

ビジュアルブロック化した DNCL の Web ブラウザ上での実行環境

岩崎 みのり, 中西 通雄

Minori Iwasaki, Michio Nakanishi

大阪工業大学情報科学部コンピュータ科学科

Dept. of Computer Science, Faculty of Information Science and Technology, Osaka Institute of Technology

Email: nakanishi.michio@gmail.com

あらまし：大学入試センター試験科目「情報関係基礎」では、疑似言語の DNCL が用いられている。DNCL の Web ブラウザ上での実行環境として PEN と WaPEN があげられる。本研究では、昨年度に本研究室で開発された PEN 用トランスレータを改良し、WaPEN の機能と統合して、ビジュアルブロック化した DNCL を Web ブラウザ上で実行できるようにした。本学部の 1 年生対象科目で使用されているプログラミングの課題を、BlocklyPEN と WaPEN の両方を使用してもらい、学生に評価をしてもらった結果、入門者にとって使いやすいとの評価を得た。

キーワード：DNCL, PEN, WaPEN, Blockly, プログラミング

1. はじめに

大学入試センターの試験科目「情報関係基礎」では、疑似言語の DNCL が用いられている⁽¹⁾。DNCL の代表的な実行環境として PEN と WaPEN がある⁽²⁾。ともに、画面上の入力支援機能のボタンをクリックし、必要に応じてキーボードから文字入力する。ただし、PEN は Java の実行環境をインストールする必要があるのに対して、WaPEN は Web ブラウザ上で実行できる利点がある。

本研究では、WaPEN に対して機能追加を行い、ブロックを組み合わせることでプログラミングができる環境として BlocklyPEN を作成した。プログラミング学習の一連の流れとして、小中学校段階で Scratch を用いたビジュアルブロックでプログラミングを体験し、高校で本研究の BlocklyPEN に引き継いで DNCL に翻訳されたコードを学び、さらにそれを WaPEN で実行する方法が考えられる。さらに、DNCL はほぼ C 言語に 1 対 1 で対応していることから、本学科で実施しているように C 言語に接続することもできる。

2. 実現方式と利用方法

2.1 実現方式

2016 年度の卒業研究では、Web ブラウザ上でブロックを組んで PEN のコードに翻訳するトランスレータが作成された。しかし、プログラムを実行するためには PEN を用意し、翻訳されたコードをコピーして PEN の画面に貼り付ける必要があった⁽³⁾。

本年度の研究では、昨年度のトランスレータを改良して、翻訳されたコードが WaPEN 上で機能するようにした(図 1)。従ってすべてを Web ブラウザ上で実行できる。

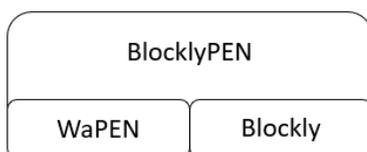


図 1 BlocklyPEN の実現方式

2.2 BlocklyPEN の利用方法

動作環境としては Google Chrome を推奨している。

プログラム作成ページでブロックを配置してプログラムを作成する(図 2)。このプログラムは九九の表を作成する 2 重ループのプログラムである。

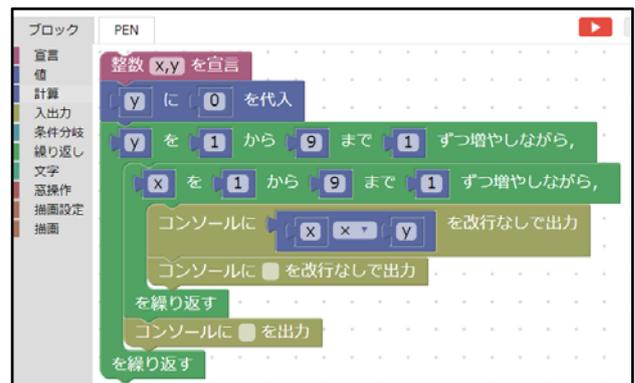


図 2 プログラム作成ページ
実行ボタンを押すと実行結果が表示される(図 3)。

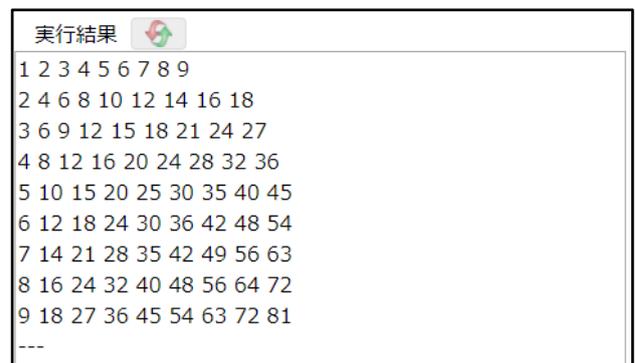


図 3 実行結果画面

昨年度のトランスレータでは描画機能が未完成だったが、WaPEN には描画命令が存在するので、対応する命令のブロックを作成した。描画命令の実行結果は、図 3 の実行結果画面の下部に別ウィンドウで表示される。図 4 は、描画範囲(サイズ 200*200)の中

にランダムな色の円を 100 個描くプログラムを作成し、その実行結果を示したものである。

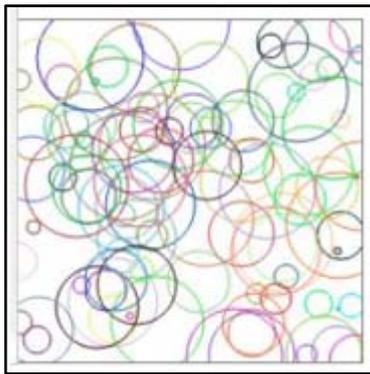


図4 描画結果

PEN のタブを押して画面を切り替えると、翻訳された DNCL のコードが表示される(図5)。



図5 プログラム翻訳ページ

また、作成したブロックは xml 形式のファイルとして保存することができ、いつでもそれを読みだすことができる。

3. 評価

本来はプログラミング未経験者に評価してもらふべきであるが、完成した時期が遅かったため、本学部のプログラミング経験のある学生 9 名に評価を依頼した。WaPEN と BlocklyPEN を使ってプログラミングをしてもらい、評価アンケートを実施した。アンケート項目は、問 1「ユーザーインターフェイスについて」、問 2「使いやすさ」、問 3「コンパイルエラーの出にくさ」を回答してもらった。各項目の評価には 4 段階評価を用いた(図 6)。

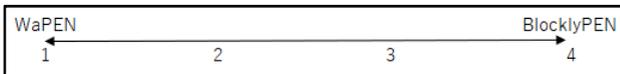


図6 評価アンケート

評価の結果、問 1 では 9 人中 7 人が 4、2 人が 3 と回答した。自由記述の欄に「色があることで見やすかった。」という意見があった。これはブロックにすることにより、プログラムの命令を視覚的にも捉えやすくなったと考えられる。問 2 では 9 人中 6 人が 4、2 人が 3、1 人が 2 と回答した。2 と回答した人は「WaPEN の方が普段のプログラミングに近かったから。」という意見であった。この人はプログラミング経験者であり、さらに「プログラミング初心者にはブロックのほうが入門しやすいのではないか。」と自由記述に記載していた。たしかにプログラミング経

験者にはブロックを組むという操作が煩わしいものであると想定されるが、初心者にとっては小学生向けのプログラミング学習環境である Scratch のようにブロックのほうが入門しやすいと考えられる。

4. 今後の課題

4.1 使用できる変数一覧の作成

プログラミング初心者には、未宣言の変数を使用するというミスがありがちである。これを減らすために、使用できる変数一覧の表をプログラム作成ページの下部に自動で作成する機能を追加することが考えられる(図 7)。

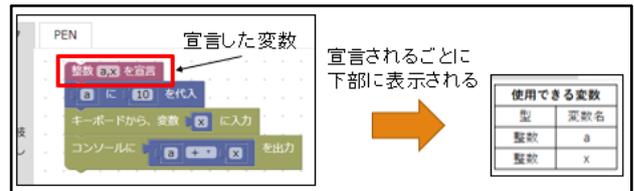


図7 使用できる変数一覧のイメージ図

変数一覧表は宣言ブロックで変数を宣言するたびに更新されていくようにする。これによって、自分で変数を宣言しなければ変数は使用できないことが認識できると期待できる。

4.2 変数宣言不要版の作成

変数宣言不要版とは、変数宣言しなくてもプログラミングができるものである。WaPEN では変数宣言不要版も公開されているが、BlocklyPEN では開発していない。これは変数宣言が必要な C 言語などを最終のターゲットとして想定した学習環境としているからである。変数宣言不要版があることによって、変数宣言が不要な JavaScript へつなぐことを想定した学習環境としても利用可能となる。

5. 結論

本研究では、ビジュアルブロック化した DNCL の Web ブラウザ上での実行環境の開発を行った。昨年度開発された PEN 用トランスレータを改良し、WaPEN の機能と統合して実装した。ブロック化したことにより、プログラミング経験者から「未経験者でも使いやすいだろう」という意見をもらった。

今後の課題として、プログラミング経験のない人に使ってもらい、初心者を使いやすいかを評価してもらふ必要がある。

謝辞：本研究の一部は、JSPS 科研費 17K01088 の助成を受けた。

参考文献

- (1) 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」, 大学入試センター (2011)
- (2) 中西渉：「DNCL の Web ブラウザ上での実行環境」, 情処学会情報教育シンポジウム, pp.160-162 (2017.8)
- (3) 辻祐介：「プログラミング初心者向けのブロックを用いた PEN 用トランスレータの開発」, 大阪工業大学卒業論文 (2017.1)