

PenFlowchartによるアルゴリズム学習

名古屋高等学校教諭 中西 渉

1. はじめに

筆者は勤務校で、PENやPenFlowchartを用いてアルゴリズム学習を行ってきた。本報告では、これらのソフトウェアを用いて行った授業実践について述べる。また、新課程の教科書に同ソフトを用いることについて考察する。

2. PenFlowchartを開発するにあたって

PENについて

PENは大阪学院大学情報学部の西田研究室と大阪市立大学大学院創造都市研究科の松浦研究室の共同プロジェクトとして作られている、初学者向けのプログラミング学習環境である。JavaのRuntimeがあればOSを問わず使用することができる。

採用されているプログラミング言語xDNCLは、センター試験「情報関係基礎」で用いられているDNCLを拡張したものである。日本語ベースなので、初心者でも意味はわかりやすい。しかしその反面、キーボードから入力・変換するのは面倒だし、表記のゆれも起こりやすい。そこでPENには「入力支援ボタン」という機能が用意されており、使うことのできる構文はすべてボタンを押すだけで入力できるようになっている。これによって入力の手間が省けるだけでなく、分岐やループのブロック構造を意識することができるようになる。

ところが、生徒にとってブロック構造は普段見慣れないものであり、すぐには自然なものとして受け入れることができない。日本語を元にしてい

ることでとっつきやすい反面、文章と違って厳密な構文があるのだということが意識されにくく、むしろ日本語の文章として読んだときの違和感の方が気になってしまうようだ。そのため、エディタ画面上でわざわざ手動でブロックを壊してしまい、エラーを引き起こしてしまう。たとえば、

```
もし a=0   ならば
| 「ゼロ」を表示する
| 実行する
```

というプログラムを

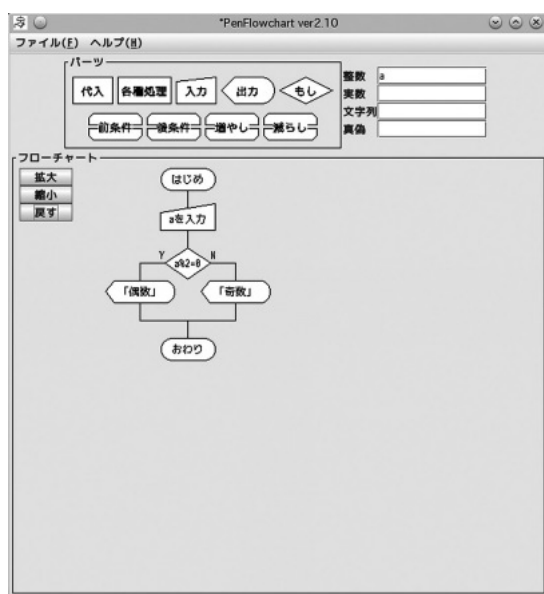
```
もし a=0   ならば
「ゼロ」を表示する
```

と修正してしまったりするのだ。おそらく日本語としては「もし～ならば、『表示する』を実行する」よりも「もし～ならば、表示する」の方が自然だと考えた上でそうしているのだが、現実にはエラーになってしまうのだからこれではいけない。

PenFlowchartについて

そこで、筆者はマウスでフローチャートを作成することによってプログラムが生成できるようにすればこのような問題を解決できると考え、2011年度の夏休みにPenFlowchartの開発に着手した。

PenFlowchartもJavaのRuntimeさえあればOSを問わず使用することができる。ドラッグアンドドロップでフローチャートの部品を並べ、式や値、条件などを編集すればプログラムが生成される。実行は連動するPENの画面で行なう。



初期のバージョンはPENと独立していたため、生成されたプログラムを手動でPENに読み込ませる必要があった。しかしプログラムを手直するたびに読み込みをし直すのは煩雑すぎるので、バージョン0.6からは直接PENの画面でプログラムを生成するようにした。実はそのためにPENのソースコードを丸ごと取り込んでいるのだが、そういったことができるのはPENがGPLで配布されているからだ。バージョン2.0ではプログラムをフローチャートに逆変換する機能を追加したが、そのための構文解析はPENのコードをそのまま利用している。

作成したフローチャートは画像として出力することもできる（形式はPNGとEPS）。これは授業用のプリントやスライド、テスト問題を作るのに

便利だろうと考えて実装した機能であり、実際にかなり役に立った。EPSはTeXで使いやすいからということで採用したが、ghostscriptがあればスライドに使うこともできる。ベクタ画像だから拡大しても大丈夫だ。

プログラムは筆者のサイトで配布している（アドレスは末尾に示す）。自由に改造できるようにソースを同梱しているため、展開した時のファイル数はやたら多いが、実行するだけならPenFlowchart.jarというファイルだけがあればよい。Windowsならこのファイルをダブルクリックすれば実行できるので、特別なインストール作業は必要ない（レジストリも使用しない）。Macでも同じように使えることは確認している。

3. 授業での使用

勤務校における実践

勤務校では情報Bの授業で、2006年度からPENを、2011年度の途中からPenFlowchartを用いてきた。開発を始めた2011年度は、PENを使った授業が数時間進んだところで唐突にPenFlowchartを紹介したこともあって、PENを使い続ける生徒が多かった（どちらを使うかは生徒に選ばせた）。あの頃はほとんど毎週のようにバージョンアップを繰り返し、使い勝手がどんどん変わっていったのだから、生徒がついてこれなかったのも無理はない。

2012年度は最初からPenFlowchartで授業を始めたので、ほとんどの生徒はフローチャートでプログラムを作成していた。しかし数名はフローチャートでなく直接PENの画面でプログラムを入力していた。見てみると入力支援ボタンをきちんと使いこなしているし、構造化も意識できているようなので、まったく問題はない。聞いてみると「いちいちダイアログを開くのが面倒」「いかにもプログラミングっぽくてかっこいい」とのことらしい。

一緒に授業をしている同僚からも好感触を得ている。PENだけで授業をしていたときは、机回巡視をしているとたびたび生徒に呼び止められ、

目を凝らしてエラーの原因を探してばかりいた。エラーの箇所にはすぐ気付くのだが、どのように修正させるかを指示するかで頭を悩ませる。入力支援ボタンを使えば簡単なのだが、そもそも彼らはそれがうまく使えないためにエラーを出しているのだ。その挙句、一行だけ直せばいいのに全部削除してしまっただけから入力をやり直してしまう。フローチャートからのプログラム生成によって、そういった効率の悪さから解放され、良いペースで授業を進めることができるようになった。

また、以前はプリントのフローチャートとプログラムを見て、まるでワープロ作業のようにプログラムを書き写す作業をする生徒が多かった。そういう生徒は構造が理解できないので入力支援ボタンを使いこなせないし、動かないことが明らかなプログラムでも平気で提出する。しかし、PenFlowchartを使うようになってからは、フローチャートを読んで作業ができる者が多くなった。実際、定期テストでは毎年フローチャートからプログラムへの（あるいはその逆の）書き換えを出題しているのだが、その問題で0点に近い点数を取る生徒の人数は格段に減少した。当初は、PenFlowchartに切り替えても文字を書き写すのが図を書き写すのに変わるだけで、フローチャートはわかってプログラムがわからなくなるのではないかと、ということを恐れていたのだが、その不安は当たらなかったようだ。

他校での利用

早い段階から専門学校や大学では使っていたのだが、高校ではそういう話をほとんど聞かなかった。前述したように頻繁なバージョンアップに対応するために、使わせる側が相当の配慮をしなくてはしなかったのだから、当然といえば当然のことではある。

しかし今は機能追加も落ち着いており、不具合の修正も進んだので、授業で使うには十分なものになっていると考えている。

4. 新課程「情報の科学」への適用

「情報の科学」では「問題の解法をアルゴリズム

を用いて表現する方法を習得させ、コンピュータによる処理手順の自動実行の有用性を理解させる」ことが学習指導要領に書かれている。そのために基本的なアルゴリズムを表現させ、自動実行させる活動が行なわれることになるだろう。

各社の教科書にもプログラムの作成や実行を行なわせる活動が記載されている。使われている言語はそれぞれ異なるが、フローチャートが記載されているのだから、PenFlowchartでその活動を行なうことはそれほど難しくない。むしろ、言語や環境個別の事情をあまり考えなくてすむ分だけ、アルゴリズムを考えることに専念しやすくなるのではないかと。

「情報の科学」は4社から合計5冊の教科書が出版されているが、ここでは実教出版の教科書を例に取り上げる。

実教出版の教科書

実教出版からは2冊の教科書が発行され、どちらも表計算のVBAを用いている。特徴的なのは、セルが変数の役割で使われていることだ。もちろんVBAでも普通に変数を使えるのだが、セルの値を操作することによって、値が変化することを観察しやすくしているのだろう。その狙いには感心させられた一方で、セルを2次元の座標で扱わなくてはいけないことがハードルになることを心配する。

記載されているプログラムをPenFlowchartに置き換えるためには、逆にセルを変数に置き換えることが必要になる。また、変数の初期化がシートで行なわれているので、プログラムではそれをコードとして明記しなくてはならない。しかしそれ以外はほとんどそのままフローチャートが使える。たとえば『302最新情報の科学』にある「交換法による並べ替え」のプログラムなら図1に示した通りである。セルの座標は1から始まるが、PENの配列の添え字は0から始まるので少し読み替えが必要ではあるが。

シミュレーションについても、プログラムで実現できる。たとえば「電子サイコロの作成」は教科書では表計算を用いているが、図2のようなプ

ログラムでも実行できる（結果は変数表示画面で見られる）。表計算ならそれぞれの結果が表示されてわかりやすいが、プログラムにも試行回数を簡単に変更できるというメリットがある。状況に応じてそれらを使い分けられることが理想だ。

ところで、同社の『303情報の科学』にはナンバープレイス（いわゆる数独）を自動で解くプログラムが記載されている（解ける問題は限られているが）。これは表計算であることがうまく生かされていて面白い。そこでJUMP編にあるそのプログラムをPenFlowchartでも作ってみたのだが、それを記載するにはこの原稿の余白は狭すぎる！実際、プログラムは150行程度であるが（変数の初期化で40行使っているので実際の処理は約100行）ネストは7段にもなっており、フローチャートの画像は縦16000ドットだ。

5. おわりに

アルゴリズムの学習ではプログラムを実際につけてみないと意味がない。そのためには何らかのプログラミング言語を選択することになるが、何が使えるかは各学校の事情によって違う。また、教員に得意な言語がある場合には、教科書のお仕着せには反発する面もあるだろう。ならばいっそ、教科書はフローチャートでアルゴリズムを示すだけにして、具体的なプログラムは別の印刷物として提供できないだろうか（冊子でなく、教員が学校で印刷するのでもいい）。そうすればさまざまな環境や言語に対応できるし、興味を持った生徒が多くの言語の存在に触れる機会を提供することにもなる。これは各社の教科書について、お願いしたいことだ。VBAやxDNCL, JavaScript, ドリトル, Rubyなどいろいろな言語が用意できれば楽しいと思うのだが。

PENやPenFlowchartで用いているxDNCLは学習用の言語である。しかし、それをもって実用的に用いられる言語に劣ると短絡してはならない。実用ということを言うのであれば、一つの言語ができるだけでは使い物にはならないし、授業で教わった程度のプログラミングが社会でそのまま使

い物になるわけではない。その意味ではVBAもJavaScriptも教科書では学習用言語なのだ。大事なのは、ここで学ぶものが最初のステップとして確かなものであればいいということだ。その先にあるものを我々教員が示せば、彼らは勝手に次のステップに進むだろう。

PenFlowchartについては、使っていただいた上で意見をいただければありがたい。筆者の作るプログラムはすべてそうなのだが、機能以外に何もない殺風景なインターフェースである。その善し悪しを自分では判断できないし、良くないとしてもどう改善したらいいかもわからない。また画像での出力についても、筆者はほとんどEPSしか使わないので、PNGのサイズなどが適切なのかわからない。筆者以外の人が使ってどう感じるかわからないので、ぜひとも教示をいただきたい。

なお、PenFlowchartは以下のアドレスでダウンロードできる。

<http://watayan.net/prog/>

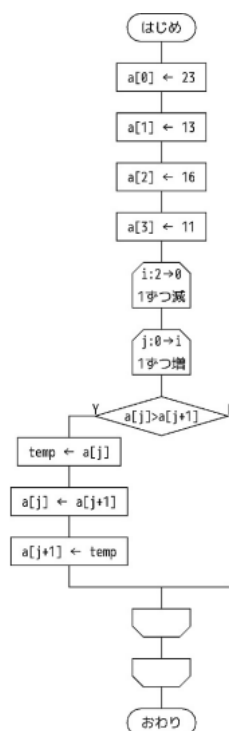


図 1

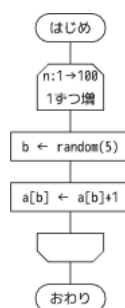


図 2