

自律型ロボット入門教材に関する研究

吉田昌春*・山崎千恵子**・吉田竹虎***

On an Introductory Learning Tool for Autonomous Robot

Masaharu YOSHIDA, Chieko YAMASAKI and Taketora YOSHIDA

ロボットとしては操作型と自律型がある。中学生など初心者が扱うものとしては前者が主流であり、技術的ものづくりとしての要素を多く含む。一方、自律型はセンサによりロボットが状況判断をして行動するものであり、プログラム学習が多くの部分を占める。ここでは、自律型ロボットの入門として車型のロボットを導入し、C言語を用いてプログラム学習の進め方を提案し、選択の時間に授業実践を行った。while文を使用した構文の理解が主であり、forward, right, left, stopなど中学生にわかりやすい命令の組み合わせによりライントレースなどのプログラムを学習させ、授業展開について一定の評価を行った。

キーワード：車型ロボット, ライントレーサ, 光センサ, C言語, プログラム構造

1. はじめに

「技術とものづくり」, 「情報とコンピュータ」でくられる技術教育の中でロボットには両者の多くの要素が包含されている。

ロボットには操作型と自律型があり、中学生の入門レベルでは主に操作型ロボットが対象になる。現に2001年度から全日本中学校技術・家庭科研究会が全国の中学生を対象にしたロボット大会を開催している。今後、ロボットは技術教育の中で伸びていくことが大いに期待される題材のひとつである。近年、ホンダが開発した二足歩行ロボットのアシモや愛知万博で人気を集めたトヨタグループ館のi-foot (搭乗二足歩行型ロボット) などの自律型のロボットが多く知られるようになってきた。このようなプログラムをからめた自律型のロボットの入門として車型ロボットが馴染みやすいと思われる。

本研究室では、光センサ, モータを組み込んだ車型ロボットに関して、中学生を対象に授業展開のあり方をした。プロセッサとしてはPIC 16F84AとADuC814を機能的に比較した上で中学生の生徒にADuC814を使用した自律型ロボットのプログラムの構造を理解させるためにどの

ように教えていけばいいのか, またADuC814をどのように生活に生かすことができるかについて研究を進める。

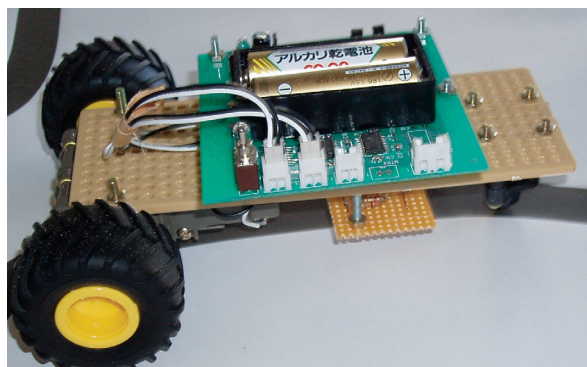


図1. 車型ロボット

2. ADuC814制御ボードを使った車型ロボット

2-1. ロボフェスタ2005 in GIFU

ロボフェスタ2005 in GIFUは、平成17年8月11日, 12日に各務原市産業文化センターで開催された。その際に行ったものづくり教室の企画の概要を先ずは述べる。

対象は中学生とし、午前中はTAMIYAの「LINE TRACING SNAIL KIT」を子どもに作らせ、製作を楽しんでもらうと同時にセンサの働きを理解させてライントレーサロボットに興味をもたせた。午後から使用した車型ロボットは、ADuC814を搭載した制御ボードを使用

* 岐阜大学教育学部技術教育講座

** 岐阜大学教育学部技術教育講座

*** 岐阜大学教育学部附属中学校教諭(H.17)

したもので、2個の反射型光センサと2個のモータを装着した。プログラムはC言語で記述するものである。プログラムを子どもたちに考えさせ、障害物回避のコースとライントレースのコース(2周)のタイムの合計を参加した子どもたちを4グループに分けて、競争させた。ただし、1つのグループに対し、大学生1名がアドバイザーになった。このロボフェスタでは、子どもたちにプログラムについて関心をもたせ、自分が考えたプログラムをすることによってロボットが自分の思い通りに動かすことができることの楽しさに触れ、プログラムを少しでも理解することができるように考えた。参加人数は以下の通りである。

8月11日は12人

(中学1年4人, 2年4人, 3年4人)

8月12日は15人

(中学1年10人, 2年2人, 3年3人)

表1 当日のスケジュール

時間	内容
9:00	受付開始
9:30	開講 挨拶 (ロボット入門としてのトレサ) スケジュールの説明 チュートリアル (ライントレーサはどうして動く) …センサの話
	作業開始
11:00	完成・ラインのコースで遊ぶ
11:30	プログラム型かたつむりライントレーサでも遊ぶ プログラムと動きの関係を学習する
昼食	
13:00	プログラムを使用したライントレーサに挑戦しよう(非かたつむりロボ) チュートリアル(プログラムの作り方) センサに無関係に50cmの四角を描く動き 自分達で課題設定とプログラムの完成・ダウンロード
	ラインに沿って走らせよう 障害物を回避して目的地へ行こう(競技) 障害物の位置とゴールの位置はラインで示す
16:00ごろ	表彰式・閉講

当日のスケジュールを表1に示す。

2個のセンサを使ってラインに沿って走るプログラム例を図2に示す。

```
#include <robo.h>
void main(void){
    int r,l;
    init();

    while(1){
        r=adin(0);
        l=adin(1);
        if(r>2000 && l>2000){forward(1);}
        else if(r>2000 && l<2000){right_forward(1);}
        else if(r<2000 && l>2000){left_forward(1);}
        else if(r>2000 && l>2000){back(1);}
    }
}
```

図2. ラインに沿って走るプログラム例

図3のようなコースで障害物を避けてゴールに到達するプログラム例を図4に示す。この場合は1個のセンサを用いる。チームごとに上の2例のプログラムを考えさせて、競争させた。3~4人を1チームとする学習で、子どもたちは互いに初対面であったが、うち解けあって相談しながら熱心にプログラムを考えていた。

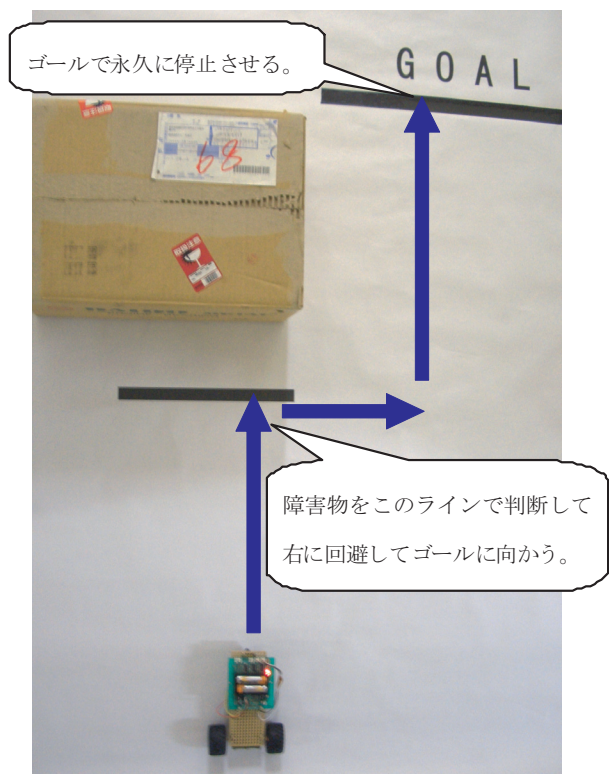


図3. 障害物を避けてゴールへ至る課題

```
#include <robo.h>
void main(void) {
    init();

    while( adin(0)>2000) {forward(1);}
    right(80);
    forward(250);
    left(48);
    while( adin(0)>2000) {forward(1);}
    stop(1);
    while(1){ } }
```

図4. 障害物を避けてゴールに至るプログラム例

2-2. ロボフェスタのものづくり教室を通じた問題点・改善点

中学1年生には短時間でC言語を理解することが少し難しかったように思われる。C言語の命令は英語に似ているため、2, 3年生には少し理解することができたように思われる。しかし、構文・プログラム構造について十分理解するには時間不足であったように思われる。

また、4つのグループに分けたことにより、一つのグループ3~4人の子どもが、一緒にプログラムを考えることができたので、お互いに案を出し合って相談することができた。また、短時間内にプログラムを理解することは難しいと思われたが、質問紙調査で少しの人がプログラムを理解したと答えている。これは、プログラムの個々の命令が理解できたものと思われる。したがって、さらに時間をかけてプログラム構造について教えていき、簡単なプログラムを作らせるような練習を多く設ければ、プログラム構文をより理解することができると思われる。

以下ではこの問題点・改善点からライントレーサのプログラム構造理解のための授業展開を提案し、授業実践を行い、省察を行う。

2-3. 中学校での授業展開

岐阜大学教育学部附属中学校3年生、17名を対象に選択の授業で実践を行った。生徒の内訳は既履修者10名、新規履修者7名であった。既習者は車型ロボットの制御を前期に学習したものである。既履修者と新規履修者での差が多少あるため、既履修者を先生役としペア学習で授業を進めていくこととした。

既履修者は、あらかじめ用意したプログラムを少し変更して、コンパイル・ダウンロードすることができる程度である。

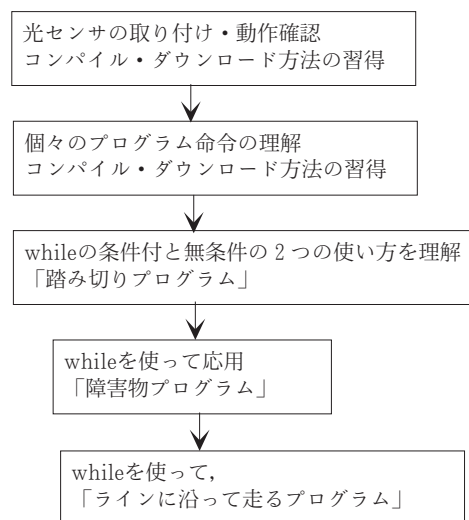
新規履修者は、まず自分のマシンの製作を行い、既履修者はそれを補助した。

生徒の車型ロボットは図1のものを使用した。ただし、反射型の光センサの数は1個にした。また、センサの取り付けは、コネクタを用いて取り外し可能とした。

2-3-1. 授業計画

授業の展開としては、50分授業を合計5時間とし、プログラムの構成について理解させることを目標として授業設計を行った。

全体的な指導構想は次のブロック図の通りである。



次に、時間ごとに使用したプログラムを示す。図5に1時間目に使用したプログラムを示す。これはセンサの動作チェックのために使用した。

```
#include <robo.h>
void main(void){
    init();
    while(adin(0)<1600){forward(1);}
    stop(1);
    while(1){ }
}
```

図5. 1時間目のプログラム

図6に2時間目に使ったプログラムを示す。こ

のプログラムを与えて、生徒に引数の値を変更させて、その意味を理解させた。

```
#include <robo.h>
void main(void){
    init();
    forward(300);
    left_forward(300);
    stop(100);
}
```

} ここを変更
させる

図6. 2時間目のプログラム

図7に3時間目に作ったプログラム例を示す。このプログラムは踏み切りで一旦2秒停止し、その後3秒進んで永久に停止させるプログラムである。

```
#include <robo.h>
void main(void){
    init();
    while(adin(0)<1600){forward(1);}
    stop(200);
    forward(300);
    stop(1);
    while(1){ }
}
```

図7. 3時間目のプログラム

図8に4時間目に作ったプログラム例を示す。このプログラムはスタートラインから障害物まで直進し、その後、それを避けてゴールまで到達して永久に停止させるプログラムである。

```
#include <robo.h>
void main(void){
    init();
    while(adin(0)<1600){forward(1);}
    right(80);
    forward(150);
    left(80);
    while(adin(0)<1600){forward(1);}
    stop(1);
    while(1){ }
}
```

図8. 4時間目のプログラム例

図9に5時間目に作ったプログラム例を示す。このプログラムはラインの左端に沿って永久に走るいわゆるライントレーサのプログラムである。

```
#include <robo.h>
void main(void){
    init();
    while(1){
        while(adin(0)<1600){right_forward(1);}
        left_forward(1);
    }
}
```

図9. 5時間目のプログラム例

2-3-2. 授業の様子

3時間目は、whileの2つの使い方を勉強し、踏み切りプログラムをみんなで考えて実際に動かさせた。

4時間目は、当初はラインに沿って走るプログラムを考えさせる予定であったが、whileの応用として障害物を避けてゴールまで到達するプログラムを考えさせることにした。

5時間目は、最初はifの構文を使用して、ラインに沿って走るプログラムを考えさせることを予定していたが、生徒の混乱を避けるために、whileを使って授業展開していくことにした。授業の最後には、ロボットは遊びのように使うことだけでなく、人が行けないところにロボットが行って人命救助を行うなど一般に人の役に立つために使われていることを紹介し、人間の生活に対するロボットの位置づけを認識させた。

4時間目と5時間目では、プログラムの日本語理解をして、生徒それぞれでプログラムを考えて作成するようにした。また、車体が軽いためにタイヤがスリップしてしまう生徒が多く見られた。5時間目はそのために重りの搭載を考えたが適切なものが見あたらなかった。

授業の様子は、生徒自身、プログラムやマシンに対して勉強したい気持ちがあるので、前向きに勉強に取り組むことができていた。

また、スムーズにプログラミングできるようにFD3.5インチにあらかじめプログラム構造を書いておき、生徒にそれぞれ配布した。

例えば，3時間目に準備したプログラムは図10に示すものである。

```
#include <robo.h>
void main(void){
    init();
    while(adin(0)<1600){forward(1);}
    stop(1);
    while(1){ }
```

図10. 3時間目に準備したプログラム

この授業展開をすることができたのは，岐阜大学教育学部附属中学校では，2年生の授業で「C言語で制御するマシン」の授業を行っており，同じADuC814のCPUを用いた株式会社イーエスピー企画から販売されている制御ボードを使用していたからである。また附属中学校の吉田竹虎教諭がプログラミングソフトウェアやダウンロードソフトウェア，ヘッダファイルを事前にサーバに書き込んで環境整備を行ったからである。

なお，図11～13に授業の様子を示す。

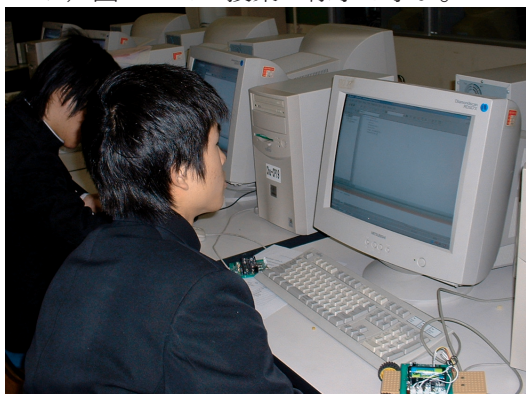


図11. 授業の様子①

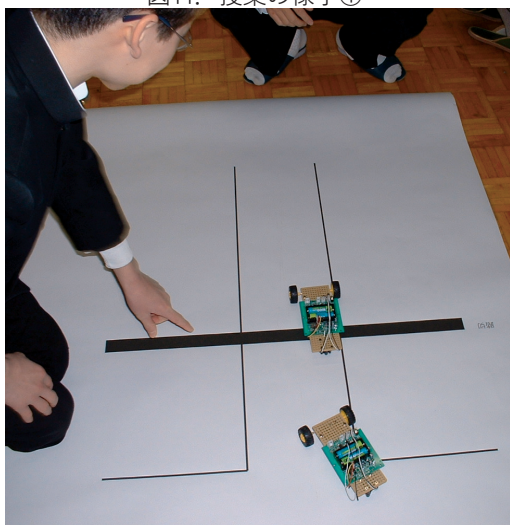


図12. 授業の様子②

図13. 授業の様子③

2-4. 質問紙調査と検討

授業開始時の質問紙調査では，図15, 16, 18, 19の結果からプログラムの命令が分からない生徒が既履修者の中にも見られた。また，新規履修者の中にはプログラムの命令が分かる生徒もいた。

また，図20, 21の結果からwhileの永久ループの使い方に関しては既履修者にも分からない生徒が何人か見られた。

2時間目で命令に少し触れることができたので，3時間目の確認プリントで，命令を理解していることが確認できた。時間(秒)に関してはほとんどの生徒が理解することができていた。

また，whileの2つの使い方についても生徒ほとんどがしっかりとまとめることができていたので，理解できていると思われる。

```
#include <robo.h>
void main(void) {
    init();

    forward(300);
    stop(100);
    right(300);
    stop(300); }
```

図14. 提示したプログラム①

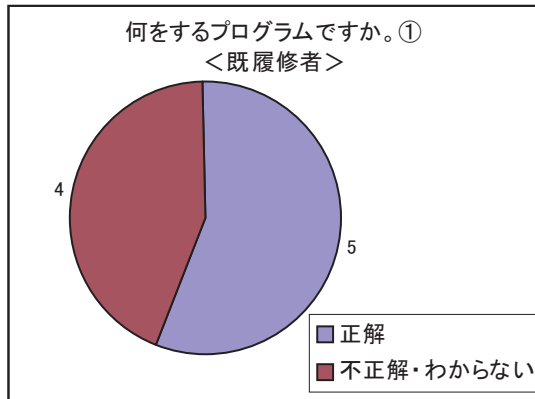


図15. プログラム理解①(既履修者)

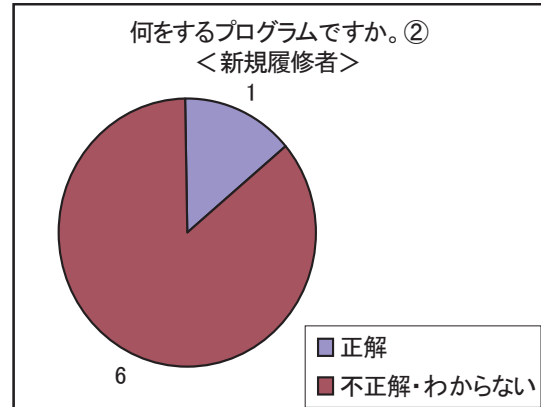


図19. プログラム理解②(新規履修者)

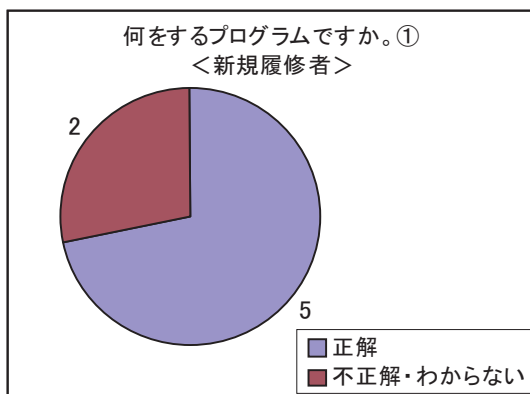


図16. プログラム理解①(新規履修者)

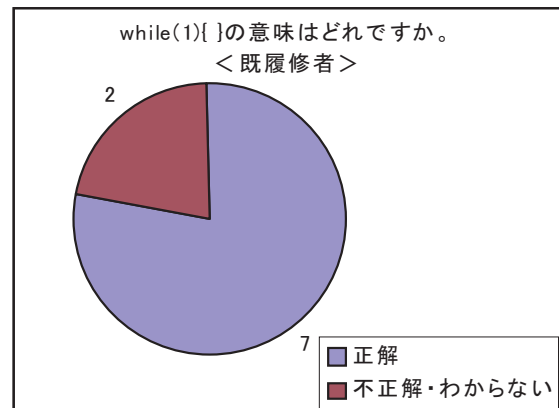


図20. while(1)の意味理解(既履修者)

```
#include <robo.h>
void main(void){
    init( );
    while(1){
        forward(300);
        left(100);
    }
}
```

図17. 提示したプログラム②

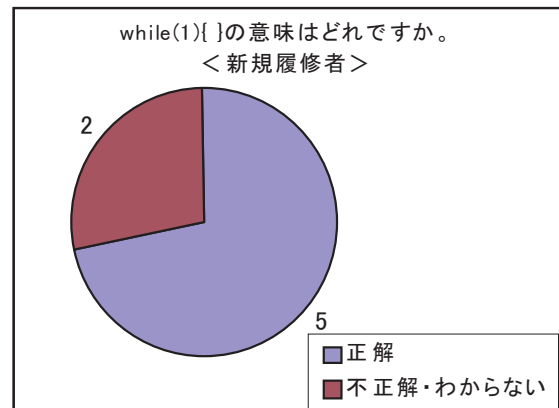


図21. while(1)の意味理解(新規履修者)

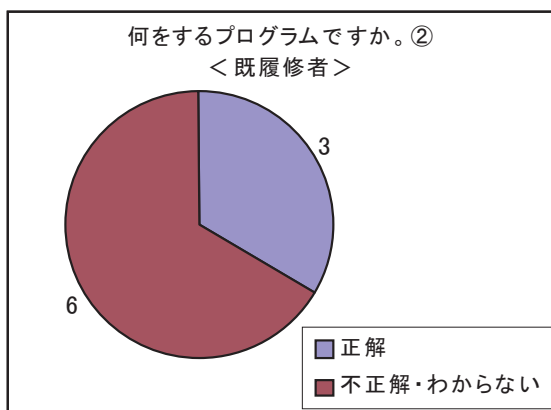


図18. プログラム理解②(既履修者)

最後の時間の始めにこれまで勉強してきたことの総合テストを行った。確認テストによる生徒の理解度を示す。このテストでは、whileの使い方の確認とleftとrightの命令の確認のために行った。テスト終了後に答えを確認し、間違っていたところを生徒に確認させた。

whileの2つの使い方について3、4時間目と勉強してきたので、そのwhile使い方の理解度を確認した。その結果が図22、23に示したも

のである。while (adin(0)>1600) の使い方については、同じものを3,4時間目に使っていたため理解が深かった。図23より、while(1){ }の永久ループについての理解が浅かった。

また、時間についての理解は、小括弧の中の100や500であると1秒、5秒とわかるのだが、1や5だと計算できずに0.1秒や1秒などと間違えてしまう生徒が多いことがわかった。

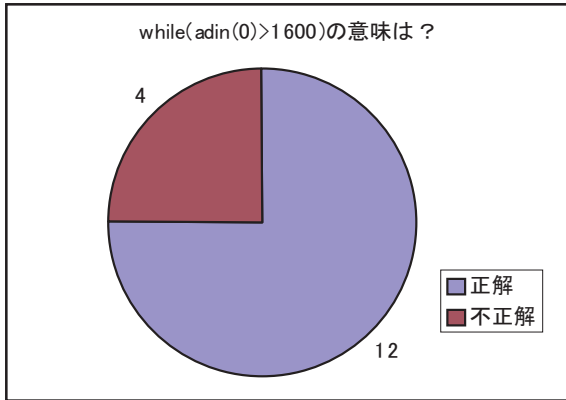


図22. while(adin(0)>1600)の意味理解

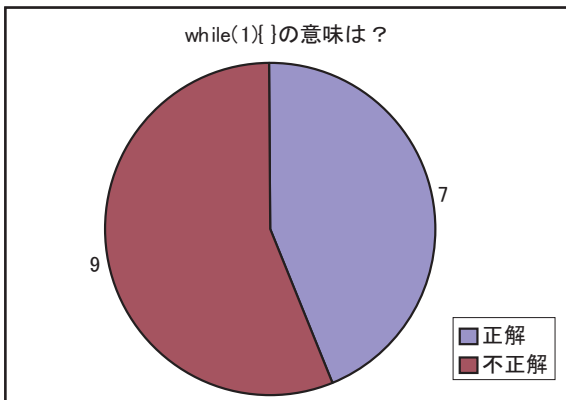


図23. while(1)の意味理解

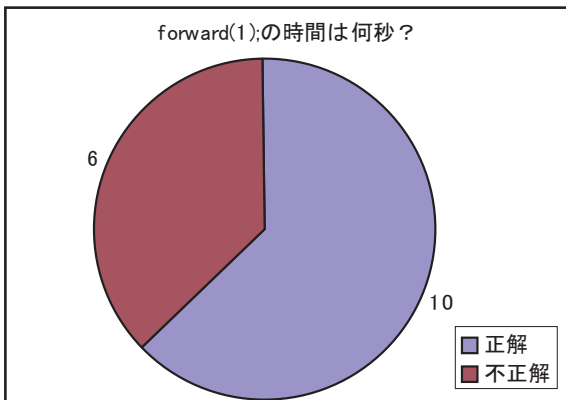


図24. 時間の意味理解①

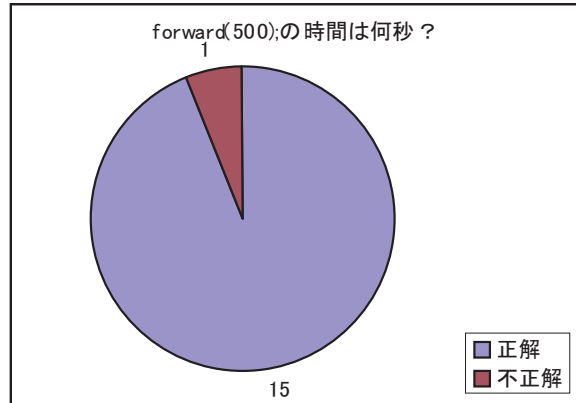


図25. 時間の意味理解②

2-4-1. 生徒が作成したプログラムについて

生徒が作成したプログラムから、どれだけの生徒がプログラムを完成することができていたかの調査を行った。

4時間目：障害物を避けてゴールまで到達するプログラムの作成

図26は、障害物を避けてゴールまで到達するプログラムの正解例を示す。

```
#include <robo.h>
void main(void){
  init();
  while(adin(0)<1600){forward(1);}
  right(80);
  forward(150);
  left(80);
  while(adin(0)<1600){forward(1);}
  stop(1);
  while(1){ }
}
```

図26. 障害物を避けてゴールまで到達するプログラムの正解例

- 8人の生徒はプログラム構造がしっかりとできていた。
- 日本語からプログラムへの理解はできているが余計なものを入れてしまっていた生徒が1人いた。
- 障害物を避けた後の、「白だったら前進しなさい」という命令がforwardになってしまっていた生徒が2人いた。whileの使い方と日本語の理解が合っていなかったためだと思われる。
- センサコネクタが外れて修理していた生徒が1人とデータを集めることができなかった生徒

が1人いた。

- やる気がなくプログラムを考えていなかったと思われる生徒が2人いた。

5時間目：ラインに沿って走るプログラムの作成

図27はラインにそって走るプログラムの正解例を示す。

```
#include <robo.h>
void main(void){
    init();
    while(1){
        while(adin(0)<1600){right_forward(1);}
        while(adin(0)>1600){left_forward(1);}
    }
}
```

図27. ラインに沿って走るプログラムの正解例

- 8人の生徒はプログラム構造がしっかりとできていた。
- 途中までのプログラムはよかったが未完成に終わっていた生徒が3人いた。
- 時間設定が間違っていた生徒が2人いた。
- やる気がなくプログラムを考えていなかったと思われる生徒が2人いた。
- 何をするプログラムであるかわからない生徒が1人いた。

以上のことから、4、5時間目のプログラムからwhileの使い方について理解した生徒が半数いたことがわかる。

このことより、半数の生徒は理解していないと思われる。全員が理解するには、生徒が教材に対して興味をもち、じっくりと考えることができるように授業を組み立てることが良いのではないかと思う。授業の展開としては、50分授業を2時間続きで行い、1つのプログラムをじっくり考え、キー入力の打ち込みに十分時間がとれるように授業計画した方が良いと思われる。

3. まとめ

授業展開を行い、一番の成果としてwhileのプログラム構文を理解させるには一つに絞って教えていくと理解しやすいことがわかった。また、while一つでもプログラムの流れ・構造について学ばせることができた。

授業展開を行ってそこから得た課題は、時間配分である。プログラムを考えて、キー入力する時間が多くかかることがわかった。だからどこに重点をおくかが問題になる。プログラムを考えさせることにこの研究では重点をおいていた。そのため、生徒にプログラムをキー入力させ、動かすことにあまり多くの時間を与えることができなかったと思う。授業の展開としては、50分授業を2時間続きで行い、1つのプログラムをじっくり考え、キー入力の時間を十分とることができるように授業計画した方が良いと思われる。

自律型ロボットの入門教材として車型ロボットは馴染みやすいと思われる。生徒の興味・関心をひく他、簡単に自分で制御することができるからである。

また、Cロボ・ADuC814の制御ボードを用いることでよりわかりやすくまた、遊び感覚だけでなく学習することも大切にすることができると思われる。さらに、ADuC814の制御ボードを使うと、センサ情報をAD変換で得られることやいくつかのプログラムができるようになる。PICの場合とは違い、ゼロフラグやポート設定等を意識する必要がないので、プログラムが簡単に作ることができる。

これらのことからADuC814の制御ボードを使い自律型ロボットの入門プログラムを行うことはよいと思われる。

4. 参考文献

- 1) 田中 繭, 高性能ライントレーサの検討, 岐阜大学教育学部技術教育講座, 平成16年度卒業研究
- 2) 吉田 幸作, C言語で制御するロボット制御モジュールの製作, トランジスタ技術スペシャル, No84, pp.123-159