

<研究ノート>

プログラミング教育における学生の学習効果

山島 一浩*

Teaching Methods that Target Learning Effect of the Students in Programming Education

Kazuhiro YAMASHIMA *

抄 録

学生が、授業を通じて、自己教育力をつけた記録である。授業で、教材の中で一つの工夫を見つけて、高校の先生方の勉強会で、新たに教える工夫を加えて教わりながら完成度を高めていった。それを小学生、高校生に公開講座で教えてきた。そして、最後に、この活動を日本教育工学会の全国大会で発表した。この一連の活動を授業から説明し、全体を記録したものである。

キーワード：大学教育、情報処理教育、自己教育力、プログラミング教育、MIT App Inventor

1. はじめに

時代によって、個人が学ぶ内容が変わり、自己教育力が試されることがままある。それが個人の教育力が試されるのであれば、どの対象のものであろうが、自己教育力を伸ばすことが重要であろう。

プログラミングは、多様な文字と数値と式を使って、人間の使ってきたものを表してきたが、その数を見ても多くあり、学生が覚えることも、多くの時間を経てプログラミング言語が移ろいでいくなかで、移り変わりの多い文書ともいえる。

大学の研究は、特定の物事について、実験、観察、調査などを通して調べて、人間の

知識を集めて考察し、その物事についての事実を深く追求する一連の過程を学ぶとすればどうか。大学における自己教育とは、どうあるべきか。それを問いたい。

自己教育力とは、基本的に「学び方」の学習の方向性を持つのだが、それ以上に、生涯にわたって学び続けるための、学習内容の社会的価値や有用性、おもしろさなどを学ぶ必要が要求される。なぜなら、基本的に何を学ぶか、また学ばないかを決定するのは個人の自由意志によるからだ。

自己教育は、大学教育では、大学教員が作成したカリキュラムを受講生に押し付けがちな形で、学生自身がカリキュラムを作成していくという形にはなりにくい。「自らデザインし、

* 筑波学院大学経営情報学部、Tsukuba Gakuin University

自ら学ぶ」、「自分で学ぶ」という行為も教育の本来の姿として考える立場からは満足できない状況が多い。

これから紹介するのは、一学生が、授業を通じて、自己教育力をつけた記録である。

大学の授業から、子供たちや高校生に指導し、高校の先生方の研修会で成果を示し、それを研究会で成果を発表した。

2. ビジュアルブロックプログラミング

ビジュアルブロックプログラミング言語とは、プログラムをテキストで記述するのではなく、視覚的な操作でプログラミングするプログラミング言語である。視覚表現でプログラミングが可能で、空間上でテキストやグラフィックシンボルを配置することでプログラムが形成される。

ビジュアルブロックプログラミングには、MIT App Inventor 以前に、Squeak や Scratch があげられる。Squeak 環境には、非開発者向けプログラミング環境が実装されていて、Morph と呼ばれる可視化に適した機構を組み込んだオブジェクトに対し、その動き属性、色属性、形属性、振る舞い属性などを変化させる手続きを、パネル状のパーツの組み合わせで表現できる。この特徴から同スクリプトシステムは、プログラミング未経験者のほかに、キーボードの扱いに馴れていない低年齢層ユーザーにも容易に扱うことができるといわれている。Scratch は、2006年にミッチェル・レズニックによって開発されたプログラミング言語学習環境である。初心者が最初に正しい構文の書き方を覚えることなく簡単にアニメーションストーリー、ビデオゲーム、インタラクティブアートを製作できる。

MIT App Inventor は教育目的のツールであり、プログラミングの経験がない人でも、手軽に Android アプリの開発が始められる。パーツをつなぎ合わせることでアプリを作る

ので、コーディングの必要がなく、バグの入り込む余地が非常に少ないことも特徴に挙げられる。

共に学習のやる気を起こさせることを意図している。

3. 授業の紹介

3. 1 授業の目的

本学のシステム開発コースの専門基礎科目群コース科目で、「表現と伝達 B」という半期の科目として行っている2単位の授業がある。

授業の到達目標は、図1で示したように、プログラムを理解し、人にわかりやすく説明できることである。この過程で、情報スキルを伸ばし、コミュニケーション能力も伸ばすという内容のものである。プログラムするというこの意味や考え方の基本を学び、課題の解決過程で、プログラムの検証や改善策の立案する力を養わせる内容である。

教材として利用したのは、MIT App Inventor である。これは、Web 上で提供している Android 対応のアプリケーションを開発する

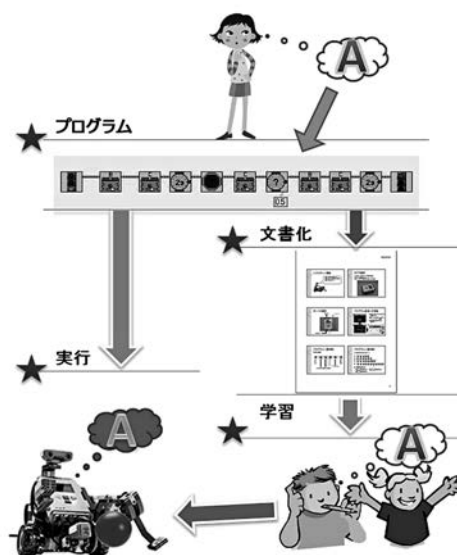


図1 授業のイメージ

ソフトウェアである。プログラミングに不慣れな人でも使いやすいように、グラフィカルやユーザ・インターフェースを使い、ビジュアル

ルオブジェクトをドラッグ・アンド・ドロップすることで Android 対応アプリを作成できるツールである。その入力には、図2のデザ

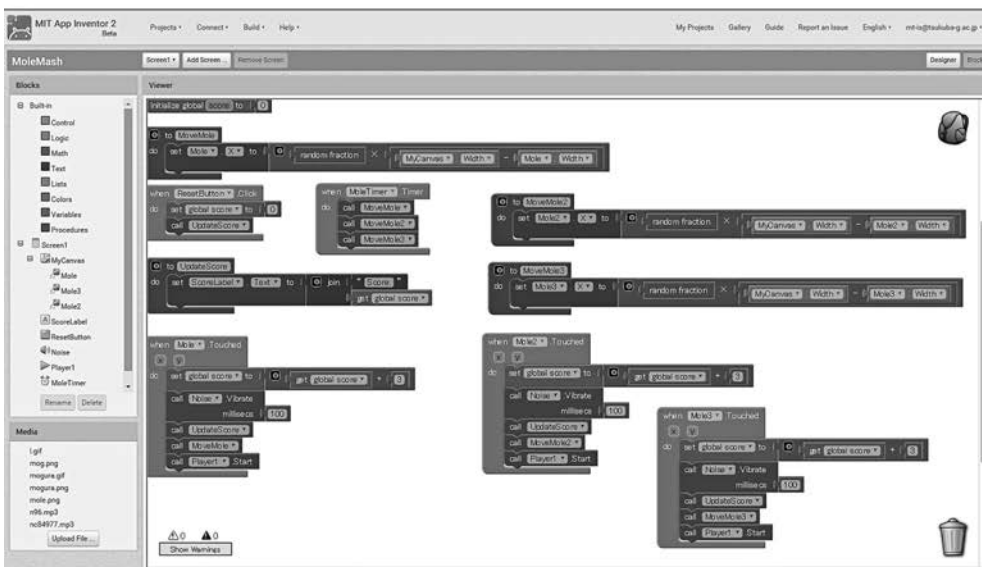
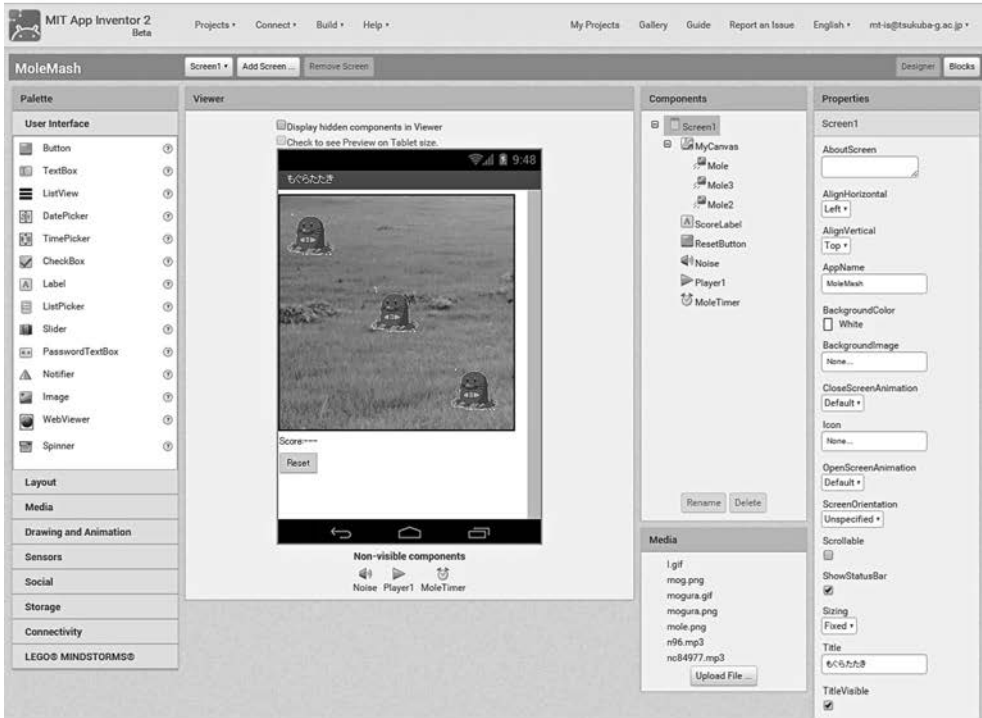


図2 MIT App Inventor のデザイン画面 (上) とプログラム画面 (下)

イン画面（上）で、Android の画面に表示される項目を登録し、その後でプログラム画面（下）でデザイン画面内に登録した項目に必要なプログラムの項目を、埋めて完成をさせる。

使用機器は、図3に示すレゴマインドストーム NXT と Nexus タブレットである。



図3 レゴマインドストーム NXT（上）と Nexus タブレット（下）

3. 2 授業の参加者の意識

この授業に参加した大学生が、表1に示す科目について、「好き」か「嫌い」かの意識調査を行った。数学は、プログラム課題で、数式を扱う際と、国語は、扱った内容をレポートにまとめる時にどうかを判断する際の指標とした。また、英語は、MIT App Inventor が、全てが英語メニューである点をどうするか判断として見ている。そして、プログラムは、プログラムの良し悪しを、図形認知は、MIT App Inventor がプログラムを図形の組み合わせであることに学習の重点をおいて見ている。

表1のような結果がでた。国語以外のものについては、全て嫌いが過半数をしめていた。

表1 受講した学生の意識調査

	数学	国語	英語	プログラム	図形認知
好き	37%	53%	16%	47%	26%
嫌い	63%	47%	84%	53%	74%

この意識調査に参加した学生全員が、苦勞した中で、最後の課題を提出して終わらせることができた。

3. 3 授業の成果

最終的な成果としては、遂行した課題の結果、学習した内容を PowerPoint に落とす。その際に留意点をそこに載せる。多くの学生が、苦勞した中で、図4のような PowerPoint を提出した。

4. 学生が参加し取り組んだ実践例

4. 1 つくばちびっ子博士

2013年夏に、子供たちへの MIT App Inventor を使った実践授業を始めた。担当した学生は、授業を受講した学生である。参加した子供は、10歳から14歳までであった。内容は、大学生が受けた内容で、提示物は、学生が提出したレポート（図4）である。それを学生が完成させた講義内容で、1時間の講義を、1日3講座開いた。各参加人数は、10人に割り当てた。

MIT App Inventor は、各説明が、英語表示であったが、プログラミング言語の構文を文字で埋め込む代わりに、構文構造に対応したブロックを直接操作して埋め込むという効果がみられた。参加した子供は、ビジュアルブロックプログラミングのプログラミングを短期間で修得できた。

図5は、左が、準備の様子と、真ん中は、指導した学生と受講生の様子である。右の写真は、作成したプログラムを Android 端末にダウンロードして操作し、ロボット相撲を楽しんでいる写真である。



図4 学生の提出した成果レポート



図5 学生が参加した講座

4. 2 つくば科学フェスティバル

2回目は11月に行われた、つくば科学フェスティバルである。市内の小中学校・高校・大学、研究機関など60団体が出展し、科学を楽しむための体験型イベントで、多くの子供たちが参加してくれた。

この場では、前回の資料が、大人向けの資料であったことから、反省をし、ロボットすもう大会にして、子供たちに難しいことを説明せずには挑んだ。この作戦は、上手く当てはまり、子供からどうしてそうなるのかとの質問に、印刷していた MIT App Inventor の画面を説明していた。

4. 3 高校の先生方の勉強会

3回目は、12月に行われた茨城県高等学校教育研究会工業部情報部会で発表をした。そ

の際に、高校の先生方から、有益な授業計画案を示してもらった。

4. 4 高校生向けの公開講座

4回目は大学で高校生向けに行った公開授業で指導をおこなった。この時間の構成は、高校の先生方のアドバイスを入れた内容になっている。

4. 5 学会での発表

学会での発表は、9月に岐阜大学で行われた日本教育工学会の第30回全国大会である。日本教育工学会が扱っている「教育工学」は、人文社会系と理工系、ならびに人間に関する学問分野を融合した学際的な学問である。

学生として、自分のやってきた事を発表した。堂々とした発表や質問にも整然と答える

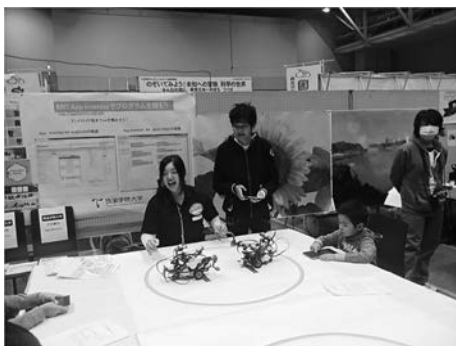


図6 学生が参加したイベント



図7 高校の先生方の教育研究会に参加

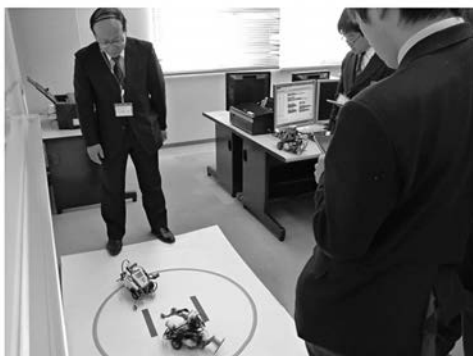


図8 高校生向けの公開講座



図9 学会での発表

さまは、実際に体験しただけに、実績を出していた。

5. 考察

人に何かを教えなければいけないとき、「効率のよい上手な教え方」を知っておけば効率がよい。学生が、授業を通じて、さまざまなことで、MIT App Inventor2を利用したことを自ら学会で発表してきた。

授業の流れとしては、ビジュアルブロックプログラミングでグラフィカルな要素やアイコン的な要素が提供され、それらをユーザが対話的に操作し、空間的にプログラムを構築する。子供から大学生に至るまで、操作の方法を教えるだけで、Android 端末やロボット制御が実現できてしまう。英語表示であったが、プログラミング言語の構文を文字で埋め

込む代わりに、構文構造に対応したブロックを直接操作して埋め込むという作業がみられた。

学生の学会発表では、子供たちの学習について、見たまま聞いたままを感覚で理解し、思ったままにプログラムのブロックを組み合わせで作成していたことを述べていた。これまでの自身の作成との比較や、やり方と内容を見て、これまでを理解しているように感じた。一人の学生の学習の流れではあるが、これらを元に、授業展開の参考にしていきたい。

6. まとめ

MIT App Inventor を使い、大学の授業から、子供たち、高校生、教員との教育活動を通じて、自らの自己教育を図った学生を紹介した。

参考文献

- 1) Mindstorms NXT、<http://mindstorms.lego.com/en-us/> (2015.9確認)
- 2) MIT App Inventor、<http://appinventor.mit.edu/> (2015.9確認)
- 3) Scratch、<http://scratch.mit.edu/> (2015.9確認)
- 4) 銭谷和磨、山島一浩、プログラミング教育を通して得られた自己教育、第30回全国大会日本教育工学会、岐阜大学、2014、pp.469-470