プログラミング学習の視覚的かつ容易な方法の提案

Online Teaching for Programming with a Visual Approach

荒木 拓\*，内山 将吾\*，川中 楓\*，西田 匠吾\*，庭野 太陽\*，榮岩 拓見\*，仲道 嘉夫\*\*

1. **概要**

初学者がプログラミングをする際，考えを整理するためにフローチャートを利用することは有用である。フローチャートでプログラムを視覚的に表現することができるからである。昨年，学習指導要領にプログラミングが追加(1)され，今後は小中学生や一般の人など，これまでプログラミング触れる機会が少なかった層の人たちも学習する機会が増えていくと思われる。そこで私たちの班は，今後重要性が認知されて,需要が上昇すると考えられるプログラミング学習において，特にフローチャートを取り入れた学習方法を提案する。

対象の言語はC言語とした。C言語は班員全員が使いこなせる言語であり，開発がスムーズに行える。多くの人が初めて学習する言語であり，文献も多く存在して学習に適している。また，利用者が多いPHPやpythonよりも型定義が明確で基本的なデータの扱い方を学ぶには良いと考えた。

図 1 学習の流れ

以上を踏まえ、私たちはフローチャートによってC言語を学習できるスマートフォン向けアプリケーションを開発することにした。スマートフォンでアプリを作成することで，パソコンを持っていないような方でも学習を進めることができ，学習を始めるハードルを低くできる。

1. **現在までの経過**
   1. **開発環境**

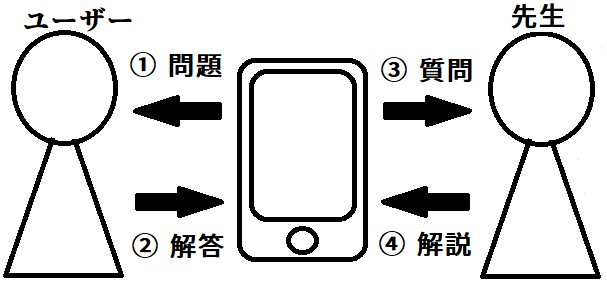
開発にはUnity(2)を使用した。Unityとは主にゲーム開発に用いられるアプリ開発プラットフォームである。Unityを使用した理由として，iOS含む複数のOSに1つのプロジェクトファイルからアプリを作成することができるという利点があげられる。AndroidStudioやXCodeを使用することもできたが，これらは対象のスマートフォンが限られてしまう。Unityは班員の経験もあったためUnityを使用することとした。

* 1. **全体像**

学習の流れを図１に示す。

1. ユーザーがアプリケーションから問題を受け取る。
2. 問題に対し、フローチャートで回答する。
3. 回答を確認し、質問があればwebを経由して質問する。
4. 先生がwebを通して解説を行う。

このような流れで、学習が行われる。



* 1. **フローチャートアプリについて**

**2.3.1ブロックを追加,削除する機能**

初めに必要なのが，フローチャートを自分で組み立てる機能である。図2の4つの矢印でカーソルを移動し，画面のどこに操作を加えるかを選択できる。右側のバーにあるボタンをクリックすることでその見た目に対応したブロックが中央の画面のカーソル部分に追加される。

現在、実装したブロックは以下である。

**表 1 実装済みブロック一覧**

|  |  |
| --- | --- |
| printf |  |
| if |  |
| for |  |

**2.3.2ブロックの中身を編集する機能**

各ブロックの中身は編集できるようになっていて，たとえばprintfであったら出力する中身，ifであったら真偽のそれぞれの条件を入力できるようになっている。

**2.3.3コード表示の機能**

自分が作ったフローチャートを実際のC言語のコードに変換する機能である。この機能によって，フローチャートのどこがC言語のコードに対応付けられているかが確認できる。

**2.3.4機能の詳細**

各ボタンと画面上の操作の対応付けはUnityの機能を使用することで実装することができた。内部的には, ボタン座標内でクリックを離す(あるいはスマホならタップして指が離れた)瞬間に指定した操作が行われる。

また，セーブなどの機能の実装に当たり、二次元座標系を取り入れた。これにより、各ブロックの位置やブロックの内容（ifの条件やprintfの内容）をまとめて保存することができるようになった。また、取り入れる前ではセーブ時に実行される処理が多く、性能の低い端末での動作が不安定だった。この問題を、座標系を取り入れることによりセーブ時の処理を大幅に削減し解決することが出来た。加えて、元から動作が安定していた端末でもセーブの際の時間が大幅に短縮されたので、アプリを利用する際の快適さが上昇した。

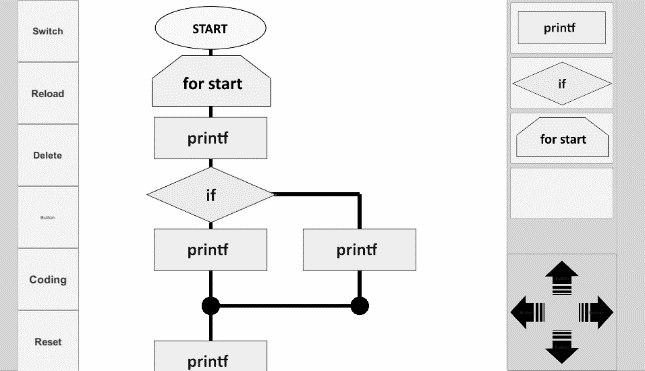


図2アプリ画面の例

* 1. **教育について**

**2.3.1教材の提供**

C言語についての解説，またフローチャートで何を作るのかというわかりやすい目標を与える必要がある。そこで，自分たちで作った課題を学習者に提供し，答えてもらった回答のどこが間違っているかを判定するシステムをアプリ内に実装する。

**2.3.2質問と回答**

課題に取り組んだり，自分で学習したりする時に出てくる疑問点を解消するために，質問を投稿する機能，質問に答える機能を実装し，学習者同士でも解決できるシステムをWeb上に実装する。

1. **問題点**

２つのブロックの間に新たなブロックを配置すると、セーブをしたときにそのブロックが認識されない。コードを書き出す機能に不十分な箇所がある。外部サービスのC言語のコンパイラ(paiza)への誘導が未実装。教育機能やサポート機能が未完成。同じ種類のブロックが複数存在している時、そのブロック同士の区別ができない（図2のprintfなど）。

1. **今後の予定**

今後は，３番で列挙した順に問題を解決していく。その後はUIの視認性やアプリの使いやすさの向上、各関数がそれぞれどのような動作をするのかわかりやすく表示することを目標に開発していく。

また，webの開発を進めて，教育として成立するように機能を充実していく。例えば現在は何か質問があった時，web上なにか質問があった時に教えることができるように掲示板やフォームのような形態のものを作っている。また, 課題に対して解答例をアップロードできるようにしたりして, ユーザー同士で課題を出し合ったりできるようなシステムなども教育の効果が望まれる。

これらを盛り込んで実際に教育に役立てるようなアプリを作る。

**参考文献**

1. 文部科学省：小学校プログラミング教育に関する概要資料（オンライン），入手先  
   〈https://www.mext.go.jp/component/a\_menu/education/micro\_detail/\_\_icsFiles/afieldfile/2019/05/21/1416331\_001.pdf〉[参照2021/06/07]