生が数学における計算機の利用や、数学におけるアルゴリズム的思考に関する素養を養うことも目標の一つにしている。

以下、本節では、例年の面接授業を前提とした「計算機演習」の概要を、授業内容、授業環境、授業方法に分けて述べる。

<sup>3)</sup> 数学類の定員は 40 名。

<sup>4)</sup> 数学、物理、化学、地球の各学類は、2007 年度に学群・学類が現在の形に改組・再編されるまでは、第一学群自然科学類という一つの学類をなしていた。自然科学類には数学、物理、化学、地球科学の各主専攻があったが、これらの各主専攻が現在の各学類の母体となっている。

<sup>5)</sup> 令和 2 年度に本学が教職免許を一括申請した内訳 [40] によると、数学類生の申請者数（延数）は、中学校教諭一種免許状（数学）が 10 名、高等学校教諭一種免許状（数学）が 20 名（数学類定員の半数: 注 3) 参照）、高等学校教諭一種免許状（情報）が 1 名であり、本学の理工系学群（生命環境学群、理工学群、情報学群）の学類の中では最も多い。

## 2.1 授業内容

授業内容は、主に以下の 4 つのパート（合計の授業回数は、初回ガイダンスを含め全 15 回）に分かれる。各パートで学ぶ内容と、面接授業の際に用いているソフトウェアを順に述べる。

**数式処理** 全 6 回。数式処理システム Wolfram Mathematica<sup>6)</sup> を用いる。Mathematica の入門的内容（数の四則演算，代数方程式の求解，グラフ描画，アニメーション），Mathematica による線形代数（ベクトルや行列の演算，行列式，固有値・固有ベクトル，行列の対角化の計算），微積分（極限值，導関数，Taylor 展開，不定積分，定積分の計算）の計算手法，プログラミング（再帰的定義，手続き型の構文）を学ぶ。

**Git, GitHub の利用および Markdown による文書作成** 全 1 回。下記の  $\text{\LaTeX}$  およびプログラミングの授業における教材やレポートのやり取りのため（第 2.3 項も参照），ソースコード管理ソフトウェア Git およびソースコードホスティングサービス GitHub [6] の利用方法と，Markdown による文書作成を学ぶ。

**$\text{\LaTeX}$  による文書作成** 全 3 回。数式を含む文書，定理番号などの引用，参考文献の引用，Beamer パッケージによるスライド作成の方法を学ぶ。

**プログラミングの初歩** 全 4 回。プログラミング言語 Scala を用いる。“Hello, world” プログラムから始まり，数値演算，関数定義，再帰呼び出し，ループ（while 文，for 文），タプル，リスト操作を学ぶ。プログラミングの題材には，連立 1 次方程式の解法や数値積分といった数値計算の要素，および，有理数や 1 変数多項式をリストで表現した四則演算といった計算機代数の要素を取り入れている。

## 2.2 授業環境

授業は，本学学術情報メディアセンターが運用する全学計算機システムのサテライト端末室（以下「サテライト端末室」と称する）でデスクトップアプリケーションを用いて実施する。本節では，サテライト端末室での端末の利用や，デスクトップアプリケーションの利用環境（具体的には履修者の Mathematica の利用環境）に関する本学の背景を述べる。

### 2.2.1 “Bring Your Own Device” (BYOD) の導入状況およびサテライト端末室の授業での利用状況

近年，各大学において「教育環境の改善やコスト削減のために，教育機関が所有または指定するモバイル端末を使い，または学生が所有するモバイル端末を使って，教育学習に使用<sup>7)</sup>」する“Bring Your Own Device (BYOD)”を導入する動きが広がっている [3]。その一方で，文部科学省の調査研究 [13] において「情報学以外の分野を専門とする学部・学科等のうち，専門教育の一部に情報教育を含む学科・課程・コース等の学士教育プログラム」に対して行われた調査のうち「教育用電子計算機システムの導入状況に関する調査」では，回答があった 998 件の大学・学部・学科・コース等の組織のうち「学生によるノートパソコン等の購入・所持は任意」としている組織が全体の 80.6% を占め，「PC の価格は低下傾向にあるが，それでも購入/所持の義務化の

<sup>6)</sup>本稿執筆時点で，もし面接授業があれば用いているバージョンは Mathematica 12.1.

<sup>7)</sup>大学 ICT 推進協議会 (AXIES) ICT 利活用調査部会が BYOD を活用した教育改善に関する調査研究 [3] を行った際に用いられた BYOD の定義。

実現はハードルが高いことがうかがわれる」ことも指摘されている [13, 表 2.3.5.1-2] .

本学においても、学術情報メディアセンターや各学群棟を中心に、学内無線 LAN や電源等の利用場所を整備し、学生がパソコンをはじめとする情報機器を持参して利用する環境が整備されているが、コロナ禍より前は、少なくとも全学、および筆者が教育に携わる数学類の範囲で、ノートパソコンを必携化するような制度は実施されていなかった<sup>8)</sup>。(2020 年度以降は、本学においても、遠隔授業に対応すべく、大学が学生に対し「個人所持のパソコン(ノート型、タブレット型)」および通信環境の整備を推奨している [39] .)

また、本学では、コロナ禍より前は、少なくともプログラミングに関連し、パソコンによる演習や実習を含む授業科目のほとんどにおいて、パソコンによる演習や実習はサテライト端末室を利用する形態が中心だったようである。コロナ禍になる前の 2019 年度の学群・学類の開設授業科目において、科目名、科目概要またはシラバスの文面に「プログラミング」という語句を含む授業科目を検索<sup>9)</sup>したところ、シラバスの内容から計算機による演習や実習を含むと思われる(本授業科目を含む)約 150 の授業科目のほとんどは、サテライト端末室が授業(演習もしくは実習)の実施場所とされていた(この場合に使用する機器が明示されている授業科目はほとんどなかったが、サテライト端末室の端末を使用すると判断するのが妥当であろう)。

以上の背景から、コロナ禍より前の本授業科目では、サテライト端末室のデスクトップアプリケーションを用いて授業を実施することを前提としていた。

## 2.2.2 履修者の Mathematica の利用環境

2020 年度初めの時点で、本学は Mathematica の「教育機関 Unlimited プログラム」を契約していた。サテライト端末室へのインストール状況については、毎学期、学術情報メディアセンターにより「全学計算機システムへのソフトウェア導入希望調査」が行われているが、筆者も、毎学期、Mathematica をインストール/アップデートするよう申請を出しており、すべてのサテライト端末室の端末に Mathematica がインストールされていた。

学術情報メディアセンターが公表していた契約内容によると、対象の教育組織に所属する学生に対しては、Student Site License Program により、Mathematica for Students が無償で提供されていた。「対象の教育組織」は、これまでの本授業科目の履修者が所属する学類のほとんどを含む。

本授業科目を面接授業で行っていた期間中は、上記の Student Site License Program の存在(対象の教育組織に所属する履修者が各自のパソコンに Mathematica for Students をインストールして使用可能であることを)を周知しており、実際に授業時間に自分のノートパソコンをサテライト端末室に持ち込んだ上で Mathematica for Students をインストールして使用していた学生は、筆者が見る限り、例年数名程度存在した。

<sup>8)</sup>本学あるいは数学類における学生のノートパソコンの所持状況に関する資料は、筆者が探した限り見つからなかったが、筆者の経験では、数学類の「卒業研究」において筆者が担当した学生(例年、4 名程度を担当)に対し、セミナーにノートパソコンを持参させて作業を行わせた際には、2018-2019 年度においても、学生のうち 1, 2 名はノートパソコンを所持していないか、所持してはいたが形式が古くて必要とされる能力が得られず、筆者が管理する大学の備品のノートパソコンを貸し出していた。

<sup>9)</sup>本学の学群・学類および大学院の開設授業科目のシラバスのうち、学外に公開されている情報は「教育課程編成支援システム(KdB) [38]」により、学内外からオンラインで内容を検索可能である。

### 2.2.3 本授業科目で用いるデスクトップ/ウェブアプリケーション

本授業科目で扱うソフトウェアを動かす計算機環境は以下の通りである。サテライト端末室の端末は Windows<sup>10)</sup> と Linux<sup>11)</sup> のデュアルブート環境であるが、Mathematica の演習は Windows で行う一方、 $\LaTeX$  とプログラミング言語の演習は、シェルによるコマンドライン環境も経験させることを目的に、Linux で行っている。

授業の教材配布やレポート回収を行うために用いるウェブアプリケーションとして、数式処理のパートにおいては、本学が契約する Learning Management System (LMS) の manaba [2] で教材の配布やレポートの回収を行う。一方で、 $\LaTeX$  とプログラミング言語のパートにおいては、ソースコード管理やソースコードホスティングサービスの利用を経験させる目的から、履修者のソースコードは各自が Git リポジトリで管理し、ソースコードホスティングサービスの GitHub [6]、および GitHub Classroom [7] を用いている。

### 2.3 授業方法

本授業科目では、各授業回の授業日より前にあらかじめ授業テキストを配布し、レポート課題を開示している。授業時間のほとんどの時間を、履修者が自習で授業内容の実習やレポート課題に取り組む時間に充て、担当教員および大学院生のティーチング・アシスタント (TA) は、履修者に対して個別に質疑応答や指導を行う形態をとっている。この授業方式は反転授業に近い方式とも言えるが、通常の反転授業では授業時間前の予習には動画による講義などが用いられるのに対し、本授業科目では、授業テキストの閲覧による自習の形態をとっている点が異なる (本授業科目と類似の実施形式の他の事例 [14] では、この授業形式を「反転的な授業」と称している)。授業時間の使い方の例外として、Mathematica や Git, GitHub などのデスクトップ/ウェブアプリケーションを最初に使う際には、サテライト端末室のスクリーンに操作方法を提示し、履修者に一斉に各ステップの操作をさせる。各ステップの操作の際には、サテライト端末室内に配置された TA が、端末の操作に関する履修者の質問に答え、必要に応じて操作補助を行っている。

履修者は、毎週の授業時間になると、あらかじめ本授業科目に割り当てられたサテライト端末室に集まり、室内に設置された端末を操作して、各自のペースで実習やレポート課題に取り組む。授業時間内の演習やレポートの解答における履修者からの質問に対応するため、TA を 10 名程度配置している。授業成績は毎回の授業に対し、レポートで評価する。

レポートの提出・回収手順は以下の通りである。数式処理のパートにおいては、履修者が Mathematica のノートブックファイルを manaba のレポート課題のページにてアップロードする形式で提出を受け付けている。Git,  $\LaTeX$ , プログラミング言語のパートにおいては、履修者がレポート用に作成した GitHub のリポジトリに成果物を push (アップロード) することで提出を受け付けている。成績の集計は、授業のすべてのパートにわたり、manaba の成績集計の機能を用いている。

<sup>10)</sup>本稿執筆時点での OS は Microsoft Windows 10 Enterprise.

<sup>11)</sup>本稿執筆時点でのディストリビューションは Ubuntu 16.04 LTS.

### 3 遠隔授業への移行と準備

コロナ禍に伴う本学の授業や本授業科目の遠隔授業への移行の状況について述べる前に、コロナ禍に対応して設けられた、国内の大学等における遠隔授業に関する制度の特例について触れておく。

大学設置基準等の規定では、「卒業の要件として修得すべき単位のうち、遠隔授業の方法により修得する単位数については、大学院及び通信制の大学を除き、60 単位（修業年限が 2 年の短期大学にあっては 30 単位、修業年限 3 年の短期大学にあっては 46 単位、高等専門学校にあっては 30 単位）を超えない」とされている。

2020 年 7 月の文部科学省からの通知 [27] では「今回（筆者註：コロナ禍）の特例的な措置として、面接授業に相当する教育効果を有すると大学等が認めるものについては、面接授業に限らず、自宅における遠隔授業や、授業中に課すものに相当する課題研究等（以下「遠隔授業等」という。）を行うなど、弾力的な運用を行うことも認められる」（留意事項つき）とされ、それらの授業の単位数は、前記の「遠隔授業の方法により修得する単位数」（上限 60 単位）に算入する必要はないものとされた。さらに、2021 年 4 月の文部科学省からの通知 [28] では、上記の特例的な措置を、今後、他の感染症や災害等の非常時においても同様に運用することが認められるものとされた<sup>12)</sup>。

#### 3.1 遠隔授業への移行の経緯

本学では 2020 年 3 月 26 日付で 2020 年度の学年暦が改正され、新年度の授業開始が例年の授業開始時期から約 2 週間繰り下げられて 4 月 27 日からとなったが<sup>13)</sup>、当初は遠隔授業への言及はなかった。ところが、4 月に入り、全学で授業形式を原則として遠隔授業に移行することになった。計算機演習は、例年、春学期に実施されていたが、端末を用いた面接授業を前提としており、即座に遠隔授業に移行することは困難であると思われたため、学類の配慮により、ひとまず授業実施時期を秋学期に延期してもらった。

その後、新型コロナウイルスの感染状況や、それに対応する大学の体制の推移を見守っていたが、6 月頃になった時点で、秋学期も遠隔授業の継続は避けられないものと判断し、「計算機演習」を遠隔授業として実施するための検討を始めた。

本学における Mathematica のライセンス契約には、ウェブアプリケーションの“Mathematica | Online”は含まれていなかったが、開発元の Wolfram Research, Inc. より、2020 年度に限り、期間限定で Mathematica | Online のライセンスが提供された旨が学術情報メディアセンターから告知された。しかし、このライセンスの提供期間は、本授業科目の授業が実施される秋学期全体をカバーするには至らないことが判明した<sup>14)</sup>。そのため、数式処理システムを利用可能な他

<sup>12)</sup>本稿執筆にあたり、本学教育推進部に確認したところ、本学の開設授業科目で、コロナ禍に伴い、遠隔授業に移行した授業科目の単位数は、同通知を根拠に遠隔授業の算定に含めていないとしている。

<sup>13)</sup>例年、春学期の学年暦は、4 月中旬から 7 月末まで講義や演習を行い、8 月上旬が期末試験期間に充てられているが、当年度は、当初予定されていた東京 2020 オリンピック開会式の期日である 7 月 23 日までに春学期の期末試験を終わらせるため、4 月から 6 月にかけて、通常は授業を行わない土曜日にも別の曜日の授業を振り替えて実施することで、学期終了を 2 週間程度繰り上げる予定だった。

<sup>14)</sup>もし、秋学期の授業期間中に Mathematica | Online を継続して利用可能な状況であれば、前年度まで使用していた既存の教材の活用も考慮すると、遠隔授業にも Mathematica | Online を採用する可能性はあったと思われる。

のウェブアプリケーションを検討した結果、7月になり、遠隔授業に用いる主たる授業環境として、クラウド型計算サービス CoCalc を使うことを決め、CoCalc のライセンスを契約した。その後、夏季休業の 8 月から 9 月にかけて教材などの準備を行い、10 月初めから翌 2021 年 1 月末にかけて秋学期の授業を実施した。

### 3.2 遠隔授業の実施方針と予想された課題

「計算機演習」を遠隔授業に切り替えて実施するにあたり、実施方針として考慮した内容は以下の通りである。

**授業内容** 可能な限り、面接授業の授業内容を変えずに授業を行う。

**授業方法** 基本的に、面接授業のときと同様に、事前に提供される教材に基づいた自習と、授業時間内の個別指導を中心に実施し、レポートの提出によって成績を評価する。

**授業環境** ウェブアプリケーションを中心とした授業環境に移行する。

今回の事例のように、特に計算機の操作を伴う授業を遠隔授業に移行する場合には、デスクトップアプリケーションを履修者の端末にインストールする授業形態も考えられた（実際、本学でも、対象となっている教育組織に所属する学生は、Mathematica for Students を利用可能である）が、今回はこのような授業形態を見送った。主な理由は、Mathematica のソフトウェアとしての容量が比較的大きく、すべての履修者の端末上で十分な使い勝手が得られるか懸念があった点や、インストーラの容量が比較的大きく、履修者の自宅のインターネット環境によってはインストールに時間がかかると見込まれた点である。必要なソフトウェアを履修者が各自でダウンロード、インストールすることは、事前学習の一部として実施することも考えられうるが、Mathematica のインストーラの容量は、筆者の勤務先で原稿執筆時点で入手可能な Windows 版バージョン 12.3.1 のインストーラで約 7.68GB であり、仮に転送速度が 10Mbps のネットワーク上で転送を行うとすると、ファイルの転送時間は（ダウンロードが順調に進んだとして）約 1.7 時間となる。想定されている履修者のほとんどが情報学を専門としていないこともあり、履修者の自宅のインターネット環境によっては、Mathematica のダウンロード、インストールを事前学習として実施するには時間的な負担が過大になったり、インストールが完了しないような状況が発生した場合、サポートに対する負担も大きくなったりすることへの懸念があり、今回はデスクトップアプリケーションの採用を見送った。

上記の方針で遠隔授業を実施しようとする際に課題となり得ると判断した点は以下の通りである。

1. 数式処理のパートにおいては、これまで Mathematica で扱っていた内容を、どのような形で CoCalc に移植して実施するか。またどのようにして Mathematica から CoCalc に迅速に移行するか。
2.  $\LaTeX$  のパートにおいては、これまでデスクトップアプリケーションのテキストエディタや  $\TeX$  環境を用いていた部分を、どのようにウェブアプリケーションに移行するか。
3. プログラミング言語のパートにおいては、これまでデスクトップアプリケーションのテキ



ストエディタや言語処理系を用いていた部分を、どのようにウェブアプリケーションに移行するか。

4.  $\LaTeX$  およびプログラミングのパートにおいて、これまで GitHub や GitHub Classroom を用いて行っていた教材の配布やレポートの回収をどのように行うか。
5. 各パートにおける成績の集計をどのように行うか。
6. デスクトップアプリケーションやウェブアプリケーションの基本的な使い方の説明を行う際、面接授業では、履修者に対し、一斉に端末の操作をさせながら操作方法を説明していた部分を、遠隔授業ではどのように行うか。
7. 面接授業では、毎回の授業の中で、個々の履修者の求めに応じ、担当教員や TA が個別に履修者が使用する端末を操作して、デスクトップアプリケーションやウェブアプリケーションの操作手順を説明する場面が見られたが、遠隔授業において、このような個別指導をどのように実現させるか。

次節以降では、以上の課題を踏まえ、今回の遠隔授業で用いた授業環境や授業内容、それらの運用方法を紹介する。

## 4 遠隔授業の授業環境

今回は、前節での検討課題を踏まえ、各授業内容に対し、授業環境として以下のウェブアプリケーションを用いた。

数式処理 クラウド型計算サービス CoCalc の数式処理システム SageMath.

$\LaTeX$  CoCalc の  $\LaTeX$  環境。

プログラミング言語 GitHub, GitHub Classroom, 遠隔統合開発環境 Repl.it.

以下、各授業環境、および今回の授業での利用の要領について簡単に紹介する。なお、GitHub については、サービスの内容はよく知られていると思われるので、本稿での紹介は割愛する。

### 4.1 CoCalc

CoCalc [34] は、計算機数学の種々の計算を行うための遠隔統合環境である<sup>15)</sup>(CoCalc による数学ソフトウェアへの入門の授業例としては Kedlaya [15] 等を参照)。数式処理システム SageMath [33] のプロジェクトの一つで、もともとは SageMath のクラウド環境として SageMathCloud と呼ばれていた。現在では、SageMath (図 1) 以外に、Python, R, Julia, C++, Haskell 等のプログラミング言語や計算システムの環境、および  $\LaTeX$  による文書作成環境を備える。 $\LaTeX$  による日本語の利用は Lua $\LaTeX$  を使うことで可能である [19] (図 2)。Web 上のフロントエンドとして、Jupyter Notebook [30], Sage ワークシート,  $\LaTeX$  エディタ等が用意されている。

今回、数式処理のパートでは、面接授業で用いていた Mathematica に代えて、CoCalc 内の SageMath を用いた。SageMath のフロントエンドには Jupyter Notebook を用いた。また、 $\LaTeX$  のパートでは、面接授業で用いていたデスクトップアプリケーションのテキストエディタや  $\TeX$  環境に代えて、CoCalc 内の  $\LaTeX$  環境を用いた。

<sup>15)</sup>CoCalc のソフトウェア基盤自体もフリーソフトウェアとして公開されており [35]、計算機仮想化基盤 Docker [4] 用のイメージ (仮想環境) も配布されている [36]。

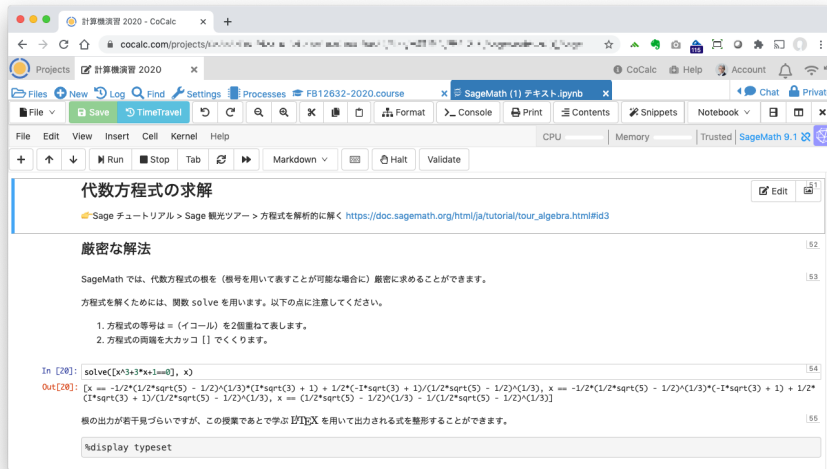


図 1: CoCalc 内の SageMath の実行画面例 (画像は一部加工済み).

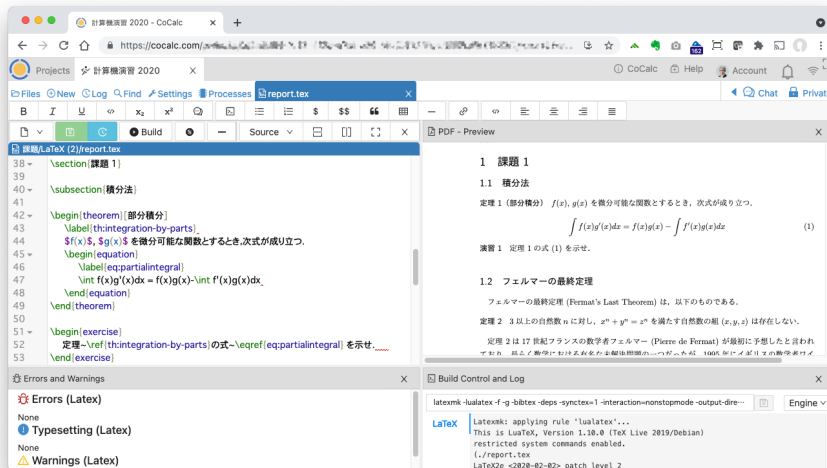


図 2: CoCalc 内の  $\text{\LaTeX}$  文書の編集画面例 (画像は一部加工済み). 日本語  $\text{\LaTeX}$  の利用により, 日本語の文書を作成可能である.

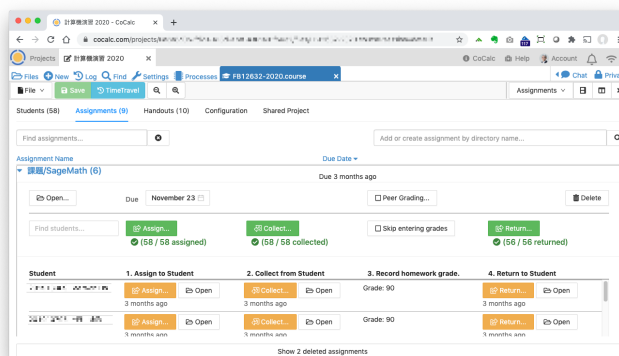


図 3: CoCalc 内の履修者管理画面の例 (画像は一部加工済み). 画面では, 履修者ごとに, あるレポート課題の出題から返却までの管理を行っている.

CoCalc のもう一つの特徴として, 遠隔授業のコース (科目) 管理 (LMS) の機能が搭載されていることが挙げられる (図 3). この機能では, 履修者名簿を作成し<sup>16)</sup>, 履修者 (の CoCalc アカウント) に教材を配布し, 履修者からレポートを回収し, 評価し, 返却することができる. レポートの回収と成績集計の方法として, 数式処理と  $\text{\LaTeX}$  のパートでは, CoCalc の LMS の機能を用いた.

#### 4.1.1 CoCalc のライセンス

ここで, CoCalc のライセンスについて触れておく. CoCalc では, アカウントを作成すると「お試し」の立場で無料で CoCalc の機能を使うこともできるが, 無料ユーザに割り当てられる計算機資源は制限されており, CoCalc を本格的に使う場合, 有料プランの契約が利便性の点で有利になるものと思われる. 記事執筆時点で, 個人用のユーザライセンスの中では比較的安価な “Hobbyist” プランの使用料が月 \$6.15 (年額 \$69.65) である.

CoCalc のライセンスは, プロジェクト単位で課金される (プロジェクトは, 同時に実行可能なアプリケーションの単位である). CoCalc を授業で用いる場合, 教員と TA は原則として授業の指導用に作ったプロジェクトを共有するので, この場合, 必要なライセンス数は 1 個である. 一方で, 履修者に対しては 1 人につき 1 個ずつ, 当該授業のプロジェクトを持たせるため, 履修者数だけの数のライセンスが必要になる.

CoCalc を授業で用いる際には, コースライセンス (Course license) という, 授業用の割引ライセンスが用意されている. 筆者がライセンスを契約した当時は, ライセンスの期間と規模によって以下の通り分かれていた.

- コースの期間による区別 (4 か月間, 1 年間)
- コースの人数による区別 (Small: 25 名, Medium: 70 名, Large: 250 名)

<sup>16)</sup> 履修者の CoCalc のアカウントとの紐付けは, コース設定において履修者のアカウントに招待状 (CoCalc アカウントの新規作成を含む) を送ることで行う.

当時の数学類においては、2020 年度の秋学期に、「計算機演習」で 60 名、「計算機数学 II」(第 2 節「数学類の計算機関連の授業科目における位置付け」を参照)で 20 名、計 80 名の履修者が見込まれた。ライセンス期間は、授業期間については 4 か月間で十分であったが、筆者が授業開始前に教材を準備する時間を考慮すると、いくつかのライセンスはすぐに利用可能にしておくことが望まれた。以上を考慮し、以下のライセンス数および期間の組み合わせで契約した (CoCalc では、複数の期間やグレードのライセンスを同時に契約し、コースの個々の履修者にどのライセンスを割り当てるかを定めることが可能である)。

- Basic Medium Course (70 名), 4 か月間 (2020 年 10 月–2021 年 1 月), \$599.
- Basic Small Course (25 名), 1 年間 (2020 年 7 月から 1 年間), \$799.

## 4.2 GitHub Classroom

GitHub Classroom [7] (図 4) は、GitHub を授業等で利用するための補助的な機能を提供するサービスである (授業における利用の一例は文献 ([5], [37]) を参照)。授業科目に対応する “Classroom” を作成し、以下の手順で利用する。

1. 授業担当教員は、あらかじめ、Classroom に履修者の情報 (学籍番号など) を登録する。登録方法として、担当教員が手動で登録する方法の他に、Learning Tools Interoperability (LTI) [10] に準拠する LMS から履修者名簿を受け取ることも可能である<sup>17)</sup>。履修者の情報と、本人の GitHub アカウントの紐付けは、履修者が GitHub Classroom の利用を開始する時点で履修者本人が行う。
2. 担当教員は、課題のひな型 (授業テキスト、課題文やレポートとなるプログラムのひな型等) を専用のリポジトリに作成し、GitHub Classroom にあらかじめ登録しておく。
3. 履修者が GitHub Classroom の課題の URI にアクセスすると、GitHub Classroom が、課題を解答、提出するための履修者の GitHub リポジトリを作成し、前項で作成されていた課題のひな型を配布する。
4. 履修者は前項で作成されたリポジトリで作業を行い、作業内容を当該リポジトリに push することで課題を提出する。
5. 担当教員は、履修者の課題のリポジトリにて提出された成果物を評価する。必要があれば講評の追記や返却物の作成などを行い、履修者に返却、通知する。

## 4.3 Repl.it

Repl.it [32] (図 5) は、オンライン統合開発環境 (IDE) である<sup>18)</sup> (授業における利用の一例は文献 ([17], [31]) を参照)。Web サイトの情報によると、本稿執筆時においては約 50 種類のプログラミング言語やスクリプト言語に対応しているようである。(Scala は、授業実施時および本稿執筆時においては beta 版の扱いのようである。) また、Git が開発環境に統合されており、

<sup>17)</sup>筑波大学が用いている LMS の manaba は LTI とは互換性を持たないので、今回、履修者名簿の作成は手入力で行った。

<sup>18)</sup>サービス名の “Repl” は、プログラミング言語の実行環境の一つの類型である Read-Eval-Print Loop (REPL) に由来するようである。

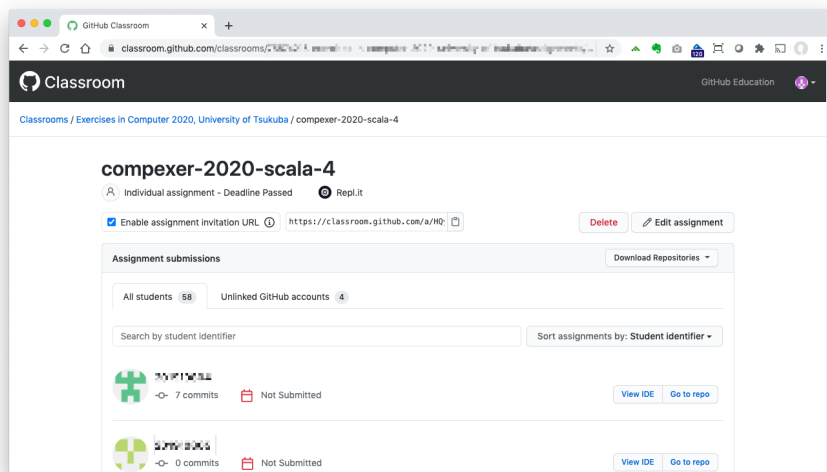


図 4: GitHub Classroom の画面例 (画像は一部加工済み)。

GitHub と連携をとることができるため, GitHub のリポジトリを Repl.it で clone (複製) して開発を行い, 成果物を GitHub に push することも可能である。

#### 4.3.1 Repl.it と GitHub Classroom の連携

今回, オンライン統合開発環境の中から Repl.it を選んだ大きな理由の一つが, GitHub Classroom との連携である。2020 年 5 月, GitHub Inc. は, GitHub Classroom と, 統合開発環境 Repl.it および Microsoft MakeCode との連携機能の実装を発表した [12]。GitHub Classroom と Repl.it を連携させて授業に用いるための主な操作手順は以下の通りである [8]。

1. GitHub Classroom で授業担当教員が新たな Classroom を作成する際, 当該 Classroom と連携させる統合開発環境として Repl.it または Microsoft MakeCode のいずれかの使用を指定することができる。
2. Classroom の中で Repl.it との連携が設定されると, 履修者が Classroom で各課題のリポジトリを作成する際に, 当該リポジトリの README.md ファイルに Repl.it へのリンクが埋め込まれる。
3. 履修者が課題のリポジトリで作業を行い, 上述のリンクをクリックすると, Repl.it の履修者本人のアカウントに課題のリポジトリが自動的に clone され, すぐに Repl.it で開発に取りかかることができる<sup>19)</sup>。
4. 履修者が Repl.it で開発した成果物については, Repl.it の Git の機能を用いて push することにより, GitHub に置かれている履修者の課題のリポジトリに開発内容が反映される。

また, GitHub Classroom と Repl.it の連携により, 通常は有償の Repl.it の機能を無償で利用

<sup>19)</sup> Repl.it のユーザ認証には GitHub アカウントを利用することも可能であり, GitHub のアカウントを所有していれば, すぐに Repl.it の利用を始めることが可能である。

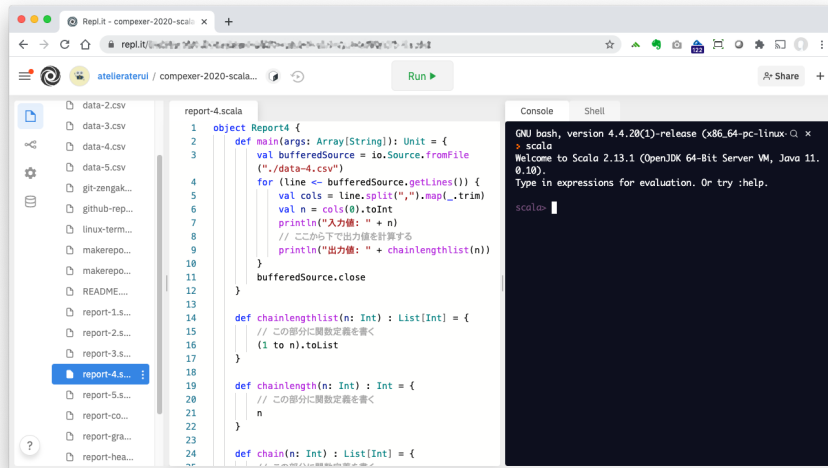


図 5: Repl.it の画面例 (画像は一部加工済み)。

可能である利点もある。Repl.it のサービス体系によると、無償の “Starter” プランでは、開発対象となるプログラムのソースコードは一般公開される。非公開の形で開発を行う際には、通常、有償の “Hacker” プランの契約が必要である。しかし、GitHub Classroom と連携したりポジトリについては、無償で非公開のままでも使えるようになっており、少なくとも Repl.it の利用開始から一定期間は、上述の Hacker プラン相当の機能が無償で利用可能な状態になっているようである<sup>20)</sup>。

これらにより、プログラミング言語のパートでは、面接授業で用いていたデスクトップアプリケーションのテキストエディタや言語処理系に代えて、Repl.it を用いた。授業テキストの配布やレポートの回収には、面接授業の場合と同じく GitHub および GitHub Classroom を用いた。成績の集計には、面接授業の場合と同じく manaba の成績集計の機能を用いた。

## 5 遠隔授業の授業内容

以上の授業環境を踏まえ、遠隔授業の授業内容としては、可能な限り、面接授業に準ずる内容と体裁で行った。

「数式処理」のパートにおける Mathematica から SageMath への移行に関しては、下記の通り、迅速な移行が可能であった。

1. 授業で用いる数式処理の機能は、両システムにおいてほぼ同一の機能が提供されており、関数名や引数、プログラムの構文の書き換え程度で、ほぼ同一の授業内容を構成することが可能であった。Wolfram Language を用いてプログラムを組む部分は Python に置

<sup>20)</sup>ただし、無償利用可能な期間は限定されていると思われる点に注意が必要である。筆者はこの点に関する明文化された情報を確認できなかったが、本稿執筆時の 2021 年 6 月になって、筆者の Repl.it のアカウントでは、それまで使っていた上述の Hacker プラン相当の機能が停止され、本来の有償である Hacker プランへの移行を促された。

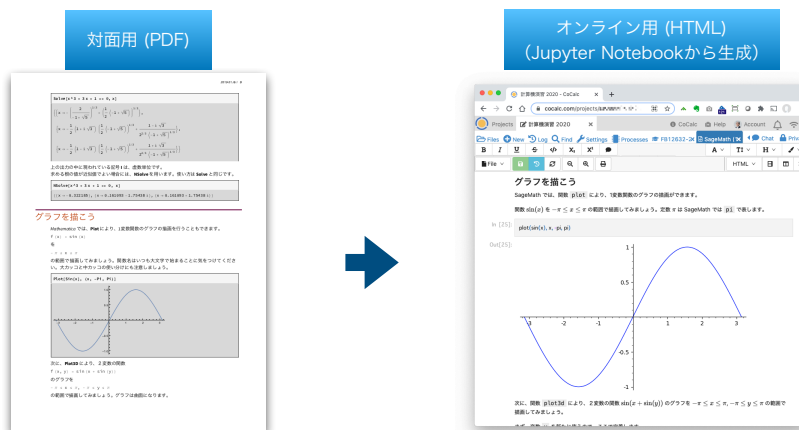


図 6: 「数式処理」パートの授業テキストの移行例．面接授業のもの（左）をほぼそのままの体裁で遠隔授業用に再構成した（右）．

き換えた．ごく少数の箇所では Wolfram Language と Python の間で互換性がない部分は、授業内容から除外した．

2. 両システムとも、フロントエンドが、計算の入出力、テキスト入力やグラフィックスの出力といった機能別の「セル」から構成されており、ほぼ同一の体裁を保ったまま、Mathematica のノートブックから SageMath の Jupyter Notebook への移行が可能であった．

授業テキストは、Mathematica の場合には、Mathematica ノートブックを PDF で出力したファイルを配布していた．SageMath では、Jupyter Notebook を HTML で出力したファイルを配布した．これにより、SageMath においても、Mathematica による授業とほぼ同一の体裁で教材を配布し、授業を行うことが可能になった（図 6）．

$\text{\LaTeX}$  と Scala については、授業内容は面接授業の際とほぼ同一の内容とした．教材は、 $\text{\LaTeX}$  については、日本語  $\text{\LaTeX}$  が、面接授業で用いられていた  $\text{p\LaTeX}$  から日本語  $\text{Lua\LaTeX}$  に切り替わったことから、例題やレポートのひな型に用いるスタイルやパッケージの設定を一部変更した．また、Scala については、操作手順に関する資料に、Repl.it に関する説明や、Repl.it と連携した操作方法といった必要事項を追記した．

## 6 遠隔授業の授業手順

遠隔授業にあたっては、面接授業における各種ウェブアプリケーションの操作手順の説明や、履修者からの質問への回答や端末の操作補助に代わる措置として、以下の措置を行った．

### 6.1 各種ウェブアプリケーションの操作手順の説明

各種ウェブアプリケーションの操作手順については、LMS の manaba に手順書を置くとともに、これまで面接授業において、操作手順を実際に見せながら説明していた部分に該当する手順については、実際の操作手順（スクリーンショット）を示した動画を作成、共有した（図 7）．動

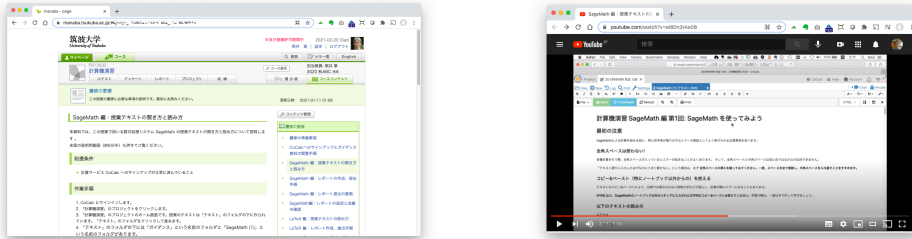


図 7: manaba に掲載した操作手順書（左）と、動画による操作説明（右）の例．

画の配信は、動画配信サービス YouTube を用いて行った．

これらの動画を作成した主目的は、オンライン双方向の遠隔授業における説明時間の短縮である．対面授業でこれらの操作説明を行っていた際は、各ステップの操作ごとに、履修者からの質問に TA が個別に対応していたため、一通りの操作手順を説明するのに 30 分から 40 分程度の時間を必要とした．一方で、オンライン双方向の遠隔授業においては、履修者からの個別の質問への対応を優先することにした．そこで、従来、全履修者に対して一斉に行っていた説明を動画化して、履修者が各自の必要性に応じて動画を閲覧し、担当教員と TA がサポートを必要とする履修者に対して個別に対応することで、遠隔授業の時間の使い方の効率化を図った．

今回作成した手順書および動画の長さを表 1 に示す．

## 6.2 リアルタイム双方向のオンラインセッションの開催

毎週の授業時間帯に、リアルタイム双方向のオンラインセッションを開催した（図 8）．毎回の授業の流れと主な内容は以下の通りである．

1. 前回授業以降、当日までに返却したレポート課題の解説
2. 当日の授業内容とレポート課題の概要説明
3. 授業内容に関する個別の質問への対応
4. 授業内容以外の質問への対応

オンラインセッションのシステムには、オンライン会議サービスの Zoom [45] を用いた．本授業に利用可能なオンライン会議サービスには、大学が契約する Microsoft 365 [21] の Microsoft Teams [22]（以下“Teams”と略す）もあり、大学では Teams の利用が推奨されていたが、本授業では、ブレイクアウトルームとリモート制御の機能を用いるため（詳細は次節を参照）、Zoom を利用した<sup>21)</sup>．

オンラインセッションでは、特に、授業内容に関する個別の質問への対応として、ブレイクアウトルームによる個別の質問への対応、およびリモート制御の機能を用いた履修者の端末の操作補助を行うことで、面接授業における履修者の個々の端末での質問対応に相当するサポートを目指した．

<sup>21)</sup> 本学では、遠隔授業等への対応策として、常勤の教員に対し、大学本部より Zoom のライセンスが提供されていた．



表 1: 各種ウェブアプリケーションの操作手順説明の手順書および動画の長さ (分:秒) . 動画の長さが記載されていないものは, 文書による手順書のみを作成し, 動画は作成していない. 詳細は第 6.1 項を参照 .

操作手順の項目	動画の長さ
1. 履修の準備事項	—
2. CoCalc へのサインアップとガイダンス資料の閲覧手順	8:10
3. SageMath 編	
(a) 授業テキストの開き方と読み方	6:40
(b) レポートの作成, 提出手順	7:55
(c) レポート提出の要領	—
(d) レポートの返却と成績の確認	—
4. L <sup>A</sup> T <sub>E</sub> X 編	
(a) 授業テキストの開き方と読み方	6:18
(b) レポート作成, 提出手順	12:08
(c) レポートの返却と成績の確認	—
5. Git, Scala 編: 教材やレポート課題へのアクセス手順	8:46
6. Git 編: レポートの編集と提出	7:59
7. Scala 編	
(a) 実習の進め方 & 第 1 回の内容	10:54
(b) レポートの作成と提出	15:02
8. Git / Scala 編: レポートの返却と成績の確認	—
9. SageMath ガイド	
(a) SageMath の起動と計算の入力	6:24
(b) ノートブックファイルの保存とファイルの操作	8:32
(c) セルの操作とテキストの入力	12:24
(d) ヘルプの参照	3:22

遠隔授業開催にあたっては, Zoom を主要なビデオ会議システムとして利用する一方, 補助的な連絡手段として, Teams に授業のチームを設け, 履修者および TA を登録した. 授業期間中, スタッフ間の連絡には主に Teams を使い, 授業時間中はオンラインセッションの実施中の補助的な連絡手段として用いた .

### 6.2.1 ブレイクアウトルームにおける個別の質問への対応とリモート制御機能による端末の操作補助

Zoom には「ブレイクアウトルーム」の機能があり, 全体のビデオ会議とは別にビデオ会議室を作り, 参加者の一部が移動しての利用が可能である. また, Zoom には「リモート制御」の機能があり, 会議の参加者が, 別の参加者の端末を遠隔操作可能になる<sup>22)</sup>.

<sup>22)</sup>Zoom のリモート制御機能に相当する機能は, Windows [23] や macOS [1] 等にも備わっている. 授業実施時には, これらのリモート制御機能との比較・検討の発想には至らなかったが, 履修者が利用するパソコンがそれぞれ異なるオ

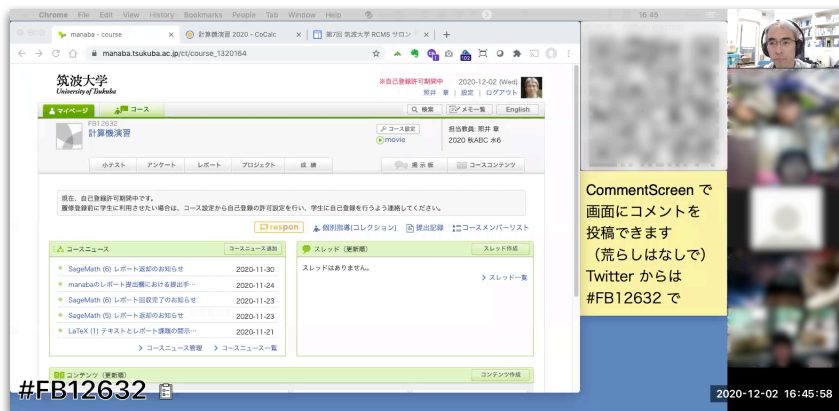


図 8: Zoom によるオンラインセッションの画面例 (画像は一部加工済み)。

授業のオンラインセッション内では、まず、授業の連絡事項の連絡、次に、前回授業以降に返却したレポート課題の解説や、今回の授業内容の紹介を行い、その後、個別の質問を募る。履修者から個別の質問がある人が申し出ると、担当教員が TA を選んで対応を依頼し、その TA と質問者をブレイクアウトルームに配置する。質問者と TA は、ブレイクアウトルームの中で質疑応答を行い、質問が終わればメインの会議室に戻ってくる。

リモート制御 (図 9) では、質問者が自分の端末での操作に困難を感じた際の申告などにより、担当教員や TA が質問者に対してリモート制御のリクエストを行う。質問者がリモート制御を許可すると、リモート制御を申請した担当教員や TA が、質問者の端末を操作できるようになる。これにより、面接授業において、履修者の端末の操作を担当教員や TA が補助するような操作が、遠隔授業でもある程度可能になった。オンラインセッションにおいて、履修者に対しては、カメラによる履修者側の映像の表示は求めなかったが、端末の操作に関する質問の際は、履修者に画面共有を求め、質疑応答は大きなトラブルなく進行した。

ブレイクアウトルームとリモート制御の利用手順の習得にあたり、事前練習を行った。担当教員と TA は、授業開始前の 2020 年 9 月に、Zoom 自体の利用の練習も兼ねて、ブレイクアウトルームの利用や、リモート制御の事前練習を行った。履修者に対しては、初回授業のガイダンス時に、グループに分かれてブレイクアウトルームに入り、各グループに配置された TA の指導により、ブレイクアウトルームの利用を経験してもらった。

## 7 遠隔授業に対する分析・反応・対応の結果

本節では、今回実施した「計算機演習」の遠隔授業に関し、以下の分析・反応・対応の結果をまとめる。

### 1. オンラインセッションの実施結果

オペレーティングシステム (OS) を用いている可能性を考慮すると、OS を選ばずに利用可能な Zoom のリモート制御機能の利用は一つの妥当な手段であったと考えられる。

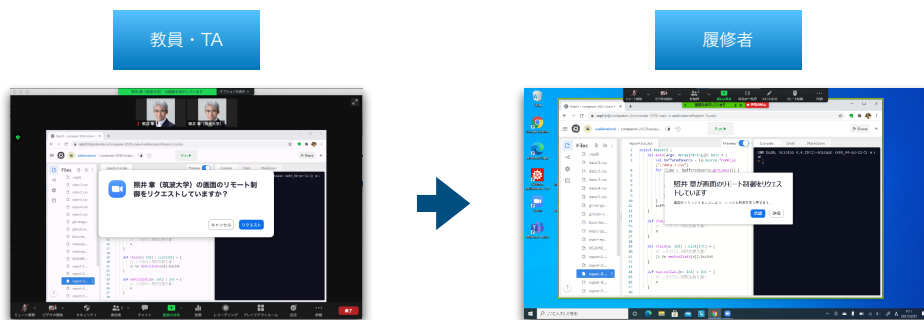


図 9: Zoom によるリモート制御(再現画面). 教員・TA が履修者の画面のリモート制御のリクエストを履修者に対して行くと、履修者の画面にリクエストが表示され、受諾の可否を求められる。

2. 各種ウェブアプリケーションの操作手順の説明資料へのアクセス分析
3. 授業時間外の質問等への対応
4. 授業に用いたウェブアプリケーションのトラブルとその対応
5. TA からの反応

### 7.1 オンラインセッションの実施結果

本年度の「計算機演習」の履修者数は約 60 名であった。履修者の学類別内訳は、数学類が 40 名強で、残りはそれ以外の学類であった。

本授業のオンランセッションは、上述の通り、2020 年 10 月から 2021 年 1 月にわたり、毎週の授業時に Zoom を用いて実施した。履修者のオンラインセッションへの参加は任意とした。初回ガイダンスには履修者のほとんどが参加していたが、2 回目以降は出席者数が減少した。年末以降、毎回の出席者数は 10 名前後であった。

オンラインセッションにおける個別の質問対応は、毎回、3, 4 名の履修者から要望があり、それぞれ Zoom のブレイクアウトルームを設け、担当教員や TA が対応した。履修者全体からすると、ブレイクアウトルームによる個別対応の件数は必ずしも多いとは言えないが、一部の履修者は、ほぼ毎回のオンランセッションに参加し、ブレイクアウトルームで質問をしていた。また、ほぼ毎回のオンラインセッションにおいて、Zoom のリモート制御の機能を用いて履修者の操作を補助する機会もあったので、今回の遠隔授業にあたって準備した Zoom の機能はその目的を果たしたと言える。

オンラインセッションに参加する履修者数が必ずしも多くなかったことの原因について、履修者の一部に尋ねたところ、「授業時間に他の用事があって参加できなかった」という意見が聞かれた一方で「操作手順を説明する動画など、非双方向型の説明資料が充実しており、オンラインセッションに参加しなくても各自で履修を進められたのではないか」という意見も聞かれた。

今回、遠隔授業で用いたウェブアプリケーションはおおむね順調に動作し(例外は第 7.4.1 を参照)、履修者から大きな不満は聞かれなかった。数式処理システムや  $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$  に用いた CoCalc については、一部の履修者から「不満というほどではないが、CoCalc の処理速度が若干遅いと

感じられる時がある」という指摘や「自分のパソコンに Mathematica や  $\text{T}_\text{E}_\text{X}$  をインストールして授業を受けてみたい」という要望も聞かれた。

本授業終了時には、今後の授業改善への利用を目的として、授業アンケートを行ったが、アンケート回収率は履修者の 1/4 程度にとどまった。授業方法に対する大きな不満はなかったが、「面接授業の方が、授業内容への理解度が増したと思うか?」という問いに対する回答は、「増したと思われる」という回答と、「遠隔授業の際とあまり変わらないと思う」という回答が拮抗していた。前者の理由としては「授業の場でわからない点をすぐに質問できるから」、「先生や TA、友人との交流がより活発になると思われるから」という意見があった一方、後者の理由としては「オンラインでの資料提供やサポートの体制が充実していたから」、「面接授業の場合でも授業内容や自分の授業への取り組み方にほぼ差異がないと思われるから」という意見があった。

## 7.2 各種ウェブアプリケーションの操作手順の説明資料へのアクセス分析結果

表 1 に挙げた、各種ウェブアプリケーションの操作手順の説明について、manaba の資料掲載ページへのアクセス状況および動画資料の再生状況を分析した。分析結果を表 2 に示す。分析の対象となった manaba の本授業科目のコースへの登録者数（授業開始当初の履修登録者数）は 57 名で、うち単位取得者数は 46 名だった。表 1 の各列の項目の内容は以下の通りである。

閲覧者数 (N) manaba の当該資料ページの閲覧者数。

閲覧回数 (P) manaba の当該資料ページの閲覧回数。ページを開設して以来の累計閲覧回数（manaba による記録）で、教材準備中のアクセスも含む。

再生回数 (V) YouTube の当該動画の再生回数。2020 年 10 月 1 日から 2021 年 2 月 1 日までの間の各動画の再生回数で、YouTube アナリティクスによる記録である。

P/N 各資料の閲覧者 1 名あたりの閲覧回数の平均値。

V/N 各資料の閲覧者 1 名あたりの動画再生回数の平均値。

V/P 各資料の閲覧回数に対する動画再生回数の割合（例えば、資料の閲覧回数が大きくてもこの値が小さかった場合は、文書資料の需要は大きかったが映像資料の需要は比較的小さかったことが示唆される）。

授業内容の各パートにおける動画の再生回数を比較すると、「5. Git, Scala 編」、「6. Git 編」、「7. Scala 編」の手順書においては、他のパートの手順書に比べ、動画再生回数が比較的大きな傾向が見られた。特筆すべき事項として、表 1 には提示していないが、YouTube アナリティクスによる時間別動画再生率（各動画の総再生回数に対する時間別再生回数の割合）の分析結果によると、操作手順の説明動画のほとんどにおいては、再生開始から 30 秒程度で動画再生率が 50% 程度に下がり、以降、動画の最後に至るまで、動画再生率は 50% 程度を維持していたが、「5. Git, Scala 編」の動画においては、GitHub から Repl.it のサイトに接続する部分の手順を説明した時間帯で、動画再生率が 100% 程度に上がり、視聴者の関心の高さが示唆された。一方で、CoCalc や SageMath,  $\text{L}_\text{A}_\text{T}_\text{E}_\text{X}$  の操作手順に関する手順書においては、文書の閲覧回数が大きなものでも、動画の再生回数は Git や Scala の操作手順書のそれらに比較して小さな傾向が見られた。

以上の結果より、ウェブアプリケーションの操作手順の動画による説明に関しては、特に Git,

表 2: 各種ウェブアプリケーションの操作手順の説明資料へのアクセス分析結果．詳細は第 7.2 項を参照．

項目番号	閲覧者数 (N)	閲覧回数 (P)	再生回数 (V)	P/N	V/N	V/P
1	56	1797	—	32.09	—	—
2	47	531	43	11.30	0.91	0.08
3 (a)	36	198	26	5.50	0.72	0.13
3 (b)	42	228	26	5.43	0.62	0.11
3 (c)	45	221	—	4.91	—	—
3 (d)	43	97	—	2.26	—	—
4 (a)	43	138	22	3.21	0.51	0.16
4 (b)	46	202	52	4.39	1.13	0.26
4 (c)	29	51	—	1.76	—	—
5	49	363	138	7.41	2.82	0.38
6	49	236	73	4.82	1.49	0.31
7 (a)	48	231	83	4.81	1.73	0.36
7 (b)	46	184	75	4.00	1.63	0.41
8	35	77	—	2.20	—	—
9 (a)	55	489	26	8.89	0.47	0.05
9 (b)	43	130	15	3.02	0.35	0.12
9 (c)	40	122	17	3.05	0.43	0.14
9 (d)	29	67	8	2.31	0.28	0.12

GitHub, Repl.it, Scala の操作手順の説明において、動画による操作手順の説明に対する需要が比較的大きく、繰り返しての視聴も含め、動画による説明の効果が高かったことが示唆された。Git, GitHub, Repl.it, Scala の操作手順の説明動画の再生回数が比較的大きかった理由の分析には、履修者に対する各種ウェブアプリケーションの難易度の調査等、今回実施されなかった調査も必要と思われるが、授業時間内外での履修者からの質問状況（第 7.3 項も参照）を見る限りでは、これらのウェブアプリケーションの操作の難度が CoCalc のそれに比べて比較的高かった可能性もあると考えられる。

### 7.3 授業時間外の対応

授業時間外は、主に担当教員（筆者）が、Teams のチャットを用いて、授業内容やウェブアプリケーションの操作方法に関する質問や相談に応じた。履修者の操作補助が必要な場合は、Zoom の会議を立ち上げ、リモート制御の機能を用いて質問に対応した。ウェブアプリケーションの操

作方法に関する質問は、特に Repl.it を使う 12 月後半以降の授業で比較的多く（後述のトラブルと関連がある）、レポートの提出期限の都合で、年末年始の大学の休業期間中にも質問への対応を行うことがあった。

#### 7.4 ウェブアプリケーションのトラブルとその対応

授業に用いたウェブアプリケーションのトラブルで最も多かったものが「Repl.it で授業の課題が開けない」というもので、その他に、CoCalc のバグに起因するトラブルが若干あった（授業への直接の影響はなかった）。

##### 7.4.1 Repl.it で課題が開けないトラブル

このトラブルは、Repl.it を GitHub Classroom と連携させた後で、GitHub の自分の課題のリポジトリにあるボタンを押しても Repl.it で課題のリポジトリが開かない、というものであった。このトラブルは Repl.it を用いた Scala の授業の 4 回で発生し、毎回、数名の履修者から質問の連絡があり、対応した。原因の詳細は不明で、系統的な対処法も不明であるが、GitHub Classroom と Repl.it を連携させる作業を何度か繰り返す等の操作を行っているトラブルが解消し、幸い、授業への大きな影響はなかった。

##### 7.4.2 CoCalc のバグとサポート対応

CoCalc を利用中に、数件の不具合の報告を TA から受けた（例えば、Jupyter Notebook の特定のセルに書き込みができない、等）。CoCalc にはサポートチケットの制度があり、TA から報告を受けた現象が担当教員の環境でも再現されることを確認し、CoCalc へサポートを依頼した。日本と開発元の時差の影響もあり得るが、日本時間の夕方に報告した不具合は 3, 4 時間ののちには CoCalc のサポートによってバグと認定され、修正の報告を受けた。サポートの対応は迅速であるとの印象を受けた。ただし、サポートへの連絡は、本稿執筆時点では英語またはドイツ語で行う必要がある。

#### 7.5 TA からの反応

秋学期の授業が終わってから、TA とも意見交換を行った。その結果、以下の指摘があった。

- CoCalc を初めて使ったが、オンラインサービスという新鮮味もあり楽しかった。一方で、CoCalc は動作が若干不安定な部分もあったので、自分が（数式処理システムを）使うのであれば Mathematica を選ぶと思う。
- $\text{\LaTeX}$  については、現在の履修者が今後卒業研究で使うことを考えると、 $\text{\TeX}$  環境を履修者にインストールさせてもよいのではないか。（筆者註：数学類では、4 年次の「卒業研究」のまとめとして「卒業研究発表会」を開催している。近年の卒業研究発表会では、学生の多くは  $\text{\LaTeX}$  と Beamer パッケージでスライドを作成して発表している。）
- Repl.it へのログインが不安定だったのは改善を要する。
- 本年度は SageMath,  $\text{\LaTeX}$  の順で授業を行ったが、 $\text{\LaTeX}$  を先に教えてもよいかもしれない。（筆者註：CoCalc では、Markdown のセル中に  $\text{\TeX}$  形式で数式を記述可能である。 $\text{\LaTeX}$  を先に習っておくと、SageMath のレポートの説明文に数式を使う際に利便性が増

すものと思われる。)

## 8 まとめ

本稿では、コロナ禍の2020年度に、筑波大学理工学群数学類の開設授業科目「計算機演習」において行った遠隔授業を取り上げ、情報学を専門としない理学系学部の学生を対象にした計算機実習を伴う授業において、遠隔授業の形態で面接授業と同等の効果を挙げる目的で行った各種の取り組みの一例を紹介した。第1節で提示した各課題に対しては、以下の結果が得られ、授業内容は面接授業とほぼ同一の内容を維持することが可能であった。

1. 数式処理や  $\text{\LaTeX}$  といった、数式を計算機上で直接的に取り扱う授業を、面接授業から遠隔授業に円滑に移行することについては、新規に導入した計算機数学の遠隔統合環境 CoCalc や、統合開発環境 Repl.it と、面接授業時より継続して用いた GitHub, GitHub Classroom を組み合わせた利用により、円滑な移行が可能であった。
2. 従来デスクトップアプリケーションを用いていた面接授業から、ウェブアプリケーションを用いた遠隔授業への切り替え（教材の書き換えや、学生のレポート回収、採点、返却に至るワークフローの切り替え）については、アプリケーションの切り替えについては、数式処理システム Mathematica を SageMath に置き換え、授業運用のワークフローの切り替えについては、CoCalc の LMS の機能と従来の manaba, GitHub, GitHub Classroom の当該機能を併用することで、これらの切り替えを短期間で実現させることが可能であった。
3. 計算機の操作方法やプログラミングを含めた指導（個別指導を含む）に関しては、従来、スクリーンに操作方法を提示して説明していた部分を動画に収録して配信し、Zoom のブレイクアウトルームやリモート制御機能を用いることにより、遠隔授業においても、面接授業と同等の指導を実現させることが可能であった。

今回、ウェブアプリケーションとして導入した CoCalc の今後の履修者による利活用については、CoCalc 全体を本格的に活用するためには有償契約が必要かもしれないが（履修者の意向にもよる）、授業内で主に利用した SageMath 自体はフリーソフトウェアとして配布されており、今後、履修者で興味を持つ者に利用を促すことも可能であろう。また、サテライト端末室が再び利用可能になった際には、SageMath と同等の機能を持つデスクトップアプリケーションとして Mathematica の存在を周知し、利用を促すことも考えられる。 $\text{\LaTeX}$  や Scala の処理系についても、サテライト端末室が再び利用可能になった際には、サテライト端末室での利用や、各自のパソコンにデスクトップアプリケーションをインストールしての利用を促すことも考えられる。

今回の遠隔授業における履修者の反応（上述）や履修状況から、遠隔授業においても面接授業と比較して学習効果が大きく損なわれることはなかったことが示唆されるが、それらの詳細な分析は今後の課題である。コロナ禍の継続により2021年度も「計算機演習」は遠隔授業で開講する可能性があるが、もし引き続き遠隔授業が行われる場合には、本授業科目の履修が履修者にとって有益な経験になるよう、本稿に述べた経験を生かしつつ、授業内容や授業方法の改善を引き続き行いたいと考えている。

## 謝 辞

「計算機演習」の遠隔授業への移行に際し、授業実施時期の移行や CoCalc のライセンス費用の拠出でご配慮下さった、筑波大学理工学群数学類に感謝いたします。

## 参 考 文 献

- [1] Apple, Inc. Apple Remote Desktop ユーザガイド . <https://support.apple.com/ja-jp/guide/remote-desktop/welcome/mac> ( 閲覧 2021-12-05 ).
- [2] 株式会社朝日ネット . クラウド型教育支援サービス「manaba」. <https://manaba.jp/> ( 閲覧 2021-06-18 ).
- [3] 大学 ICT 推進協議会 ICT 利活用調査部会 . BYOD を活用した教育改善に関する調査研究: 結果報告書 ( 第 1 版 ). 大学 ICT 推進協議会 , 2018 年 3 月 . [https://axies.jp/report/ict\\_survey/2016survey/](https://axies.jp/report/ict_survey/2016survey/) ( 閲覧 2021-10-01 ).
- [4] Docker, Inc. Docker: Empowering App Development for Developers. <https://www.docker.com/> ( 閲覧 2021-06-18 ).
- [5] J. Fiksel, L. R. Jager, J. S. Hardin, and M. A. Taub. Using GitHub Classroom To Teach Statistics. *Journal of Statistics Education*, **27** (2), 110–119, 2019.
- [6] GitHub, Inc. GitHub. <https://github.com/> ( 閲覧 2021-06-18 ).
- [7] GitHub, Inc. GitHub Classroom. <https://classroom.github.com/> ( 閲覧 2021-06-18 ).
- [8] GitHub, Inc. Manage coursework with GitHub Classroom. GitHub Docs. <https://docs.github.com/en/education/manage-coursework-with-github-classroom> ( 閲覧 2021-06-18 ). 日本語訳あり .
- [9] 飯尾淳 . プログラミング教育における反転授業の実施 . 情報教育シンポジウム論文集 , 2021, 209–215 , 情報処理学会 , 2021.
- [10] IMS Global Learning Consortium. Learning Tools Interoperability. <https://www.imsglobal.org/activity/learning-tools-interoperability> ( 閲覧 2021-06-18 ).
- [11] 石田晴久 . 大学教養課程における大人数の計算機実習 . 情報処理 , **9** (1), 36–38, 1968.
- [12] A. Jones. Code in the browser with GitHub Classroom. GitHub Blog, May 26, 2020. <https://github.blog/2020-05-26-code-in-the-browser-with-github-classroom/> ( 閲覧 2021-06-18 ).
- [13] 情報処理学会「超スマート社会における情報教育の在り方に関する調査研究」事業実施統括委員会 . 「超スマート社会における情報教育の在り方に関する調査研究」平成 28 年度報告書 . 文部科学省先導的 大学改革推進委託事業「超スマート社会における情報教育の在り方に関する調査研究」, 文部科学省 , 2018 年 3 月 . 国立国会図書館インターネット資料収集保存事業 ( WARP ), 2021 年 8 月 5 日 . [https://warp.da.ndl.go.jp/info:ndljp/pid/11710739/www.mext.go.jp/a\\_menu/koutou/itaku/1386892.htm](https://warp.da.ndl.go.jp/info:ndljp/pid/11710739/www.mext.go.jp/a_menu/koutou/itaku/1386892.htm) ( 閲覧 2021-11-14 ).
- [14] 河村一樹 . 講義レスによるプログラミング実習教育の試み . 情報処理学会研究報告 , **2015-CE-128** (21), 7 pages, 2015.



- [15] K. Kedlaya. Math 157: Introduction to Mathematical Software. University of California, San Diego, 2018. <https://github.com/kedlaya/math157> ( 閲覧 2021-11-28 ).
- [16] 喜多一, 岡本雅子. 写経型プログラミング学習と反転授業. 第 60 回システム制御情報学会研究発表講演論文集, 4 pages, 2016.
- [17] N. Kodagoda, A. Gamage, K. Suriyawansa, B. Jayasinghe, S. Rupasinghe, D. Ganegoda, T. Jayalath, and A. Kurrupu. Innovative use of Collaborative Teaching in Conducting a Large Scale Online Synchronous Fresher's Programming Course. 2021 IEEE Global Engineering Education Conference (EDUCON), 891–896, IEEE, 2021.
- [18] 厚生労働省. 中華人民共和国湖北省武漢市における原因不明肺炎の発生について. 報道発表資料, 2020 年 1 月 6 日. 国立国会図書館インターネット資料収集保存事業( WARP ), 2021 年 8 月 2 日. [https://warp.da.ndl.go.jp/info:ndljp/pid/11714667/www.mhlw.go.jp/stf/newpage\\_08767.html](https://warp.da.ndl.go.jp/info:ndljp/pid/11714667/www.mhlw.go.jp/stf/newpage_08767.html) ( 閲覧 2021-11-07 ).
- [19] Lua $\TeX$ -ja プロジェクト. Lua $\TeX$ -ja. <https://ja.osdn.net/projects/luatex-ja/> ( 閲覧 2021-06-18 ).
- [20] 道越秀吾, 丸野由希. オンライン授業におけるプログラミング教育の実践と評価. 2020 年度情報処理学会関西支部支部大会講演論文集, 4 pages, 2020.
- [21] Microsoft Corporation. Microsoft 365 ドキュメント. <https://docs.microsoft.com/ja-jp/microsoft-365/> ( 閲覧 2021-06-18 ).
- [22] Microsoft Corporation. Microsoft Teams 管理ドキュメント. <https://docs.microsoft.com/ja-jp/MicrosoftTeams/> ( 閲覧 2021-06-18 ).
- [23] Microsoft Corporation. リモート アシスタンスと簡単接続を使用して、リモートで PC の問題を解決する. <https://support.microsoft.com/ja-jp/topic/%E3%83%AA%E3%83%A2%E3%83%BC%E3%83%88-%E3%82%A2%E3%82%B7%E3%82%B9%E3%82%BF%E3%83%B3%E3%82%B9%E3%81%A8%E7%B0%A1%E5%8D%98%E6%8E%A5%E7%B6%9A%E3%82%92%E4%BD%BF%E7%94%A8%E3%81%97%E3%81%A6-%E3%83%AA%E3%83%A2%E3%83%BC%E3%83%88%E3%81%A7-pc-%E3%81%AE%E5%95%8F%E9%A1%8C%E3%82%92%E8%A7%A3%E6%B1%BA%E3%81%99%E3%82%8B-cf384ff4-6269-d86e-bcfe-92d72ed55922> ( 閲覧 2021-12-05 ).
- [24] 文部科学省. 各教科等の指導における ICT の効果的な活用に関する参考資料: 算数・数学科の指導における ICT の活用について. 2020 年 9 月. 国立国会図書館インターネット資料収集保存事業( WARP ), 2021 年 8 月 1 日. [https://warp.da.ndl.go.jp/info:ndljp/pid/11710739/www.mext.go.jp/a\\_menu/shotou/zyouhou/mext\\_00915.html](https://warp.da.ndl.go.jp/info:ndljp/pid/11710739/www.mext.go.jp/a_menu/shotou/zyouhou/mext_00915.html) ( 閲覧 2021-11-07 ).
- [25] 文部科学省. 高等学校学習指導要領(平成 30 年告示). 2018 年 3 月. 国立国会図書館インターネット資料収集保存事業( WARP ), 2021 年 8 月 1 日. [https://warp.da.ndl.go.jp/info:ndljp/pid/11710739/www.mext.go.jp/content/1384661\\_6\\_1\\_3.pdf](https://warp.da.ndl.go.jp/info:ndljp/pid/11710739/www.mext.go.jp/content/1384661_6_1_3.pdf) ( 閲覧 2021-11-07 ).

- [26] 文部科学省．中学校学習指導要領（平成 29 年告示）. 2017 年 3 月．国立国会図書館インターネット資料収集保存事業（WARP）, 2021 年 8 月 1 日．[https://warp.da.ndl.go.jp/info:ndljp/pid/11710739/www.mext.go.jp/content/1413522\\_002.pdf](https://warp.da.ndl.go.jp/info:ndljp/pid/11710739/www.mext.go.jp/content/1413522_002.pdf)（閲覧 2021-11-07）.
- [27] 文部科学省高等教育局大学振興課．本年度後期や次年度の各授業科目の実施方法に係る留意点について．事務連絡，令和 2 年 7 月 27 日．国立国会図書館インターネット資料収集保存事業（WARP）, 2021 年 8 月 1 日．[https://warp.da.ndl.go.jp/info:ndljp/pid/11710739/www.mext.go.jp/content/20200727-mxt\\_kouhou01-000004520\\_1.pdf](https://warp.da.ndl.go.jp/info:ndljp/pid/11710739/www.mext.go.jp/content/20200727-mxt_kouhou01-000004520_1.pdf)（閲覧 2021-11-29）.
- [28] 文部科学省高等教育局長．大学等における遠隔授業の取扱いについて（周知）. 3 文科高第 9 号，令和 3 年 4 月 2 日．国立国会図書館インターネット資料収集保存事業（WARP）, 2021 年 8 月 1 日．[https://warp.da.ndl.go.jp/info:ndljp/pid/11710739/www.mext.go.jp/content/20210402-mxt\\_daigakuc01-000014531\\_1.pdf](https://warp.da.ndl.go.jp/info:ndljp/pid/11710739/www.mext.go.jp/content/20210402-mxt_daigakuc01-000014531_1.pdf)（閲覧 2021-11-29）.
- [29] 長瀧寛之．情報処理入門科目における反転授業形式の授業実践．情報処理学会研究報告，**2018-CE-143** (22), 1–9, 2018.
- [30] Project Jupyter. Jupyter Notebook. <https://jupyter.org/>（閲覧 2021-06-18）.
- [31] M. M. Rahman, R. Paudel, and M. H. Sharker. Effects of Infusing Interactive and Collaborative Learning to Teach an Introductory Programming Course. 2019 IEEE Frontiers in Education Conference (FIE), 1–8, 2019.
- [32] Replit, Inc. Repl.it. <https://replit.com/>（閲覧 2021-06-18）.
- [33] The Sage Group. SageMath — Open-Source Mathematical Software System. <https://www.sagemath.org/>（閲覧 2021-06-18）.
- [34] SageMath, Inc. CoCalc: Collaborative Calculation and Data Science. <https://cocalc.com/>（閲覧 2021-06-18）.
- [35] SageMath, Inc. CoCalc: Collaborative Calculation in the Cloud (source code). <https://github.com/sagemathinc/cocalc>（閲覧 2021-06-18）.
- [36] SageMath, Inc. CoCalc Docker image. <https://github.com/sagemathinc/cocalc-docker>（閲覧 2021-06-18）.
- [37] G. Sprint and J. Conci. Mining GitHub Classroom commit behavior in elective and introductory computer science courses. *Journal of Computing Sciences in Colleges*, **35** (10), 76–84, 2019.
- [38] 筑波大学．教育課程編成支援システム（KdB）マニュアル．<https://www.tsukuba.ac.jp/campuslife/tool-manual-kdb/>（閲覧 2021-10-01）.
- [39] 筑波大学．キャンパスライフ: パソコン等の準備について．<https://www.tsukuba.ac.jp/campuslife/calendar-ceremony/pc/index.html>（閲覧 2021-10-03）.
- [40] 筑波大学．卒業者の教員免許状取得の状況（教育職員免許状一括申請集計表【令和 2 年度】）. <https://www.tsukuba.ac.jp/education/ug-courses-tt-programs/>（閲覧

2021-11-07).

- [41] 筑波大学．筑波大学学群学則（平成 16 年法人規則第 10 号）. <https://www.tsukuba.ac.jp/about/disclosure-ho-kisoku/s-04/>（閲覧 2021-11-06）.
- [42] 筑波大学．令和 3（2021）年度履修要覧．<https://www.tsukuba.ac.jp/education/ug-courses-directory/>（閲覧 2021-11-06）.
- [43] 米沢宣義，志村武，南敏．多人数初心者向きプログラミング教育システムの基本設計について．情報処理学会論文誌，**27** (1), 96–102, 1986.
- [44] 吉川遼．プログラミング系授業におけるオンライン型反転授業の導入と実践．名古屋文理大学紀要，**21**, 5–14, 2021.
- [45] Zoom Video Communications, Inc. Zoom（ビデオ会議，ウェブ会議，ウェビナー，画面共有）. <https://zoom.us/>（閲覧 2021-06-18）.