

信号機を題材とした小学生向けプログラミング教材の開発

大賀 瑞穂^{1,a)} 鈴木 優^{1,b)}

概要: 2020年度より、新たに全国の小学校でプログラミング教育が必修化された。必修化の目的は、子どもに「プログラミング的思考」を身につけさせることと、コンピュータに意図した処理を行うよう指示することができるということを体験させながら、身近な生活でコンピュータが活用されていることに気づかせることである。しかしながら、このねらいを全て満たす既存のプログラミング教材は存在しない。そこで、本研究では、プログラミング教育必修化のねらいを達成できるプログラミング教材として、タブレット端末とタンジブルなユーザインタフェースである信号ブロックを用いた信号プログラミング教材を開発した。この教材では身近な生活に使われているコンピュータである信号機を題材とし、新たなビジュアルプログラミング言語として数直線タイプを実装した。

1. はじめに

2020年度から小学校でプログラミング教育が必修化された。必修化の目的は、子どもにプログラミング的思考を身につけさせることであり、コードや言語などの技術を習得することではなく、思考法を育てることである。プログラミング的思考とは、自分が意図する一連の活動を実現するために、どのような動きの組合せが必要であり、1つ1つの動きに対応した記号を、どのように組み合わせればいいのか、記号の組合せをどのように改善していけば、より意図した活動に近づくのか、といったことを論理的に考えていく力である。また、小学生において、コンピュータに意図した処理を行うよう指示することができるということを体験させながら、身近な生活でコンピュータが活用されていることに気づかせることも必修化のねらいである [1]。

これまでに多くの教材が開発されている一方で、既存教材には後者のねらいを達成できる教材が存在しない。そこで、本研究では、身近な生活にあるコンピュータの存在に気づき、コンピュータに意図した処理を行うよう指示することが体験できる教材の開発を目指す。

2. 既存教材

プログラミング的思考が学べる様々な教材がある。小学生向けに開発されてきたプログラミング教材の種類を、次の5つに分類した。

アプリ系

パソコンやタブレット端末でプログラミング学習用のゲームやクイズに取り組んだり、自由にプログラミングしてアニメーションやゲームを作成したりする教材。

センサ系

通信機能や各種センサなどが仕込んである小さな高性能パーツを、日用品や工作と組み合わせて、いろいろな仕組みを作る教材。

ロボット系

形や動く仕組みを自由に考えてロボットを組み立て、機能はパソコンやタブレット端末で機能をプログラミングしてロボットを完成させる教材。

ガジェット系

球体やキューブ型など多彩なロボットやドローンなどを、コントローラの操作やプログラミングで動かす教材。

アンプラグド系

電子機器を使わずに、遊びやゲーム仕立てのアクティビティ、アナログなゲーム、絵本や物語などで、プログラミングの考え方を学ぶ教材。

また、アンプラグド系以外の他4種類にはビジュアルプログラミング言語が用いられていることが多い。ビジュアルプログラミング言語とは、プログラムをテキストで記述するのではなく、視覚的なオブジェクトでプログラミングする言語であり、次の3種類に分類される。

ブロックタイプ

指示の書かれた小さなブロックをつなぎ合わせてプログラミングする言語。

¹ 宮城大学, Miyagi University

^{a)} p1920046@myu.ac.jp

^{b)} suzu@myu.ac.jp

フロータイプ

フローチャートのように、指示や機能、条件のアイコンを配置して線でつないでプログラミングする言語。

独自ルールタイプ

上記のどちらでもなく独自ルールで作られたプログラミング言語

既存のプログラミング教材は数多く存在する。Scratch[2]はキャラクタをプログラミングして動かし、アニメーションやゲームなどを自由に作成できる教材であり、アプリ系のブロックタイプに分類される。MESH[3]はセンサで状態を確認することや自動で動く仕組みをつくる教材であり、日用品や工作に付け足せば、IoTの仕組みを手軽につくることができる。センサ系のフロータイプに分類される。embot[4]は本体をダンボール工作で組み立てるロボットキットである。ロボットのプログラミングをして腕などを動かすことができる。ロボット系の独自ルールタイプに分類される。Sphero[5]はタブレットやスマートフォンから操作、プログラミングして球体のロボットを動かす教材であり、ガジェット系のブロックタイプに分類される。Robot Turtles[6]はコンピュータを使わないカードゲームである。カードを並べてキャラクタの動きをプログラムして宝石を取る内容である。アンプラグド系に分類される。

また、関連研究においても様々な教材が開発された。まなべー [7]は命令カードを並べ、宝箱を取りゴールを目指すプログラミング教育ゲームである。アプリ系のブロックタイプに分類される。操作カードによるプログラミング体験システム [8]は、タンジブルな操作カードを使い、スマートフォンのカメラを通してマップ上の爆弾に当たらないようにコインを集めるプログラミング教育ゲームである。スマートフォンを用いるが、プログラミング自体はアンプラグド系に分類される。

プログラミング教育のねらいは、コンピュータに意図した処理を行うよう指示することができるということを体験させながら、身近な生活でコンピュータが活用されていることに気づかせることであるが、この2点を達成できる教材は存在しない。本研究では、このねらいが達成できる教材の開発を目指す。

3. プログラミング教材のデザイン

3.1 教材の設計指針

本研究の目的を達成できる、かつ既存教材が抱える問題点を解決し、より理解しやすく、より良い学びができるようにするため、次の教材設計の指針を設ける。

- (1) 身近な生活にコンピュータが使われていることに気づくことができる
- (2) コンピュータに意図した処理を行うように指示することを体験できる
- (3) 入出力のインタフェースが理解しやすい

指針1と2はプログラミング教育のねらいを満たすために、指針3は小学生が理解しやすい教材の形態を目指すために設けた。

3.2 教材の形態

3.2.1 教材の題材

指針1を満たす教材の題材として、次の3つの理由から信号機を選択した。1つ目は小学生でもルールを知っている身近でコンピュータが使われているものであること、2つ目はプログラムの内容が容易であること、3つ目はプログラミングの基本である順次、分岐、反復の3つを学べることである。

1つ目について、小学生でも青色は進む、黄色と赤色は止まるとルールを知っている。信号機は外でよく見かける、身近でコンピュータが使われているものと言える。

2つ目について、信号機の基本的なプログラムは、それぞれの秒数を指定しLEDが青・黄・赤の順に光る、比較的容易な内容である。既存教材として挙げたScratchやembotなどは、中学校で学ぶ負の数や座標などの知識が必要であり、プログラミング学習に取り組む前にこれらについて学習する必要がある。信号機ではこのような事前知識を必要としない。

3つ目について、順次、反復、分岐はプログラミングの基本構造の3要素である。順次処理とは、この処理の次はこの処理、というように順々に処理を実行していくことである。反復処理とは、ある処理を繰り返し実行することである。分岐処理とは、ある条件によって実行する内容を変えることである。これらの要素はプログラミングを考える際に重要な考え方で、プログラミング的思考を身に着けるためには欠かせない要素であり、信号機はこれらの要素の学びに最適な題材である。信号機では、青・黄・赤の順に光る部分が順次処理に、青・黄・赤の順に光った後、また始めに戻り青・黄・赤と光る部分が反復処理に、信号機の色が赤になると、交差点の対の信号機が青に変わる部分が分岐処理に当たる。

3.2.2 教材の端末

指針2を満たす教材の形態として、タブレット端末を使用する。文部科学省が推進するGIGAスクール構想によって、タブレット端末は児童全員が個別に使用できる標準的な学習用具となっている。令和3年7月末時点の端末利活用状況等の実態調査によると、全国の公立の小学校などの96.2%が「全学年」または「一部の学年」で端末の利活用を開始している [9]。コンピュータに意図した処理を行うように指示する体験を実現するため、児童が使い慣れたコンピュータであるタブレット端末を教材の端末の一部として使用する。

3.2.3 教材のインタフェース

指針3を満たすために、教材のインタフェースとして

GUIとTUIの両方の考え方を取り入れたものを採用する。インタフェースを考えるにあたり、GUIとTUIについて説明する。GUI(グラフィカルユーザインタフェース)は、コンピュータの画面上に表示されるウィンドウやアイコン、ボタンなどを使い、マウスなどのポインティングデバイスで操作できるインタフェースである。一方、石井が提唱したTUI(タンジブルユーザインタフェース)は、形のない情報を直接触れることができるようにした、実体で表現されたインタフェースである[10]。GUIのメリットは物理表現に制限がないことが挙げられる。例えば、モニタの中のキャラクタが自由に空を飛んだり色を変えたりすることができる。TUIのメリットは物理的表現の直接操作性、直感的な理解容易性が挙げられる。しかしながら、TUIには、物理表現に大きな制限を持つというデメリットがある。例えば、プラスチックで作られたプラモデルが空を飛んだり色を変えることは難しい。このデメリットをカバーするためにGUIとTUIを組み合わせることで、TUIのデメリットをカバーしつつ、両者のメリットを得られることができる。この両者のメリットを得るため、入出力のインタフェースの一部を直接操作可能なタンジブルタイプにし、GUIと融合させた教材を開発する。

4. 信号プログラミング教材の開発

4.1 システム概要

本研究で開発した信号プログラミング教材は、信号機のプログラミングを行い、プログラミングの思考を学ぶことを支援する教材である。プログラミングには後述する数直線タイプのビジュアルプログラミング言語を使用する。車をスムーズに走行させるため、車の渋滞や事故が起こらないように繰り返し信号機をプログラミングする。

4.2 システム構成

本教材のシステム構成を図1に示す。本教材はタブレット端末1台と、タッチパネル付き液晶ディスプレイ1台とそれに接続する映像投影用のパソコン1台、複数の信号ブロックから成る。タブレット端末は信号機のプログラミングを行うタブレット(以下、プログラミングタブレット)、タッチパネル付き液晶ディスプレイは道路と車が表示されるディスプレイ(以下、マップ表示モニタ)である。信号ブロックをマップ表示モニタ上に置くと、信号ブロックの位置と種類が検出される。

4.3 信号ブロック

信号ブロックとは信号機を模したブロック型の入出力インタフェースである(図2)。信号ブロックのシステム構成を図3に示す。信号ブロックは、青・黄・赤のLEDと各電球を制御できるボタン、ブロックの底面には、タブレットで位置検出を行うための静電マーカを備える。内部には

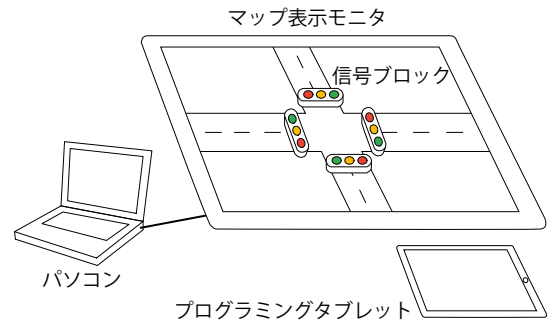


図1 システム構成



図2 作成した信号ブロック

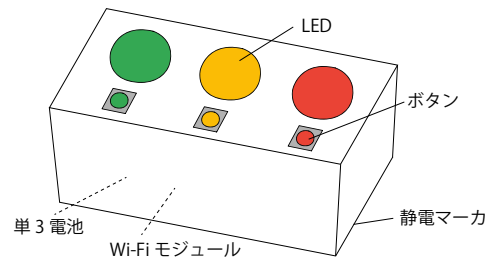


図3 信号ブロックのシステム構成

タブレットと通信するためのWi-Fiモジュール(ESP32)と、単3電池2本を搭載する。

信号ブロックの位置検出には、青木ら[11]がリング型物理オブジェクトに実装した静電式マーカを用いる。静電式マーカとは、物理オブジェクトに導電部を複数配置し、その配置パターンを工夫することで、静電容量方式のマルチタッチスクリーンが取り付けられたタブレット上でのさまざまな操作を可能とする操作方式である。本研究では、3点式のマーカを作成する。認識点同士の距離やその関係性から、それぞれのブロックを認識する。

4.4 通信方式

本教材はOSC通信という通信方式で、プログラミングタブレットと信号ブロック内のWi-Fiモジュール、およびマップ表示モニタと信号ブロック内のWi-Fiモジュールが通信する。プログラミングタブレットが作成した信号のプログラムを信号ブロックへ送信し、信号ブロックがマップ

表示モニターへ光った LED の情報を送信する。

4.5 数直線タイプのビジュアルプログラミング

4.5.1 数直線タイプの概要

本研究では、数直線タイプのビジュアルプログラミング言語を開発した(図4)。数直線タイプとは、数直線上にプログラムを書き込む吹き出しをつけることでプログラミングする、新たなビジュアルプログラミング言語である。

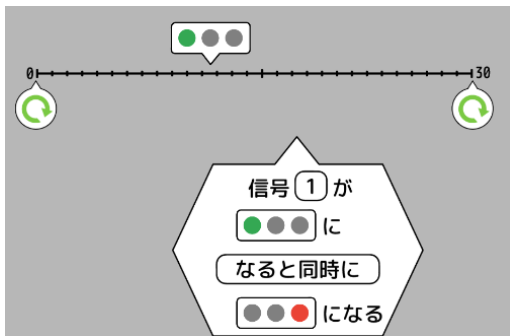


図4 数直線タイプのビジュアルプログラミング

ビジュアルプログラミング言語のブロックタイプやフロータイプで信号機の動作を表現すると、1つ1つコードを読まないとのタイミングで電球の色が切り替わるのかわかりにくい問題点がある。数直線タイプは、色の切り替わるタイミングが視覚的にわかりやすい点と、Scratchやembotのように事前知識を必要としない点の2つのメリットを持つ。

4.5.2 吹き出しの種類と使い方

吹き出しは順次吹き出しと反復吹き出し、分岐吹き出しの3種類を作成した(図5)。

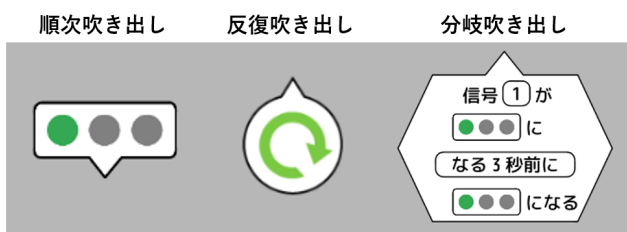


図5 吹き出しの種類

吹き出しは3種類とも、吹き出しのツノ部分をスワイプすると、吹き出しを左右に動かすことができる。実行のタイミングは吹き出しの位置によって変えることができる。

順次吹き出しは信号機の部分をタップし青・黄・赤の色の選択をする。

反復吹き出しは2個セットで使用する。それぞれ吹き出しの位置を移動でき、数直線上の左にある方が反復の開始位置、右にある方が反復の終了位置となる。この2つの反復吹き出しで囲った部分のプログラムが繰り返し実行さ

れる。

分岐吹き出しは、プログラミングする信号ブロックとは別の信号ブロックの情報によってプログラムを実行する。分岐条件の対象となる「信号ブロックの番号」と、「条件の信号の色」、「条件」、「実行内容」をタップで選択する。「条件の信号の色」と「実行内容」は順次吹き出しと同じく、信号機の部分をタップし青・黄・赤の色の選択をする。「信号ブロックの番号」は『1』『2』『3』『4』の4パターン、「条件」は『なると同時に』と『なる3秒前に』の2パターンを用意した。

4.5.3 プログラミング方法

前述した3種類の吹き出しを使い、数直線に吹き出しを付けてプログラミングする。プログラミングタブレットの画面の右上部に、「送信」ボタンがあり、それぞれの数直線上に「削除 OFF」「順次」「くり返し」「条件」ボタンがある(図6)。

「順次」ボタンを押すと順次吹き出しが、「くり返し」ボタンを押すと反復吹き出しが、「条件」ボタンを押すと分岐吹き出しが、押したボタンと対応する数直線上に追加される。「削除 OFF」ボタンを押すと、ボタンの文字が「削除 ON」になり、その状態で吹き出しをタップするとタップした吹き出しを削除することができる。「削除 ON」の状態では「削除 ON」ボタンを押すと、ボタンの文字が「削除 OFF」になり、吹き出しの削除はできなくなる。吹き出しを追加したり削除したり、吹き出しの内容や位置を変更してプログラミングを行う。最後に「送信」ボタンを押して信号ブロックに作成したプログラムを送信する。

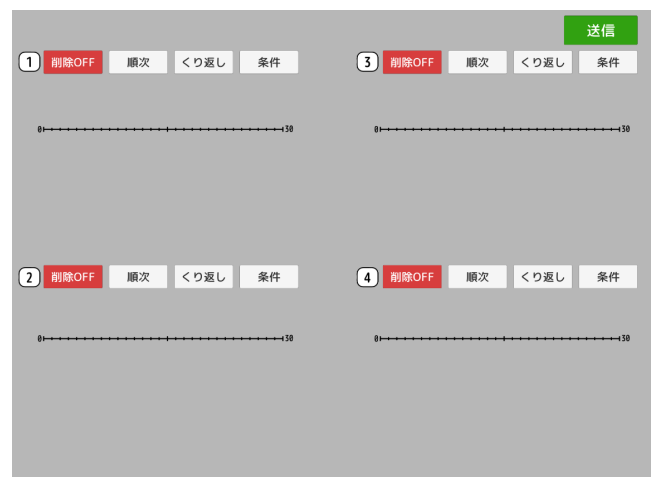


図6 プログラミング画面

4.6 操作方法

4.6.1 操作方法の種類

本教材は、信号ブロックのボタンを直接押して制御するボタンモードと、プログラミングで信号ブロックを制御するプログラミングモードの2種類がある。

多くの人はプログラミングする際、無意識に次のステップを踏んでプログラミングしている。まずはじめに実装する内容を考え、次に実装をするため順次、分岐、反復により表現できるアルゴリズムを考える。そして、最後にプログラミングを行う。しかしながら、実装内容を考えることや、それをアルゴリズム化することは、プログラミング経験の乏しい児童にとってハードルが高い。本教材では、児童が取り組みやすいよう、実装内容を考えるステップとアルゴリズム化のステップを、ボタンモードでサポートする。最後に、ボタンモードで考えた実装内容をもとに、プログラミングモードでプログラミングを行う。

まず児童はボタンモードで遊び、次にプログラミングモードで遊ぶ。児童はボタンモードでどのような制御をすればいいのかプログラムを考え、プログラミングする前にアルゴリズムなどの情報を整理する。次に児童はプログラミングモードで信号ブロックをプログラミングする。車をスムーズに走行させるため繰り返しプログラミングを行い、渋滞や事故が起こらないように信号機をプログラミングするという遊び方である。

4.6.2 ボタンモード

ボタンモードとは、信号ブロックのボタンを直接押して制御するモードである。ボタンモードは任意の数の信号ブロックとマップ表示モニタを用いる。児童ははじめにマップ表示モニタ上の道路の交差点に信号ブロックを置き、信号ブロックのボタンを直接押して手で電球を点灯させる。その信号機の電球の色に従って画面上の車が動く。信号ブロックは、赤色のボタンを押すと他のLEDは消灯し赤色のLEDが点灯する。黄、青も同様にそれぞれの色のLEDのみが点灯する。

4.6.3 プログラミングモード

プログラミングモードとは、プログラミングで信号ブロックを制御するモードである。プログラミングモードは任意の数の信号ブロックとプログラミングタブレット、マップ表示モニタ(図7)を用いる。はじめに児童はプログラミングタブレットで信号ブロックをプログラミングし、プログラミングをし終わったら、信号ブロックをマップ表示モニタの道路の交差点に置く。児童がマップ表示モニタで「スタート」ボタンを押すと信号ブロックが光り出し、信号ブロックの情報に従い車が走ったり停止したりする。車に渋滞や事故が起こると、マップ表示モニタの画面左上のライフが1つ減る。残り時間が0になるまでライフを減らさないように、繰り返しプログラミングする。

5. おわりに

本研究ではプログラミング教育のねらいを達成できる教材について検討し、信号機を題材としたプログラミング教材の開発を行った。新たに数直線タイプのビジュアルプログラミング言語を考案し、タンジブルなインターフェースで

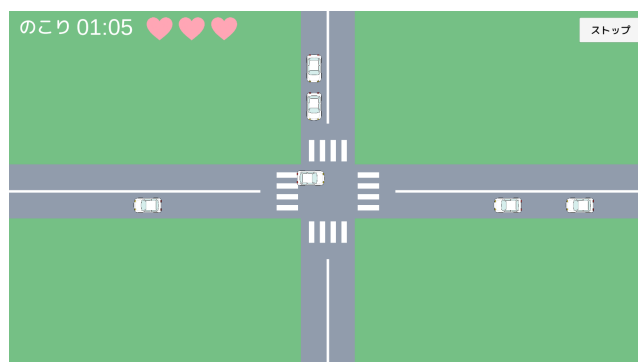


図7 マップ表示モニタの画面

ある信号ブロックを用いた、信号機のプログラミングを行い、プログラミング的思考を学ぶことを支援する教材を開発した。

今後は、実際に児童に本教材を体験してもらい、小学生に対し研究目的が達成できるか評価実験を行う。また、矢印式信号や歩行者信号などの追加を行い、応用的な部分についても学習支援できるよう、さらなる学習者の支援を行いたい。

参考文献

- [1] 文部科学省. 小学校プログラミング教育の手引(第三版). https://www.mext.go.jp/content/20200218-mxt_jogai02-100003171_002.pdf.
- [2] Mitchel Resnick, John Maloney, Andrés Monroy-Hernández, Natalie Rusk, Evelyn Eastmond, Karen Brennan, Amon Millner, Eric Rosenbaum, and Jay Silver. Scratch: Programming for all. *Communications of the ACM*, Vol. 52, No. 11, pp. 60–67, 2009.
- [3] Sony Marketing Inc. Mesh. <https://meshprj.com/jp/>.
- [4] e craft. embot. <https://www.embot.jp/>.
- [5] Sphero Edu. sphero. <https://sphero-edu.jp/>.
- [6] Thinkfun. Robot turtles. <https://www.thinkfun.com/robot-turtles/index.php>.
- [7] 間辺広樹, 並木美太郎, 兼宗進. 小学生向けプログラミング学習教材「まなべー」の開発と教育効果. 情報処理学会研究報告, Vol. 2015-CE-131, No. 6, pp. 1–8, 2015.
- [8] 小原楓, ミークライサージャッポン, スリプラプルチャイカントポーン, 平井佑樹, 金子敬一. 操作カードによるプログラミング体験システムの開発と評価. 情報処理学会研究報告, Vol. 2016-CE-133, No. 1, pp. 1–8, 2016.
- [9] 文部科学省. 端末利活用状況等の実態調査(令和3年7月末時点)(確定値). https://www.mext.go.jp/content/20211125-mxt_shuukyo01-000009827_001.pdf.
- [10] 石井裕. 仮想と現実の融合: 3. タンジブル・ビット-情報と物理世界を融合する, 新しいユーザ・インタフェース・デザイン-. 情報処理, Vol. 43, No. 3, pp. 222–229, 2016.
- [11] 青木良輔, 宮下広夢, 井原雅行, 大野健彦, 千明裕, 小林稔, 鏡慎吾. くるみる: 複数導電部もつ枠型物理オブジェクトを用いたタブレット操作. 研究報告ヒューマンコンピュータインタラクション(HCI), No. 6, pp. 1–8, 2011.