

1

基  
般

## 小学校における全校的な プログラミング学習活動とその意義



阿部和広（青山学院大学 社会情報学部）

政府が、2020年からコンピュータ・プログラミングを初等教育段階から義務化する方針を打ち出したことで、世の中のプログラミング学習に対する関心がにわかに高まっている。これに対し、文部科学省は有識者会議を開き、意見のとりまとめを行っているが<sup>☆1</sup>、何を、どのように行うのかについては、まだ一般に周知されていないのが現状である。このような状況の中で、品川区立京陽小学校では、2013年から現在まで継続的に全学年・全学級でのプログラミング学習を実施してきており、貴重な先行事例となっている。本稿では、それに至る背景と、その意義について解説する。

### 初期のプログラミング学習の 発展と衰退

小学校の授業でコンピュータ・プログラミングが用いられるようになったのは、決して最近の話ではない。1970年代のマイクロコンピュータ、いわゆるマイコンの登場により、コンピュータは、個人でも所有可能な価格になり、卓上に載る大きさとなった。さらに、この上でBASIC言語のインタープリタが動くようになったことで、Burrhus Frederic Skinnerのプログラム学習（行動主義心理学のオペラント学習から発展したもので、プログラミング学習とは異なる）に基づくティーチングマシンをコンピュータに置き換えたCAI（Computer Aided Instruction）を導入したり、自作したりする

☆1 文部科学省初等中等教育局教育課程課教育課程企画室、小学校段階におけるプログラミング教育の在り方について（議論の取りまとめ）平成28年6月16日、[http://www.mext.go.jp/b\\_menu/shingi/chousa/shotou/122/houkoku/1372522.htm](http://www.mext.go.jp/b_menu/shingi/chousa/shotou/122/houkoku/1372522.htm)（2016年9月1日参照）

ことが可能になり、小学校でも用いられるようになった。我が国でも1980年代には、いくつかの事例が発表されている（川村、小森<sup>1)</sup>など）。しかし、これは研究者や教員が教材作成のためにプログラムを開発したもので、児童自身にプログラムさせたものではない。

一方、Seymour Papertらが、1960年代に開発したLOGO言語は、構築主義（constructionism）の考えに基づいて、児童がプログラムを書くことを通して数学の諸概念を発見し、学ぶことを目的としている<sup>☆2</sup>。LOGOでは、画面に表示されるカメの画像やカメ型のロボットに命令することで、幾何図形を描くことができる。これを用いた実践は、1980年代に盛んに行われている（図-1）。

たとえば、金本<sup>2)</sup>は、プログラミングを通して、児童に消費者的ではない主体的な態度が養われたと指摘し、教師主導型のCAIを批判している。また、同時期には、BASICによるプログラム作成を取り入

☆2 <http://el.media.mit.edu/logo-foundation/>

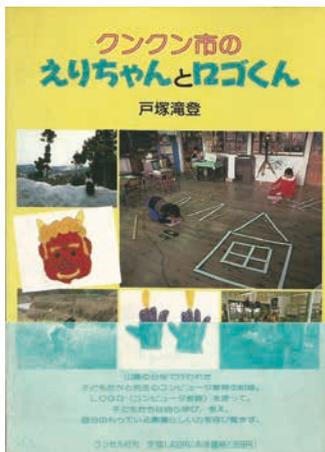


図-1 LOGOの書籍の例「クンクン市のえりちゃんとロゴくん」（戸塚 滝登，1989）

れた実践も行われていた（岡森，西谷，柳本<sup>3)</sup>）。

1990年代に入ると，パッケージソフトウェアやインターネットの利用が一般的になり，これらを利活用することが，児童に求められる能力と考えられるようになった。2002年の学習指導要領改正で，情報教育重視がうたわれた一方で，プログラミングには触れられていない。教員に求められる能力も，コンピュータ操作やそれを用いた指導となっている（伊藤，山崎<sup>4)</sup>）。Apple社のマルチメディア作成環境であるHyperCardや，LEGO社のロボット教材，Mindstormsを用いた実践など，一部の例外はあったものの，小学校でのプログラミング学習はほぼ実施されなくなった。

すでに便利なアプリケーションがあるのだから，これからの情報化社会では，プログラムの作り方よりも，使い方を学ぶ方が重要とする考えは，一見すると合理的なように見える。しかし，使い方の学習に限定することは，児童が試行錯誤しながら，ものづくり（構築）を通して学ぶこと，さらには，主体的な学習態度を身に付けることというプログラミング学習のメリットを失うことになる。

とはいえ，プログラム学習が根付かなかった理由の1つには，当時のプログラミング言語の限界もあった。森本<sup>5)</sup>は，ミスタイプ，空白忘れ，入力不足（引数忘れ）などによるエラーの問題を指摘している。このようなエラーは，入力補完などの開発支援環境のなかった当時のテキスト型プログラミング言語では避けるのが難しく，キーボード入力に不慣れな教員や児童のモチベーションを下げることに繋がったと考えられる。また，プログラミング言語自体が英単語によるものも多く，扱うためには専門的な知識が要求されたことも導入を難しくした。

## ビジュアルプログラミングとプログラミング学習の復活

Alan Kayは，1970年代から80年にかけて，LOGOの思想を発展させる形で，ユーザフレンドリーなGUIを持つオブジェクト指向型プログラミ

ング環境のSmalltalkを開発する。しかし，当時のSmalltalkはXEROX社がライセンスを独占しており，高価なワークステーション上でしか動作しなかったため，本来の目的であった教育現場ではほとんど使われることがなかった。その後，1996年に，一般的なパーソナルコンピュータで動作し，オープンソースソフトウェアで無償のSmalltalkであるSqueak<sup>☆3</sup>が公開され，さらにビジュアルプログラミング環境のEtoysが備わったことで，再びプログラミング学習の気運が高まることになる。

従来のプログラミング言語では，キーボードからの文字入力でプログラムするのに対し，ビジュアルプログラミング言語では，画面上の視覚的な要素を，マウスなどのポインティングデバイスで組み立てることでプログラムする。これにより，前述のエラーをなくすことができ，導入の手間も大幅に容易になったため，非専門家の教員や児童でも簡単に扱えるようになった。これを受けて，2002年には，京都大学，京都市教育委員会などが協力して，京都市下の公立学校でSqueak Etoysを用いたプログラミングの授業を行うALAN-Kプロジェクトが発足する<sup>☆4</sup>。また，CSK社（現SCSK社）が行う子供のワークショップ活動のCAMPや，日本ヒューレット・パッカー社のCSR活動のHP-Squeakers（現在はスーパーサイエンスキッズとしてNPO法人化）でも，Squeak Etoysが用いられた。HP-Squeakersは，各地の学校への機材の寄贈や社員ボランティアの派遣を行い，特に東京都杉並区立和田小学校では，サーバと無線LANで接続された可搬型のノートPCによる普通教室での活用など，先駆的な取り組みが行われた。筆者もこれらの活動を支援するとともに，Squeak Etoysの日本語化などを行っている<sup>☆5</sup>（図-2）。

Squeak Etoysでは，LOGOと比べてマルチメディア機能が拡張されるとともに，カメラに限らず，任意のオブジェクトを並行に動かすことができる。こ

☆3 <http://squeak.org/>

☆4 <http://www.cm.is.ritsumeai.ac.jp/akp/>

☆5 <http://squeakland.jp/>



図-2 CAMPでのSqueak Etoysワークショップの様子  
 (「Squeakers」(Jim Shasky, 2002)より)

れにより、モデル化とシミュレーション能力が向上し、児童が科学と数学を発見的に学ぶ環境としての能力を備えるようになった。また、メディアを作り出すためのメディア、すなわちKayのDynabook<sup>☆6</sup>としての側面もあった。Squeak Etoysを導入した学校では、1980年代のLOGOと同様の効果を確認できたが、その先まで理解して使われたケースは非常に限られた。

## 外部の力による部分的な導入の限界

前章で紹介したさまざまな活動は、いずれも教員が自ら行うというより、外部の研究者や組織が支援して児童・生徒にプログラミングを行わせる形をとっていた。

外部の支援で行う活動は、児童の興味、関心を引き出すという点においては効果的だが、一過性のイベントになりやすく、継続性に問題があった。

限られた時間内に終わらせるために、本来は基本の導入のために用意したチュートリアルが、目的として使われることもあった。たとえば、Squeak Etoysの代表的なチュートリアル、「車の運転」は、車やハンドルの絵(オブジェクト)をペイントツールで描き、そのスクリプト(メソッド)を定義して、



図-3 典型的な「車の運転」<sup>☆7</sup>

自律的に動かすというものだ。その過程で、児童は基本的なツールの使い方やフィードバック制御などを学ぶ。大切なのは、ここで学んだことを発展させて、自分のアイデアでさまざまなプロジェクトに取り組むことである。しかし、現実に全国で行われていたのは、「車が走って楽しかった。おわり」である。結果として、Squeak Etoys = 車を走らせるソフトとして認識されることになった(図-3)。

筆者らも、普及を急ぐあまり、あまりにも丁寧な指導書の作成と授業支援を行ってしまい、これが学校や教員、児童らに対して、受動的な態度をとらせることにつながった。授業形態も、従来と同じ、教員主導の一斉授業として行われることが多く、児童の主体性を引き出す形になりにくかった。

科目は、総合的な学習の時間や特別活動があてられることが多く、それ以外の教科や単元とのつながりも希薄だった。学校側としても、現行学習指導要領の総則にある「情報手段に慣れ親しむ」を満たせばよいとの考えがあったように感じられる。

授業が終わった後、児童の感想としてよく聞かれるのは、「もっとやりたい」である。しかし、学校のパソコン教室は閉じられており<sup>6)</sup>、この声に十分応えることができなかった。

<sup>☆6</sup> アラン・ケイが予測した「子供たちのパソコン」：ITpro, <http://itpro.nikkeibp.co.jp/article/COLUMN/20140327/546447/>

<sup>☆7</sup> <https://www.youtube.com/watch?v=inJESDMMgTs>

## プログラミング学習を日常化するために

MIT メディアラボの所長だった Nicolas Negroponte は、Papert の Children's Machine, あるいは Kay の Dynabook の実現として、世界中のすべての子供たちにノート PC を行きわたらせようという One Laptop Per Child (OLPC) プロジェクトを 2005 年に開始する<sup>☆8</sup>。これはデジタルデバイド解消のため、発展途上国にインターネット接続可能で安価（目標は 100 ドル）な端末を配る運動と誤解されているが、実際は構築主義に基づく教育プラットフォームの普及を目指したものである。そのために Squeak Etoys などがプリインストールされている。OLPC は、その原則の 1 つに、子供がノート PC を自己所有できることを掲げており、子供たちが自由に学べる環境と位置付けていた。これを Negroponte は、「ノート PC が学校にあるのではなく、ノート PC が学校になる」と言っている。

また、2006 年には、Mitchel Resnick らにより、LOGO と Squeak Etoys を発展させたプログラミング環境である Scratch が公開される<sup>☆9</sup>。

Scratch は、Squeak Etoys と比べて、より洗練されたユーザインタフェースを持ち、子供たちが好きな、ゲーム、アニメーション、音楽づくりに特化したデザインとなっている。これは、Papert の「子供たちは彼らにとって個人的に意味のあるものを組み立てているときのみ、それを知的に行っている」という考えによる。また、SNS (Social Network System) と統合されているのも特徴の 1 つで、時間や空間にとらわれない子供たちの協働作業を支援している。

OLPC は、当初の想定より採用国が少なかったことなどもあり、ノート PC の価格も 189 ドルから下がらず、約 280 万台の出荷にとどまっている。しかし、1 人に 1 台のコンピュータ (1:1) という考

えは、Eben Upton らによって 2012 年に発表された、非常に安価（当初は 25 ドル、現在は 5 ドルから）な教育用パーソナルコンピュータ、Raspberry Pi が登場する素地となった<sup>☆10</sup>。

一方、Scratch は、2013 年 12 月に、アメリカの Barack Obama 大統領が、国民に対して、プログラミングを学ぶことの重要性を説いた影響などもあり、1,300 万人の登録ユーザ（2016 年 9 月現在）を数え、1 年に約 1.8 倍のペースで増えている。また、Raspberry Pi の標準 OS、Raspbian にもプリインストールされている。

## 小学校における全校的なプログラミング学習活動

2011 年 3 月 11 日に発生した東日本大震災は、多くの団体が子供たちへさまざまな支援活動を行うきっかけともなった。NPO 法人 CANVAS は Google の支援を受けて、2012 年に「プログラミングワークショップ in 東北」を実施した<sup>☆11</sup>。この活動に筆者も参加し、宮城県の 26 カ所、約 600 人の子供たちを対象に Scratch を用いたワークショップを行った。一連のワークショップでも、過去の実践と同様に児童の積極的な学習態度が観察され、一定の成果はあったが、パソコン教室への Scratch のインストールや、教員の研修までは行えず、筆者らが帰ると活動が終わってしまうのも同様だった。

この活動を受けて、Google から 5,000 台の Raspberry Pi の寄贈を受けられることになり、2013 年 10 月に、筆者が監修者となり、CANVAS を中心に PEG (Programming Education Gathering) プロジェクトが開始された<sup>☆12</sup>。PEG では、単に機材を寄贈して終わりにするだけでなく、各地に活動の拠点を作り、指導者の研修も行うことで継続性を担保できるようにした。

その PEG プロジェクトのパートナーの 1 つが、

☆8 <http://one.laptop.org/>

☆9 <http://scratch.mit.edu/>

☆10 <https://www.raspberrypi.org/>

☆11 <http://canvas.ws/project/programmingtohoku>

☆12 <http://pegpeg.jp/>

品川区立京陽小学校である。京陽小学校では、全校児童約 350 人と全教員に 1 台ずつの Raspberry Pi Model A が配られ、図書室を改装したメディアルームでは、自分の Raspberry Pi をつないで動かすことのできるディスプレイ、キーボード、マウスなどが常設されている。それぞれの Raspberry Pi は校内のサーバに無線 LAN で接続され、作品のファイルを共有できるようになっている。また、希望すれば、書道セットや絵の具セットのように自宅への持ち帰りも可能である。

支援に際して、筆者らは児童の前に出ないことを心がけた。児童と接するのはあくまでも学校の先生の役割であり、我々支援者は裏方に徹するべきである。これは、支援者が直接教えることで、先生や子供たちの主体性を損なった過去の活動の反省による。その一方で、先生の指導案作成への助言、技術的な質問の対応、サンプルプログラム作り、校内研究会への参加などは積極的に行っている。

最初は戸惑いがあった教員、めずらしい道具や環境に興奮していた児童も、今ではすっかり落ち着き、通常の教科、単元の学習の手段として、教科書やノート、文房具と同じように、プログラミングが授業の中に自然に溶け込んでいる (図-4)。

京陽小学校の活動を通し、ついに子供たちのプログラミング学習が日常化する機会が訪れたと考えている。プログラミングが特別なものではなく、基本的なリテラシーとして、すべての児童と教員との間で共通のものとなったとき、小学校にいったいどのような変化が起こるのか。この事例は、それを、ごく普通の公立の小学校で行ったという世界でもあまり例を見ないものである。



図-4 日常化するプログラミング (図工のクレイアニメも専用ソフトを使わずに自作している)

活動は、2014 年度から本格的に始まり、2016 年度で 3 年目に入った。その詳細は、本小特集のほかの記事をお読みいただきたい。

#### 参考文献

- 1) 川村憲雄, 小森孝彦: 教授過程の CAI 化, 日本教育心理学会総会発表論文集 (25), pp.572-573 (1983).
- 2) 金本良通: 小学校におけるプログラミング教育のための事例研究: 三年生の場合, 福島大学教育実践研究紀要 (11), pp.23-32 (1987).
- 3) 岡森博和, 西谷 泉, 柳本朋子: 小学校におけるプログラミング指導について LOGO と BASIC を中心として, 大阪教育大学紀要.V, 教科教育 36 (1), pp.39-56 (1987).
- 4) 伊藤一郎, 山崎謙介: 初等中等学校における情報教育の課題, 情報処理学会研究報告コンピュータと教育 (CE) 2000 (51 (2000-CE-056)), pp.71-78 (2000).
- 5) 森本哲郎: 子どもの「コンピュータ活用能力」の実態について: LOGO による学習を通して, 教育情報研究, 日本教育情報学会学会誌 5 (2), pp.28-38 (1989).
- 6) 阿部和広: 幸せなパソコン教室のために, 情報処理, Vol.55, No.6, pp.598-601 (June 2014).

(2016 年 9 月 5 日受付)

阿部和広 ■ abee@squeakland.jp

青山学院大学 社会情報学部 客員教授。Etoys と Scratch の日本語版を担当。2003 年度 IPA 認定スーパークリエイター。専門は構築主義と初等教育におけるプログラミング教育。