

初学者の対話型学習を促す プログラミング教育支援システムの現状と今後の計画

Current Status and Future Prospects of Programming Education Support Systems to Promote Interactive Learning for Beginning Learners

若林 遥大, 川戸 聡也
Haruto WAKABAYASHI, Toshiya KAWATO
米子工業高等専門学校
National Institute of Technology, Yonago College
Email: kawato@yonago-k.ac.jp

あらまし: プログラミング教育の効果的な実施のために対話型学習に着目し、システムを独自開発している。本システムは、統合開発環境にチャット機能を統合し、プログラミングしながらの手軽なコミュニケーションを可能にする。これにより、対話型学習を支援し、主体的な学びの促進や円滑な授業進行を通じて効果的な教育を実現する。本稿では、システムの概要および現状を踏まえた今後の計画について検討する。
キーワード: プログラミング教育, 教育支援システム, 対話型学習

1. はじめに

プログラミングの技術を身に付けるためのプログラミング教育が広く行われている。本教育を効果的に実施するため、様々な手法が提案され、その実現を補助する多様なシステムが開発されている。また、教育現場では対話型学習が広く行われている。対話型学習は、ディスカッションといった学習者相互のコミュニケーションに重きを置く学習方法であり、主体的な学びの促進や円滑な授業進行に有効である。

対話型学習はプログラミング教育においても有効であると考えられる。特に初学者は、つまずくと学習意欲を失いやすく疑問を気軽に相談できる環境が必要である。対話型学習とプログラミング教育が一体となることで、理解度に差が生じることや学習意欲の低下を防ぎ、意見交換や共同作業を通じて幅広い視野を身に付けることにも繋がる。しかし、既存のシステムやサービスでは、対話型学習とプログラミング教育に必要な機能が別々であるため、切り替えやユーザ登録が必要となり学習者の負担となる。

そこで、これらの負担を軽減するため、教育支援システムを独自開発している。本システムは、コードエディタとコンパイラからなる統合開発環境にチャット機能を統合し、プログラミングしながら、チャットによる手軽な対話を実現する。本稿では、本システムの現状と今後の計画を報告する。

2. システムの概要

本システムのユーザインタフェースを図1に示す。コードエディタ、コンパイラ（コンパイルを伴わないコードの実行環境も含む）、チャットの3つの機能を同一画面に同時に表示している。各機能の表示比率や文字サイズは、利用者の環境や利用方法により変更可能である。また、画面左端のボタンからコードの実行と結果の表示、エディタの初期化などを行うように設計しており、直感的な操作が可能である。

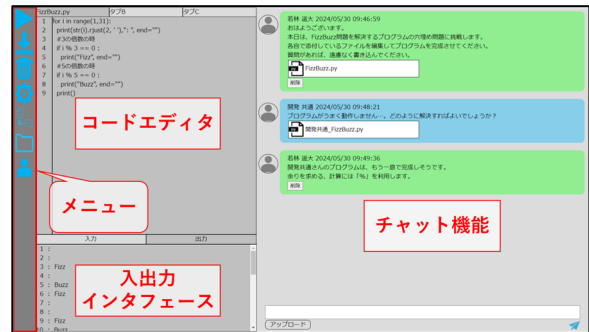


図1 ユーザインタフェース

本システムは Web アプリケーションとして構築されている。コードの実行はサーバで一貫して処理することで、利用者はコードの作成から実行までを Web ブラウザのみで完結できる。これにより、複数のアプリケーションのインストールや切り替え、ユーザ登録といった手間を省くことができる。なお、現状では C 言語と Python の実行に対応している。

本システムのチャットでは、テキストやファイルを教師と学生の間で送信および閲覧できる一般的なチャットの機能に加え、コードエディタとの連携が可能である。まず、チャットで送られたソースコードをコードエディタで直接開いて、編集および実行できる。また、コードエディタで作成したソースコードをチャットで送信できる。これらにより、コードの作成中にファイルを他者に共有しながらチャットを介した相談に活用するなど、対話型学習を支援できるようにしている。なお、個人間のチャットには現状対応できていない。

3. システムの試用

3.1 概要

本システムの利便性や有用性、改善点を洗い出すことを目的とした試用を行った。対象は、プログラ

ミングの初学者である高専2年生33名であり、所要時間は15分とした。内容はFizzBuzz問題を解くソースコードをPythonで作成するものとし、最後にアンケートを実施した。アンケートでは、多肢選択の各項目について「思う」から「思わない」まで5段階で回答を求めつつ、自由記述も設けて意見を求めた。

3.2 アンケート結果

アンケートの各項目で得られた結果を図2に示す。本システムの最大の利点である(A)必要な機能が1つに集約されている点、(B)環境構築が不要である点がそれぞれ便利であるかについて、いずれの結果も8割以上が肯定的な回答であった。自由記述では「自宅学習をする際に学校と同じ環境を簡単に準備できるのが便利」、「普段行っている面倒な手順が省略されていて、効率よく学習できる」という回答もあり、必要な環境が簡単に利用できることは、利用者の負担軽減に繋がることが確認できた。

同様に、(C)チャット機能が統合されている点、(D)各機能間でファイルのやり取りが不要である点についても、8割以上が肯定的な回答であった。自由記述では「チャットを介した情報共有が簡単にできる」、「ソースコードを素早く共有できて便利」という回答もあった。この点については、今後も各機能間の連携を拡充し、さらなる利便性の向上を目指す。

一方で、システムの現状について(E)初学者向けであるか、(F)十分な機能を有しているか、(G)実際の授業に取り入れたいかについては、6割から7割ほどが肯定的な回答であった。他の項目と比べると肯定的な回答が少ない。その要因として既存のシステムやサービスに比べると機能が不足している点や見劣りする点が挙げられる。自由記述でも「まだ機能が不十分である」という意見があった。この点は指摘を真摯に受け止め、機能の拡充を通して、見劣りしないシステムを目指す。

なお、(C)から(G)の回答については、肯定的な意見の中でも「4」（どちらかといえば思う）の割合が高くなっている。これは、試用時間が短いために回答に迷いが生じた可能性が考えられる。

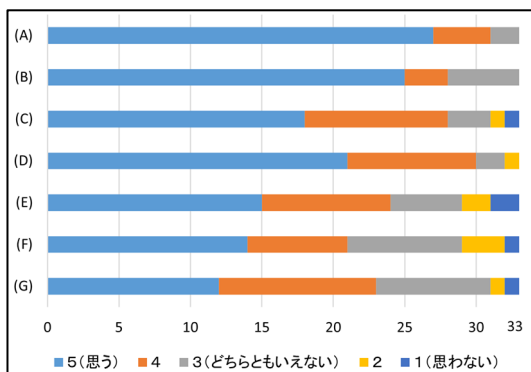


図2 アンケート結果

4. 現状の取り組みと今後の計画

アンケート結果を踏まえて改善に取り組んでおり、

特に重視している点について現状と今後を述べる。

4.1 授業支援機能の拡充

個人間のチャットが欲しいとの意見があった。学生間および教師と個別の学生とのコミュニケーションを促進する上で必要不可欠であると考えており、現状開発中である。しかしながら、教師が学生同士のチャットを監視や制限する機能について、状況によっては必要だと考えている。この点は実際に授業を行う教師から意見を得た上で検討する。

また、ソースコードの空所補充問題を作成し、解答可能なシステムの開発を計画している。コードを作成する試験などでの活用をはじめ、空所補充後にコードを実行可能とすることで、変数名や数値を細かく変更するなど、コードの動作の初歩を覚える段階の支援にも繋げたい。

加えて、現状のコードエディタは、シンタックスハイライトや予測変換機能が実装できておらず不便であるとの意見もあった。この点は改善すべきであると認識しており、実装を目指す。なお、初学者に入力補助機能のないエディタをあえて使うメリットも一定数あると考えるため、その点も考慮したい。

4.2 利用可能な言語とコンパイラの拡充

C言語とPythonのみに対応しているため、Javaなどの利用者の多い言語や、高校の共通教科情報科でも利用されているJavaScriptの実装を検討し、幅広いプログラミング言語に対応したいと考えている。

また、現状のコンパイラは、サーバの能力などを考慮してコードの実行前に入力を登録しておく必要があり、不便であるという意見もある。入力が対話的なコンパイラを実現し、さらなる利便性の向上を目指す。加えて、ソースコードのエラー個所を表示する機能の追加及び日本語化についても検討する。

4.3 再試用

機能の修正や拡充が一段落した際には、改めて試用する。今回は、15分程度しか時間を確保できなかった。その結果、一部の機能について詳しく紹介や試用ができず、正しく評価できない面もあった。今回は試用時間を拡大し、より適切な評価ができるように手法を再度検討したい。また、対象者についても人数や属性を検討して広く試用を行いたい。

5. おわりに

開発中のプログラミング教育支援システムについて報告した。問題点の解決や機能の追加を通して、利便性や有用性を向上していく。そして、本システムの利用により、プログラミング教育の場で広く対話型学習が取り入れられることを目指す。

参考文献

- (1) 若林遥大, 岩尾朋哉, 川戸聡也: “双方向コミュニケーションを重視した初学者向けプログラミング教育支援システムの開発”, 情報処理学会第85回全国大会講演論文集, pp.877-878 (2023)