

# プログラミング初学者にみられる例題プログラムの解釈

中京大学情報科学部 土屋孝文 小林晴純 田中昭子 西村美香  
(tsuchiya@sccs.chukyo-u.ac.jp)

## 1. はじめに

プログラミングの学習は、例題と解説、それらを基に解決可能と期待される複数の課題を1セットとし、このセットを繰り返しながら段階的に進められる場合が多い。そこには「例題や解説から得られた新知識を利用し、例題とほぼ同じ解法に従ってプログラミングすれば、少々遠回りするだけで解が得られるはずの課題」に、多様な躓きや行き詰まりを示す初学者たちも存在する。本稿は協調学習環境の運用事例(大学2年次31名)を基に、このような初学者の知識獲得や問題解決過程に関する探索的な検討を行う。対象は、再帰的プログラムの基礎(言語はProlog)である。

## 2. 例題からの知識利用と課題の問題解決過程

初学者のプログラム理解を扱った認知的研究(たとえばMayer(1989))は、理解に必要な知識を、プログラミング言語に特有な構文的知識、変数や制御構造などの概念的知識、プログラム実行(実行戦略)に関する知識、プログラムの実行環境に関するシステム知識に分類する。学習者は、段階的に与えられる例題と解説を通して、これらの知識を次第に豊かに獲得していくと期待される。

一方、課題プログラムを作成するには、これらの知識のほかに、プログラムを構成していくための方略、いわば「プログラムの作り方」に関する発見的な知識が必要となる。しかし、例題は、そのプログラムが、どのように作られたのかを直接には表現していない。代表的な教科書も、例題とその実行過程を示しているだけで、プログラミングの方針や、具体的な生成手続きについて述べているものは少ない。例題が解と解を導くための解法手続きを示している学習場面(たとえば初等数学)と比較すると、プログラミングの学習場面の特徴の一つは、課題を与えられた学習者自身が、効果的な解法方略を自分の問題解決経験から学習しなければならない状況にあることと考えられる。我々も、この点に考慮しながら指導を行っているが、現状の指導では、多くの初学者から「例題プログラムが正しく解を出力することはわかるが、正しく解を出力するプログラムを書くにはどうしたらよいかわからない」という声が聞かれ、多様なパフォーマンスがみられる。

Pirolliら(1994)は、このような学習場面におけるLispの初学者モデルをプロダクションシステム(ACT-R)によって構成した。彼らは初学者によるプログラム生成方略にトップダウンなプランニングを仮定し、例題プログラムの表層構造を利用した適切な構造類推がプランニングをガイドするとしている。このモデルでは、解決経験で利用された手続き的知識の効率化によって、解決方略の学習を説明

する。また、初学者の課題解決パフォーマンスにみられる多様性を、主に課題解決に関係する手続き的知識の準備状態の差異によるものと捉え、差異をもたらす学習者の例題解釈方略を分析している。

学習の支援には、このようにプログラミングを知識の運用や問題解決過程ととらえた上で、実際の学習者の課題解決にみられる学習上の問題点を抽出して、その対応を検討することが重要と考えられる。その一つは、課題出題者が仮定、期待している課題問題解決(理想的な学習者モデル)と初学者による実際の問題解決との差異を比較していくことである。だがPirolliらのモデルは、我々が関わる学習場面をみるには、解決方略知識の仮定や期待が高く、学習者がなぜそのようにプログラムを書き止まってしまうのか、書けないのか、その実際や理由を簡単には説明できない。我々は、まず、例題から利用できたはずの知識の見過ぎの様子や、時には自力で解消できない失敗に陥りがちな不安定で弱い問題解決の実際などを、我々自身の仮定や期待と比較しながら、明らかにしていく必要があると考えている。

そこで、本稿では協調学習環境の運用記録にみられる学習場面の実際に関する検討(反省)を報告する。以下では、まずプログラミングの学習を対象にした協調的学習環境の設計と実践を述べる。次に再帰的プログラムの代表的課題を用いた運用事例より、課題解決に対する我々の期待と比較しながら、初学者にみられた知識獲得と問題解決過程について探索的に検討する。

## 3. Prologプログラミング協調学習環境

土屋ほか(2004)は、Prolog言語に関する様々な学習資源の利用支援を目的に設計された、ネット上の学習環境を報告した。そこでは、学習者が振り返って例題中の知識を確認できるよう、例題プログラム中の各構文要素に、学習者集団に依存した小さな解説が関係付けされている。また、仲間同士の情報交換による問題解決支援に注目し、課題解決に成功した学習者に課題解決ヒントの提供を求め、ヒント集の共有を行っている。今年度の実践では、学習者に電子メモを与え、例題の解説や課題解決過程について重要と思われる情報の記録を推奨した。

解説講義では、例題プログラムの解釈や実行過程に加え、再帰的Prologプログラムに関する手続き的な構成法を教授し、例題の具体的な生成過程を示した。この構成法は、大まかに、(A)問題を境界条件と再帰条件に分割するなど全体の問題構造をとらえる部分、(B)それぞれの部分にあたる具体的な内容を課題に応じて発見的に探索する部分、(C)その手続きを具体的なプログラムコードに表現する部分からなる。

#### 4. 運用結果にみられる実際の問題解決

課題解決に利用される知識のタイプを、要素知識（例：構文知識；リスト記法、概念的知識；リスト構造、再帰）、実行過程に関するもの（例：呼び出し制御の実際、実行中におこる解の合成）、構成法中の3部分に関するもの（構造化、問題分析、オペレーション）に分類した。Prologでは、共有変数の単一化を手続き的にとらえる(C)のタイプの知識獲得に難しさが予想されるが、(A)や(B)については、他の言語を利用しても同様の難しさが予想される。また、例題からの完全な知識獲得と構成方略の使用を仮定すれば学習者が課題を遠回りでも比較的容易に解けると期待できるなら、その課題が学習者に求める解決は主に(B)の発見と考えられる。

実践では、各例題の説明に続き、プログラムが未完成のままでも数日後に課題の提出を求めた。次回には、課題を完成できない参加者それぞれに指導を行った。本稿では、member/2（リストの要素の判定）を例題とする課題last/2（リストの最後尾要素の判定）、append/3（2つのリストの結合）を例題とする課題del/3（リスト要素の削除）に関する実践について報告する。

協調環境での活動を自由とした上に、課題完成には指導の効果も大きく、今回の実践では、メモとヒント、未完成プログラムや完成後の内省記述に未提出や未記入の場合もある。また、この運用記録は個人の問題解決過程の全ステップをとらえているわけではない。それでも、この記録には、課題の内容と学習状況に依存して各個人が重要とみなした知識や、それぞれの問題解決エピソードが間接的に反映されていると考えられる。表1は、参加者31人について、課題の正答率と、参加者が自分のプログラミング過程に関して行った内省的記述にあらわれる知識の分類結果である。del/3は、予想していたほど容易に解かれる課題ではなかった。

##### [我々の予想と学習者の比較]

課題last/2の難しさに関する我々の期待は、リスト要素が一つの場合を境界条件とする問題分析や、それを表現するオペレーション（コード）の発見だった。課題del/3では、解決に必要なオペレーション知識が例題append/3の中心的な解説内容のため、主にdel/3そのものの問題分析が期待された。

表1をみると、実際には両課題とも、これらの知識のほかに、既にそれまでの例題で確認したはずの知識が振り返られていることがわかる。実践場面でも、例題から得られるはずの新知識を、課題時の指導から獲得している様子がみられた。例題の説明は期待されていたほどの効果をもたないまま、学習者

は不十分な準備で課題解決にとりかかってしまっており、課題が例題からの知識獲得の機会となっていたとみられる。del/3の場合、自分たちだけでは、まだ例題の新知識をうまく解釈できなかつたり、問題分析中に含まれる解の発見的な探索に失敗したりしていたと考えられる。この課題を「例題の新知識だけに注目できればできる（仲間で協力すればできる）」と考えていた我々の仮定は強すぎ、もう少し適切な準備が必要な状況だった。

##### [問題解決過程の様子]

課題last/2では、リストに関する要素知識が不十分なまま例題member/2について表層的な構造類推を行ったと考えられる、構文的に誤った未完成プログラムがみられた（未完成中の26%）。課題del/3の記述にも、例題append/3からの構造類推を行っていることと解釈できる事例が多くみられる。表1中の実行過程知識には、解の正答性を実行によって確認している記述が含まれている。課題プログラムは2行程程度の記述量しかないため、たとえ類推がかなり表面的なものであっても、いくつもの候補を生成し、実際に実行してテストを行えば、やがて正答に至ると予想されるが、実際の正答率は低い。ここから学習者の多くは、教示した方略よりもむしろ、プログラムパターンをボトムアップに生成、実行、修正する方略に基づいており、生成方略には主に構造類推が利用されているが生成や修正は予想に比べて弱いと考えられる。

#### 5. 今後の課題

これらの検討を通して、例題導入時や振り返り時に、それ以前の例題や課題と異なる新知識を強調、確認する学習支援が必要とみられる。また、実際の学習者の弱い問題解決過程を考慮し、問題解決過程を修正、ガイドしていく知識の提供を検討したい。経験的になされる例題と課題の組み合わせや例題間の構成についても、学習者の実際に応じた再構成を行う必要がある。

#### 文献

- Mayer, R. E. (1989) Learning a programming language. In M. J. Carroll (Ed.), *Interfacing thought*, The MIT Press, 61-79.
- Pirolli, P. & Recker, M. (1994) Learning strategies and transfer in the domain of programming, *Cognition and instruction*, 12(3), 235-275.
- 土屋孝文, 岡崎加奈, 谷志穂 (2004). Prolog プログラムの協調的学習環境の設計と運用 - 学習初期の問題解決過程について, 日本認知科学会 第 21 回大会発表論文集.

表 1 課題ごとの活動結果

	正答提出 (%)		内省記述 (人)				
	指導前	指導後	要素知識	実行過程	構造化	問題分析	オペレーション
last/2	80.7	90.3	14	7	0	8	11
del/3	25.8	100	7	6	0	9	11