

子供の創造的活動とプログラミング学習

阿部 和広

青山学院大学 客員教授

[キーワード]

プログラミング教育, ものづくり教育, 構築主義, Scratch, Raspberry Pi

1. はじめに

世界的な産業構造の変化と人材育成の要請を受け、プログラミング教育への注目が高まっている。米国では大統領が国民にその重要性を訴え、英国では小学校から義務教育が始まった。日本でも政府方針として初等中等段階からの教育の充実がうたわれている。その一方で、MIT のパパートやレズニックは、プログラミングを新しい表現や学習の手段と捉え、学習者の能動性を重視する構築主義を提唱している。ここでは、その考え方と実践について紹介する。

2. デジタルネイティブと創造的活動

近年、デジタルネイティブ(digital native)という言葉を目にする機会が増えた。これは、生まれながらにして、デジタル機器に親しんでいる人々の意である。マサチューセッツ工科大学(MIT)の Mitchel Resnick は、このような人々について以下の様に述べている。

「今世界中で若いひとたちがますます多くの時間を、デジタルメディアと共に過ごしています。彼ら彼女らはコンピューターや電子玩具、ゲーム機、そして携帯電話に囲まれて育ち、デジタル機器を多様な活動で使いこなしています」¹⁾

これは日本の街中や電車の中でも、よく見かける光景である。幼児が教えてもないのに、タブレットのロックを解除し、アプリケーションを起動して遊んでいたという話も聞く。しかし、Resnick は、このように続けている。

「若い人たちの多くはただデジタルメディアを使っているだけで、デジタルメディアを使っての何かの創造は行っていないのです」¹⁾

デジタル機器の多くはインタラクティブであり、それを使って通信したり、情報にアクセスしたりするにはタッチパネルなどのユーザインタフェースを介して操作する必要がある。デジタルネイティブと呼ばれる人々は、この操作に長けている。しかし、そこで行っていることは、他の誰かが作ったコンテンツの消費に過ぎない。たとえば、ソーシャルゲームで遊ぶ、動画を視聴するなどがそれに当たる。

歌声合成ソフトウェアを使った作曲や、センサーやマイクロコントローラを用いた電子工作を行って、作品を動画共有サイトで発表したり、展示会に出展したりと言った例外はあるものの、自分で新しいコンテンツを作り出したり、さらにはメディア自体を作り出すようなことは稀である。

3. 小学校での ICT 活用

一方で小学校における ICT 活用はどうだろうか。平成 23 年 4 月に実施された学習指導要領には以下のように書かれている。

第 1 章 総則

第 4 指導計画の作成等に当たって配慮すべき事項

(9) 各教科等の指導に当たっては、児童がコンピュータや情報通信ネットワークなどの情報手段に慣れ親しみ、コンピュータで文字を入力するなどの基本的な操作や情報モラルを身に付け、適切に活用できるようにするための学習活動を充実するとともに、これらの情報手段に加え視聴覚教材や教育機器などの教材・教具の適切な活用を図ること。

つまり、児童に求められているのは、ICT に慣れ親しみ、基本操作と情報モラルを身に付け、教材・教具を使うことである。具体的には、総合の学習の時間などで、キーボードからのローマ字入力やペイントソフトウェアを使ったマウス操作、インターネット検索による調べ学習、各教科における CAI(Computer Aided Instruction)の利用などにとどまっている。ここにプログラミングは含まれていない。

たとえば、絵と文字を組み合わせたカレンダー作りを行うとしても、それは基本操作の習得を目的としたものであって、その作品の創造性を評価する訳ではない。理科の単元で風とゴムの働きアニメーションを見ることがあっても、そこで扱う物理現象を児童がモデル化し、シミュレーションすることを期待しているわけではない。

4. 表現手段としてのプログラミング言語

問題を解決する手段や、プログラミング言語の習得ではなく、プログラミング(あるいはデバッグ)という行為そのものが子供たちの心的枠組み(schema)の構築につながるという発想は 1960 年代に MIT の Seymour Papert によってもたらされた。これは、発達心理学者、Jean Piaget の構成主義(constructivism)を発展させた構築主義(constructionism)として知られる。その実践のために作られたのが LOGO 言語である 2)。

LOGO は、画面上のカメ(現実のロボットのこともある)をプログラムして動かし、その軌跡で様々な図形(turtle graphics)を描くための言語である。

たとえば、1 辺の長さが 100 の正三角形を描きたいとすれば、カメを 100 歩進め(forward 100)、120 度右に曲がること(turn 120)を 3 回繰り返す(repeat 3)。このプログラムを書くとき、子供たちは正三角形の幾何的性質を試行錯誤(バグとデバッグ)を通して考え、その概念を獲得する。それが数式で表されていない、プログラムとして等価なものが表現できるのであれば、その子は正しく理解していると言える。

これは、教師が教える正しい科学的知識(3 辺の長さが等しく、それぞれの頂点の内角が 60 度の三角形)を無批判に受け入れることとは異なる(面白いことに、大人にこのプログラムを書かせると、かなりの割合で turn 60 とする)。

重要なのは、子供たちは、正三角形を描きたいのではなく、家や花、鳥などを描きたい(創造したい)ということだ。そのために「たまたま」正三角形が必要だったに過ぎない。学びは遊びから生じる。

1968 年に LOGO を見た Alan Kay は、これからインスピレーションを得て、1972 年に「すべての年齢の子供

たちのためのパーソナルコンピュータ」1) すなわち, Dynabook を着想し, それが Smalltalk, GUI, オブジェクト指向などにつながる。

Papert の指導を受けた Rasnick は, Kay の Squeak Etoys や Randy Pausch が開発した Alice, Alexander Repenning の AgentSheetsなどを参考に Scratch 3) というプログラミング環境を開発する。Scratch には以下の特徴がある。

- ・文法エラーの生じないビジュアルプログラミング(文字入力とプログラミングの間に本質的な関係はなく, 文字入力に習熟していない段階での苦痛を軽減する)
- ・メインルーチンのないスクリプトの並列実行(現実の世界は非集中的で複数の事物が並列に動いている)
- ・属性値やスクリプト変更の実行中即時反映(コンパイルやリンクが必要なのは計算機の都合である)
- ・メッセージ送信によるオブジェクトの連携(現実の事物はメッセージを交換することで処理を進めている)

プログラミングを通して, 子供たちが数学や科学を自ら学ぶことを目指した Papert や Kay に対して, Rasnick は, それらを含んだ上で, プログラミングを, 単なる職業訓練ではなく表現の新しい形式や学びの新しい文脈と定義している 4) . そのため, Scratch には, アニメーションやゲーム, 音楽と言った要素が取り入れられている。ただし, それらは, CAI によくあるように何かを達成した報酬として与えられるのではなく, 自分で作り得る可能性として示されており, アイデアをプログラムとして表現しなければ何も起こらない。

Scratch は, 子供たちの遊びを観察することから得られた「想像, 作成, 遊び, 共有, 振り返り, 想像…」という螺旋構造(Creative Learning Spiral)を支援するために作られている。

そのため, 動画共有サイトによく似た作品共有サイト(<http://scratch.mit.edu/>)が用意され, 世界中の子供たちが協働して作業する環境として機能している。共有された作品数は 740 万個, 登録ユーザ数は 480 万人である(2014 年 12 月現在)。これらの作品はクリエイティブコモンズ(CC BY-SA)にしたがって, 自由に改変できる(これをリミックスと呼ぶ)。

5. 子供たちのためのパーソナルコンピュータ

文部科学省による「平成 26 年度 学校における教育の情報化の実態等に関する調査結果」によると, 2015 年 3 月 1 日現在の教育用コンピューター1 台あたりの当たりの児童生徒数は全国平均で 6.4 人である。

教育用コンピューターは, パソコン教室に据え置きになっていることが多く, 子供たちは自由に使うことができない。紙や鉛筆であれば, いつでも文字を書き, 絵も描けるが, パソコンを使えるのは許可された特別な時間だけである。これを日々の創造的活動の手段と考えるのは, 教職員にとっても児童にとっても難しい 5) 。

これを変えるには, パソコンを特別視せず, 当たり前のように常にあり, 子供たちが誰から指図されることもなく, 自然に使うような環境や雰囲気醸成する必要がある。

MIT メディアラボの創設者である Nicholas Negroponte は, Papert の構築主義を実践するには「子供 1 人に 1 台のノートパソコン」が必要と考え, 2006 年に OLPC(One Laptop Per Child)という NPO を立ち上げ, 単価 100 ドルを目標とするノートパソコン XO を開発して, 各国政府に導入を働きかけている(導入台数は 2014 年現在で 300 万台弱。価格は 189 ドル)。これは Papert の Children's Machine や Kay の Dynabook の実現ともいえる。

ケンブリッジ大学の Eben Upton は, スマートフォンやタブレットの台頭により, 子供たちがプログラミングを学ぶための手軽なパソコンが無くなっていることに危機感を覚え 6) , 20 ドル(モデル A+), 25 ドル(モデル A), 35 ドル(モデル B, B+)という極めて安価なワンボードコンピュータ Raspberry Pi を開発した。Raspberry Pi は, スマートフォン用の SoC を搭載し(ARM11 700MHz, 256~512MB RAM), USB 2.0 や HDMI, 100BASE-T と言った標準

的なインタフェースを備えている。また、HDD の代わりに SD カードを用い、Linux をはじめとするオープンソースソフトウェアが用意されている。また、GPIO(汎用入出力端子)が用意されているため、センサーやアクチュエータの接続も容易である。

標準 OS の Raspbian は X Window と LXDE によるデスクトップ環境を備え、Python や C/C++, Ruby, Scratch など豊富なプログラミング言語を使うことができる。

Raspberry Pi は 2015 年 6 月現在で 600 万台が出荷されている。

5. 日本での実践事例

Scratch は、日本でも各地の小中高校や大学、NPO のワークショップ、企業のプログラミング教室などで使われているが、中には必ずしも Rasnick の考えに沿っていないものもある。また、Raspberry Pi も、教育目的というより、Linux が動作する GPIO を備えた安価なマイコン基板として、電子工作マニアに使われている傾向がある。

ここでは、本来の目的に近い実践例として、東京都品川区立京陽小学校での事例を紹介する。

京陽小学校は、教職員数 25 名、児童数 342 名、14 学級の公立小学校である。2014 年度の研究主題として、「デジタルテクノロジーの書き手を育てる ～豊かな言語能力の育成を目指して～」を掲げ、4 月から全校児童に OS と Scratch が SD カードにインストールされた Raspberry Pi モデル A が配布されている。これは、Google の支援を受けて NPO 法人 CANVAS が実施しているプログラミング教育支援プロジェクト「PEG -Programming Education Gathering-」から寄贈されたものである。また、2014、2015 年度の東京都教育委員会言語能力向上拠点校、品川区教育委員会 ICT 活用実践校でもあり、その支援も受けている。

図書室を改装したメディアルームには、電子黒板 1 台と 30 台分のディスプレイとキーボード、マウス、USB ハブなどが設置され、児童は各自の Raspberry Pi を接続して使う。ディスプレイなどは廊下にも置かれており、そちらにつないで使うこともできる。それぞれの Raspberry Pi は無線 LAN で接続され、NAS に保存されたファイルに Samba でアクセスできる。自宅への Raspberry Pi の持ち帰りも適宜行われている。

京陽小学校では、総合の学習の時間ではなく、国語や算数、理科と言った通常の教科の単元に Raspberry Pi と Scratch を用いている。一例を以下に示す。

- 1 年生 市民科「パソコンをつかおう」(パソコンの基本操作)
- 2 年生 算数科「かけ算(1)」
- 3 年生 理科「風やゴムのはたらき」
- 4 年生 国語科「言葉から風景を想像しよう～短歌をデジタル画で味わおう～」
- 5 年生 音楽科「音楽づくり」
- 6 年生 市民科「自分たちの学校」、「プレゼンテーション力をつける」

たとえば、3 年生の理科「風やゴムのはたらき」では、ゴムの力で走る車を使った実験に先立ち、ゴムを伸ばす長さや本数と車が走る距離との関係について、その仮説を画面上で車を動かすプログラムとして児童が作成して動かし、それを発表したり、他の児童のプログラムと比較したりした後に、実験の結果と比較して考察する。これは、初歩的なモデル化とシミュレーションを行ったと考えられる。

4 年生の国語科「言葉から風景を想像しよう～短歌をデジタル画で味わおう」では、いくつか用意された短歌の中から、自分が好きなものを一首選び、その短歌から想起される色と風景の写真、効果音を組み合わせたマルチメディア作品を作る。これも他の児童と見せ合ったり、前で発表したりする。

5年生の音楽科「音楽づくり」では、いくつかの気持ち(感情)を決め、あらかじめ用意されたサンプルとなる小節(長調や短調, スタッカートなど)を組み合わせて、ふさわしい音楽を作る。聴き合ったり、発表したりするのは他と同じである。

ここでのポイントは、いずれも子供たち自身が自分なりにプログラムを書いて表現していることである。単に動画を観たり、音楽を聴いたり、短歌を読んだりするだけでなく、新しいコンテンツを自分たち自身で作り出している。教員が説明していない機能やテクニックを使ってプログラムする児童もおり、それを友達同士で教え合う光景も日常化している。NAS に保存された他の児童の作品を開いて見ることも普通に行われている。最初は同じだったデスクトップの壁紙も今はさまざまなものになっている(方法は誰も教えていない)。

また、言語能力の育成もテーマのひとつであり、プログラムで書いたものを、言葉として言えることも重視している。これはひとつのものを複数の視点から見て表現することにつながる。

5. 初等教育におけるプログラミングの今後

前述のように、現行の学習指導要領では小学校でプログラミングを行うことは求められていない(中学校では2012年4月から技術・家庭科の単元「プログラムによる計測・制御」が必修となつている。これはGPIOを持つRaspberry Piと相性が良い)。

2015年6月に改訂された内閣の「世界最先端IT国家創造宣言」では、初等・中等教育段階におけるプログラミングに関する教育の充実に努めることが謳われており、その工程表には、文部科学省と総務省がNPOや企業と協力して、2019年度より全国で展開すると明記されている。

しかし、これはIT人材育成の立場に立ったものであり、Papertの構築主義やResnickの言う「表現の新しい形式や学びの新しい文脈」としてのプログラミングとは若干の乖離がある。また、学校現場においてもプログラミング教育に対する理解が十分とは言い難い。そのような状況で、子供たちによりよい学習環境を提供するために情報教育関係者にも一層の努力が求められるだろう。

以上

【参考】

- (1) 阿部和広, 小学生からはじめるわくわくプログラミング, 日経BP社(2013)
- (2) シーモア・パパート(著), 奥村貴世子(訳), マインドストーム 子供、コンピューター、そして強力なアイデア, 未来社(1982)
- (3) Maloney, J., Burd, L., Kafai, Y., Rusk, N., Silverman, B. and Resnick, M., Scratch: A Sneak Preview, Second International Conference on Creating, Connecting, and Collaborating through Computing, Kyoto, Japan, pp.104-109(2004)
- (4) Mitchel Resnick, Learn to Code, Code to Learn, <https://www.edsurge.com/n/2013-05-08-learn-to-code-code-to-learn> (2013)
- (5) 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 June 2014, pp.598-601(2014)
- (6) Eben Upton, Gareth Halfacree, Raspberry Pi ユーザーズガイド, インプレスジャパン(2013)