

# 初年次教育における Scratch とロボットを活用した プログラミング学習の取り組み

矢野 弘<sup>1)</sup>, 崔 龍雲<sup>1)</sup>, 篠宮 紀彦<sup>1)</sup>

1) 創価大学大学院工学研究科

choi@soka.ac.jp

## A Programming Class for Beginners using Scratch and Robots

Hiroshi Yano<sup>1)</sup>, Yongwoon Choi<sup>1)</sup> and Norihiko Shinomiya<sup>1)</sup>

1) The Graduate School of Engineering, Soka Univ.

### 概要

社会システム全般にわたって共通の関心事である初年次教育において、大学新入生のプログラミング学習に焦点を当て、Scratch とロボットを活用した効率的な授業に挑戦している。授業では、初心者を取り組みやすいプログラミング言語と言われている Scratch によるロボット操作システムを導入し、決まったコース走破を目標とし、少人数のチームワークで、それに競技方式の得点を競い合うことにより、自主的で協調性に富んだ解決策を生み出させることが目標である。本稿では、初年次教育のプログラミング授業に実機として使用しているロボット操作システムの構成と、そのシステムを取入れた授業の取り組み方、及び授業実施前後のアンケート調査からプログラミング学習に対する意識や能力向上の変化について述べる。また、ロボット操作システム導入による授業の実施・運用の結果から得られたその考察についても詳述する。

### 1 はじめに

社会システムを構築する組織を維持する上で、その大小に関わらず、必要不可欠な要素の1つがその構成員の初年次教育である。近年、特に叫ばれている初等教育での情報教育のみならず、家族の小単位での親子の情報教育[1]から新入社員の教育まで初年次教育への関心が高まっている。これらの対応には、かつて広く紹介されてきた PBL (Project Based Learning)[2]を始めとする、Active Learning というキャッチフレーズのもとで、各種組織にあったプログラミング学習のモデルが改良・開発され、試行錯誤が繰り返されている。特に、情報教育の学習内容の中で欠かせない項目が効率的なプログラミング教育である。初年次構成員の個々は、プログラミングに対する基礎能力や経験度合が異なるはずなので、その教育手法を一括にまとめるのは容易ではない。

そこで本研究では、本学（創価大学）情報システム工学科の新入生を対象に実施している初

年次教育の実例をあげ、その目的や取り組み方、及びその効果についてまとめている。特に、プログラミング教育においては、ロボットなどの実機道具を取り入れることにより、座学とは違った学習効果が期待できる[3-5]。本学では、プログラミング経験度合の異なる初心者を取り組みやすくするため、Scratch[6]を用いてロボット操作を行い、決まったコースでの競技方式を導

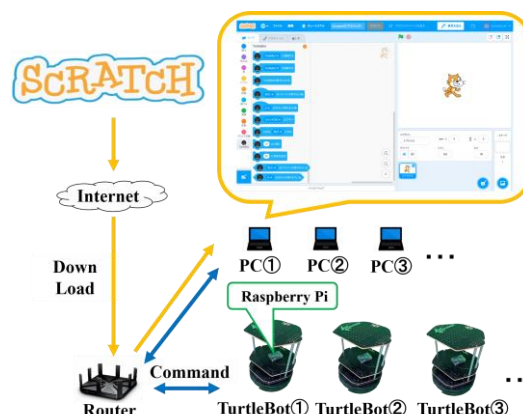


図1 Scratch による  
ロボット操作システムの構成図

表 1 TurtleBot 操作用の Scratch ブロック群

| ブロック図 | 処理内容       |
|-------|------------|
|       | ロボットに接続    |
|       | ボタン押し判定    |
|       | ブザー音を鳴らす   |
|       | LED の点灯・消灯 |
|       | 前後進        |
|       | 回転         |
|       | バンパーの状態判別  |

入することにより、少人数チーム(3, 4 人)毎の学習効果の極大化を図った。

本稿では、2019 年の前期に行われた授業内容とロボット操作システムの構成について述べた後、授業実施結果と考察、ロボット操作システムの改善点について述べる。

## 2 Scratch によるロボット操作

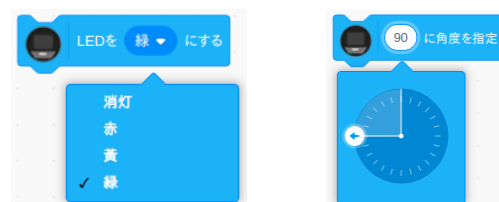
### 2.1 ロボット操作のためのシステム構成

Scratch はブロックを組み合わせてプログラムを作成することができるため、ロボット操作のプログラムを作成したことのない未経験者や初心者の新入生でもなじみやすい特長がある。これを活かしつつロボットの制御プログラムが作成できるようにするためには、図 1 に示すようにロボット操作システムを工夫して構築する必要がある。このシステム構築[7]については、すでに実機による動作確認や 2019 年度授業現場での運営チェックが実施されている。

ロボットは、自律ロボットの研究・開発のための platform である TurtleBot (Yujin Robot 社製)を使用する。このロボットは、前後進・回転や LED の点灯、ブザーを鳴らすことや接触センサーによる物体との接触検知をすることができる。また、カメラやマイクなどのセンサーが取り付



図 2 プログラミング例



(a) Pull Down から選択 (b) GUI 操作

図 3 マウス操作による値の指定方法

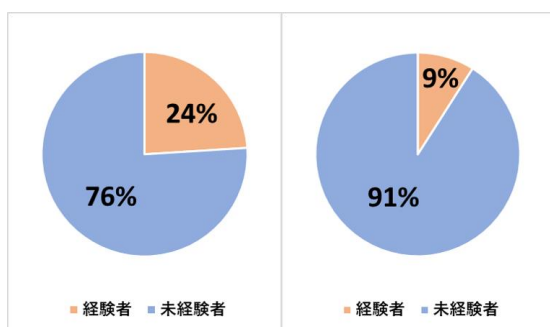
表 2 学習内容

|                    |
|--------------------|
| グループディスカッション       |
| Word でレポート作成       |
| PowerPoint でスライド作成 |
| プレゼンテーション          |
| プログラミング            |
| ロボット操作             |

けられるように設計されているため、ロボットの拡張性に富んでいる。そのため、学生たちのプログラム能力の成長に応じて拡張できる特長が活かせる。さらに、TurtleBot には、Raspberry Pi (Single board computer)を搭載させることにより、Router を介して PC から IP アドレスが割り振られた Raspberry Pi にアクセスすることができる。そのため、PC から操作対象の TurtleBot を指定することができる。これにより、Scratch のブロックの命令と TurtleBot のセンサー情報は、Command 形式での送受信が可能となる。

### 2.2 TurtleBot 操作のための Scratch ブロック群

表 1 は、図 1 に示す PC の画面上で使用する TurtleBot 操作用の Scratch ブロック群を示す。例えば、PC から操作する TurtleBot の Raspberry Pi に接続するブロックや指定した番号のボタンが



(a) Visual Based Language (b) Text-Based Programming

図 4 入学前のプログラミング経験者の割合

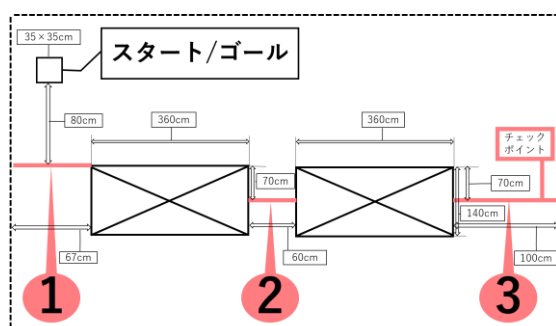


図 5 タイムアタックコース俯瞰図

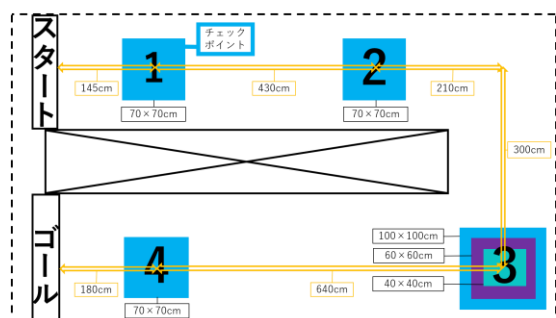


図 6 テクニカルコース俯瞰図

押された際に実行されるブロックなどがある。例えば、これらのブロックを用いて TurtleBot のボタンを押したとき、LED を緑に点灯させ、ロボットを 90 度回転させたい場合は図 2 に示すようなプログラムになる。このプログラムを PC 上で実行することにより、TurtleBot の操作が可能となる。

また、操作対象者は、キーボードで値を入力することに慣れていないことが考えられるため、直感的な値の指定方法が必要である。例えば、TurtleBot を操作する際、LED の色や回転角度などの値を指定する場合がある。これを実現するために図 3 に示すブロックを作り、マウス操作による値指定ができるようにした。図 3(a)は、

表 3 授業計画

| 回数 | 内容                   |
|----|----------------------|
| 1  | Scratch の説明と動作確認     |
| 2  | Scratch によるロボットの動作確認 |
| 3  | チームの方針と個人の役割決め       |
| 4  | ロボットの開発とテスト          |
| 5  | ロボットの開発とテスト          |
| 6  | ロボットの開発とテスト          |
| 7  | 計画と戦略の修正検討           |
| 8  | ロボットの開発とテスト          |
| 9  | ロボットの開発とテスト          |
| 10 | ロボットの開発とテスト          |
| 11 | 競技大会実施               |

候補の中から値を指定するための Pull Down である。これにより、選択肢から値を指定することができる。図 3(b)は、GUI (Graphical User Interface)操作で値を指定する方法である。これにより、矢印を目盛りに合わせることで値を指定することができる。ここでの指定範囲は、-165~180 に設定できるようにした。これらの値指定ブロックにより、半角全角入力の誤りや例えば、10000 のような単純誤入力によるエラーが避けられる。

### 3 初年次教育に本システムの導入

本学の初年次教育では、グループで 1 つのテーマに取り組むことで、学生 1 人ひとりがチームの中で何をすべきかを明確にし、協働性や企画性、及び礼儀やポジティブな思考を身に付け、新生活に慣れると共にスムーズな人間関係を築かせることなどが達成目標であり、表 2 に示すような学習内容が含まれている。これらのことを踏まえて、本ロボット操作システムを初年次教育のプログラミング学習への導入に焦点を当て、その応用手法と内容、及びその授業運用の効果について調べた。

#### 3.1 プログラミング学習の意識とその経験

本学情報系の新生 70 名を対象にプログラミングに対する意識調査を行った。これは、情報系の学生と言ってもプログラミングの初心者

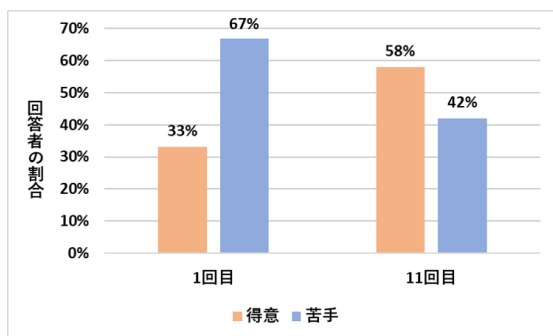


図7 プログラミングに対する意識調査

であるとの前提で、苦手意識やプログラミング経験の度合を調べた。図4には、大学入学前にプログラミング経験した割合を示す。図4(a)より、Scratchなどのビジュアルプログラミングを経験したことのある学生は、全体の24%である。そして、図4(b)のテキストプログラミングを学んだことがある学生は、全体の9%である。このことから、多くの学生がプログラミング未経験者であることが分かった。これは、授業実施後の意識変化の判断基準になり得る。

### 3.2 プログラミング学習への取り組み

初年次学生の授業に工夫・開発された本ロボット操作システムをどのように適用し、特にプログラミング学習にどう活かせるのかについて述べる。今回の取り組みでは、2種類のコースによるロボットの競技方式を採択し、ロボットのもつ機能の範囲内で達成できる内容にした。

図5にはその1種目のタイムアタックコースの俯瞰図を示す。これは、スタートからゴールまでの時間を競う競技である。ロボットは、ゴールするまでにチェックポイント①②③を全て通過する必要がある、各ポイントの通過ごとに得点が与えられる仕組みとなっている。学生達は、チェックポイントをどのような経路で通過するかを考え、プログラムを作成する必要がある。

図6にはもう1つのテクニカルコースの俯瞰図を示す。これは、スタートからゴールまでに設定されている地点1~4の各課題の達成度を競う競技で、課題ごとに得点が設けられている。そのため、ロボットは課題を達成する時間よりも精度が大事であり、それらを考慮したプログ

ラムの作成が必要となる。

### 3.3 授業の実施方法

学期中の授業では、全11回(各90分)がプログラミング学習と各コースでの実機テストであり、残りの4回はガイダンスやレポート作成、プレゼンテーションである。火曜日と木曜日に約12名のグループに分けられ、3~4名が1チームになっている。各グループにはTA(Teaching Assistant)が配置され、授業中の学生の質問に対応できる体制となっている。

表3には授業ごとに行った内容を示す。1回目と2回目は、学生にScratchの扱い方を説明し、学生はScratchによるロボット操作に慣れてもらう。3回目は、学生に図5、6に示すコースを使用した2種類の競技について説明した後、学生自らが各競技に取り組む3、4人のチームを決める。そして、編成したチームのメンバー同士で、競技を達成するための戦略や計画、個人の担当を話し合い、その内容をレポートにまとめる。4~6回目は、チームで3回目に立てた計画を元に、プログラムの作成や競技コースを使用した走行実験を行う。7回目は、チームで計画の進捗や走行実験の結果から計画や戦略の修正を行い、レポートにまとめる。8~10回目は、チームで見直した計画を元に、競技大会までプログラムの作成や走行実験を繰り返す。11回目は、各チームの成果を披露してもらうために、実機による競技大会を実施する。

## 4 授業の実施結果とその考察

### 4.1 プログラミングに対する苦手意識の変化

図7は、授業の1回目と11回目にプログラミングに対する苦手意識についてアンケート調査を実施した結果である。プログラミングが得意だという人の割合が25%増加していることが分かる。その具体的な理由には、プログラミングが楽しい、または、好きだからという回答が多かった。一方、プログラミングが苦手だという人も存在しており、その理由には、C言語のようなテキストプログラミングが苦手だからとい



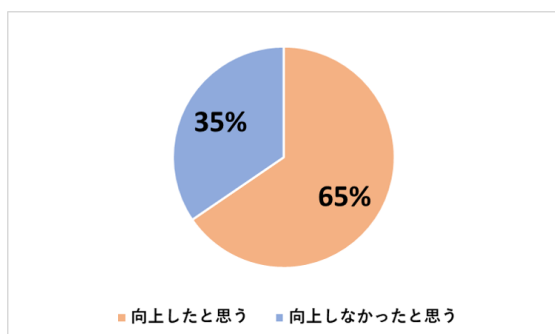


図 8 プログラミング能力に対する意識調査

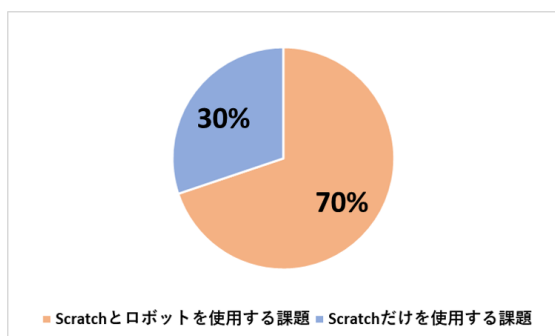


図 9 取り組みたい課題内容調査

う回答が多かった。

#### 4.2 プログラミング能力向上の意識調査

図 8 は、授業の 11 回目以降に学生自身が競技課題を通してプログラミング能力が向上したと思うかについてアンケート調査を実施した結果である。競技課題に取り組んだ結果、プログラミング能力が向上したと思う割合は 65% である。その理由には、チーム内のメンバーと話し合いながら取り組めたからという回答が多かった。一方、プログラミングの能力が向上しなかったという理由には、役割分担により、Scratch にあまり触れなかったからという回答が多かった。このことから、学生全員が Scratch を扱えるように、チーム全員でプログラムの作成を分担するよう促すことや Scratch 上にロボットの動作確認用のシミュレーションを作成するなどの授業環境を拡張させる必要があると考える。

#### 4.3 ロボットを使用した課題に関する意見

図 9 は、授業の 11 回目以降に学生が Scratch だけ、または Scratch とロボットを使用する課題のどちらを取り組みたいかについてアンケート調査を実施した結果である。Scratch とロボット



(a) タイムアタックコース



(b) テクニカルコース

図 10 プログラムの構成要素数

を使用する課題に取り組みたい人の割合は 70% である。その理由には、(1)プログラミングの結果をロボットの動きで確認できるから、(2)ロボットが動く楽しいからという回答が多々あった。このことから、学生達はロボットを動かすことやプログラムの挙動を確認することでその楽しさや達成感などのポジティブな感情を抱くのではないかと考える。一方、Scratch だけを使用する課題に取り組みたい理由には、競技コース使用時間が限られるといった空間的な制約があることやチーム内の役割担当上、Scratch にあまり触れなかったからという回答が多かった。このことから、競技コース使用時間は、競技コースの数を増やすなどの空間を確保することで解決可能である。

#### 4.4 コースにおけるプログラム分析

競技大会実施後、各チームが作成したプログラムを収集し、プログラムの構成要素を分析した。図 10(a)にはタイムアタックコースの各チームで使用されたプログラムの構成要素数を示す。9 チーム中の 6 チームがブロックを組み合わせただけの順次処理のみでプログラムを作成していた。しかし、A\_10、B\_6 はロボットを前後進させる際に、反復処理を使用していた。A\_6 はバンパーを使用した動作を実現するために分岐処理を使用していた。このことから、反復処理



図 11 接触センサーを使用した処理例



図 12 速度ブロックを使用した処理例

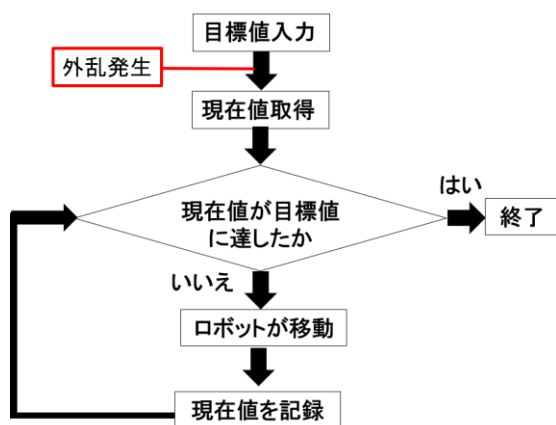


図 13 前後進・回転処理のフローチャート

や分岐処理を使用したチームが存在した理由には、競技課題を達成する戦略のため、もしくは、プログラミング能力が向上したからなどの要因が考えられる。

図 10(b)にはテクニカルコースの各チームで使用されたプログラムの構成要素数を示す。11チーム中7チームがブロックを組み合わせただけの順次処理のみでプログラムを作成していた。A\_7とB\_7は、プログラムを見やすくすることや走行実験後のプログラムの修正の効率化する

ことのために、チェックポイントごとのプログラムを関数化していた。また、B\_9は変数を使用することでプログラムの処理がどこまで実行されたか判定していた。その判定結果からバンパーを使用した割込み処理の内容を変更できるようにしていた。このことから、関数や変数を使用したチームが存在した理由には、競技課題を達成する戦略や作業の効率化のためや、プログラミング能力が向上したからなどの要因が考えられる。

以上の各チームが作成したプログラムの分析結果から、競技課題を達成する上で、分岐処理や変数などのプログラム要素を使用したチームが少ないことが分かった。これにより、次の授業では、例えば、分岐処理を使用しなければ達成できない課題を設定することにより、学生が競技課題を通してプログラムの要素を理解できるのではないかと考える。

以上の各チームが作成したプログラムの分析結果から、競技課題を達成する上で、分岐処理や変数などのプログラム要素を使用したチームが少ないことが分かった。このことから、次の授業では、例えば、分岐処理を使用しなければ達成できない課題を設定することにより、学生が競技課題を通してプログラムの要素を理解できるのではないかと考える。

## 5 システム運用上の課題とその対策

本システムを本学初年次教育に導入した結果、予期せぬシステムの課題が3点見つかった。以下にその課題内容とその対策について述べる。

### 5.1 送受信情報の遅延

授業中にPCとRaspberry Pi間の情報送受信の遅延が発生する場合があった。この原因は、複数台のPCがRouterへ一斉に接続し、同時に9~11台のロボットを操作したからだと考える。そのため、Scratchによるロボット操作システムを使用する際は、Routerへ同時に接続するPCの台数や操作対象であるロボットの台数を制限する必要がある。今回は、チームごとに使用時間をずらすことで対応した。

## 5.2 接触センサーによる方向転換処理

競技大会を実施したとき、ゴールまで到達できないチームが存在した。その理由は、ロボットが競技コース上の机や壁に衝突していたからである。この原因として、ロボットの前進中に接触センサーを使用した分岐処理を実行できなかったためだと考える。例えば、ロボットの前進処理と接触センサーが机や壁に衝突した場合、後進・回転する処理を組み合わせたプログラムは、図 11 のようになると想定された。しかし、分岐処理は、直前のブロックの処理が完了するまで実行されない。そのため、図 11 のプログラムでは、ロボットが移動中に物体との接触を検知したにもかかわらず、ロボットは机や壁に衝突してしまう。この課題を解決するためには、速度を指定するブロックを新たに作成する必要がある。このブロックを用い、図 12 のようなプログラムを作成することで、障害物に衝突したときの対処が可能になる。

## 5.3 ロボットの移動処理の誤動作

図 13 に授業中にロボットが指定する値で正しく前後進や回転をしない状況を示す。その理由は、学生がロボットを手動で移動することにより、外乱が発生し、現在値の取得で誤差が生じていたからである。この課題を解決するためには、目標値が入力されたときに現在値を初期化する必要がある。

# 6 おわりに

本稿では、初年次教育における効率的な授業実現のために、Scratch によるロボット操作システムの構成と本システムの本学初年次教育への導入方法、および授業実施結果について述べた。授業実施結果より、本システムを使用した本学情報系の新生のプログラミングに対する苦手意識やプログラミング能力が改善されたことが分かった。その一方で、競技課題のプログラムを作成する際、分岐処理や変数などのプログラム要素を使用しないチームが多かったことからプログラム要素の技能を深める競技コースの発

掘や課題設定には改善の余地がある。

今後は、分岐処理や変数を使用する競技課題の考案やシステム課題の改善、および発展的なプログラミング能力向上のためにアップグレードされた内容で後期授業を実施し、その結果を精査してまとめたい。

## 参考文献

- [1] 久郷 紀之，崔 龍雲，親子プログラミング学習における発話インタラクションの分析，情報処理学会第 81 回全国大会，pp.519-520，2019.
- [2] PBL 教材 洗礼 WG，PBL(Project Based Learning)型授業実施におけるノウハウ集(2017 年 7 月改訂案)，2017.
- [3] 大熊 一正，恐神 正博，杉原 一臣，高校生を対象としたプログラミング体験授業において実機ロボット制御を取り入れる効果，情報処理学会第 80 回全国大会，pp.501-502，2018.
- [4] 加藤 聡，小谷 猛房，富永 浩之，LEGO ロボットとゲーム課題を題材とする問題解決型のプログラミング演習ー事前学習から事後発表までを含む理系高校生への教育実践ー，情報処理学会研究報告，Vol.2010-CE-105，pp.1-10，2010.
- [5] 高岡 詠子，伊川 英男，ロボット制御による Java 言語プログラミング教育支援システム Jesle，情報処理学会研究報告，Vol.2003-CE-70，pp.9-16，2003.
- [6] Resnick, Mitchel et al., Scratch: programming for all. Commun, ACM, vol.52, pp.60-67, 2009.
- [7] 久郷 紀之，Scratch で@Home プログラムを作ろう，RoboCup@Home Education Challenge @ RoboCup JAPAN OPEN 2017 Final，2017.