

目次

第1章 序論

第2章 プログラミングに対する意識調査

- 2.1 概要
- 2.2 調査結果
- 2.3 プログラミングとそのモチベーション

第3章 ビジュアルプログラミングツール 「Visual-Poyogramming」について

- 3.1 概要
- 3.2 使用方法
- 3.3 フィードバック

第4章 今後の課題・考察

付録A ソースコード

- A.1 index.html
- A.2 style.css
- A.3 code.js
- A.4 script.html

第1章

序論

現代において、情報技術は生活に欠かせない技術となっている。これまでは人力や原始的な構造で動いていた様々なものが情報技術により制御され、駆動している。

/* 実例 */

その情報技術の発展にエンジニアの存在は欠かせず、専門的な技術を持ったプログラマたちが今の社会の基盤を創り上げたと言って問題はないだろう。それほど現代と情報技術は深く関わっている。しかし時代は進み、必要な人材も変化しているのではと私は考えた。まだ情報技術と深く関わっていない分野、言うなれば新たな市場を発見し、開拓できるような人材がこれからの発展に有用だと言えるのではないか。/* 具体例。Suicaとか？ */無論、基礎的な情報技術の革新は限界には未だ遠く、通信速度、処理速度、小型化、軽量化など、より高性能な技術に向けて開発が進んでいる。それは近年のPCやポータブルデバイスの進化からも見て取れる。しかし、これらの技術者はコアであるがゆえに他の分野との関わりが薄い。通信と電波等、近い分野ならその限りではないが、新たな分野の開拓に進むのは難しいだろう。またアイデアがあったとしても、それを実現できる環境ではない。

情報技術というものは学問の一つでありながら一つのツールであり、様々な分野に応用できる多くの可能性を持っている。第一次産業である農業を例にとってみる。まず生産段階で、ITを利用したセンサや機械による農作物の管理が可能である。降雨量や湿度から栄養、水分量を調節することができる。流通に関しても、現在Web上のサービスにより多くの作物が売買されている。ローカルな特産品と現地に行かずに情報を得られるインターネットは相性がよい。このように多くの分野にITを組み込むためには、様々な分野で仕事をする人間が、基礎的な情報技術、もしくは情報技術に対する一定の理解を持っていることが必要であるのではないか。

そこで、本研究では「プログラミングをより理解しやすくするにはどうするか」を主題として、プログラミングに対する意識調査、そしてビジュアルプログラミングツール

「Visual-Poyogramming（以下VP）」の実装を行った。調査はプログラミング初学者が理解しづらい場面、箇所はどこか、そして学習意欲を削ぐ要因は何かについてのアンケートを実施した。またツールの作成に関しては、プログラムの構造をビジュアル化し、複雑なコーディングを可能にするよりはむしろアルゴリズム構築を基礎を学んでもらうことを目的とした。

第2章ではプログラミングに対する理解と、モチベーションを重視したアンケートによる考察を行う。また、先行研究や実際に稼働している初学者向けのサービスと合わせ、どのような形式、特徴を持ったものが初学者に優しく理解しやすいのかも考慮する。

第3章では実装したビジュアルプログラミングツール「VP」の解説を行う。構成や実装した機能、また利用者によるフィードバックを述べる。

第4章では全体を総括して今後の課題を述べる。本ツール「VP」はプロトタイプであり、今後実装するツールに向けどのような機能、デザインが望ましいのかをこれからの情報教育の展望と合わせて考察する。

第2章

プログラミングに対する意識調査

2.1 概要

この章ではプログラミング初学者が実際にプログラミングをどう捉えているかについて調査し、問題解決に何が必要かを考える。調査はアンケートを実施し、プログラミングをどのレベルまで理解しているか、またプログラミングに対しどのような感情を感じているか、作業が停滞する原因は何かを調べた。

2.2 調査結果

まず、プログラミング技術に関するアンケートについて考える。項目は以下のとおりである。これらの質問について、自身があるかを回答してもらった。回答はできる、おそらくできる、不安がある、できないの4段階とした。またこの回答を数値化し、その平均値を合わせて記載しておく。なおこのアンケートの対象はSFCにおけるプログラミングの初学者である。プログラミングの経験が半年以下の学生が多くを占める。

	回答
1, 「変数について、人に説明できるか」	2.5
2, 「代入について、人に説明できるか」	2.75
3, 「条件分岐 (if文) について、人に説明できるか」	3.375
4, 「while文について、人に説明できるか」	2.25
5, 「for文について、人に説明できるか」	1.75
6, 「プログラミングの構造は簡単だと思うか」	2.0
7, 「プログラミングは好きか」	2.0

表 1

今回、質問の形式を「理解しているか」ではなく「人に説明できるか」とした。理由としては「できる」と「おそらくできる」の差別化である。ある程度自身の技術として持っていないと人に説明できるとは選びづらく、「理解している人」と「なんとなく理解している人」を区別することができる考えた。

結果としては変数、代入に関する回答が条件分岐に関する回答より自信がないということになった。これは変数、代入に不安がある人が多いのではなく、if文なら説明できるという回答の人が多かったためである。条件文に関する処理は他の部分に比べわかりやすいと感じるようである。それと反してwhile、for文の理解は浅い生徒が多かった。条件文はプログラムの上部から流れていく動作に反する動きをしないため理解しやすいが、繰り返しの処理が見えないのが分かりにくさの原因となっているのだろう。

また構造は難しいかという質問に対し、やはり難しいと感じる生徒が多かった。上記の結果と合わせると、やはり繰り返しの処理や複数のif文の組み合わせが難しいと感じられるようである。それぞれの構文への理解と、その組み合わせによる処理が可視化できれば解決に近づけると考えられる。

次にプログラミングに対する意識についてのアンケートについて考える。こちらは具体的な事例を回答として受け取った。

- 1, 「プログラミングにおいて複雑だと感じるのはどのような箇所、場合か」
- 2, 「プログラミングにおいて、何度も躓くのはどのような箇所、場合か」
- 3, 「プログラミングにおいて、
モチベーションが下がると感じるのはどのような箇所、場合か」

表 2

/* 上記のアンケート結果に対する集計 */

2.3 プログラミングとそのモチベーション

/* アンケート結果からのモチベーションの維持に対する考察 */

第3章

ビジュアルプログラミングツール 「Visual-Poyogrammingについて」

3.1 概要

本ツール「Visual-Poyogramming」（以下VP）は、マウス操作によるプログラミングを可能にし、初学者にアルゴリズム構築に基礎を学んでもらうことを目的として作成している。

3.2 使用方法

/* 図を利用した使用方法 */

3.3 フィードバック

/* フィードバックと、それによる改善点 */

第4章

今後の課題・考察

/* 今回の実装からの改善点 */