

IT マスターによる小中学校向けプログラミング教育に関する考察

神山 司¹

概要：厚生労働省による IT マスターとよばれる制度がある。これは小中高校の頃から子供たちが IT に慣れ親しむための制度であり、IT に関する熟練技能者として登録された IT マスターを小中高校からの要請によって派遣し授業を行うという事業である。現状ある IT マスターの授業の中でもプログラミングに着目すると小学校向けのロボットを用いたもの、中学校向けのカラーカードを用いたものとあり、これらは所謂プログラミング言語を用いないものである。IT マスターにおけるプログラミング授業と一般的な大学におけるプログラミング言語を用いた講義と比較する。

キーワード：IT マスター、プログラミング授業、ロボットプログラミング

Report of programming education by IT Master for elementary and junior high school

TSUKASA KAMIYAMA¹

1. はじめに

労働生産性を上げるための手段の一つに ICT を有効利用することが上げられる。ICT を有効利用するためには ICT を使いこなす人材が不可欠である。この人材を育てるには小中学校の世代から ICT に慣れ親しむ環境を作る必要がある。

そのため厚生労働省の事業に IT マスター [1] が出来た。

2. IT マスター制度

ICT 関連の特に優れた技能者を“IT マスター”として認定及び登録を行い小中学校へ派遣し講習などを行うことで児童及び生徒に ICT に触れる機会を作っている。これを IT マスター制度としている。

IT マスターが指導するのは以下の 5 つの職種になる。これらの職種は技能五輪全国大会もしくは若年者ものづくり競技大会で競技として実施されているものである。

- ウェブデザイン

- IT ネットワークシステム管理
- グラフィックデザイン
- オフィスソフトウェア・ソリューション
- ロボットソフト組込

実務経験や所有資格などの要件を満たした技能者が各都道府県の地域技能振興コーナーへ申請することで IT マスターとして認定、登録される。

IT マスターの講習をする場合は各小中高等学校が地域技能振興コーナーに派遣を依頼することができる。

3. IT マスターによる授業

IT マスター事業では以下の内容の事業が用意されている。

- 小学生向け
 - － ロボットプログラミング
 - － グラフィックデザイン
- 中学生向け
 - － プログラミング
 - － グラフィックデザイン
 - － IT リテラシー

¹ NPO 法人インターネットスキル認定普及協会

- ウェブデザイン
- 高校生向け
- IT リテラシー
- ウェブデザイン

ここではプログラミングの授業に着目する。

3.1 小学生向けロボットプログラミング

小学生向けのロボットプログラミングについて詳細を述べる。2017年度においては88回ほど授業として行われている。

基本的には小学生に対しては45分×2の90分間となる。小学生向けのロボットプログラミングではNAOというロボットとタブレットを用いる。

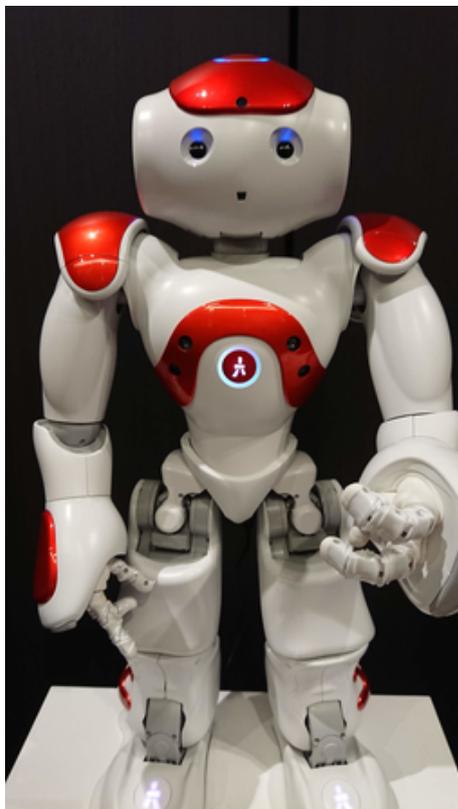


図1 NAO

教材はプログラミング言語に依存しない問うことを前提に設計されている。図2のようなビジュアルプログラミングの環境が用意されており、タブレット上のウェブブラウザ上で動作する。ここで作成したプログラムをロボットの中にあるHTTPサーバに送信することでロボットを喋らせたり動かしたりできる。

IT マスターロボットプログラミング

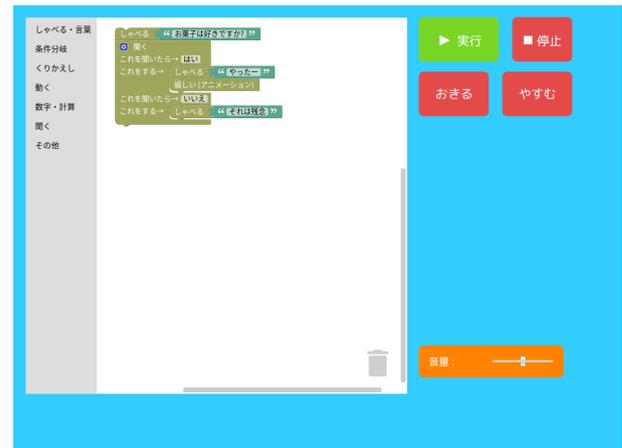


図2 IT マスターロボットプログラミングの画面

授業で使用するロボットは4台、タブレットを8台であったが、最近では講師用に使用していたロボットも児童が使うという運用に変更された。これによりロボットが5台、タブレット10台という構成になった。

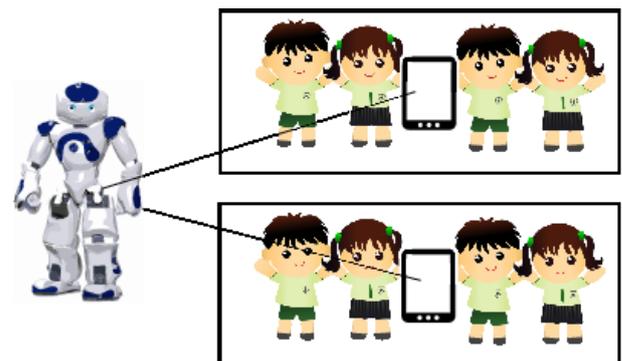


図3 NAOとタブレットの共有

例えば30人のクラスであれば6人ずつの班に分かれて、それぞれの班にロボットが1台、タブレットが2台となる。6人の班では3人ずつのグループに分かれて各グループでタブレットを共有する。

机を使用せずにロボットの周りに車座のようになって授業を行う。これはロボット(NAO)が高価なため机の上からの落下などをなくすためである。

3.2 授業内容

これらを使って以下の内容の授業を行う。

- 処理：喋る、アクションをする
 - 条件分岐：NAOが聞いた言葉を条件に使う
 - 繰り返し：回数での繰り返し、及び抜ける
- 演習はロボットを喋らせる“処理”から始まり、条件分岐、繰り返しと続いていく。
- 最後に繰り返しの内容までを含めて“朝起きてから学校

にいくまでロボットにチェックしてもらおう”プログラムを作成する。例えば「朝ごはん食べた?」の問いをロボットが行い、ユーザ(児童)が「はい」や「いいえ」で答えて、その結果によって再び「朝ごはん食べた?」の問いを繰り返すようなプログラムである。

3.3 中学生向けプログラミング

中学生向けプログラミング教材について説明する。これは2018年10月に初めて実施された。

中学生に対しては50分×2の100分間の講習となる。

生徒がプログラミング言語を習得する必要がなくロジックの組み立てを出来るようにプログラミング言語に依存しないような教材になっている。これは小学生向けのロボットプログラミングと同様である。

実際の演習では色のついたカードを用いる。各色に表示や条件分岐、繰り返し等を割当てて生徒がそれらを並べることでプログラミングを行う。

カラーカードを使用するのは生徒全員分のスマートデバイスを用意することが不可能であっても、カラーカード程度であれば全員分の配布は可能であるためである。生徒がスマートデバイスを交代で使うとしても、プログラムの確認時以外はスマートデバイスなしでプログラムを作成できる。

図4のように並べられたカラーカードをITマスターがスマートデバイスで撮影することでそのデバイス上でプログラムが実行される。教室の証明などで色の認識がうまくいかない時は認識ではなくマニュアルでカラーカードの並びを入力することが可能である。



図4 カラーカードを用いたプログラミング

カラーカードの凡例は以下のようになる。

- 青 : 数を表示 (繰り返しの中では何回目か)
- 赤 : 繰り返し (50回)
- 黄 : もしも2で割り切れるなら
- 水色 : もしも3で割り切れるなら
- 紫 : もしも5で割り切れるなら
- 緑 : もしも15で割り切れるなら
- 黒 : そうでないなら
- 茶 : Fizz を出力
- ピンク : Buzz を出力
- オレンジ : FizzBuzz を出力
- 灰色 : 何もしない

これらを組み合わせることで課題のプログラムを作成していく。

また、中学生向けのスライド教材にはウェブアプリ版があり、ブラウザ上で生徒たちが送信したプログラムの実行結果をスライド内で見ることができるようになっている。



図5 生徒の作ったプログラムをスライドに表示

3.4 授業内容

小学生同様に、処理、条件分岐、繰り返しの解説を行う。その後カラーカードを用いた演習となる。

- 1~50の数を表示する
- 1~50の偶数を表示する
- 1~50の奇数を表示する
- 1~50の3の倍数を表示する
- FizzBuzz 問題

“1~50の数を表示する”場合は以下のようにカードを並べる。

結果



図6 1~50の数を表示する

このカラーカードを撮影し、サーバに送ることで結果がスマートデバイスに表示される。この場合は50個の数字となる。また、カードの並びをJavaScriptでコード化したものも同時に表示される。この場合は以下のコードが表示される。

```
for (var i=1; i<=50; i++)  
  this.print(i);
```

最後の課題は FizzBuzz 問題となっている。以下のルールで表示を行う。

- 1~50 の数を出力する。ただし
 - 3 で割り切れる場合は数の代わりに Fizz を出力する
 - 5 で割り切れる場合は数の代わりに Buzz を出力する
 - 15 で割り切れる場合は数の代わりに FizzBuzz を出力する

この課題の前に “3 の倍数を表示する” をしているので Fizz の表示はほぼ問題なくすすめるのだが、条件分岐の順番により想定している出力結果を得られない生徒が大半のようである。以下のようなコードを生成するカードの並びが正解になる。

```
for (int i=1; i<=50; i++){  
    if ( i%15 == 0 )  
        printf(" FizzBuzz\n");  
    else if ( i%3 == 0 )  
        printf(" Fizz\n");  
    else if ( i%5 == 0 )  
        printf(" Buzz\n");  
    else  
        printf("%d\n", i);  
}
```

しかしながら、以下のようなコードを生成するようにカードを並べる生徒が多い。

```
for (int i=1; i<=50; i++){  
    if ( i%3 == 0 )  
        printf(" Fizz\n");  
    else if ( i%5 == 0 )  
        printf(" Buzz\n");  
    else if ( i%15 == 0 )  
        printf(" FizzBuzz\n");  
    else  
        printf("%d\n", i);  
}
```

このようなプログラムを考える生徒に対しては、少し示唆を与えることで正解にたどり着く者が多いようである。

このように小学生向けと異なり、より論理的思考を身に付けるような教材になっている。

4. 一般的なプログラミング教育との比較

ここでは大学などで行うプログラミングの講義との比較を行う。

学生がプログラムを始めるための環境の一例として以下のものがあげられる。

- OS(実行環境)
- エディタ

• コンパイラ

IDE(統合開発環境)を用いる大学もあるが、ここでは gcc などのコンパイラを使用することを想定している。

C プログラミングの入門用の講義のカリキュラムの一例を載せる。

- (1) 講義概要
- (2) UNIX コマンド
- (3) 変数・標準入出力 (1)
- (4) 変数・標準入出力 (2)
- (5) 条件分岐
- (6) 繰り返し (1)
- (7) 繰り返し (2)
- (8) 配列・マクロ (#define)
- (9) 乱数・数学ライブラリ
- (10)関数
- (11)ポインタ
- (12)文字と文字列
- (13)ファイル入出力
- (14)構造体

上記にあるような大学での 6 回相当までを、IT マスターの講習においては 小学校で 90 分、中学校で 100 分で教えていることになる。C 等の場合は その文法的な要素も習得する必要があるが、ビジュアルプログラミングの環境においてはそのコストが不要であるということも大きい。

中学生向けの授業では若干の示唆を与えることで生徒自身で FizzBuzz のプログラムを完成させている。この FizzBuzz は大学の講義で試験やレポートの課題になっているものである。

IT マスターの教材に限らず Scratch などであっても同様であるが、若年世代が論理的思考などを身に付けるためのプログラミングという意味ではビジュアルプログラミングは強力な手段の一つであるということがわかる。

5. まとめ

厚生労働省の事業の 1 つである IT マスター制度で行っている ICT の授業について報告した。

この制度の目的は ICT を有効利用できる人材を若年者のうちから育てるということであるが、大学のプログラミングの講義で試験などに使われる課題のプログラムを中学生が作成している状況を見ると、この目的に合致した制度であるということがわかる。

参考文献

- [1] : ものづくりマイスターデータベース [IT マスター]
<https://www.monozukuri-meister.javada.or.jp/mm/mm/contents/shokai/it-index.html/index.html>.