

レゴ学習システムを使用したプログラミング実習の導入研究

名古屋市立工業高等学校 情報科

加藤ひとみ

1. はじめに

近年、小学校での必修化が話題となっているプログラミング教育だが、工業高校としての目的は、プログラミングに関する職に就くために必要であるのはもとより、プログラミング的思考を育むことや、コンピュータを活用した問題解決能力の身に付けることなどが挙げられる。本校では言語によるプログラミングを1年生から3年間実施しているが、中には言語プログラミングに苦手意識を持つ生徒も少なくない。そこで言語プログラミングが苦手な生徒にも取り組みやすいよう、レゴロボットを使用したプログラミング学習システムを2018年より実施している。導入・実施から5年が経過し、ロボットの故障が相次ぎ、部品を含めた本体の製造が中止されたため、新しいシステムである「SPIKE」を導入し、より発展したプログラミング実習を行うことを計画した。

2. レゴ®エデュケーション「SPIKE」について

今回導入したレゴ®エデュケーション SPIKE™プライムは、STEAM（科学・技術・工学・芸術・数学）教育用のプログラミング教材である。レゴのブロックとセンサ、モータ、プログラミングツールを組み合わせることで論理的思考や問題解決能力を育むことを目的としているシステムである。本校でこれまで使用していた教育版レゴ®マインドストーム®EV3との違いは以下の点が挙げられる。

- ・Scratch ベースのビジュアルプログラミングが中心であるため、直感的で学びやすい
- ・AI分野で使われる Python 言語にも対応しており、より高度なプログラムも可能
- ・入出力ポートが6つあり、複数のセンサを組み合わせることが可能

また、教育版レゴ®マインドストーム®EV3は販売が中止されており、今後はこれまでEV3で行ってきた実習を順次SPIKEを使用したものに変更する予定である。



教育版レゴ®マインドストーム®EV3



レゴ®エデュケーション SPIKE™プライム

3. SPIKE を使用したプログラミング実習

SPIKE の学習では、専用のアプリを使用してプログラミングを行う。

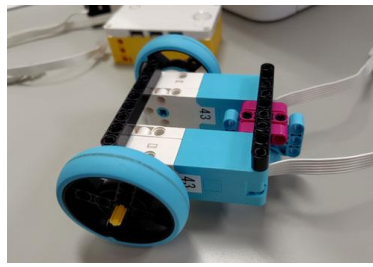
今回は2年生の授業内容である迷路を脱出するためのプログラミングを実装してみる。

① SPIKE の組み立て

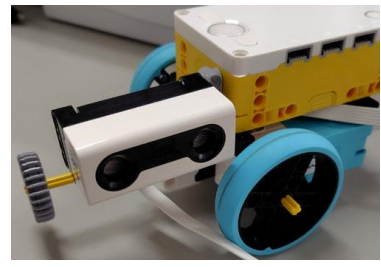
迷路を脱出するためには、タイヤを取り付け、前進するロボットを多数のパーツから選んで最適なロボットを組み立てる。また、壁を認識する「超音波センサ」と、壁に当たったことを認識する「タッチセンサ」の2つを使用するため、それらを取り付ける。



モータ部分



タイヤ周り



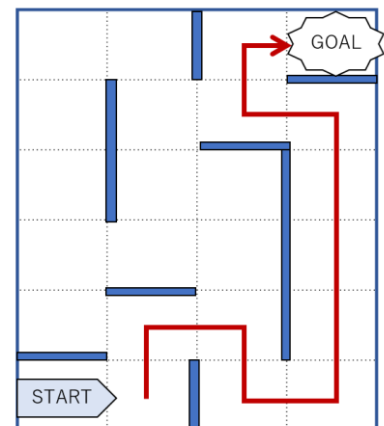
超音波センサとタッチセンサ

② アルゴリズムを考える

今回は右図のような迷路を使用する。

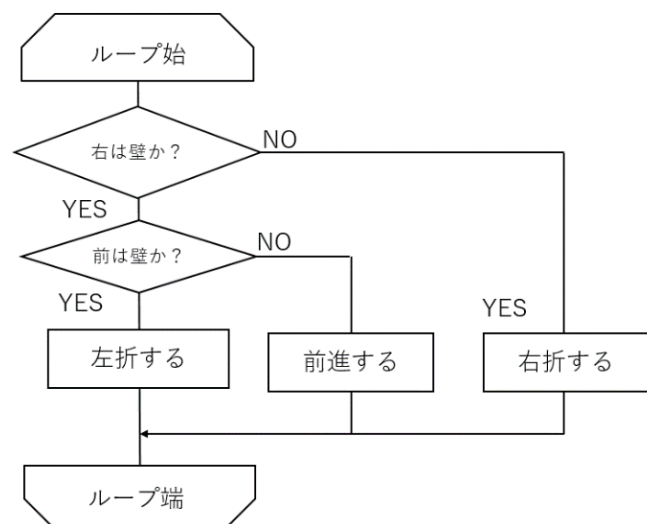
この迷路では、以下の前提条件のもと制作する。

- ・ 進行方向は、前後左右の4方向のみ
- ・ スタートとゴールの位置は固定
- ・ 3方向を壁に囲まれたエリアは、スタートとゴールのみ



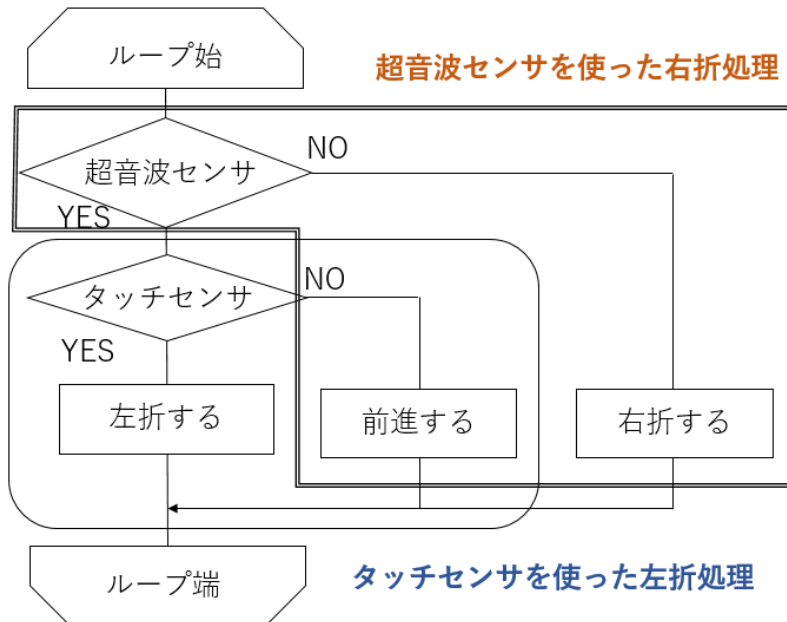
まず、迷路を脱出するためのアルゴリズムを検討する。

ここでは、ロボットが前進したときの判断条件をフローチャートに起こしてみる。



今回は、迷路を脱出するアルゴリズムに「右手法」を採用する。
進み始めた際、右側に壁があるかを判断する。もし右に壁があり、前に壁が無ければ「前進」し、前に壁があれば「左折」する。また、右側に壁が無い場合には、「右折」をする。

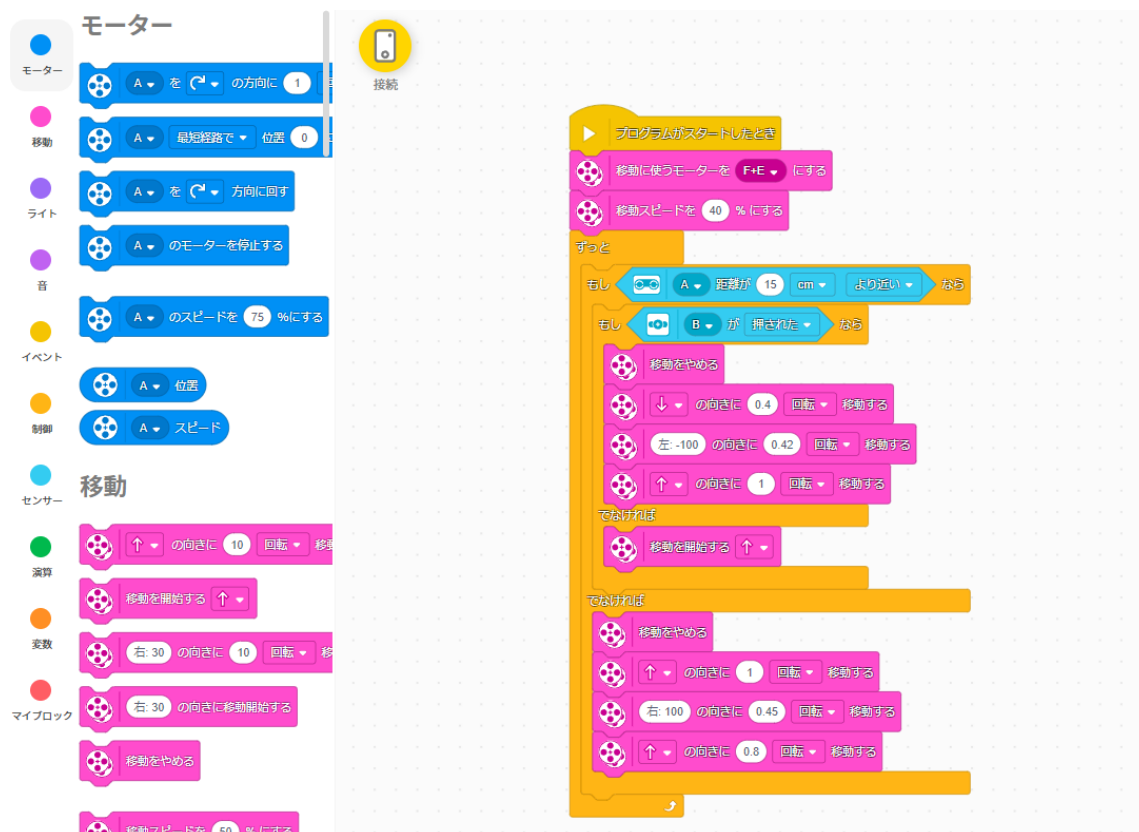
この時、超音波センサを使うことで右に壁があるかを判断し、タッチセンサを使うことで前に壁があるかを判断させる。



③ ビジュアルプログラミング

アルゴリズムを元に、プログラミングを行う。

まずは SPIKE の特徴であるビジュアルによるプログラミングを行う。



ビジュアルプログラミングでは、画面の左側に処理内容の一覧が用意されており、それを繋ぎ合わせて具体的な数値を入力することで完成する。言語プログラミングが苦手な生徒であっても、アルゴリズムさえしっかり考えておけば、それに合う処理パーツを探して直感的に配置するだけでもプログラムを完成させることができる。

④ 言語プログラミング

ビジュアルプログラミングでもプログラムは作成できるが、情報科では言語プログラミングを行っているため、Pythonを使用したプログラミングに書き換えてみる。

● プログラム全体

```
import runloop,motor_pair,force_sensor,distance_sensor
from hub import port

async def main():
    motor_pair.pair(motor_pair.PAIR_1,port.F,port.E)
    while True:
        if distance_sensor.distance(port.A) < 180:
            if force_sensor.force(port.B) > 0:
                await
motor_pair.move_for_degrees(motor_pair.PAIR_1,144,0,velocity=-280)
                await motor_pair.move_for_degrees(motor_pair.PAIR_1,175,-
100,velocity=280)
                await
motor_pair.move_for_degrees(motor_pair.PAIR_1,360,0,velocity=280)
            else:
                motor_pair.move(motor_pair.PAIR_1,0,velocity=400)

        else:
            await motor_pair.move_for_degrees(motor_pair.PAIR_1,250,0)
            await motor_pair.move_for_degrees(motor_pair.PAIR_1,175,100)
            await motor_pair.move_for_degrees(motor_pair.PAIR_1,330,0)

runloop.run(main())
```

⑤ テスト

プログラミングが出来たら、組み立てたレゴロボットにプログラムを転送し、実際に走行させてみる。上手く動作しない場合、アルゴリズム及びプログラミングを見直し、何度もデバッグして完成させていく。



⑥ 生徒による検証

これまで本校で使用していた教育版レゴ®マインドストーム® EV3 を使用して迷路のプログラミングを作成してきた生徒に、今回導入した SPIKE を使って同じようにプログラミングを行ってもらった。



始めて触れるため、さまざまな点で苦労していたが、最終的には正しく動作するプログラムを作成することができた。以下に生徒の感想を記載する。

- ・ Scratch ベースのビジュアルプログラミングでプログラムを作ることができたので、直感的に配置することができ、作成しやすかった。
- ・ SPIKE のパーツが分かりやすいため、簡単にロボットを組み立てることができた。
- ・ SPIKE で使用するための Python のライブラリが分からず、言語プログラミングに苦労した。

実際の生徒の声を聞き、今後授業で導入する際には、言語プログラミングの理解度を深めるための内容が必要だと感じた。

4. まとめ

本研究では、これまで実習で使用していた機器の数台を新しい機器へ更新することができた。解決したい行動に対して、ブロックを組み立てる（機構を考える）ことから実際に動作させ、デバッグやアルゴリズムの再検討など一連の流れを学習できるロボットプログラミングには、より探求的な学習に繋がることを改めて実感した。また、言語プログラミングが苦手な生徒がビジュアルプログラミングから行うことによりプログラミングに対する苦手意識を無くし、徐々に言語プログラミングへ移行していくことで、自主的・自発的に学習に取り組めるのではないかと考えた。情報教育において新しい実習機器を使って先端の学習することは非常に有意義ではあるが、予算の関係で定期的に更新をしていくことが難しいのが現状である。今回の助成金により機器の更新ができ、より充実した授業内容を考案することができた。

謝辞

本研究を遂行するにあたり、公益財団法人天野工業技術研究所理事長川幡長勝様から多大なご支援を頂きましたことをここに記して謝意を示します。ありがとうございました。