

Web 上のプログラミング学習環境の紹介

名古屋高等学校 中西渉 *

2017 年 11 月 21 日

1 はじめに

現行の「情報の科学」の教科書では、プログラミング言語として主に Excel の VBA と JavaScript が取り上げられている。いずれも情報教室に特別なソフトウェアをインストールする必要がない*¹ことが、それらが選ばれた理由であろうと考えられる。

筆者は勤務校の授業で PEN[1] という初学者向けプログラミング学習環境を使い始め、その後それにフローチャートを付け加えた PenFlowchart[2] を開発して使っている。しかし、これらは Java アプリケーションであるため実行ファイルをインストールする必要があり、また Java のランタイムライブラリを必要とするので、学校や自治体によっては導入できないことがあると聞いている。

プログラミング学習環境が Web ブラウザ上で実行できるものであれば、そのような問題は解決されると考えられる。そこで本報告では Web ブラウザ上で実行できて無料で利用できる主だったプログラミング学習環境を紹介するとともに、前述した PEN, PenFlowchart を HTML5 に移植した WaPEN を開発したことについて述べる。

2 各プログラミング学習環境

2.1 テキストによるプログラミング

2.1.1 Bit Arrow

校種を問わず多くの現場で用いられているドリトル [3] は東京書籍の教科書「情報の科学」でも取り上げられているが、やはり Java アプリケーショ

ンであるということでインストールの必要がある (Windows, Mac 版は Java のランタイムライブラリが同梱されている)。ブラウザ上で実行できるお試し版として Java アプレット版が公開されているが、これの使用は今となってはさらに難しい。

ブラウザ上でプログラミングできる環境 Bit Arrow[4] はドリトルの他、JavaScript, 簡易 C に対応している。JavaScript は記述を簡単にできるよう工夫されており、`getElement...` などといったものを書くことなく、表示の変更や移動などが簡単にできたりする。また、作成したプログラムはサーバ上に保管され、QR コードでアドレスを示してスマートフォンなどで実行することもできる。

学校で利用する際には申請すればクラス ID が交付され、各生徒はログインしてプログラムを作成・保存し、教員は生徒のプログラムを閲覧・実行して確認できる。生徒にはパスワードは与えられないが、これは管理の手間とのトレードオフによる選択であろう。

2.1.2 paiza.IO

paiza.IO[5] はギノ株式会社が公開しているオンライン実行環境であり、多くの言語 (C, C#, C++, Java, JavaScript, PHP, Perl, Ruby, Python, Haskell, ...) に対応している。ブラウザ上でコードの編集、実行ができる。

手元のコンピュータ上に環境を用意しなくても様々な言語のプログラミングを試せることが利点である。勉強会などでの利用は想定しているが、授業で使うことについてはどうかかわからない。筆者は生徒の自習や部活での利用には適していると考えている。

* watayan@meigaku.ac.jp

*¹ 筆者の勤務校では端末の OS が Linux なので Microsoft Office はインストールされていないのだが。

2.1.3 Monaca

Monaca[6] はスマートフォンのアプリがブラウザ上で作成できる環境で、アシアル株式会社が運営している。HTML5 で作ったアプリは画面上だけでなく、デバッガのアプリをインストールすることによって自分のスマートフォンで動作確認できる。

プロジェクトなどをサーバ上に保存するためログインの必要はあり、その際にメールアドレスが必要となる。学校での利用も想定しているということで、高校や大学などでの利用実績もある。

2.2 ブロックによるプログラミング

2.2.1 Scratch

既に多くの実践で取り上げられている Scratch[7] は、MIT メディアラボのライフロングキンダーガーデニンググループが開発したプログラミング環境である。

現在公開されている Scratch 2.0 は Flash アプリケーションであるが、次に公開される 3.0 は HTML5 で開発されているという [8]。Flash Player は 2020 年に提供終了される予定であり [9]、既にいくつかのブラウザでは明示的に許可しないと Flash をブロックする状況になっているので、この変更は待ち遠しい。

2.2.2 プログラミン

文部科学省が運営しているプログラミン [10] があるが、これは Flash アプリケーションなので、2020 年以降どうなるのかが気になるところである。

2.2.3 micro:bit の JavaScript ブロックエディター

Micro:bit 教育財団が提供する小型ボード型コンピュータ micro:bit[11] の開発環境の一つ、JavaScript ブロックエディターは、シミュレータがついているので実機がなくても動作を確認できる。また、ブロックと JavaScript の編集画面を行き来し、どちらでもプログラムを編集できる。

2.2.4 スモウルビー

Ruby ゲーム開発用ライブラリ Smalruby[12] のためのエディタは、ブラウザ上でブロックと Ruby のコードの画面を行き来してどちらでも開発できるようになっている。手間はかかるがサーバに置けば他のマシンからもブラウザで開発ができるのだが、

実行環境はローカルマシンにインストールする必要があるようだ。

3 WaPEN について

3.1 PEN → PenFlowchart → WaPEN

大学入試センター試験「情報関係基礎」の問題には DNCL[13] というプログラミング言語が用いられている。大阪学院大学情報学部西田研究室と大阪市立大学大学院創造都市研究科松浦研究室の共同プロジェクトとして開発された PEN は、DNCL の拡張である xDNCL を用いた初学者向けプログラミング学習環境である。

素のままの DNCL には入力構文がなく、数値の扱いにも曖昧な点があるなど、そのままプログラミング環境を作るには難がある。そこで xDNCL では入力構文や繰り返しの種類を増やし、変数宣言を付け加えて「型」を明確にしている。

PEN での実習をしていると、テキストエディタ上でたとえば「もし～を実行する」の「～を実行する」を削除してエラーを引き起こすというようなことが多く見受けられた。それを防ぐために、フローチャートを作成することでプログラムを生成する機能を PEN に付け加えたものが PenFlowchart である。

PEN, PenFlowchart は Java アプリケーションなので、冒頭に述べたようにインストールが必要である。そこでそれらの機能（全部ではないが）を HTML5 で移植して、ブラウザでプログラムが作れるようにしたものが WaPEN (Web-aided PEN) である。

3.2 機能の違い

操作が PEN, PenFlowchart と大体同じになるように作成したが、次の点は異なっている。

- 変数表示画面を実装していない。
- 入力支援ボタンを構文に関するものだけに限り、関数の入力などは右クリックのメニューで選べるようにした。
- フローチャート画面ではパーツの追加をドラッグ&ドロップでなく、右クリックのメニューで選ぶようにした。

- フローチャートのカット&ペーストは未実装。

3.3 PEN の xDNCL との違い

移植をしきれていないことや、フローチャートでの表現方法が思いつかないことにより、WaPENでは実装されていない構文がある。逆に最後にあげた配列への代入は PenFlowchart と WaPEN で独自に実装したものである。

- 複数の代入文をコンマでつないで 1 行に書く構文がない。
- 関数や手続きを作る構文がない。
- File I/O の命令がない。
- 「そうでなくもし」(else-if に相当) の構文がない。
- `a ← [1,2,3]` のように、配列にまとめて値を代入できる。

3.4 想定している使用環境

WaPEN は筆者のサイト*2と GitHub*3で公開している。これをダウンロード（あるいは `git clone`）して `sample.js-dist` を `sample.js` にコピーすればいい（ついでにサンプルプログラムや `index.html` に手を加えてもいい）。ローカルマシンに置いた `index.html` をブラウザに読み込ませてもいいし、適当な Web サーバにアップロードして Web アプリケーションとして使うこともできる。

職場の Web サーバが使えないのであれば、月 500 円程度でレンタルサーバを借り、そこに教材などと一緒に置いて授業で利用するのもいいのではないだろうか [14]。この件に限らず、自由に使える Web サーバがあると何かと便利である。

3.5 使用実績

2017 年 8 月に行われた情報処理学会の教員免許更新講習において、センター試験「情報関係基礎」の問題を解く演習で WaPEN を使用した（当時フローチャートは未実装）。会場となっていた東京大学の実習室の端末に PEN や PenFlowchart をインストールするのが面倒だったこと、変数の宣言を不

要とする設定*4で演習を行いたかったこと、配列の初期設定など試験問題中のプログラムに現れない部分をサンプルプログラムとして提供できることなどが、これを選択した理由であった。

勤務校ではプログラミングを 3 学期に扱うため、授業ではまだ使用していない。

3.6 余談：PEN とブロックプログラミング

Web ブラウザ上の環境ではないが、PEN にも OpenBlocks[15] を使ったブロックプログラミングの環境の実装 oPEN[16] があり、GitHub で公開されている*5。一方、Google が提供しているライブラリ Blockly[17] を使って、タブレットにも対応したものを作ろうという構想はあるらしい [18]。

4 おわりに

ブラウザ上で実行できるプログラミング学習環境をいくつか取り上げたが、これらが使われることによってインストール版の価値が損なわれるわけではない。完全にオフラインの状況でも使えたり、ハードウェアやローカルマシンの資源にアクセスしやすいというような利点があるので、適宜使い分けのが良いのだろう。

ところで、筆者は WaPEN を「素材」であると考えている。たとえば次のような「改造」はわりと簡単にできるのではないだろうか。そのような、いろいろな使い方をしてもらえれば作者として幸甚である。とはいえ、そういったことをしやすくするためにはどのライセンスを選択するといいいのか、まだ決めあぐねているのだが。

- CGI や PHP を利用してボタン一つでプログラムが提出できるようにする。
- 入出力部分を書き換えて、与えた入力に対して予期通りの出力が得られるかを自動でチェックするようにして自動採点する。

*2 <http://watayan.net/prog/#wapen>

*3 <https://github.com/watayan/WaPEN>

*4 PEN はそのような設定ができるが、PenFlowchart はできない。

*5 <https://github.com/xDNCL/oPEN>

参考文献・サイト

- [1] 中村亮太, 西田知博, 松浦敏雄. 初学者向けプログラミング学習環境 PEN. <http://www.media.osaka-cu.ac.jp/PEN/>.
- [2] 中西渉. 自作プログラム (PenFlowchart) . <http://watayan.net/prog/#penflowchart>.
- [3] 兼宗進. プログラミング言語「ドリトル」. <http://dolittle.eplang.jp/>.
- [4] 兼宗進, 並木美太郎, 長慎也. オンラインプログラミング環境 ビットアロー (Bit Arrow) . <http://bitarrow.eplang.jp/>.
- [5] paiza.IO. <https://paiza.io/ja/>.
- [6] Monaca Education. <https://edu.monaca.io/>.
- [7] Scratch – Imagine, Program, Share. <https://scratch.mit.edu/>.
- [8] Scratch 3.0. https://jp.scratch-wiki.info/wiki/Scratch_3.0.
- [9] Flash & the future of interactive content. <https://blogs.adobe.com/conversations/2017/07/adobe-flash-update.html>.
- [10] 文部科学省. プログラミン. <http://www.mext.go.jp/programin/>.
- [11] micro:bit. <http://microbit.org/ja/>.
- [12] Smalruby. <https://github.com/smalruby/>.
- [13] 独立行政法人大学入試センター. センター試験用手順記述標準言語 (DNCL) の説明. http://www.dnc.ac.jp/sp/albums/abm.php?f=abm00004841.pdf&n=H23_dnc1.pdf.
- [14] 鹿野利春. 教科「情報」の授業 on クラウド. 全国高等学校情報教育研究会 第6回全国大会, 2013.
- [15] MIT Sheller Teacher Education Program – Open Blocks Download Page. <http://education.mit.edu/about-old/open-blocks-download-page/>.
- [16] 主原佑記, 赤井昭仁, 中村亮太, 松浦敏雄. Openblocks を用いたプログラミング学習用ソフトウェアの開発. 情報処理学会 コンピュータと教育研究会 (CE) 研究報告, Vol. 2014-CE-124, No. 9, 2014.
- [17] Blockly. <https://developers.google.com/blockly/>.
- [18] 西田知博. 初学者向けプログラミング学習環境 PEN と教材に関する研究. PhD thesis, 大阪大学, 2017.