

プログラミング教育のための発展的な教材作成の実践と考察 - Java 言語による 3 目並べを題材として -

大庭孝則

神奈川県立相模原総合高等学校

Practice and consideration for developing teaching materials for programming education
- Tick-tack-toe using the Java language in the curriculum -

Takanori OOBA

Abstract

今後の社会は人工知能やインターネットを用いた大きな変革がなされると予測されている。第 4 次産業革命と呼ばれるそうした変化に向けて、学校教育ではプログラミング教育が重視されるようになってきた。この報告では、Java 言語の基礎を学んだ学習者が、次のステップとして取り組むプログラミング学習用の教材作成について考察した。教材に用いたのは新たに Java 言語で作成した「3 目並べ」という古典的なゲームであるが、コンピュータと人間が対戦できるようになっている。コンピュータをどのように考えさせるかという視点でアルゴリズムを組み立てることは、人工知能への興味を喚起する学習につながるのではないかと考える。

Key Words: Programming Education, Java language, Making teaching materials, Tic-tac-toe

1. プログラミング教育

高等学校での情報教育は、共通教科情報科を中心に行われている。共通教科情報科は「社会と情報」と「情報の科学」の 2 科目で構成されるが、プログラミング教育の観点からすると物足りなく感じられる点がある。というのは「社会と情報」では、プログラミングは学習内容として扱われておらず、「情報の科学」では「問題の解決と処理手順の自動化」を通じてプログラミングが取り扱われているものの、「情報の科学」の履修率は 2 割以下（2016 年度）と低い¹⁾ので、高等学校のプログラミング教育は充実しているとはいえないためである。

今後の社会は、進化した人工知能を用いて様々な判断をしたり、インターネットを通じて身近な機器が情報通信を行ったりする第 4 次産業革命によって大きな変化がもたらされると予測されている。近年の我が国の政策では、プログラミングをはじめとする情報の科学的な理解の重要性が指摘されていることにはそうした背景があると認識している。現在の神奈川県では、県立高校改革実施計画が進行中である。大庭が勤務する神奈川県立相模原総合高等学校も、計画の中で「プログラミング教育研究推進校」の指定を受け²⁾、2016 年度から 2018 年度までの 3 年間にわたり、プログラミング教育に関する研究開発を行っている。コンピュータを活用して、論理的思考

力を身につけ、協働して問題解決に取り組むことのできる人材を育成することがねらいである。

2. Java 言語によるプログラミング教育

プログラミング教育研究を推進するため、神奈川県立相模原総合高等学校では 2017 年度より、共通教科情報科の科目を従来の「社会と情報」に代えて「情報の科学」を設置した。また総合学科高校として、プログラミングを扱う選択科目を充実させた。商業科の科目「プログラミング」は 2003 年の開校当初から設置していたが、2017 年度より新たに「発展プログラミング」「アプリケーション作成入門」「ロボット制御入門」等の学校設定科目を開設し、プログラミング教育の推進を図っている。

取り組みの中心となっている科目は選択科目の「プログラミング」で、この授業は Java 言語によるプログラミングを扱っている。使用している教科書は「最新プログラミング オブジェクト指向型言語」³⁾だが、掲載プログラムはすべて統合開発環境の Eclipse を用いて作成されている。Java 言語を含めて使用ソフトウェアは無償で提供されているため、学校のコンピュータ教室など、コンピュータの台数が多い場合でも導入しやすい。教科書の後半では、Swing を利用した GUI プログラムの作成を扱っているため、この教科書で学習すれば、GUI を用いた実

用的なプログラムの作成の基礎を身につけることができる。そこでこの教科書を使って Java 言語プログラミングの基礎を学んだ学習者が、次のステップとしてどのような教材で学習するのが望ましいか、また教員が自作したプログラムを学習者が学びやすいように教材化するにはどのような手順が必要か、神奈川県立相模原総合高等学校の学校設定科目「発展プログラミング」での教材作成を例に考察した。

3. Java 言語による 3 目並べプログラム

教材を作成するにあたって考えたことは、生徒が興味を持って遊べるゲームを題材にするということであった。「遊べる」というのがポイントで、ゲームなら生徒たちは繰り返して実行するだろうし、繰り返しプログラムを実行するうちに、プログラム内容を自発的に発展させることができるのではないかと期待があった。思いついたのは、幼少時に遊んだ「3 目並べ (×ゲーム)」である。

「3 目並べ」ゲームというのは、2 人のプレイヤーが行うもので、通常は紙に「井」の字のようなゲーム盤を書いて行う。プレイヤーの 2 人は空白のセルに「 」と「 × 」を交互に書き込んでいく。縦・横・斜めのいずれか 1 列に 3 個自分のマークを並べると勝ちとなるものである。

このゲームを Java 言語のプログラムで表すにはどうしたらよいだろうか。選択科目「プログラミング」の教科書に扱われている範囲の技術で作成することを前提に、次のような方針を定めた。

まず、外見を作る。そのために Swing のフレーム (JFrame) を作成する。

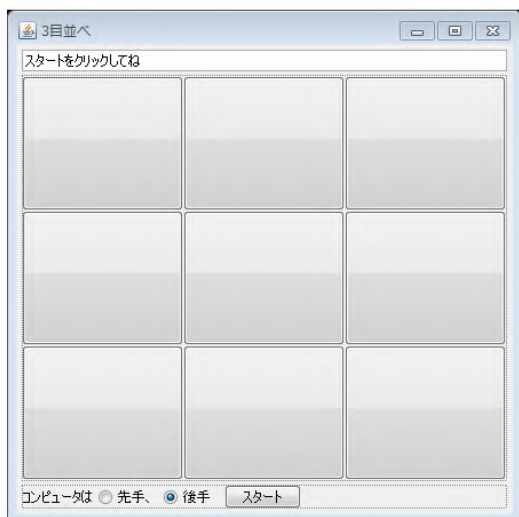


図 1 3 目並べプログラムの外見

フレームの上段 (North) にはメッセージ表示用のテキストフィールド (JTextField) を配置する。フレームの中央 (Center) には中間コンテナとしてパネル (JPanel) を配置し、その中に「 」や「 × 」を表示するためのボタ

ン (JButton) を 9 つ配置する。9 つのボタンは 3 行 3 列に揃えて表示するため、中央の JPanel のレイアウトには 3 行 3 列の GridLayout を設定しておく。フレームの下段 (South) にはもう一つパネル (JPanel) を配置して、「コンピュータは」と表示するラベル (JLabel)、先手・後手を選択するためのラジオボタン (JRadioButton) とゲームのスタートボタン (JButton) 1 つを配置した。



図 2 配置したコンポーネント

次にプログラム内で使用する変数を考えた。プログラムを簡潔にするため、変数はなるべく少なくするように考えて、次の 3 つとした。

- (1) 9 つのセルの状態を保持する整数型配列
- (2) 現在何手目かを示す整数型変数
- (3) ゲームが終了したかどうかを示す整数型変数

最後に行ったのがプログラムのコーディングである。各ボタンのクリックに対応するイベント処理とは別に、追加したのは次の 5 つのメソッドだった。

- (1) コンピュータが打つセルを、0 ~ 8 のセル番号で返すメソッド。
- (2) あるセルが空白かどうかを判定するメソッド。
- (3) マークの並びからゲーム終了を判定するメソッド。
- (4) 9 つのセルに「 」 「 × 」を表示するメソッド。
- (5) 1 つのセルに文字を与えて表示するメソッド。

このうち (1) がコンピュータの思考ルーチンにあたるメソッドである。最初は単純に、空のセルを見つけて 0 ~ 8 のセル番号を返すだけのメソッドとした。返り値が 9 の場合は打つ場所がないことを示している。

```
private int think(int n){
    int c;
    for(c=0; c < 9; c++){
        if(mas[c] == 0) break;
    }
    return c;
}
```

リスト 1 初期段階の思考ルーチン

プログラムの仕組みは、9つのセルに図3のように番号を振り、整数型の配列 mas[0] ~ mas[8] に各セルの状態を保持させることによる。配列 mas の値が 0 ならば空、1 ならば「 」, 2 ならば「 x 」を示す。コンピュータが手を打つということは、配列 mas のいずれかに 1 か 2 の値を代入することで実現している。

0	1	2
3	4	5
6	7	8

図 3 9つのセルに与えた番号

この段階で実行してみると、人对コンピュータの対戦はできるものの、コンピュータは勝敗を度外視して、ただ空のセルに打ってくるだけなので、弱い。人が同じマークを 2 つ揃えて、あと一手で勝つ段階になっても、無関係のセルに打ってくる。

そこで同じマークが 2 つある列を調べ、マークが 2 つあれば同じ列のセルに打ってくるようにメソッドを追加した。

この chk2koma メソッドを使って、同じマークが 2 つ

```
private int chk2koma(int n){
    int[][] kPtn = {{0,1,2},{3,4,5},{6,7,8},{0,3,6},{1,4,7},{2,5,8},{0,4,8},{2,4,6}};
    for(int i=0; i < 8; i++){
        if(mas[kPtn[i][0]]==0 && mas[kPtn[i][1]]==n && mas[kPtn[i][2]]==n)
            return kPtn[i][0];
        else if(mas[kPtn[i][0]]==n && mas[kPtn[i][1]]==0 && mas[kPtn[i][2]]==n)
            return kPtn[i][1];
        else if(mas[kPtn[i][0]]==n && mas[kPtn[i][1]]==n && mas[kPtn[i][2]]==0)
            return kPtn[i][2];
    }
    return -1;
}
```

リスト 2 同じマークが 2 つある列の空きセル番号を返す chk2koma メソッド

ある列を探し、見つからない時だけ空のセルに打つよう、think メソッドを改良すると、コンピュータは「強く」なり、油断すると人が負けるようになった。

3 目並べは最大で 9 手までしかない単純なゲームである。突き詰めれば、常に最善手を持つアルゴリズムを作り出すことができるだろう。しかしこれ以上の改良は学習者に任せることとして、教材としてはこれで完成とした。

4. Java 言語プログラムの教材化

教員が自作したプログラムを教材化する場合に、最初に行うのは、授業で学習者にどのような作業をさせるか、学習者は作業を行いながらどのような課題を解決するのかを決定することである。その際のヒントになるのは、プログラムの作成過程で教員自身が直面した課題であり、これらをどのような手法で解決したかということは、教材作成の有力な手掛かりになる。ただし教員が試行錯誤した制作過程をなぞるだけでは適切な学習教材にはならない。学習者に対する課題の提示、問題解決の道筋を見出すことが必要である。

こうした考えに基づき、「3 目並べ」プログラムは 4 回の授業で扱うこととし、各回の授業テーマを次のように定めた。

- 第 1 回、Swing コンポーネントでゲームの外見を作る。
- 第 2 回、ゲームの勝敗を判定する機能を付加する。
- 第 3 回、コンピュータ対人の対戦ができるようにする。
- 第 4 回、コンピュータの思考ルーチンを強化する。

各回のテキストは 4~8 ページの内容にまとめて授業で配付することにした。テキストを作成する際に心掛けたことは、1 回の授業ごとに学習者が実行可能なプログラムを作成するという点であった。プログラムの機能が不十分でも、起動させて GUI の操作ができるものを作った。たとえば第 1 回の授業では、実行させるとゲームの外見を表示するだけのプログラムを作成した。セルをク

```

private int think(int n){
    int c = 9;
    if(chkEnd() > 0) return c;
    if( n==0 ){ // コンピュータが先手なら
        if((c=chk2koma(1))>=0){ // をチェック
            return c;
        }
        if((c=chk2koma(2))>=0){ // xをチェック
            return c;
        }
    }else{
        if((c=chk2koma(2))>=0){
            return c;
        }
        if((c=chk2koma(1))>=0){
            return c;
        }
    }
    int rnk[]={4,6,2,8,0,7,5,3,1,9};
    for(c=0; c < 9; c++){
        if(mas[rnk[c]]==0) break;
    }
    return rnk[c];
}

```

リスト3 強化した思考ルーチン

リックしても反応はないが画面にはゲーム盤が表示される。第2回では人対人の対戦だが、ゲームの勝敗を判定できるようにした。第3回では人対コンピュータの対戦ができるようにして、第4回ではコンピュータ思考するように改良した。

各授業ごとに実行可能なプログラムを作成するようになったことには2つの意味がある。一つは学習者に達成感をもたせるためであり、もう一つは学習者に未完成のプログラムを操作しながら、次に追加すべき機能とそれを実現するためのアルゴリズムを考えさせるためである。授業では、その授業での課題を解決し、実行可能なプログラムが出来上がったところで、次はどこを修正し補強するかを考えさせた。

5. おわりに

有史以来、人類はさまざまな物を発明した。コンピュータは人間の発明品の中でも最高のものの一つであろう。プログラミングは、人類最高の発明品であるコンピュータを人の考え通りにコントロールする技術である。

学校でのプログラミング教育は、コンピュータを活用して、論理的思考力を身につけ、協働して問題解決に取り組む人材を育てることが目標である。それを実現するには、学習者がコンピュータを自由に操る喜びを実感することが、ぜひとも必要であると考えられる。実際のプログラミングには集中力と持続力が求められる。コンピュータを操る喜びこそが、プログラミングに対するモチベーションを維持する力になるからである。

3目並べプログラミングの授業で、人対コンピュータの対戦ができるプログラムを作成したのは第3回の授業でのことであった。この段階では、コンピュータは空のセルを探して、その場所に打ってくるだけである。

次の授業では、コンピュータの思考ルーチンを強化して、ある程度「強い」3目並べプログラムを完成させる予定であった。そこで次の授業への課題として、生徒たちにヒントを出した。

「コンピュータを強くするための一つの考え方が、同じマークが2つある列を見つけた場合、マークが自分のものなら空のセルに打てば自分の勝ちでゲーム終了、マークが相手のマークであっても、空のセルに打たないと自分が負けになる。いずれにせよ2つ同じマークがある列の空のセルには必ず打たなければいけない。」

しばらくたって、普段は寡黙な生徒の一人が「やった。コンピュータが負けないアルゴリズムができた」と声を上げた。「どれどれ」と向かい側の席の生徒は立ち上がって彼のディスプレイを覗き込んだ。

別のある生徒は、休憩のときに「先生のプログラムはカドが空いているとそこに打ってくるんですが、中央を優先したほうが勝率が上がりませんか」と話しかけてきた。

生徒とのやり取りの中で気づいたことは、コンピュータの思考ルーチンのアルゴリズムを考えることは、コンピュータを操る実感を得るのに適したものであるということだった。コンピュータが考える手順をプログラミングし、そのプログラムを実行させる。作成者が正しく考えてプログラムすれば、コンピュータは想像以上の働きをする。プログラムを作った本人を、コンピュータが負かしてしまうことさえあるのだ。こうした体験は、将来、人工知能のような大量のデータに基づいた思考ルーチンの開発につながることも可能であろう。このような可能性を育むためにも、教材作成者には、生徒にとって身近で興味深く、よりプログラミングのモチベーションを高めるような教材開発を行うことが求められると考える。

授業で使用したテキスト教材は、神奈川県立相模原総合高等学校のホームページ⁴⁾からダウンロード可能である。

参考文献

- (1) 文部科学省, 教科書の種類数・点数・需要数(平成28年度用)。
- (2) 神奈川県教育委員会, 県立高校改革実施計画(全体), 2016。
- (3) 中澤興起ほか7名, 最新プログラミング オブジェクト指向型言語, 実教出版株式会社, 2017。
- (4) 神奈川県立相模原総合高等学校ホームページ, <http://www.sagamiharasogo-ih.pen-kanagawa.ed.jp/>