

## 浜松キャンパス学童保育におけるプログラミング教育

著者	安原 裕子
雑誌名	技術報告
巻	21
ページ	1-6
発行年	2016-03-30
出版者	静岡大学技術部
URL	<a href="http://doi.org/10.14945/00009491">http://doi.org/10.14945/00009491</a>

# 浜松キャンパス学童保育におけるプログラミング教育

安原 裕子

静岡大学 技術部 情報支援部門

## 1. はじめに

筆者は、2012年から本年度まで、静岡大学（以下本学）浜松キャンパスで夏季に実施されている学童保育において、参加する小学生を対象に“パソコン教室”と銘打たれた企画を継続して開催してきた。2013年からはプログラミングをメインとした構成を取っており、約2時間、小学1年生から6年生の児童が取り組む。

学年、一般知識、パソコンレベルなど、個人ごとに大きな振れ幅がある中で、教材となるツールの選択やテキスト作成、授業の進め方など、試行錯誤を重ねてきた。本稿では、これまでの実施内容やテキスト、アンケート結果などを報告するとともに、課題や今後の目標を整理する。

## 2. 経緯

### 2.1. 静岡大学浜松キャンパス学童保育

本学浜松キャンパスでは、夏季と春季の長期休暇期間に、小学生児童を対象とする学童保育が実施されている。2010年度頃から学内で本格的な議論がなされるようになり、2010年12月には教職員の意見交換会が開催され、当時筆者の子供が未就学児であったことから参加した。2011年3月に、8日間、試行として開催され、結果2011年7月から本格実施されるようになった。

小学1年生から6年生までの児童、約40名定員で、本学教職員児童だけでなく、地域の児童も申し込むことが可能である。大きな特徴としては、何よりも大学キャンパス内で実施されることであり、本学の教職員が協力し、体験授業や見学などが実施されている。

### 2.2. 企画提供の経緯

学童保育の開設について、前述のように筆者自身が一般の教職員に比べると身近なニュースとして捉えていたこともあり、本学内で実施されるなら、自身が何か協力できることはないだろうかと考えたのが、企画提供の第一歩であった。

学内の学童保育関係教職員やその他様々な方の協力を得、2012年の夏から企画提供を開始した。以後本年度まで、毎年夏に開催してきた。

## 3. 企画内容検討

### 3.1. 検討

筆者の得意とする分野や勤務場所の設備などから、コンピュータに触れてもらう、というのは容易に考えられたが、最近では家庭にも小学校にもパソコンを始めとする情報端末は当たり前のように存在する。ただパソコンに触り遊んでみるだけでは何も面白味がない。小学校では多少パソコンを用いて絵を描いたり表を作ったりしているようだが、それとは異なることがしたい。パソコンが、ただ文章を書いたり表を作ったりするだけのものではないことを知り、様々な可能性を体感し、創

造力をふくらませることはできないかと考えた。

時間は午前中の2時間強、休憩時間やアンケート記入時間を取ると、実質2時間弱である。この短い時間内で、子どもたちにコンピュータの楽しさを体感してもらうべく、これまでに実施してきた企画経緯を下表に示す。

表1 企画内容

年度	タイトル	参加者数	利用システム
2012	パソコンで音楽を作ってみよう♪	27	muphic <sup>[1]</sup>
2013	プログラミングで遊ぼう	26	プログラミング <sup>[2]</sup>
2014	SCRATCH であそぼう!	34	SCRATCH <sup>[3]</sup>
2015	プログラミング GOOD!	19	Code.org <sup>[4]</sup>

### 3.2. 1年目の反省とプログラミングの導入

1年目となった2012年夏は、パソコンにより親しむことを主目的として、muphicというソフトウェアを利用した。詳細は後述するが、十分に楽しめたものの、もう少し学術的要素を加えられないかと考えた。

これからの世代は、コンピュータの仕組みを学習するよりも、コンピュータやすでにある物をどのように使うかを考える、また、コンピュータを使って新たな何かを創造していく世代である。後述する文科省プログラミング<sup>[4]</sup>のサイトにはこのよう次のようにある。

「自分で何かを作れるということ。そのためには具体的な手順を考え、ひとつずつ実行していく必要があること。プログラムという人工言語は、そういったことに気づき、理解するためのツールとして非常に有効です。」

自分が何を作りたいのか、それを作するためには何が必要なのか、それをどのように使えばいいのかなどは、プログラミングに関わらず必要な思考であるが、プログラミングを学習することで養うことができる。また、うまくいかず何度も試行錯誤することで問題解決力も育むことができる。

最近では、Webブラウザを用いて直感的な操作で簡単にプログラミングできるシステムが多数存在することから、2013年からはそれらを利用することにした。

## 4. muphic

2012年はmuphicを情報学部の教育用端末にインストールして利用した。muphicは、仙台高専情報システム工学科の力武研究室が作成されたソフトウェアで、音楽を作る、物語を作る、という2つからなる。音楽を作る方は、色の線が音の高さになっており、動物スタンプを配置することで音楽を作ることができる。物語を作る方は、用意された様々なスタンプを好きに配置し、複数画面を作ることによって物語を作ることができる。

スタンプを配置するだけなので、低学年でも非常に簡単にできたが、絵を作るのに必死になってしまい、それだけで満足してしまう児童が多く見られた。最初は、音楽付きの物語を作るという目標を立てていたが、2時間で物語も音楽も両方作るのは無理があることがわかった。それでも3画面くらいの素晴らしい作品を作った児童がおり、最後に皆の前で発表をしてもらった。

初回としては、大変楽しく終えることができたが、もう少し学問的要素がほしいという反省があ

った。また、時間配分がうまくいかなかったのも反省点である。

## 5. プログラミン

2013 年は、文科省のプログラミンを利用した。これは Web ブラウザがあれば実行できるため、先の muphic より準備の手間を格段に省くことができた。

いくつもの”プログラミン”というアイコン状のツールを組み合わせることで、オブジェクトを動かしたりすることができる。前へ進むには 1、後ろへ進むには-1、右を向くには 90 度回転、というように、自分の思うように動かすにはどのツールをどのように使わなくてはいけないかを考える必要がある。動かす対象であるオブジェクトや背景も多数用意されており、発想次第で非常に面白いコンテンツを作ることができる。2 時間という枠内では終わることができない児童が多発した反面、低学年ではプログラミングの概念がまだ理解しにくく、単にオブジェクトが左右に動くだけのような単純な動きで終わってしまう傾向が見られた。

プログラミンは、かわいいアイコンを用いてプログラミングへの敷居を低くしており、かなり直感的に操作できる。オブジェクトに”プログラミン”を重ねて設定する方式であるため、オブジェクトを増やしたり複雑な処理をしたりしようとすると、画面がごちゃごちゃしてしまうのが欠点である。作成したプログラムを保存、公開することができる。保存すると URL が発行され、自宅からも作品を修正、閲覧することができるので好評だった。

## 6. SCRATCH

2014 年は、MIT メディアラボの SCRATCH を使用した。これは前年のプログラミンの元にもなっているシステムで、とても良質なコンテンツである。プログラミンと同様、Web ブラウザがあれば利用でき、複数のツールを組み合わせることでプログラミングを行う。ちなみにこの年から筆者が講師を務めている。

前年度、低学年のプログラミングに対する理解が及ばなかった反省があった。物を動かすときの考え方を理解してもらうため、当時子どもたちの間で流行っていた「あたりまえ体操」を取り入れて説明した。「右足出して、左足出すと、歩ける」というものだが、つまりオブジェクトを歩かせるには「歩く」という命令ではなく、「右足を出す」「左足を出す」「それを繰り返す」という命令をしなくてはいけない、プログラムとは命令の集まりなのだ、ということの説明した。また、SCRATCH 上で簡単な動きを一緒にやってみたことも前年度との違いである。

始めたときから薄々感じていたのだが、男女差を特に感じたのはこの年だった。男子は自由に好きなように次から次へと試してみるが、女子は非常に保守的で冒険をしない傾向が強い。特に高学年女子は、隣の児童と同じことしかしない。各自の空間の中で作るのだから、思い切ってやってみることができない。これは非常にもったいないことだと感じた。

## 7. Code.org

SCRATCH が良質だったため、2015 年も続けてもよかったが、女子にもう少しアプローチできないかと思い、Code.org を採用した。これは、SCRATCH やプログラミンよりも、もう少し大きな単位のツール、つまり「歩く」「投げる」がすでにコード化されたツールを使う。少しずつステップアップしていけるようにステージが用意されており、パズルを解く感覚でステージを進めていける

ので達成感を得ることができる。

今回は女子目線で、アナと雪の女王のステージを利用した。モニタにキャラクターが見えただけで、子どもたちの興味を惹けていることがよく見て取れた。20 ステージ用意されているので、前半は1ステージずつ一緒に進めていった。数ステージ進めると要領のいい児童は先に進みたがるので、後半は自由に進めさせ、講師は20 ステージまで、全員の進捗を確認しながら解説した。

20 ステージまで終えた児童は、新たな雪の結晶を描いてみたり、色を変えてみたり、または他のコードを使ってゲームを作ってみたりと、様々に楽しんでた。今回は高学年女子が少なかったが、角度を変えたりキャラクターを変えたりと、各自が思い思いに試すことができていた。

## 8. アンケート結果・感想より

以下に、各プログラムのアンケート結果や感想を抜粋して記す。一部、わかりやすいように表現や誤記を修正している。

感想を俯瞰して、指示されるのではなく、自分で考えて作り上げる体験をすることで、子どもたちが得たことはとても大きいのではないかとあらためて感じる。時間が足りなかったり、うまくできなかったり、一瞬のミスで削除してしまったりして悔しい思いをしている姿も見え、向上心が次へ繋がっていく様子がわかった。

### 8.1. muphic の感想

- 自分で自由にお話しや音楽を作るのは初めてで、上手にできたと思う。全部がすごく楽しかったからまたやりたいです！
- 一回できたけど間違えて消してしまってショックだった。けど楽しかった。
- もっと時間がほしかったです。
- 自分なりの絵や音楽ができたので、とても楽しかった。また、やってみたい。
- 音楽や文章はどうやってかこうかなあ どんな音を作りたいかなあと思っていたけど 慣れてきたからよかったです。
- もっと時間がほしかった。もっと長いお話を作りたいかった。
- (先生より) とても楽しい時間でした。『パソコンは苦手』と思っているわたしでもていねいに教えていただいたおかげでとても楽しくパソコンにさわることができました。子供たちの思い切りのよさや、やわらかい考え方には脱帽です。あっという間の2時間でした。

### 8.2. プログラミンの感想

- 最初はうまくパソコンに命令をだすことができなくて、うまく絵がかけませんでした。でもだんだんうまくなって、最後はとてもいい絵がかけました。またやりたいです。
- 天才っていわれたからうれしかった。(筆者注：大人には思いつけない発想のプログラムを作っていたので、みなで絶賛したのが印象に残った様子)
- プログラミンはいろいろなパターンで作れるし、自分でストーリーも考えられるから楽しかった。
- 物語を作るためにどんなキャラクターを使ったりすればいいか、またどんな背景にすればいいかなど、考えるのがとても楽しかったです。
- 自分の思いどおりの動画がつくれて楽しかった。
- いろいろ自分で工夫をするところがおもしろかった。

### 8.3. SCRATCH の感想

- パソコンをあんまり操作できなかつたけど、想像よりおもしろかつたし説明も簡単だつた。でもちょっとむずかしかつた。
- 今日プログラミングをしてパソコンがとっても好きになつたし できるようにもなりました。またいろいろなところでやってみたいです。
- ロケットがねこにふれたらスコアを増やすプログラムを作ってもうまくいかなかつたので、次回は成功させたいです。
- 最初はむずかしそうと思つたけど楽しかつたし、またやってみたいと思つました。
- 完成まではできなかつたけれど、パソコンの楽しさなどが分かつたのでよかつた。先生の話やパソコンは楽しかつたので、家でもまたやりたいです。
- □を押したら動くスクリプトが面白かつた。背景や人物を自分で決め、作れたことが楽しかつたので、またやってみたいです。今度のときは、もう少し時間があるといいです。
- 今度はもっとすごいものを作りたい。

### 8.4. Code.org アンケート結果

以下に、本年度開催時のアンケート結果をいくつか示す。毎年ほぼ同様の良好な反応を得ることができている。

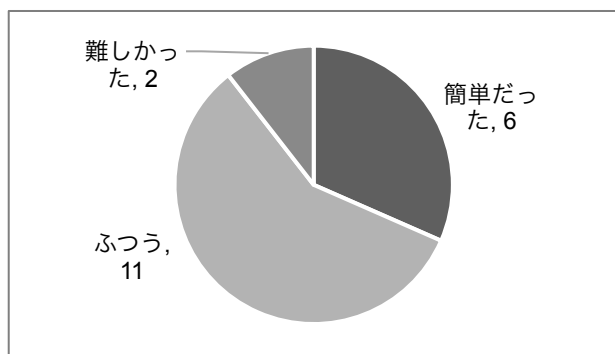


図1 Q.プログラミングは簡単でしたか?

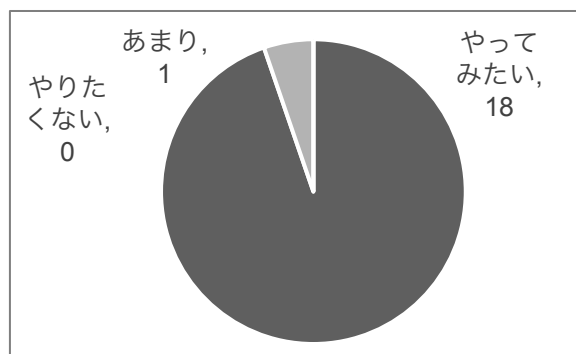


図2 Q.もう一度やってみたいと思つますか?

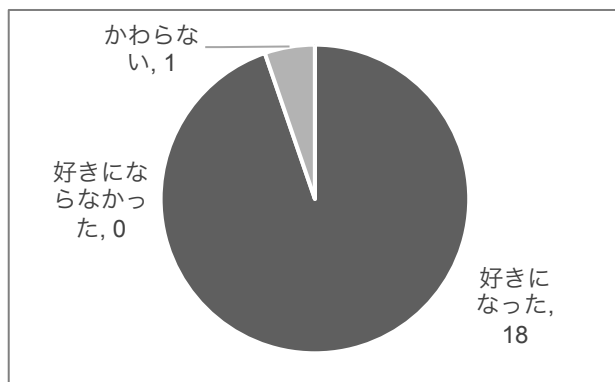


図3 Q.パソコンのことがもっと好きになりましたか?

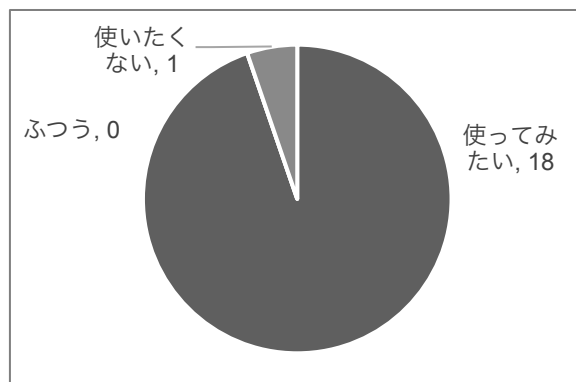


図4 Q.もっとパソコンを使つてみたいですか?

### 8.5. Code.org の感想

- ゲームはむずかしかったけど 楽しくできました。
- エルサのプログラムが、簡単なものと難しいのもあって面白かったです。
- もっと時間がほしかったです。
- ヒントがあったから簡単にできた。
- 家では2年生からパソコンを全くやってないけどできた。
- キャラクターが思いどおり動くのがおもしろかった。
- 今日やって面白かったのもっとやってみたいと思った。
- 全部クリアーできた。

## 9. まとめ

学童保育で行う上での難しさが3点ある。1つは学年の差。2つはパソコンレベルの差。3つは興味、関心度の差である。一般的に行われるプログラミング教室のように、興味やレベルがある程度粒度の揃った集団ではないことから、必然的に進度差が大きく出てしまう。どのレベルに合わせるか、目標設定をどこに据えるかは、毎年悩みどころである。全体的な進捗度合いを見ながら、個々のレベルに応じた声かけや対応が必要だが、答えを言うのではなく、一緒に考え、やる気に繋がられるよう注意しなくてはいけない。これらは今後の課題である。

プログラミングを取り入れたことで、児童の取り組み方が変わったことを強く実感した。ある程度与えられた環境の中ではあるが、自身が考えたものを作り上げる喜び、自身にしか作れないものを作る喜びを体感することができた様子を見て、筆者はじめ開催側スタッフも嬉しく思っている。また、思い通りにいかず、ウンウン唸ったり、他のツールを使って試行錯誤してみたりする経験はとても大切なものであり、その結果うまく行ったときの喜びは何よりも大きいし、結果うまくいかなくても、次こそは！と思う気持ちが重要なのだと思う。子どもたちの、大人にはできないだろう柔軟で自由な発想には毎回驚かされ、感心させられる。筆者としては、このような場を1年に1回であっても提供することで、多様な感性や意識を解放できればよいと考えている。

今回紹介したようなプログラミングツールはどれも楽しく児童に人気があるが、プログラミング以外の情報教育、例えば本年度加えてみたネチケットなども、今後は取り入れていきたいと考えている。情報社会を生き抜くための様々な知識を与えることができればと思う。

児童の楽しそうな様子を見ることを励みに、次年度以降も継続して実施していきたいと考えているので、引き続きのご理解、ご協力を賜りたい。

## 参考 URL

- [1] プログラミン、文部科学省 <http://www.mext.go.jp/programin/> (2016.1.18 閲覧)
- [2] muphic、仙台高専情報システム工学科 力武研究室  
[http://rikitakelab.com/netcommons/?page\\_id=29](http://rikitakelab.com/netcommons/?page_id=29) (2016.1.18 閲覧)
- [3] SCRATCH、MIT メディアラボ <https://scratch.mit.edu/> (2016.1.18 閲覧)
- [4] Code、Code.org <https://studio.code.org/> (2016.1.18 閲覧)