

SaaS環境によるプログラミング初学者教育の試み

A Programming Class for Beginners with SaaS

田中克明[†], 鈴木令子[†], 山崎秀記[‡]

Katsuaki Tanaka[†], Reiko Suzuki[†], Hideki Yamasaki[‡]

katsuaki@cio.hit-u.ac.jp, suzuki@cio.hit-u.ac.jp, yamasaki.hideki@r.hit-u.ac.jp

一橋大学情報基盤センター[†]

一橋大学大学院商学研究科[‡]

Center for Information and Communication Technology, Hitotsubashi University[†]

Graduate School of Commerce and Management, Hitotsubashi University[‡]

概要

SaaSとして提供されるGoogle Appsを用い、Webサイトの作成、サイトへのアンケートフォームの設置、フォームを通して収集したデータの処理、およびそのためのWebサービス呼び出しなどを含むプログラムの作成を、授業の一部として実施した。

本稿では、この授業に関連し、プログラミングを学習する環境としてSaaSを利用するメリットの検討、授業で利用したGoogle Apps Scriptを含めたGoogle Appsの概要、授業の目的と実施内容を述べる。

キーワード

SaaS, オンラインプログラミング, Google Apps

1 はじめに

プログラミングなど計算機を操作する授業では、そのための環境を学習者が利用する端末上に用意する必要がある。また、学習者に自習をうながすためには、演習室などを提供するだけでなく、学習者所有のPCにもプログラミング環境一式をセットアップし自習環境を整えることが望ましい。しかし、学生がPCを保持している場合が多くなったとはいえ、プログラミング初学者にプログラミング環境のセットアップを要求することは難しく、結果として、あらかじめ用意された端末環境以外において学習を行うことは困難であった。

これに対して、Webを通したサーバサイドのサービス作成のためのプログラミングを対象とし、サーバサイドにプログラミング環境を用意する試みがなされている[1][2]。これらの手法は端末室を用意するよりは十分に負荷は低い、授業目的に合うように環境を開発し整

備を行う必要があるなど、直接の授業内容以外の部分の用意に相応の時間を要する。

環境の構築に時間を要しない方法としては、Microsoft Excelに組み込まれているVisual Basic for Application (VBA)を用いるなどの方法が考えられる。しかし、学習者の立場から見るとMicrosoft Excel固有の環境を学ぶという印象が強く、「プログラミング」を学習しているという意義が伝わりづらいことが、問題点として挙げられよう。

近年のクラウドの発展にともないX as a Serviceとして、Software as a Service (SaaS)、Platform as a Service (PaaS)、Infrastructure as a Service (IaaS)などの形でさまざまなサービスが提供されている。クラウドをプログラミングに関連して用いる場合、PaaSをあらかじめ記述したコードを実行するための環境として利用する、IaaSとして提供された仮想マシン中にOSから環境を構築して実行環境を構成するといった、実行

環境としての利用が多くなされている。一方、プログラミング「環境」は SaaS に該当すると考えられ、例えば Cloud9 IDE¹ のような Web ベースの統合環境が SaaS として存在する。

一橋大学では、学生向けのメール環境として 2008 年より Gmail を導入しており、メール以外のいくつかのサービスを統合した Google Apps も利用可能となった。時間の経過とともにサービスも安定し、授業での利用が現実的になり、近年、Google Apps を教育に用いる検討がなされている [3]。そこで、2011 年度より、Google Apps のうちいくつかのサービスを組み合わせつつ、JavaScript に準じた言語体系を持ち開発環境が SaaS として提供されている Google Apps Script を用いて、プログラミング初学者向けの授業を実施した。本稿ではその内容を報告する。

2 プログラミング学習における環境

2.1 一橋大学情報系科目におけるプログラミング授業の現状

一橋大学では、情報に関連した科目を共通教育科目の中で「情報基礎」「計算機概論」などとして開講している。「情報基礎」は、学部生を対象に開講しており、通年で 24 クラスが開講され、ほぼ全学生が学部 1、2 年次で履修を終える。「情報基礎」「計算機概論」とも、授業は学生 1 人ずつに Windows7 がインストールされた端末が用意された演習室にて行い、実施内容は、ある程度まで担当する教員の裁量に任されている。

2.1.1 情報基礎におけるプログラミング導入

現状の情報基礎においては、半数以上の講義でシラバスに少なくとも一コマプログラミングを取り入れている。3 章にて述べる以外に、取り入れ方は様々であるが、次のような目的に沿って行われている。

1. Excel における VBA

Microsoft Excel のマクロ記述言語である VBA を用い、操作手順を命令文の列で記述できること、分岐・繰返しといった基本的な制御構造の理解

2. Java+NetBeans

統合開発環境における GUI をもつアプリケーション開発手順の理解

3. C 言語によるプログラミング

プログラミングを用いた「情報科学」という学問分野の基本概念理解

2.1.2 計算機概論におけるプログラミング

計算機概論は、情報基礎を履修した後の基本的プログラミングを目的とし、年間 16 クラスを開講している。実施内容は、情報基礎同様ある程度まで担当教員の裁量に任されており、C、Java などいくつかの選択肢がある。ここでは Java を利用した授業例を示す。

● 使用環境: Java, NetBeans

● 内容: 基本的な値と型・変数、制御構造（分岐・繰返し）、配列、アルゴリズムの巧拙：離散指数・整列における計算時間の比較、メソッド、ウィンドウズアプリケーションの作成（GUI 部品、イベントの取得）

● 課題: 簡易テキストエディタ、ファイル入出力

2.1.3 VBA と NetBeans の授業における利点と欠点

上述した情報科目におけるプログラミングの授業を通して、VBA と NetBeans の利用上のメリット、デメリットがいくつか浮かび上がった。

VBA のメリットとして、Excel 内部から VBA を用いることによりスプレッドシートの操作やセル配列のイメージと結びつけてプログラムの動作を理解することが容易であることが挙げられる。命令と Excel 上の具体的な操作を対応づけられプログラミングのイメージが掴みやすく、学生のプログラミングに対する敷居を下げられる。また、Excel さえあれば、プログラミング環境を構築する必要が無いことも利点である。

VBA のデメリットのひとつは、エラーメッセージなどが不十分でありデバッグが難しいことがある。また、Excel が「計算機」としての機能を補うため、データ型への意識、あるいは浮動小数点演算の誤差など、プリミティブな「プログラミング」を学ぶことが難しいこともデメリットとして挙げられる。

一方、NetBeans の利点は、Windows アプリケーションの作成を目的にした上で、部品の配置、プロパティ設定が容易に行え、イベントハンドラの記述に専念することが可能であることが挙げられる。さらに豊富な入力補完機能があり、エラーメッセージも充実していることから、マニュアルを別途参照する手間が省ける。また、無償のオープンソースソフトウェアであり学習者の PC へのインストールが可能であることもメリットである。

NetBeans 利用のデメリットとしては、開発環境そのものを習得するハードルが挙げられる。プログラミングに必要な開発環境の操作手順習得のために 1 回（90 分）をあてている。誤操作を行った際の修復が困難であり、授業中、開発環境の操作に関する疑問への対応に相応の

¹<http://c9.io/>

時間を必要とすることも問題である。また、このような利用上の手間などにより、無償で配付されているとはいえ、自らの PC にインストールし活用する学生は少数である。

2.2 プログラミング実習における環境の割り当て方式

2.1.3 でも検討を行ったように、授業にてプログラミングを取り上げる際、その環境をどのように用意するかということは、授業実施のために重要な事項の 1 つである。

プログラミングを計算機を用い実習を通して学習するためには、プログラムを作成する環境およびプログラムを実行する環境が計算機上に必要である。そのためには、学習者が利用できる物理的なインタフェースを用意し、インタフェースを通して計算機上のエディタなどの開発環境や実行環境を利用することが出来ればよい。これら割当て方には、例えば、以下のような方式が考えられる。

1. 学習者ひとりひとりに開発・実行環境として PC を用意し、PC ごとに必要なソフトウェアをインストールする
2. 学習者ひとりひとりに開発・実行環境として端末を用意し、端末はネットブートなどによるクライアントとして一括管理する
3. 実行環境を Web サーバなどに共通化し、学習者には開発環境が用意された端末を割り当てる
4. 実行・開発環境を Web サーバなどサーバ上に用意し、学習者にはサーバにアクセスする環境を割り当てる
5. 実行・開発環境にサービスとして提供されている環境 (SaaS) を用い、学習者にはサービスにアクセスする環境を割り当てる

1. の 1 人 1 端末方式では端末の管理にある程度の費用が発生し、管理維持にも労力が必要である。そのため 2. のようなシンクライアント方式が多数の端末を設置する多くの演習室で採り入れられている。本学でもこの形式で演習室を提供している。この場合、開発環境を新規に導入する際、原則としてオリジナルとなる 1 式分のディスクイメージ作成に注力すれば良いのだが、端末を利用する利用者 ID ごとに環境設定を必要とする事項が残ることもあり、授業進行中に受講者が設定作業を行う必要が生じる場合が多い。また、演習室以外で自習を

行いたい場合には、演習室同様の環境を学習者ごとに設定する必要がある。

3. は Web サーバを用い、Web サーバ上で動作するプログラム、CGI などを作成する場合などに用いられる形式である。エディタなどの開発環境を手元の端末に用意し、作成したプログラムを ftp などを利用してサーバに転送、サーバ上でプログラムを実行させる。手元の端末には最低限テキストエディタなどがあればよく、実行環境などをサーバ側で一元管理できる点では負担が低い。また、実行される環境への依存性が高いコンパイラ部分をサーバ上に設置する方式 [4] もある。授業実施者による環境管理が容易であり、演習室の端末以外でもサーバにアクセスできれば手元にはエディタが用意できればよく環境設定の手間がかからないなどの利点がある一方、記述したコードをサーバに転送する手間や、デバッグの困難さが伴う。

4. は、主に Web を通したサーバサイドのサービス作成のためのプログラミングを対象とし、サーバサイドにプログラミング環境を用意する方式 [1][2] である。その他、実行・開発環境は端末に持たせるが、プログラムソースコードの管理をサーバに行い、学習の進捗状況などを把握する仕組みも提案されている [5]。これらは 3. と同様に学習者に必要とされる環境設定が少ない上に、コード記述、デバッグなどの環境も Web を通して提供されることから、学習に集中することが可能である。一方、環境を構築するためには、Web ベースのエディタやデバッグ環境、学習者を認証する仕組みなど、さまざまな仕組みの構築を行わなければならない。個別の授業ごとにサーバハードウェア、サーバ上で動作するソフトウェアなどを用意すると、それなりのコストを要する。

5. が本稿で報告する SaaS 方式である。実施形態は 4. と同じであるが、サーバのハードウェア、サーバ上で動作するソフトウェア、すべてサービスとしてあらかじめ提供されているものを用いる。学習者は Web ブラウザさえ動作すればプログラミング環境を用いることができ、授業を行う側も、特別な環境の用意を行う必要がない。

2.3 Google Apps

今日、Web ブラウザとネットワーク接続を用意すればソフトウェアによるサービスを利用できる SaaS としてはさまざまなサービスが提供されている。その先駆けともなったサービスがメール (Gmail) に代表される Google Apps である。Gmail 以外にも、一般的なオフィススイートに準じた、スプレッドシート、ワードプロセッサ、プレゼンテーションツールなどが Web ブラウ



図- 1: フォーム編集画面例



図- 3: Google サイト編集画面例

	A	B	C
	タイムスタンプ	ユーザー名	作品ページのURL
1			
2	2012/05/31 13:06:19	@g hit-u.ac.jp	https://sites.google.com/a/g hit-u.ac.jp/
3	2012/05/31 13:06:20	@g hit-u.ac.jp	https://sites.google.com/a/g hit-u.ac.jp/
4	2012/05/31 13:06:40	@g hit-u.ac.jp	https://sites.google.com/a/g hit-u.ac.jp/
5	2012/05/31 13:07:19	@g hit-u.ac.jp	https://sites.google.com/a/g hit-u.ac.jp/
6	2012/05/31 13:07:32	@g hit-u.ac.jp	https://sites.google.com/a/g hit-u.ac.jp/
7	2012/05/31 13:08:30	@g hit-u.ac.jp	https://sites.google.com/a/g hit-u.ac.jp/
8	2012/05/31 13:08:55	@g hit-u.ac.jp	https://sites.google.com/a/g hit-u.ac.jp/
9	2012/05/31 13:09:59	@g hit-u.ac.jp	https://sites.google.com/a/g hit-u.ac.jp/
10	2012/05/31 13:09:04	@g hit-u.ac.jp	https://sites.google.com/a/g hit-u.ac.jp/
11	2012/05/31	@g hit-u.ac.jp	https://sites.google.com/a/g hit-u.ac.jp/

図- 2: フォーム入力データの記録例

ザを経由して利用可能なサービスとして提供されている。これらは、Google ドライブ（2012年6月現在、「ドキュメント」からの移行が順次進んでいる）の各ファイルを編集する機能としてアクセスできる。

本稿では、この Google Apps に含まれるいくつかのサービスを組み合わせ、Google Apps Script を用いて、情報基礎の授業においてプログラミング初学者向けのプログラミングの基礎の授業を行った。以下に、Google Apps に含まれるサービスのうち、本稿で述べる授業にて用いた機能の概要を説明する。

2.3.1 スプレッドシート・フォーム

複数のシートをもつスプレッドシートである。データは Google のサーバ上に保存され、複数のユーザが同時にシートの編集を行うことができる。編集、閲覧を行えるユーザは、後述する共有機能により制御が可能である。

また、スプレッドシートをデータの記録先として用いる Web 上のフォームを作成・編集、公開する機能が提供されている（図-1）。フォームから入力されたデータは、1回の入力がシートの行、フォームの各項目がシー

トの列に対応付けて記録される。フォームから入力されたデータがスプレッドシートに記録された例を図-2に示す。

スプレッドシートのセルへの変更は、編集履歴として記録される。編集履歴にはどのユーザが、いつ行ったものかという情報が全て記録され、また、この履歴により、スプレッドシートを任意の時点の状態に戻すことができる。

また、Microsoft Excel における VBA などに相当する機能として、Google Apps Script が用意されており、セルやシートの操作を行うことができる。Google Apps Script については 2.3.3 にて後述する。

2.3.2 サイト

Google ドライブとは別に、Web サイトを構築、公開する機能が「サイト」として提供されている。「サイト」は、Google が提供するサーバを用いて Web サイトを構築できるサービスであり、Web ページ管理機能、オンラインでの簡易な Web ページエディタなどを備える。ページ編集画面例を図-3に示す。本授業では用いなかったが、ページの HTML ソースコードを直接編集することもできる。なお、Google Apps 環境では、Web サイトの共有範囲として同一の Google Apps ドメインのアカウントを持つユーザを選択できることから、学生の間にのみ公開範囲を限定した Web サイトの作成が容易である。

サイトには、2.3.1 で述べたフォームに記入画面を埋め込むことができる。フォームから入力されたデータは、先に述べた通り、スプレッドシートを通して確認、編集することができる。

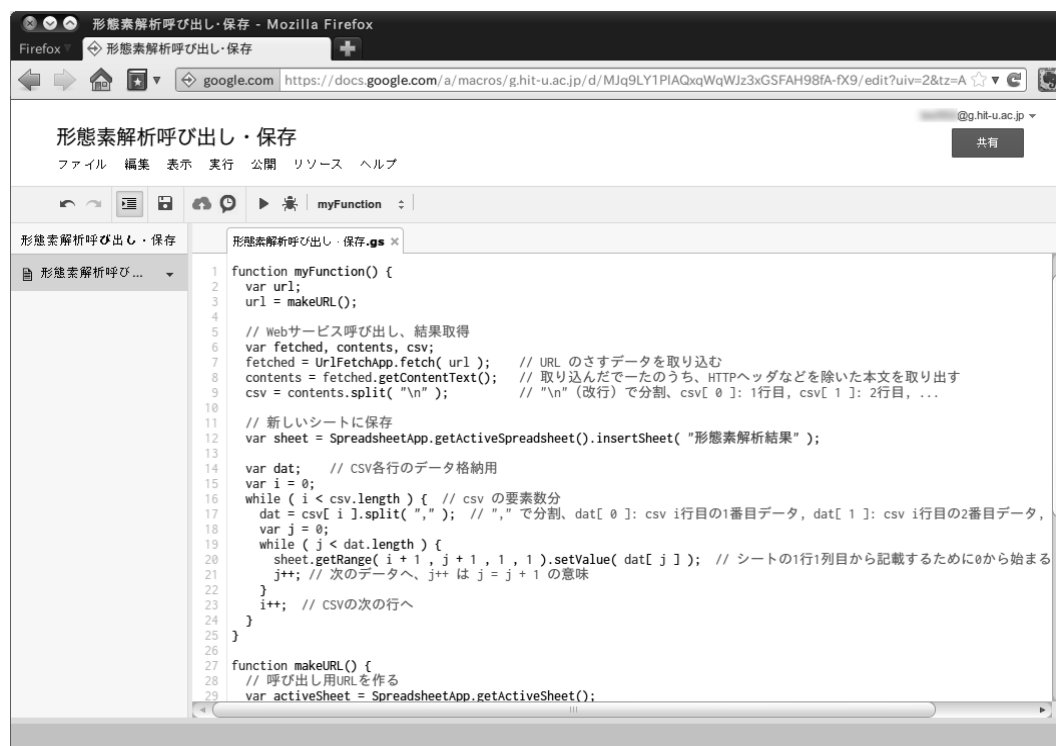


図- 4: Google Apps Script エディタ画面例

2.3.3 Google Apps Script

Google Apps Script (以下, AppsScript) は Google Appss 環境の一部として提供される, プログラミング, プログラム実行環境である². 図-4 に AppsScript エディタの画面例を示す.

以前より, Google はプログラム実行環境として PaaS である Google Apps Engine を提供していたが, プログラミングエディタなど統合環境を含む SaaS として, 2009 年から Google Apps Script が利用可能となった.

AppsScript にて記述したプログラムの実行は Google のサーバにてなされ, 手元の Web ブラウザを動作させている端末側にはスクリプト実行時の負荷はない. なお, 1 つのスクリプトで利用可能な計算時間の総量は, 特に明示されていないがある程度の制限がなされているようであり, 計算時間が長時間に渡るような処理を行わせるには適さない.

AppsScript の記述言語は JavaScript を基にしており, 文法や基本的なメソッドは JavaScript に準じる. AppsScript には GoogleApps 環境で提供される各種サービスを実行する機能が与えられている. 例えば, スプレッドシートではセルや複数のセルに渡る範囲の値の読み込みや書き込み, シートの追加, 削除といった操作などを行える. サイトでも, 各ページの記述内容の書き換え, ページの追加, 削除などの操作が行える. その

他, Google の翻訳機能呼び出しテキスト翻訳を行うといったことも可能である.

AppsScript 環境を用いて記述したプログラムはエディタ画面から起動できる他, 時刻を指定して実行する, スプレッドシートのメニューへ組み込みメニュー呼び出しにより実行する, サイトへガジェットとして組み込みサイト上のコンテンツとして実行することなどが可能である.

2.3.4 共有範囲の設定

以上に述べた各機能で作成したデータや Web サイトは, 一般公開できる³ほか, 他の Google アカウント保持者を選んで閲覧用に公開したり, 共同編集者として指定したりすることができる. とくに Google Apps for Education 環境など法人向けの Google Apps 環境では, ドキュメントや設置したサイトの共有範囲として, 同一の Google Apps ドメインのアカウントを持つユーザへの公開を選択できる. 同一の Google Apps ドメインを利用するユーザを公開範囲として選択すれば, 学生の間にのみ公開範囲を限定したドキュメントや Web サイトの作成を容易に行える.

²<https://developers.google.com/apps-script/>

³本学では, 学生用の Google Apps 環境において「一般公開」を選べないよう設定を行っている.

3 授業内容

3.1 概要

一橋大学における「情報基礎」授業のうち1クラス、商学部、経済学部、法学部、社会学部に属する学部生約40人を対象とし、Google Apps 環境を用いた学生各個人のサイトコンテンツの作成、アンケートフォームの設置、収集した回答の簡単な解析を2011年度より実施した。

本稿で述べる授業では、コンピュータおよびコンピュータネットワークに関する基礎的な学習を通し、情報の社会がどのような技術によって形作られているのかを理解することを大きな目的とし、Webに関連したインターネット技術の概観を中心とした学習を行った。

Information and Communication Technology (ICT) の Communication については、授業開始時のアンケートなどから、学生が常日頃から接している部分であることが分かっているため、その仕組みを説明し、機能を理解するためのいくつかの演習を中心に行った。一方、Communication ではない部分、コンピュータを計算機として用いることにより、人間に代わって情報を処理することができることについては、日常生活では意識することがない部分であることから、その仕組みを構築する手法があることを実際に体験する学習に比重をおくこととした。

これを踏まえ、プログラミングに関する学習では、変数の概念や制御構造など基本的なプログラム記述に必要な概念を学びつつ、「Webを通して情報を集め、集めた情報に対し、人手では手間がかかるとされる処理を計算機を用いて行うこと」を作成するプログラムの目標として設定した。そのために、学生自身のサイトを作成しそこに入力フォームを設けて情報を集め、集めた情報を AppsScript により外部の Web サービスと連携して解析し、結果をスプレッドシートに保存する、という流れをとった。なお、単純な計算による解析であればスプレッドシート内で行うことが合理的であることから、スプレッドシートで行うには適さない、自由記述文章の形態素解析を Web サービスと連携した解析としてとり上げた。

3.2 フォームを用いたデータの収集

まず、Google Apps のサイト機能を用い、各自の Web サイトを作成した。自由にサイトを作ることによってサイト機能の操作に慣れつつ、写真と短文の組み合わせを1つの作品単位とし、自分の作品の写真を他の授業参加者の写真と入れ替えた作品を複数作ることを通して、自

分以外のサイトを見て回る機会を設けた。この際に、自身のサイトの URL を教員が設けたフォームに入力、入力した URL がスプレッドシートを通して一覧できるよう、フォーム入力内容が記録されるスプレッドシートを受講者全員で共有するよう設定した。共有したシートの例が図-2である。

続いて、感想を記入するフォームを作成、サイト内の写真と短文の組み合わせを掲載したページ内に設置した。解析対象として数個の感想が集まるよう、相互に感想を書くためのグループ分けを、フォームを通してサイトの URL を集めたシートを学生が各自の端末上で表示している間に、各自のサイト URL の横に番号を記入することにより行った。フォームやスプレッドシートに他のユーザが記入した内容は、ほぼリアルタイムに各自が閲覧しているスプレッドシートに反映される。

3.3 AppsScript での処理目標

アンケートとしてフォームを用いて収集した自由記述文書を、外部の Web サービスと連携して形態素解析し、スプレッドシートに記録することを最終的な目標とした。例えば Yahoo!Japan が提供するサービス⁴など、一般の Web サービスとして用意されている形態素解析サービスでは、利用のために登録が必要である。学生に授業での1回の利用のために登録を要求することは現実的ではないと考え、文を渡すと形態素解析結果を返す Web インタフェースを持つサービスを独自に用意した。サービスは REST 形式を模し、HTTP の GET メソッドで要求を受け取り、結果を CSV で返すものとした。XML を利用する方法に比べ、交換するデータの直感的な理解を容易にするためである。初学者向けではなく、より高度な利用を目指すのであれば、AppsScript に用意されている SOAP 関連のサービス⁵を用いることも可能であろう。

AppsScript を用いた Web サービスの利用に先立ち、Google マップ⁶など URL 中に引数を記すことで、その挙動の変化がわかりやすいサービスを用い、URL に GET メソッドで処理されるパラメータを付加し、パラメータにより Web サービスが応答として提示する情報を変化させられることを学習した。また、Web から情報を取得しスプレッドシートに格納する手法の学習を目的とし、AppsScript を用いて東京電力が提供する電力使用量 CSV⁷をスプレッドシートに取り込む方法、データを確認する演習を行った。

⁴<http://developer.yahoo.co.jp/webapi/jlp/ma/v1/parse.html>

⁵https://developers.google.com/apps-script/service_soap

⁶<https://maps.google.co.jp/>

⁷<http://www.tepco.co.jp/forecast/html/images/juyo-2012.csv>



図- 5: AppsScript エディタでの入力候補表示例

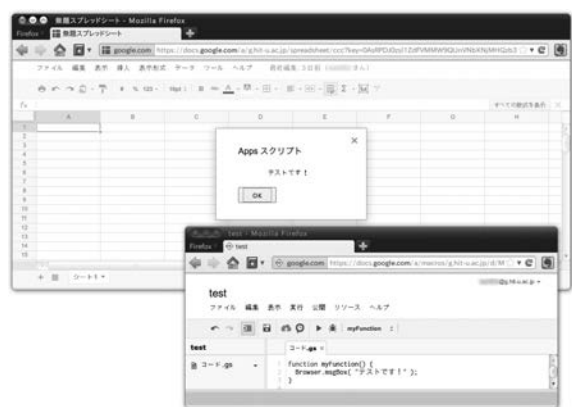


図- 6: AppsScript 実行例

3.4 AppsScript でのプログラミング

AppsScript は GoogleApps のスプレッドシートメニューから記述を行うことから、スプレッドシートと結びつつ、プログラミングの授業内容を行った。

3.4.1 メッセージボックスの表示

まず、人手によるスプレッドシート操作とは異なっていることができることを学習するため、メッセージボックスを表示するコードの記述を行った。Web ブラウザ上で JavaScript を利用するには `window.alert()` を用いるが、AppsScript では `Browser.msgBox()` を用いる。なお、AppsScript エディタでオブジェクト名、メソッド名などを入力する際には、入力途中でそれぞれの候補が表示される (図-5)。

エディタからは実行する関数を指定してコードを実行することができ、処理結果はエディタとは独立して AppsScript エディタを呼び出したスプレッドシート上に表示される (図-6)。

3.4.2 変数・配列の利用

AppsScript エディタをスプレッドシートのメニューから呼び出して実行することを踏まえ、変数、配列と

いったプログラミング上の概念を、スプレッドシートと結びつけて説明を行った。セルが言語上の変数に相当し、シートの範囲指定が配列に相当するものとして、AppsScript からセルや範囲の値の読み出し、値の設定を行いつつ、変数への値の設定と値の利用を並行して演習を行った。

3.4.3 制御構造

制御構造のうち、繰り返し処理については、変数などと同様にスプレッドシートと結びつけ、セルを連続して読み出す、連続して値を設定するなどの処理を記述することにより、視覚的に理解が行えるよう演習を行った。

分岐はスプレッドシート上で直接的な表現を行うことが難しいと考え、スプレッドシートとは結び付けず、先の `Browser.msgBox()` と入力を受ける `Browser.inputBox()` を組み合わせ、入力値に対して出力値を変化させるコードの記述をとりあげた。

3.4.4 外部サービスの呼び出しと結果の格納

先に述べたように、Web 上の単純な CSV ファイルの取得とスプレッドシートへの格納を行った後に、形態素解析を行う Web サービスの呼び出しと得られたデータの格納を行った。

Web サービスの呼び出しでは、フォームに入力された内容を記録したスプレッドシートから、解析対象としたいセルの内容を取得、形態素解析 Web サービス呼び出し用の URL を取得した内容を引数として生成し、URL が示す内容の取得を行った。AppsScript において URL を指定して URL の示す内容を取得するためには、`UrlFetch Services`⁸ を用いた。

なお、取得したデータを一度配列に格納し配列をスプレッドシートに転記する方が効率的だが、スプレッドシートのセルを直接操作するイメージに近いほうが直感的に理解しやすいと考え、CSV から取得した値を1つずつセルに設定するサンプルコードを受講者に示した。

4 考察

4.1 本実践におけるプログラミング学習の位置づけ

3.1 にて述べたように、本稿で取り上げた授業におけるプログラミングの学習では、Web などをとおしたコミュニケーションの裏に隠れ日常の生活では意識する

⁸https://developers.google.com/apps-script/service_urlfetch

機会が少ない「計算機が（人間に代わって）情報を処理すること」について、実際に体験することに比重をおいた。

Google Apps のスプレッドシートを用いると、複数人が同時に同じシートを編集することができる。たとえば、授業中に 40 人で共同編集を行うと、眺めている間にセルに値が記入されていく。本稿で述べた形態素解析を行いシートに記録する AppsScript のコードにより、自ら書いたコードが、文書中にあらわれる単語の一覧を別の共同編集者に入力してもらっているかのように、動作することを体験することが可能であった。

このように、ここで取り上げたプログラミングでは、計算機上でプログラミングを行うことにより何ができるかを主たる目標とし、たとえば浮動小数点数の誤差など、計算機の内部に関する学習は取りあげなかった。

4.2 Web サイト構築に関する学習

Web 上で動作するプログラムの作成を行う学習には、Web サーバに HTML ファイルをおき、クライアントとなる PC からアクセスする、サーバ・クライアントモデルを体験する側面もある。つまり、サーバに ftp を用いてファイルを置き、アクセスすることでファイルの内容が表示されたり、プログラムが実行されることを学習の目的とすることも考えられる。

しかしながら、近年、多くの Web サイトは Web ベースのインタフェースを持つ Contents Management System (CMS) を用いて管理されており、ftp などを利用して Web サイトを管理する必要性は多くはない。また、Web サーバを構築する時も、そもそも CMS が標準でインストールされていることが増えてきている。

これらのことから、本授業では、文科系学部における学部初年度で「ファイルを転送する」ことなどを実習する重要性は高くないと判断し、SaaS として提供される Web サイトやプログラミング環境を利用しつつ、Web サイトの仕組みについてはサーバ・クライアント方式として P2P 方式などと絡めながら、サービス提供におけるアーキテクチャとしてとりあげた。

なお、Google サイトでは、HTML の直接編集が行える他、画像は Web ブラウザ経由ではあるが「アップロード」を行う必要があり、これらを入り口に Web サイト構築の詳細を説明することも可能であろう。

4.3 AppsScript を用いたプログラミング授業の可能性

AppsScript は、コード記述を行うエディタ以外にも、デバッガ、プロファイラを備えるなど、開発環境が順

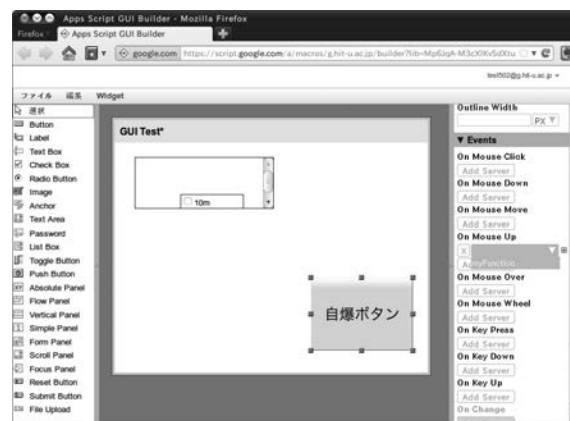


図-7: AppsScript ユーザーインタフェース構築例

次整備されている。また、Google Apps 以外の各種の Google API を利用するための拡張など、AppsScript で利用することが可能な機能も随時追加されている。

例えば、まだ正式な機能とはなっていないが、GUI 画面でダイアログボタンの配置、ボタン操作時などのイベントに対する実行コード記述を行える「ユーザーインタフェースを構築」機能も持つ（図-7）。これにより、2.1.2 に述べたような GUI に基づくイベント駆動型のプログラミングについて授業を行うことも可能である。

Google Apps の機能を多用して授業を行うことにより、学習結果まで Google Apps に依存してしまはないかという懸念もある。この点、「サイト」機能は比較的低機能の CMS であり、むしろ現在の高機能な CMS を利用するための導入となるであろう。また、AppsScript は JavaScript をベースにしているため、汎用性が高いと考えられる。用いるオブジェクト名称などがブラウザベースで動作する JavaScript とは異なるが、同様の機能を異なるオブジェクト名称などで記述することになるのは、言語というより利用するフレームワークに依存する部分であり、AppsScript に限らずプログラミングの授業では問題となる事項である。このように、学習環境に過度に依存せず応用が利くよう、授業を組み立てることに留意することは、SaaS を用いるかによらず常に必要であろう。

4.4 プログラミング学習環境に SaaS を用いる利点と欠点

プログラミングの学習環境に SaaS を用いる利点として、まず、2.2 にも述べたが、授業ごとの環境構築、環境設定が不要であることが挙げられる。サーバサイドの環境構築も不要である。

端末環境も Web ブラウザが動作すればよく、例えば、若干の不便は伴うものの、タブレット型端末でもコード

の記述と記述したコードの実行が可能である。自習のための環境整備が不要となることから、授業時間外にプログラミングの実習を行うことも相対的に容易となる。

Google Apps 環境では、Web ブラウザへの依存性も十分に注意がなされているように見受けられた。本稿で述べた授業において挙動の違いが生じたのは、Internet Explorer を用いた際の iframe タグの扱い、同じく Internet Explorer を用いてテキスト入力を行う際のバックスラッシュの扱いのみであった。

その他にも、いくつか懸念事項が挙げられる。たとえば、サーバサイドで動作することから、授業でサービスへ同時アクセスを行った際などの安定性が挙げられる。現在までのところ、2011 年度前期には AppScript エディタが不安定な状態になることがあったが、Web ブラウザにてリロードを行うことにより解決可能であった。2011 年後期以降同様のトラブルは発生していない。本授業を実施した学生がプログラミング初学者でありほかの環境との比較ができないであろうことから、プログラミング環境としての AppScript の利用しやすさや安定性について調査を行っていないが、スプレッドシートについて Microsoft Excel と比較した利用しやすさや安定性について調査を行ったところ、動作が不安定であるという選択肢を選んだのは 40 名中 1 名であった。

また、これ以上に大きな事項としては、サービスの一貫性が挙げられる。SaaS として提供されている以上、SaaS の利点を活かすべく、サービスの内容は随時拡充、変更される。例えば、Google ドキュメント画面デザインは 2011 年 8 月にうちに水色を基調としたものからモノクロ基調のものに変更され、AppScript にプロジェクトの概念が導入されるなどの変化があった。

とはいえ、Google Apps は、サービスの利用者が多数いることなどから、必要な機能が突然提供されなくなる可能性は低く、十分なアナウンスの後に実施されると期待できる。そのため、たとえば授業資料を用意してから授業を行うまでの間に大規模な見た目や機能の変更が行われるようなことはなく、大きな問題にはならないが、常に注意は必要であろう。また、本学では Google Apps for Education を利用する契約を行っており、大規模なインタフェイス変更などの導入タイミングは、Google Apps 管理者がコントロールすることができた。その他、利用規約の変更などにも、常に注意を払う必要がある。

5 おわりに

本稿では、SaaS として提供されている Google Apps の各サービスと、その一部として提供されており Google のサーバサイドでコードの実行もなされる Google Apps

Script を組み合わせ、プログラミング初学者向けの授業を実践し、その報告を行った。

SaaS をプログラミング環境として用いることにより、授業時に環境の設定などを行う必要がなく、また、学習者も演習室以外を利用しての自習が容易となる。2011 年度より開始し、半期で 40 人、計約 120 人が受講したが、これまでのところ大きな問題はない。また、Google Apps のサービスであるスプレッドシートやサイト、サイトへ埋め込めるフォームなどを連携させ、Web 上でのデータ収集から解析までの一連の流れとして学習をすすめることができた。

現在、本学の「情報基礎」におけるプログラミング初学者向けの授業においてのみ Google Apps Script を利用しているが、今後、「情報基礎」に続けて開講する「計算機概論」においても、プログラミング環境として SaaS 環境を利用することが可能か、発展の検討を進める予定である。

参考文献

- [1] 兼宗, 長: “文科系大学におけるサーバサイドプログラミング授業の試み”, 情報処理学会研究報告コンピュータと教育研究会, pp. 141–148 (2006).
- [2] M. Đanić, D. Radošević and T. Orehovalčki: “Evaluation of student programming assignments in on-line environments”, Proc. of 22nd Central European Conference on Information and Intelligent Systems, pp. 111–116 (2011).
- [3] Y. Wang and B. Jin: “The application of saas model in network education – take google apps for example”, Proc. of 2nd International Conference on Education Technology and Compute, Vol. 4, pp. 191–194 (2010).
- [4] A. Malinowski and B. M. Wilamowski: “Web-based c++ compiler”, Proc. of the American Society for Engineering Education 2000 Annual Conference, pp. 1–7 (2000).
- [5] 三浦, 國藤: “Subversion を用いた学習管理システムの構築とプログラミング初学者への適用”, 情報教育シンポジウム予稿集, pp. 155–162 (2006).