

(研究ノート)

スマートフォンアプリ作成演習の試み

Trial of Programming Education of 'Making Smart Phone Applications'

高尾 哲康

Takao Tetsuyasu

「義務教育段階からのプログラミング教育等の IT 教育を推進する」と 2013 年 6 月発表の政府の成長戦略[1]に盛り込まれた。これを受けて、文部科学省において 2020 年から初等中等教育段階からのプログラミング教育を必須化することが検討された。この背景として、IT 人材不足が影響しており、これは国内だけにとどまっていないことがある。本学においても、特に経営情報専攻の教育課程において、キャリア面の充実のためにコンピュータリテラシー教育の他にプログラミング教育の必要性も大きいと考えられる。今回、1 年次の必須科目である経営情報概論において、プログラミング教育の一環としてスマートフォンアプリの作成を試みた。本稿では、この試行について、学生のアプリ作成進捗状況やアンケート結果などをもとに今後のプログラミング教育のあり方について検討した結果を報告する。

キーワード：アプリ、スマートフォン、プログラミング教育、ソフトウェア開発環境

1. はじめに

2016 年 4 月、文部科学省は 2020 年からの実施を目標に小学校におけるプログラミング教育を必修化すると発表した[2]。その目的はどのような業種・職種においても普遍的に求められる能力として「プログラミング思考」があり、これを育成することであるとしている。現在、パソコンやスマートフォン、タブレットなどの情報機器が非常に身近になってきており、生活に欠かせないものとなってきている。また情報技術もますます進歩していくと予想されている。その一方で、情報システムの開発・運用・保守を行なう技術者をはじめとする IT 関連分野の人材不足が現実となっている。2016 年 6 月に経済産業省が発表した、「IT 人材の最新動向と将来推計に関する調査結果」[3]によると、2020 年に 36.9 万人、2030 年には 78.9 万人の IT 人材が不足すると予測している。今後、「第 4 次産業革命」とも言われる、人工知能 (AI) や IoT (Internet of Things、モノのインターネット) などが将来の社会のあり方を大きく変えていくと予想されている。

プログラミング教育には、アルゴリズム・論理的思考や創造性・問題解決能力の育成とともに、

コンピュータを動かすために必要なコーディング技法（プログラミング言語を用いて、問題を解くための手順であるアルゴリズムを記述する技法）を学ぶことが含まれている。今後、実施にあたり具体的な施策が固まるとみられる。

今回、本学の現代社会学部1年後期の必須科目である経営情報概論において、プログラミング教育の一環としてスマートフォンアプリの作成を試みた。この科目は専攻教員持ち回りによるオムニバス形式となっており、第4回目の講義にて実施した。授業後に実施した学生へのアンケート結果などから今後のプログラミング教育のあり方を検討した。

2. スマートフォンアプリについて

本学学生のスマートフォン（スマホ）普及率はほぼ100%であり、機種はiPhone系とAndroid系でほとんどとなっている。しかし、パソコンについては、入学者に対して斡旋PCプログラムが用意されているが、留学生を中心に所有しないことが多く、所有率100%には到っていないのが現状である。本学の情報センターではノートPCの貸し出し制度を用意しているので、必要な学生は一時的に借用することができる。また、コンピュータ教室の運営については大学で設置する据置型PCを縮小していく方向にあり、コンピュータリテラシー教育はノートPC中心として行なわれている。このような現状の中で、プログラミング教育を少しでも成果あるように実施するためには、スマホやタブレット向けのアプリ作成が中心となる。現実問題としてはスマホだけでコーディング作業を伴うプログラム開発にはハードルが高い。現在主流となっているPCによるスマホアプリ開発をするにしても、ソフトウェア開発環境（Visual Studio、Eclipse、Android Studio、Xcodeなど）のインストール・設定やその使用法を覚えるだけでもかなりの時間が必要となる。授業の1コマ分（90分）の時間内でプログラミング未経験者がアプリの動作確認まで行なえるようにするためには余分な作業をなるべく減らし、プログラミング作業のみに専念できる別の手段が望まれる。

3. クラウド型開発環境 Monaca[4]

コーディング作業を含めてすべてのプログラミング作業をWebブラウザから行なえる、クラウド型のソフトウェア開発環境（<http://ja.monaca.io/>）を利用した。準備するものは、ウェブブラウザGoogle Chromeが動作するPCとスマホ（iPhone、Android）またはタブレット（iPadなど）の2つだけである。最低限の利用には、無料のBasicプラン（作成可能なプロジェクト3つまで）に登録してアカウントを作成する必要がある。アカウント作成にはメールアドレスとパスワードを入力することで登録完了となる。あとはいい例題を用意することだが、せっかくのスマホアプリなので、スマホのハードウェアのGPS機能を活用したアプリとした。GPS機能を使えば現在の位置の経度・緯度データが得られ、Google Maps API（Application Program Interface）を利用して経度・緯度データから住所情報を求めて表示させるアプリとした。プログラム入力量もできるだけ少な



図 1. 現在位置情報取得アプリの完成画面



図 2. 実機のスマホにて現在位置表示アプリを起動

くした結果、合計 30 行、半角文字で 1,090 文字（空白文字を含む）となった（コメント行 7 行分で、半角文字換算で 220 文字分を含む）。図 1 にそのプログラム（赤枠内部分を入力する）を、図 2 にスマホで実行させた画面を示す[5]。

わずかな量のプログラム（プログラミング言語は JavaScript と HTML）ではあるが、プログラミング言語の構文（文法）と意味を理解し、コメント行があるとはいえ、プログラム内での動作の意味を理解しようとするだけでかなりの時間が必要となる。今回はプログラミング言語に関する説明はせず、「経営情報概論 2016」テキスト[5]の説明にしたがってクラウド開発環境のセットアップとプログラム動作についての簡単な説明のみにした。プログラム開発という作業がどのようなものであるかを体験してもらうために授業時間の大部分を入力作業にあてることにした。これは授業における学生からの質問数の増加や学習意欲・学習効率を高めることをめざし、授業における一体感や競争原理を打ち出す「アクティブラーニング」にはほど遠く、きわめて孤独な作業となる。プログラミング教育は初等教育においてカリキュラムに取り込もうとしている段階であり、また学生間のレベルの差も大きいのが現状である。そのため、「アクティブラーニング」は教育方法が十分成熟した教育分野で有効であり、プログラミング教育分野ではまだなじめていないのではないかと考える。操作方法や入力方法などに関する質問は個別に受け付けているが、学生固有の環境で発生する問題のことも多く、他の学生のためになると思われる、よくある間違いや注意点のみを教室内の全学生にフィードバックすることになる。

4. 実施結果

アプリの授業時間内での作成状況および学生（94 名）へのアンケート結果を図 3 に示す。約半数の学生がアプリ完成または完成に近いという結果となった。完成までの過程では半数以上が難しいと回答している。初めてアプリを作成したと自由記述に明記する学生も多く、難しいと感じた原因の大部分は入力が大変であると回答している。入力する文字の全角・半角の区別（アルファベットや数字、記号は必ず半角文字）、空白文字であっても全角空白文字を使わないこと、入力する文字列の意味がわからないために入力作業そのものが苦痛と感ずることになるのであろう。全角文字がプログラムのどの部分で使えるかはプログラミング言語の構文（文法）を理解しないと判断できない。そうであっても、語彙（単語）と文法で構成される言語というものは日本語や英語などの自然言語、プログラミング言語などの人工言語を問わず、習得に時間がかかるものである。

クラウド型開発環境サイトへの登録はメールアドレスとパスワードのみではあるが、一部の学生ではこの登録作業でつまずいたと回答している。これはインターネット上でさまざまなサイトでのサービスを受ける場合に必要な作業のひとつであり、どのサイトであっても作業量に大きな違いはない。プログラミング以前のコンピュータリテラシーの問題であると考えられる。

プログラミングは人間が行なう知的作業の一種なので、職業として選択できるか、今後も長く続けられるかについては、作業そのものが楽しめるかどうかが重要である。約半数はふつうと回答しているが、楽しめたと回答した学生も 1/3 いた。これはプログラミング教育の今後に十分期

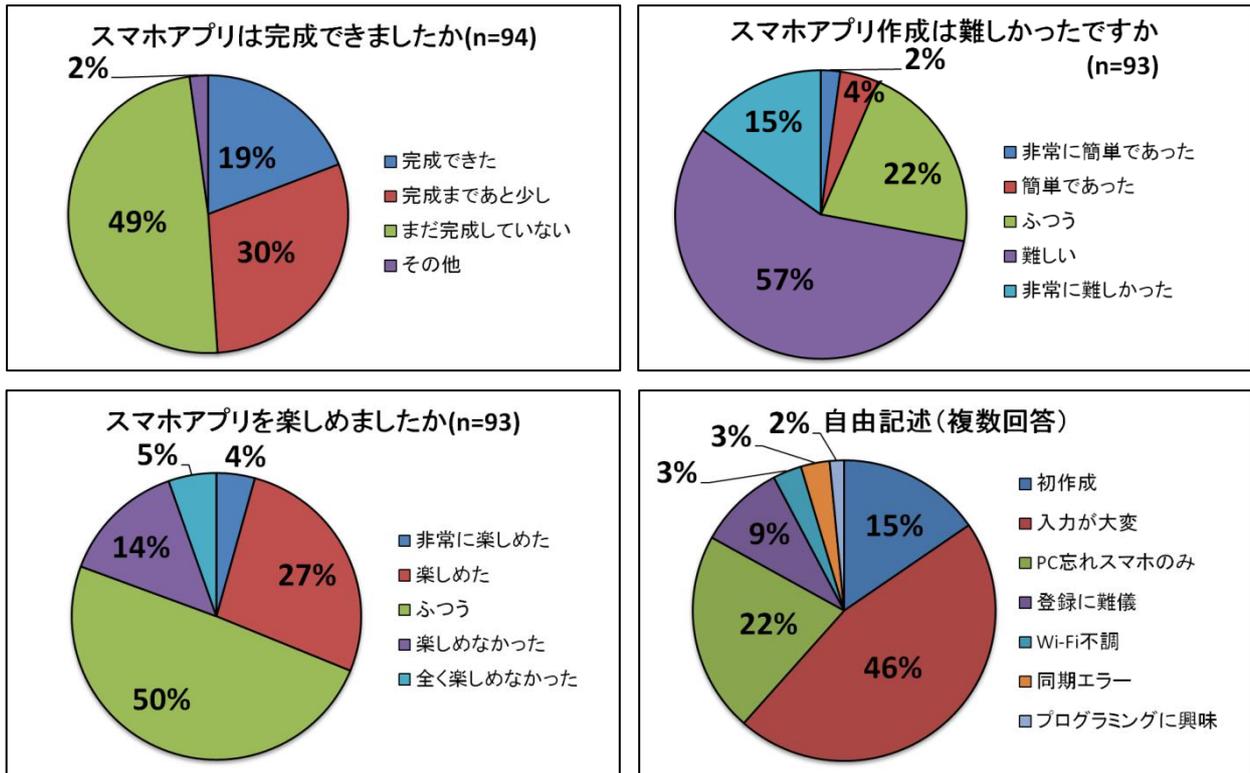


図3. アプリ作成状況およびアンケート結果

待できる割合であると考える。

アプリを完成させたかほぼ完成させた学生(46名)に注目した回答結果を図4に示す。「非常に簡単であった」と「簡単であった」の割合は多くなり、「非常に難しかった」の割合は減るが、それでも難しいと回答する学生が半数以上である。図3と比較して、アプリを完成できたかどうかにかかわらず難易度について似たような割合になったことは面白い現象である。また、楽しめたかどうかについても、難易度の場合とほぼ同様の結果となっており、約半数はふつうであると回答している。楽しめたと回答した学生の割合が増加した一方で、ほぼ完成できたにもかかわらず、楽しめなかったとした回答も1割以上ある。達成感よりはプログラムの入力作業のつらさ、困難さが印象強かったものと思われる。入力作業が一段落してもその後の完成まではエラーとの戦いであり、1文字の間違いで動かないことが多いことも影響したと思われる。また、テキストを見ながら意味も分からずに苦勞して入力して完成できたとしても自分にとって得られるものはあまりないと判断したと考えられる。誰でも初めてのプログラムは教科書に書いてある通りに自分の手で入力して動作させたものであると説明してはいるが、なかなか実感してもらえていないようである。完成に到った学生は(たまたま入力ミスがなくて)幸運であったか、あるいは必ず完成させてやるという(テキストと1文字1文字チェックする)しぶとさを発揮したのであろう。

MITで開発されたScratch[6]という初心者向けのプログラミング言語学習環境がある。ブロックを組み合わせて積み上げるような視覚的な感覚で学べ、プログラミングを楽しめるようになっている。しかし、十分実用的なプログラムを開発するにはそれなりの構文およびデータ構造を含めた文法をもつ言語が必要となる。なるべく早くプログラミングを楽しめる段階に到達できるか

どうか今後もアプリ開発を続けていきたいかどうかにかかっている。今回の試行では授業時間内に完成まで到達できることを最優先とした。

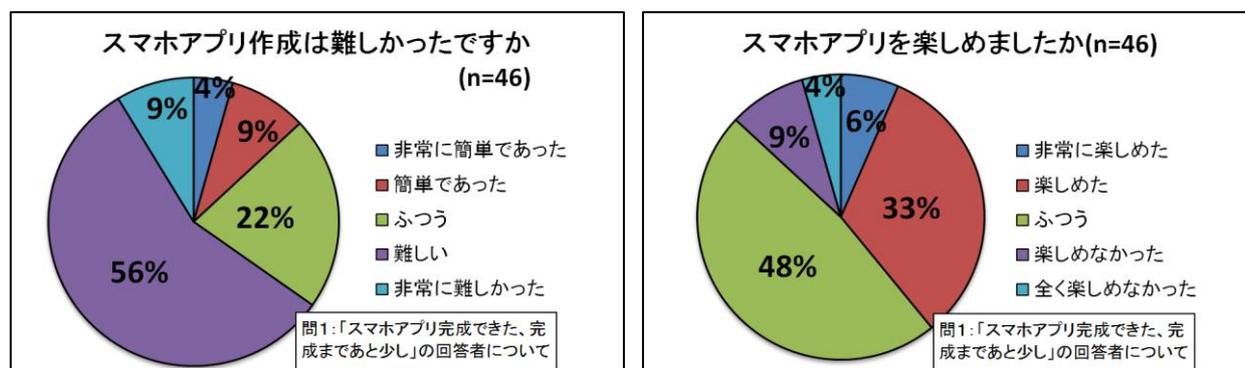


図4. アプリほぼ完成に到った学生のアンケート結果

5. まとめ

初心者向けプログラミング教育の一試行としてスマホアプリ作成演習を実施し、その結果と考察を報告した。今後は、よりよい教育内容・方法について検討を重ねていく予定である。

参考文献

- [1]政府の成長戦略（素案）：<http://www.kantei.go.jp/jp/singi/keizaisaisei/skkkaigi/dai11/siryou1-1.pdf>
- [2]「小学校段階における論理的思考力や創造性、問題解決能力等の育成とプログラミング教育に関する有識者会議」文部科学省：http://www.mext.go.jp/a_menu/shotou/zyouhou/detail/1375607.htm
- [3]「IT人材の最新動向と将来推計に関する調査結果」経済産業省：
<http://www.meti.go.jp/press/2016/06/20160610002/20160610002.pdf>
- [4]「Monaca で学ぶはじめてのプログラミング ～モバイルアプリ入門編～」, アシアル株式会社、生形可奈子、岡本雄樹、アシアル株式会社、ISBN 978-4-86584-099-5、(2016)
- [5]「第4回 情報と企業経営：「情報システムのツウになろう2」、高尾哲康、pp.19-24、「経営情報概論 2016」、富山国際大学現代社会学部経営情報専攻、(2016)
- [6]Scratch：<https://scratch.mit.edu/>