

# 自動販売機を題材にした QRコード入力によるプログラミング体験アプリの開発

## Development of a Simulation Software for Programming by QR Code Inputting Based on Vending Machine

テーマ：インターネット技術とその応用  
指導教員：松本 章代

教養学部 情報科学科  
1157227 齋藤 綱太

### 1. 研究の背景

パソコンやスマートフォンはもはや我々の生活の中で欠かせないものになっている。しかし、それらのデジタル機器を制御するプログラミングについて全く知らない人が多数である。情報科学科に入って初めてプログラミングに触れたという学生もいる。そのため情報科学科に入ってからプログラミングが苦手だと気づいた学生もいるのが現状である。これらの大きな要因としてまだプログラミングは身近なものでなく、プログラミングを体験する機会がほとんどないからであると考えられる。そこでプログラミング体験アプリを開発し、プログラミング未経験者に体験してもらう。体験してもらうことでプログラミングに関心を持ってもらえればプログラミングの普及につながると考えた。

### 2. 関連研究

伊藤 [1] は授業科目として「Scratch」をカリキュラムに取り入れた。Scratchとはプログラムコード打ち込ませることなく、コマンドブロックと呼ばれるブロックを順番に並べ処理することで、未経験者でも簡単にプログラミングを体験することができる。この報告によると、体験者に実際のプログラムコードを打ち込ませるのではなく、コマンドブロックを並べるというよりわかりやすい方法でプログラミングさせることで、授業時の質問は生産的かつ前向きになり、“命令事態をどう書けばいいかわからない”という白旗状態に陥ってしまう学生がほぼ皆無だったと報告されている。

本研究のプログラミング体験でも、プログラムコードを打ち込ませずにプログラミングをさせる方法を取り入れたい。相違点としては、自動販売機の動作をプログラミングをするという題材に沿ったプログラミング体験をさせる点である。

### 3. 研究の目的

#### 3.1 先行研究の検証結果

先行研究では、プログラミング未経験者を対象にしたプログラミング模擬体験アプリの開発がテーマとなっている [2]。この研究では OpenCV\*1 [3] を使って色紙の色をタブレット端末のカメラで認識することによりプログラミングを模擬体験できる Android アプリが開発された。この研究の評価実験結果によると、色認識を使ってプログラミングをすることでプログラムコー

\*1 OpenCV とはオープンソースの画像処理・画像認識用のライブラリであり、これを用いることによってコンピュータビジョンに関する高度な画像処理機能を容易に利用できるようになる。

ドをタイプしてプログラミングするよりもハードルを下げることができ、色認識というユニークなインターフェースを用いることでプログラミング未経験者に対して興味を持たせることができたという結果がでている。

#### 3.2 到達目標

本研究でもタブレットのアプリによるプログラミング体験を考えた。タブレットのアプリを採用することで、体験者はプログラミングをする環境を整える必要もなくなり、短時間でのプログラミング体験が可能になる。また、体験者にプログラミングをする目的意識をもってもらうために、なにか1つ題材を定め、それに沿って体験させたいと考えた。生活の中で身近なものを例にあげてプログラミングできれば未経験者も興味を持ちやすいのではと考え、私たちが普段から使っている自動販売機を題材にすることにした。入力インターフェースについては、未経験でも簡単にプログラミングができる方法を取り入れたいと考えた。当初は、文字認識を用いた入力を検討していたのだが、認識精度や認識時の環境状況に左右されてしまう点から、この案は実用的ではないという結論に至った。そこで、これらの問題をクリアできる QR コードを用いた入力インターフェースの実装することに決定した。

到達目標としては、以上で説明したプログラミング体験アプリを開発し、最終的に、ビジュアルプログラミング言語\*2の App Inventor2 (以下 AI2) で同様のシステムを開発し、入力インターフェース部分で評価実験することを目指す。評価対象として AI2 を採用した理由としては、インタフェース部分での検証を狙っているため、それ以外の部分は共通させる必要がある。そのため、実行を同じタブレット環境で検証できる AI2 を評価対象にすることに決めた。

### 4. システム概要

このアプリの特徴として、体験者にプログラムコードを打ち込ませずに、事前に与えられた処理 (図1参照) をユーザーが選択して入力できるようにする。体験者には以下の4つが行われた時に動く処理をプログラミングしてもらう。

- お金が投入された時
- 購入ボタンが押された時
- ジュースを出した時

\*2 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、視覚的な操作でプログラミングするプログラミング言語である。

<ul style="list-style-type: none"> <li>・効果音がなる</li> <li>・ボタンの色が変わる</li> <li>・<math>G + \text{投入された金額} \rightarrow G</math></li> <li>・<math>G</math> をモニターに表示する</li> <li>・<math>0 \rightarrow G</math></li> <li>・<math>G - \text{選択されたジュース代金} \rightarrow G</math></li> <li>・<math>(G \div 100)</math> の商 <math>\rightarrow S</math></li> <li>・<math>(G \div 100)</math> の余り <math>\rightarrow A</math></li> <li>・<math>(A \div 50)</math> の商 <math>\rightarrow S</math></li> <li>・<math>(A \div 50)</math> の余り <math>\rightarrow A</math></li> <li>・<math>(A \div 10)</math> の商 <math>\rightarrow S</math></li> <li>・100円を <math>S</math> 枚出す</li> <li>・50円を <math>S</math> 枚出す</li> <li>・100円を <math>S</math> 枚出す</li> </ul>
例外処理
<ul style="list-style-type: none"> <li>・<math>G &lt; 0</math> の場合 (*1)</li> <li>・<math>G \geq 0</math> の場合 (*1)</li> <li>*1 が選択された時は以下の処理を追加で入力する</li> <li>・<math>G + \text{選択されたジュース代金} \rightarrow G</math></li> <li>・選択されたジュースを出す</li> </ul>

図 1. 入力する処理の一覧



図 2. 実行画面

● 返却レバーが回された時

処理の入力画面と自動販売機の実行画面を遷移させながら、プログラムの動作を体験してもらう。入力した処理は実行画面上の右側（図 2 参照）に実行中の処理が時間差で表示されていく。このような表示にすることによってユーザーはどの処理が現在動いているかをリアルタイムに確認することができる。QR コードによる入力インターフェースについては、オープンソースの ZXing<sup>\*3</sup> を用いて実装する。

### 5. 評価実験

評価実験は以下の 2 グループ、各 10 人ずつに分けて行う。

- 1 回目に本アプリを体験してから、2 回目に AI2 を体験する A グループ

- 1 回目に AI2 を体験してから、2 回目に本アプリを体験する B グループ

グループに分けることで、体験する順番によって実験結果が偏ることを回避するためである。

実験は以下の手順で進める。

<sup>\*3</sup>ZXing (ゼブラクロッシング) は一次元・二次元バーコード処理ライブラリであり、多くの QR コード読み取りアプリで採用されている。

表 1. A グループの結果

評価段階	1	2	3	4	5
体験前	0	6	1	1	2
本アプリ体験後	5	3	0	2	0
AI2 体験後	4	3	1	2	0

表 2. B グループの結果

評価段階	1	2	3	4	5
体験前	3	2	3	1	1
AI2 体験後	4	3	3	0	0
本アプリ体験後	3	6	0	1	0

1. 体験する前にプログラミングに対するアンケートに記述してもらう。
2. 1 回目の体験が終わった後にも、体験する前と同じ質問を含めたアンケートに記述してもらう。
3. 2 回目の体験が終わった後にも、体験する前と同じ質問を含めたアンケートに記述してもらう。
4. 最後に各体験を比べて、良かった点と悪かった点および簡単な感想を記述してもらう。

実験の結果によるプログラミングへの興味の変化を表 1, 表 2 に示す。評価段階は、1 がいい、2 がどちらかと言えばいい、3 がどちらでもない、4 がどちらかと言えばいい、5 がいいである。

この実験結果によると、両グループともに、プログラミング体験後は評価段階の 5 に該当する人はいなかった。A グループで、体験前に評価段階の 1,2 にあてはまる人が 60 % だったが、本アプリ体験後では 80 % に増えた。B グループで、体験前に評価段階 1,2 にあてはまる人が 50 % だったが、本アプリ体験後では 90 % に増えた。この結果から、本研究でのプログラミング体験を通して興味をもたせることができたのではないかと考えられる。

### 6. まとめ

QR コード入力を用いることでプログラミング時間の短縮をすることができ、体験者はなにをすればいいか見失うことなく体験することができた。自動販売機を題材にしたことで、“プログラムが裏でこんな動作をしているのには驚いた”や“プログラミングをより身近に感じる事ができた”との意見があったので、題材に沿ったプログラミング体験の効果はあったと感じることができた。今後の課題としては、“プログラミング画面と実行画面を切り替えながら動作を確認することがやりづらい”との意見もあり、画面の大きさといった制約もあるために検討を続ける必要がある。

#### 参考文献

- [1] 伊藤一成: プログラミング, 何をどう教えているか: Scratch を用いた授業実践報告, 情報処理, Vol.52, No.1, pp. 111-113(2011).
- [2] 鈴木 健太: Android タブレットを用いた色認識によるプログラミング模擬体験アプリの開発, 東北学院大学教養学部卒業論文 (2014).
- [3] OpenCV 公式サイト, <http://opencv.org/>