

プログラムを言葉で表現する協調的活動

- 基本アルゴリズムの学習場面を例に -

土屋孝文*1・松井浩紀・江口準啓・大場智之・長江悠太

Email: tsuchiya@sist.chukyo-u.ac.jp

*1: 中京大学工学部

◎Key Words アルゴリズム、プログラミング、学習支援

1. はじめに - 学習支援環境の概要

情報系基礎科目「アルゴリズムとデータ構造」の学習には3種の領域知識、すなわち、対象問題のアルゴリズムの理解、アルゴリズムに対応するコーディング（プログラム生成）、プログラムの実行過程とアルゴリズムの対応（プログラム解釈）の理解が相互に関係している。本研究は各領域について、以下の点に考慮しながら、学習者に応じた支援環境の設計と運用を検討している。

- ① 学習者の説明や仮説生成を活動の中心におく。
- ② 演習室施設を利用し、活動結果の集計/共有/比較や、学習者間コミュニケーションなど協調的な学習環境を活用する。
- ③ 活動を通して得られた知識を既有知識と関係づけながら理論的知識へと橋渡しする
- ④ 先輩が自分自身の学習経験に基づき後輩の学習支援環境設計に関わるサイクルを継続する。

表1は、基本アルゴリズムの理解とC言語によるコーディングを対象に、主に本学会の研究報告事例を参考にしながら、初学者向けに試作運用している代表的ツールである。たとえばC言語構文辞書ツールはプログラム列をハイライトすると、数行程度の関連解説をポップアップする。スタイルチェッカーはK&Rスタイルなど代表的スタイルに具体的に違反している部分を示すフィルター型ツールである。初学者向け構文エラーツールは、実際のエラー事例（エラールール）との適用から修正部をアドバイスする。

2. コーディングの支援

基本アルゴリズムの理解については、①自分の(素

朴な) 解法の内省とあわせて、学習対象の(賢い) 解法の具体的な操作事例を観察して仮説的な説明を行う学習環境の運用(表1(B)、図1)、②自己説明や感想の共有環境の運用、③自己説明のレベルや誤答の分析から、コーディングへ続く学習環境の検討を行っている。基本的なアルゴリズムに関する運用事例では、事前に知識を準備せずとも、学習者は具体的な操作事例の観察を通して、背景の操作手続きルールを自然に模倣、推論できる⁽²⁾。

一方、コーディングについては、アルゴリズムの模倣ができて、アルゴリズムの自己説明(自然言語による表現)やプログラムの生成が難しい初学者への支援が必要である。後者のプログラム生成について、手続き的に重要な部分に穴のあるプログラムを完成させる課題を準備して、適切な知識運用を促すように段階的なヒント(関連知識プロンプト)を提示する支援法(表1(C))には、自分のプログラムを生成するよりも先に解を表示させてしまう行動が多くみられた。学習者自身がより能動的に問題解決に関与しながら、不十分な関連知識の確認を自然に行える環境が課題となる。

そこで、基本的な問題処理知識に対応する、小さく具体的な例題集を準備し(表1(B))、一旦これらを学習した後の学習者に、課題に応じた適切な知識(ヒント例題)の利用を促すこととした。学習者自身がいつでも例題コードに立ち戻って学習できること(学習者自身で一定の抽象的知識を得られる良質な例題であること)を考慮しながら、次元配列に対する二重ループにより値の交換を繰り返す、いわゆる基本ソートのコーディングまでに9カテゴリ、14の代表的例題プログラムを用意した⁽²⁾。

プログラム生成時の支援に対して、本稿ではプログラム生成前のプラン作成にあたるアルゴリズムの自己説明支援と説明共有環境の試作について、報告

表1 学習活動の支援レベルと初学者向けツール例

支援レベル	初学者向けツール運用事例
(A) 問題解決支援、情報収集支援	C言語構文辞書ツール、C言語スタイルチェッカー 質問コメント共有BBS
(B) 思考誘発、フィードバック	基本アルゴリズム観察・仮説テストツール 構文エラーツール、代表的例題プログラム集
(C) 知識獲得・運用支援	変数の説明と手続き列によるアルゴリズム表現ツール 段階的ヒント提示法によるコーディング支援ツール

を行う。以下では基本ソートを例に、これまでの支援の問題点を整理したあと、一定の表現形式を用いた協調活動について述べる。

3. アルゴリズムの理解と表現

基本ソートの理解では、自分の解法を対象化する活動として、5つの数を並びかえてみるページを利用し、その操作履歴を参考にしながら自分の解法を説明、共有する。続いて学習対象ソートに関する動画を観察する。

図1は対象のソート法に従った一連の操作を問うクイズページである。初期系列が動的に生成され1手ごとに正誤フィードバックがなされる。学習者は正答までクイズを繰り返したあと、手続きの説明と感想を共有する。2015年度は83.2% (全149名)のユーザから肯定的な評価を得た。



図1 観察・仮説生成用動画と仮説テストページ・選択ソート

これまでの共有表現形式には、表現を制約しないよう、フローチャートや疑似コードなどの形式的表現ではなく自然言語を用いた。しかし記述例をみると、自分なぞれる手続きであっても、手続きを適切に言語化できている説明事例は少なく、また、変数や処理内容などプログラムプランとして適当な情報を含む記述には遠い。仲間の記述例を参照、評価する活動も少ない。一方、図1については、表示画面を共有しながら、お互いの手続き案を説明しあう活動が多くみられる。このように仲間同士に自然な説明活動が起きやすい表現の提供が望ましい。

4. 仲間に向けたプログラムの表現

一定の表現形式をプロンプトに、日本語でプログラムを緩やかに表現する活動を検討した。表現形



図2 選択ソートの記述例

式は、(A) 変数とプログラム中の役割および(B) 手続きの処理のリストである。プログラミングの学習には、ループカウンタや値の交換時に一時的に使用される変数など、プログラムに特有な変数の役割(典型的な利用例)に注目する必要がある⁽¹⁾。

図2は、説明記述ページの例である。上部にはプログラムに必要な変数とプログラム中の役割を記述する。下部には手続きを順序付けて、制御文も含めて緩やかに自然言語で記述する。投稿されたページは、一覧ページから参照可能である。

基本ソートに関する運用に、63.6% (全74名)のユーザから肯定的な評価を得たが、実際に適切な記述が行われたかをみると、平均量は4行程度で、テキスト表現の難しさに関する感想が多い。一方、この活動では前年に比べ、仲間と相談したり仲間の記述例を参照したりする機会の増加がみられ、仲間の例を参考に記述を行ったとする回答は83.8%になっている。さらに教員は、記述から部分的に推論できるアルゴリズムにコメントを与えたり、複数の表現を比較・評価したり、学生の記述事例を利用することで学生に応じた働きかけができた。これらは言葉による説明表現を共有する活動の効果と考えられる。

5. おわりに

部分的にせよ自分の理解に関する一定の言語表現ができれば、そこから仲間とのコミュニケーションを通じた理解が期待できる。より豊かな初期表現を作りだすための支援の検討が課題である。

参考文献

- (1) Sorva, J.: "A roles-based approach to variable-oriented programming", Human Technology, Vol. 4 (1), pp.62-74 (2008).
- (2) 土屋孝文, 齋藤真琴, 鹿内拓哉, 原田翔一, 松井浩紀, "例題を利用するプログラミング支援環境-基本アルゴリズムの学習場面を例に-", PCカンファレンス論文集, pp.9-10 (2015).