

プログラミング教育に関する見える化の提案と評価

三宅 新二† 劉 渤江†

岡山理科大学 情報理工学部 情報理工学科†

1. はじめに

情報教育においてプログラミングは不可欠であるが、初学者にとってはプログラムの基本的な動きを把握するのは難しい。

プログラムの理解に関しては、変数の値と関連付けて動作を理解する必要がある。また、問題がどのように手順化され、プログラムと対応付けられるのかも理解する必要がある。

問題をアルゴリズム（処理手順）と対応付け、プログラム動作を考える必要があり、初学者にはハードルが高くなっていると考えられる。

いきなり複雑なプログラムを学習する必要はなく、シンプルな問題が、どのように手順化され、各手順がどのようにプログラミングされているか、確実に理解できることが大切である。

本研究では、図1に示すように、問題を読み解き手順化する部分、変数を確認する部分を追加し、手順化された処理、各変数をプログラムと対応付けること、プログラムとプログラムの実行履歴を対応付けることによる「見える化」を提案する。これにより、学習者が自分のペースで確実に理解することを目指す。

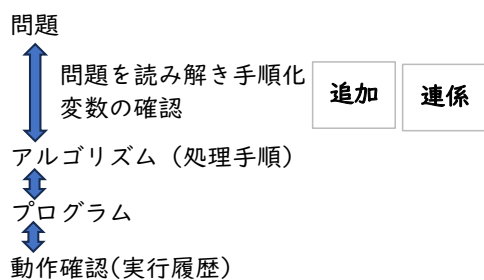


図1 見える化の概要

2. プログラミング教育における課題

与えられた問題をプログラミングする場合、以下の課題があげられる。

- どのように手順化されているか理解できない。
- 処理手順を、どのようにプログラミングしていいかわからない。
- 必要な変数と、その変数をどのようにプログラムで利用すべきか、処理手順との対応が分からない。

実際にプログラミングを教える場合でも、本研究の提案と同様に各情報を関連付けて説明している。しかし、講師のペースで説明するため、学習者が理解できる前に説明が進んでいる。また、学習者が自分で考えることなく説明を受け入れるため、動作を理解できていない場合があると考えている。

このため、問題を読み解き、処理を手順化し、プログラミングする流れを、関連付けて提示し、学習者のペースで繰り返し、反復学習できるツールを提案する。

このツールを通して、問題を手順化し、手順化した処理をプログラミングし、作成したプログラムの動作確認を行い、正しく対応できていることを理解させる。

講師により、簡単な操作説明を行い、学習者が自らのペースで動作確認を行うことで、プログラムに対する理解が深まると考える。

3. プログラミングの理解

今までの我々の研究では、プログラムと変数の値を連動させ、学習者のペースで、ステップ単位で実行状況を確認することにより理解を深めることを目指していた。[1][2]

しかし、プログラミングを行う場合は、問題を読み解き、手順化を行い、変数を検討し、プログラムを作成する必要がある。

プログラムを考える場合、プログラムの動作を変数の値と関連付けて検討することが重要である。手順と対応付け、変数の値を確認しながらプログラム動作を考えることが、プログラミングにとって不可欠である。

問題を手順化し、それぞれの処理ステップが、どのようにプログラミングされるか、プログラ

Proposal and evaluation of visualization related to programming education

† Miyake Shinji, Liu Bojiang

Department of Information Science and Engineering,
Faculty of Information Science and Engineering,
Okayama University of Science



図2 提案ツールの動作イメージ

ムの対応する部分がどこか等, 相互に関連付けることで理解を深められると考えている。

4. 見える化の実装

問題を読み解き, 手順を整理し, 手順をプログラムするための変数を考え, 各手順, 変数から処理の流れをイメージしてプログラムを作成すること, プログラムの動きを変数の値と連動して確認することが大切である。

本研究では, 図2に示すように, 問題を読み解き手順化し, 各変数に着目できるよう連携して着色し, 動作確認できるよう配慮し, 実装を行った。

全体の制御は, 動作履歴から行う。各フレームを連動して提示できるよう全体を制御する。

学習者が自分の操作によって, 処理を確認できるよう配慮している。学習者のペースで繰り返し動作確認できることで, 理解を深められると考える。

5. 実装機能の検証

様々な問題に対するアプローチを汎用的に行えるのが理想であるが, 効果把握のためにも簡単なサンプルを作成し, 評価する。

図2は, 1 から N までの和を求める問題であるが, ユークリッドの互除法, 最大値を求める等の問題についても対応予定である。

また, プログラムへの理解を深めるために, 以下の確認を可能とする。

- ・問題を読み解き, 処理を手順化する。
- ・プログラミングに必要な変数を確認する。
- ・プログラムの動作を変数の値と関連付ける。

- ・プログラムと動作履歴を関連付ける。

講義で利用するだけでなく, 学習者アンケートを取り, 以下の観点で検証予定である。

- ・手順を考えることができるか。
- ・その手順をプログラミングできるか。
- ・プログラムを動作確認できるか。

6. おわりに

初学者が躓きやすい点を明確にし, 早い段階で理解を深めておくことが重要であると考えている。

提案ツールによる「見える化」によって, プログラム作成の一連のステップへの理解が深まることを検証予定である。学生による試行, 評価を行い, 効果を確認する。プログラムに対する確実な理解が, その後のプログラミングに対しても効果的であることも確認したい。

プログラム作成における課題解決の一助となることを目指している。

参考文献

- [1] 三宅新二, 立石佑実, 劉渤江: “学習者の気づきを促す学習支援コンテンツについて -処理手順とプログラムの連動によるアルゴリズム学習コンテンツを例に-”, 情報処理学会 第84回全国大会(2022)
- [2] 三宅新二, 北川文夫, 劉渤江: “プログラム実動作の見える化 -プログラム動作の理解に着目した初学者向け教材の開発を通して-”, 情報処理学会 第85回全国大会(2023)