

実践報告

ICTを活用した幼児教育の充実

— 幼児向け三択クイズの修正・機能追加 —

Enhancing Early Childhood Education Utilizing ICT: Modifications and additions to the multiple-choice quizzes for preschoolers

石部忠之 国際学院中学校高等学校

前回、Scratch を活用したタブレットPCによる幼児向け三択クイズの制作手順を示し、その基本構造を紹介した。今回はその続編として、幼児が実際に触ることを想定した機能追加と改良を行った。幼い子どもにとってより扱いやすく、インタラクティブな部分についても力を入れてプログラムをブラッシュアップしていきたい。

キーワード: ICT、タブレットPC、幼児教育

1. はじめに

前回の報告では、三択クイズプログラムの処理手順などを中心に作成してみたが、今回は機能の追加や充実に取り組んでみた。幼児にとって、より親しみやすい画像や、想定外の操作に対応するためのプログラムの改良、キャラクターに動きを加えたり、効果音を追加したりすることなどで、魅力あるアプリケーションになるように心掛けた。前回のコードに修正や追加を行うことで、上記機能の実装をした。

2. クイズの制作

2-1 想定外の操作に対応

前回のプログラムで、正しく動作するようにはなったが、幼児がタブレットを操作することをイメージすると、不必要な操作（タップ）などをすることも考えられる。そこで、幼児の予期せぬ操作に対しても正常に動作するように、プログラムの修正を行う。そのため、入力が可能かどうかを判定するためのフラグ用変数を追加する。

ちなみに、フラグとはFlag（旗）のことで、状態を表す変数として、しばしば用いられる。今回は「入力可能な状態か」「入力不可能な状態か」を「1」か「0」で判断する。プログラム言語によっては、真理値（BOOL型）を扱うことができるものもあるので、その場合は「True」、「False」で処理することが一般的である。

今回のプログラムでは問題を出した後、3匹のスプライトたちが解答をそれぞれいうので、その間は入力不可能な状態（=0）とし、3匹目のスプライトの解答が終わったあとに、入力可能な状態（=1）とする。そして、いずれかのスプライトがクリック（タップ）

されたときに、もう一度入力不可能な状態 (= 0) に戻す、といった処理を追加する。
以下に、修正後の各スプライトのコードを示す。(図 1、図 2、図 3、図 4)

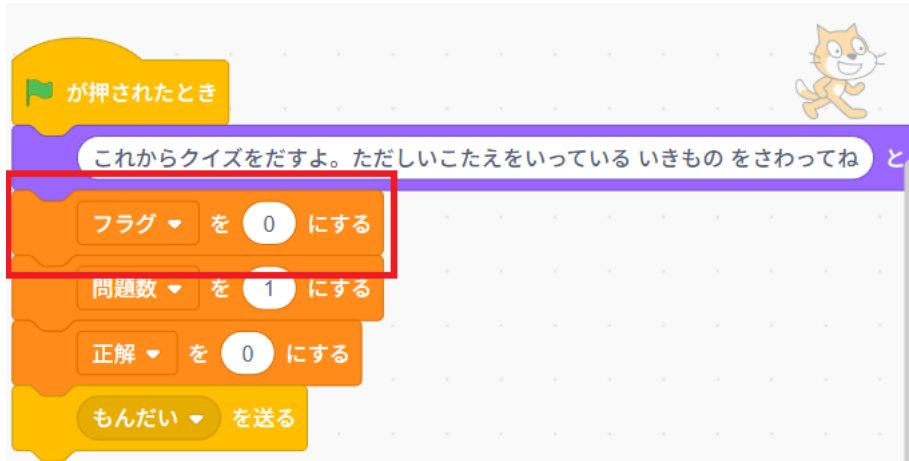


図 1 修正後のネコ (スプライト1) のコード画面



図 2 修正後のカニ (Crab) 用のコード画面



図 3 修正後のカエル (Frog) 用のコード画面



図 4 修正後のテントウムシ (Ladybug1) 用のコード画面

以上の修正を加えたことで、問題文を読み上げている最中などに誤ってクリック（タップ）した場合も、反応することなく、正しく動作することが確認できた。

2-2 効果音の追加

正解したスプライトをタップしたときに、効果音を加えることによって、より魅力的なプログラムになる。効果音については、デフォルトで用意されている音源も豊富にあるので、今回はその中からいくつか選択して利用することにしたが、外部音源を取り込むことも可能なので、その点については、機会があれば試してみたいと思う。

ゲーム開始の効果音、正解、不正解の場合の効果音の追加について、以下にコードを示す。

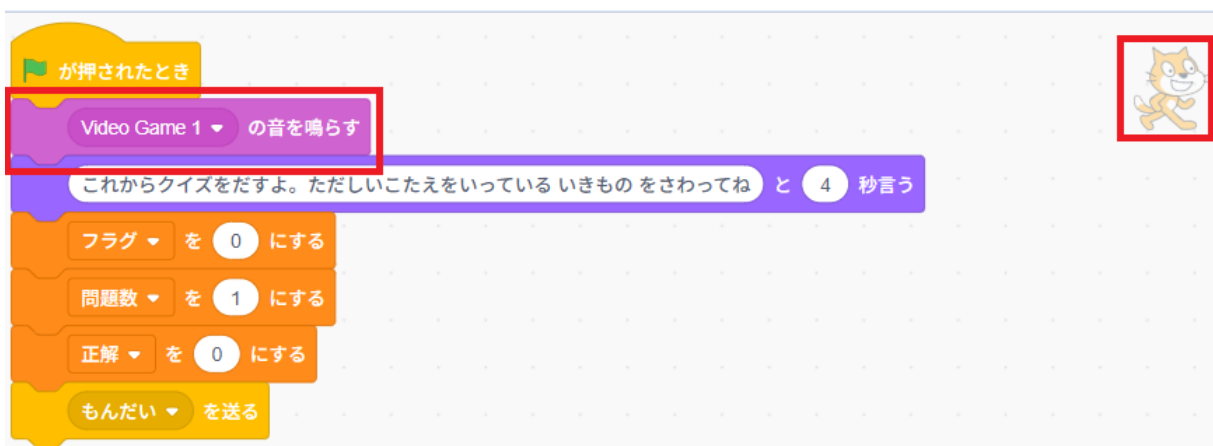


図 5 ゲーム開始の効果音のコード画面

正解、不正解の際のコードはカニ (Crab)、カエル (Frog)、テントウムシ (Ladybug) はすべて同様であるので、カニのコードを示す。



図 6 正解、不正解のときのカニのコード画面

2-3 スプライトの追加

更に、タップされたキャラクターが動くことによって、アニメーションのような効果が得られる。今回用いているスプライトには、それぞれ2つずつのデザインが用意されているので、タップされたときに、別のデザインに切り替えて表示して、次の問題に進む際に元のデザインに戻すコードを追加する。

これについても、カニ (Crab)、カエル (Frog)、テントウムシ (Ladybug) はすべて同様であるので、カニのコードを示す。

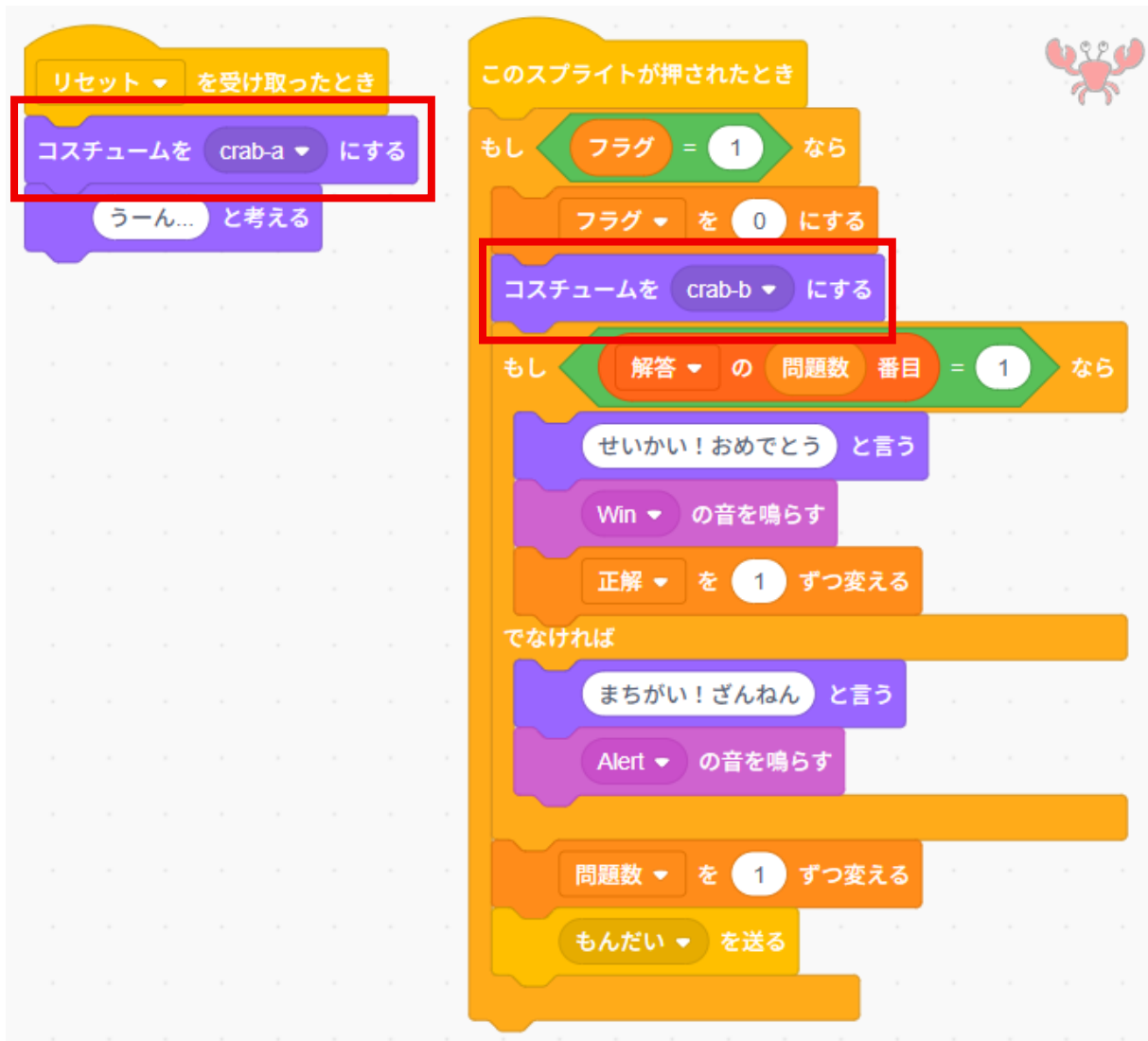


図 7 スプライトのコスチュームを切り替える

2-4 全問正解の場合の処理

最後に、全問すべて正解していた場合のみ、特別感を出すための処理を追加する。この際に用いるのが、「もし～ならば～、でなければ～」というブロックである。このブロックによって、条件分岐を行うことができるので、正解数が 10 問の場合の処理と、それ以外の処理に分けて実行する。なお、今回は出題数は 10 問に固定したので、コードの中にも定数「10」を入れたが、出題数の変更も考えると「(変数) の長さ」を用いた方が望ましい。



図 8 全問正解の場合のコード画面

エラー処理、効果音挿入、コスチューム切り替え、全問正解処理などを追加することで、新たな動きが加わり、より楽しめるようにコード等の修正を行うことができた。

修正後のプログラム

<https://scratch.mit.edu/projects/1193910580/embed>

3. おわりに

今回の取り組みでは、前回制作した幼児向け三択クイズに対し、機能追加やプログラム修正を行うことで、より魅力的で操作しやすいアプリケーションとなるよう改善を進めた。Scratchにはまだまだ多くの機能や可能性があり、今後もアプリケーション開発の観点から幅広く研究を続けていきたい。また、実際の教育現場からの声やニーズを反映させた開発にも、積極的に取り組んでいく必要性を感じている。

著者の利益相反：開示すべき利益相反はない

文献

- 1)石部忠之(2025) ICTを活用した幼児教育の実践に向けてータブレットPCを用いた幼児向け三択クイズの制作ー. 国際学院埼玉短期大学研究紀要 2025; 55: 1-8.