

VR ゲームコントローラーを題材としたプログラミング教育教材

石井 敬太[†] 尾上 楓真[†] 浅岡 正教[‡] 河西 紀明^{††} 河並 崇[†]

金沢工業大学[†] 一般社団法人 FAP[‡] 合同会社 DMM.COM^{††}

1 はじめに

近年, VR, AR, MR など xR 技術の進歩により, 安価でかつ本格的な xR 環境を身近に体験できる機会が増えてきている. 特にゲーム分野において, VR への注目度は高く, 多くの作品がリリースされてきている. 一方で, その開発の担い手の育成が急務である. しかしながら, 一般に VR ゲームの開発は高い専門性が要求されるため, プログラミング教育において, 特に初心者向けのプログラミング教育教材は少ないのが現状である.

そこで本研究では VR ゲーム開発に興味を持たせることを目的に, VR ゲームコントローラーの開発を題材としたプログラミング教育教材の提案を行う.

本研究で用いる VR ゲームコントローラーは, イギリスの公共放送局である BCC が主体となって作った教育用マイコンボードである micro:bit を用いる. VR ゲームを進めるためには, ゲームプレイヤーが自身の micro:bit に新しい機能をプログラムしないと進めないようにすることにより, ゲームを楽しむのと同時にプログラミング技術を学ぶことができると考えた.

2 教材プラットフォーム

教材のプラットフォームは図 1 のように, プログラミング環境と VR ゲーム実装環境に分ける. プログラミング環境では micro:bit のセンサーの値や, 文字列などを VR ゲーム実装環境の micro:bit に送信する. VR ゲーム実装環境の micro:bit はプログラミング環境の micro:bit から受け取ったデータをシリアル通信でパソコンに送り, パソコンの Unity によって開発されたゲームがデータを受け取り, ゲームに変化を与える.

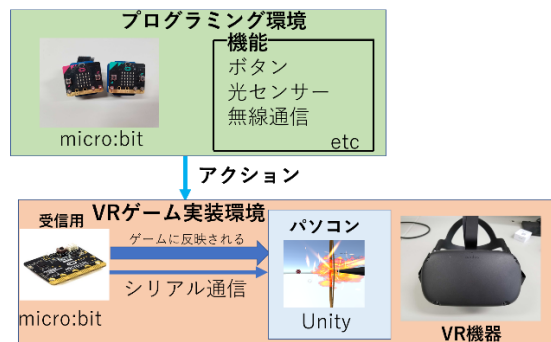


図 1 教材プラットフォーム

2.1 プログラミング環境

VR 専用コントローラーとして利用する micro:bit は, ビジュアルプログラミングが可能であるため, 子どもでもプログラミングが行いやすい. 加えて, 開発環境が Web

Programming education materials based on VR game controllers

[†] Kanazawa Institute of Technology

[‡] FAP association

^{††} DMM LLC

ブラウザ上で動作するため, 環境構築が不要なこと, オフラインエディタを利用すればオフライン状況でもプログラミングが可能なことなど, プログラムを行うための準備が簡単である. また, ボタンや光センサー, 加速度センサー, 電波強度などを利用したプログラムを容易に書くことができるため, ゲームコントローラーとの相性がとても高いと考え選定した.

2.2 VR ゲーム実装環境

VR ゲーム実装環境は, 本研究においてはプログラミングの対象とはせず, あらかじめ決められた文字列などに応じてゲームに変化をつける実装としている. VR ハードウェアとして oculus quest を使用し, ゲーム開発エンジンとしては oculusSDK が配布されている Unity を使用した. Unity はマルチプラットフォームに対応しており, ストアでほかのユーザーが作成した素材や部品などが簡単に手に入れることができる. oculus quest は単体で VR を楽しめる機器であるが, 今回はパソコンと通信を行うため, oculuslink 機能を使用してパソコンと接続する.

3 教材の流れ

教材の流れを図 2 に示す. まずステージをクリアする際に必要になる機能を課題として提示する. その後チュートリアルで機能を学び, プログラムを書き換える. 書き換えたプログラムでゲームを遊び, ステージをクリアする. クリアするとまた次のステージで必要になる機能を課題として提示する. 以上のサイクルを繰り返すことで, プログラムを学習する.

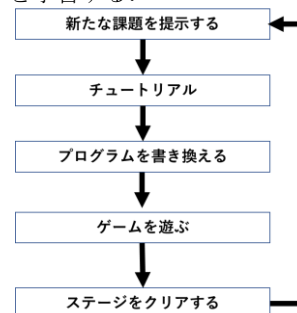


図 2 教材の流れ

表 1 課題と目的と難易度

課題	目的	難易度
矢を番える	ボタンを押すプログラムを組めるようになる	易しい
矢を放つ	ボタンを2種類使うプログラムを組めるようになる	易しい
矢の種類を変更する	矢の変更方法がわかる	易しい
コントローラーを振ってパワーをためる	変数の使いかたが分かるようになる	普通
一つのプログラムで2種類の矢を使う	変数と条件分岐を利用した状態遷移がわかる	難しい
micro:bitの距離で矢の飛距離を変える	通信強度の使い方がわかる	難しい

教材として学びやすくするために、1サイクルにおけるプログラムの変更を少なくし、サイクルを多くすることで、プログラムの経験が少ない人であっても学びやすいように工夫している。教材でステージの変化などによる教材の課題と目的と難易度の一例を表1に示す。次節以降で具体的な教材を示す。

3.1 矢を番える・放つ課題

ゲームを開始した直後はコントローラーのボタンを押しても何も変化が起きない。そこで、図3のようにプログラムに変更することで、ボタンを押した際に、文字列を送信することができ、矢を番えることが可能になる。この時の変化を図4に示す。この課題によって文字列を送信することでゲームに変化を与えることができることを理解させる。この後、矢を放つ課題によって定数を用いて一定のパワーで発射することを学ぶ。



図3 矢を番えるプログラム

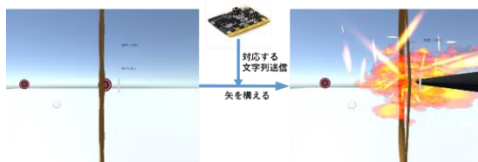


図4 送信された文字列によっておこる変化

3.2 パワーをためる課題

先の課題ではパワーを定数として利用しているが、この課題では変数を利用する。プログラムとしては図5のようになる。また、ゲーム実装環境側で変数の値によって弓の横のゲージが増減するようになっている。その時の変化を図6に示す。この課題によって変数の使い方を理解してもらうことを目標とする。



図5 パワーをためるプログラム

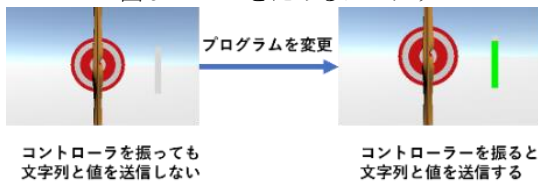


図6 送信された文字列と値によっておこる変化

3.3 放つ矢を変更する課題

この課題では図7に示すような2種類の的を用意し、的の色によつて的を破壊できる矢の種類を変えなければならなくなっている。そのため、ゲームプレイ中に矢の種

類を変更する必要がある。矢の種類を変えること自体は、送信する文字列を変えるのみなので簡単に行うことができるが、micro:bitの2つのボタンだけでは実装できず、変数を用いて飛ぶ矢を変更するなど、条件分岐を利用したプログラムを学ぶことができる。

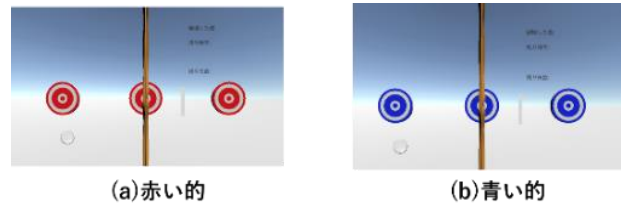


図7 的の種類

3.4 電波強度によってパワーを変更する課題

この課題では、コントローラーをより弓のイメージで扱うために、矢のパワーを2つのmicro:bit間の電波強度によって変更するようなプログラムに変更する。これにより、図8のように弓を現実にくくようにmicro:bit同士を離れた距離に応じて矢が飛ぶようになる。この課題によって電波強度の扱い方を理解することができる。



図8 弓を引く構え

3.5 その他考えられる課題

micro:bitの他のセンサー利用として、光センサーを用いて明かりの強さによつて的を照らす範囲を変えたり、温度センサーによつてパワーや属性を変えたりする課題などが考えられる。また、これらも含め、これまでの課題の組み合わせによるコンボ技(連続攻撃)などを組み込むことで、より複雑な教材にすることが可能である。

4 予備評価

micro:bitに関する別の講習を受けた18名の小学生を対象に、本教材の通信強度を利用したVRコントローラーのデモを体験してもらいアンケートを行った。各内容に対して1~5段階で評価してもらい当てはまる場合を1とし、当てはまらない場合を5とした。アンケートの内容と平均値を表2に示す。

表2 アンケート結果

プログラミングに興味を持ったか?	1.17
弓矢のデモを見て興味を持ったか?	1.39
コントローラーをもっと改造してみたいと思ったか?	1.39

アンケートの結果からプログラミングに対して興味を持たせることが出来たことが分かった。また、コントローラーを改造してみたいかという項目も概ね高い評価を得たことから、教材としての将来性があることがうかがえた。

5 まとめ

本研究ではプログラミング教材としてVRゲームに注目し、VRゲームのコントローラーをプログラムする教材の開発を行った。

今後の展開として、VRゲーム実装環境として、DMM VR Connectとの連携をはかり、より本格的なゲームを用いたプログラミング教材開発を行う予定である。