

# 自己説明を促す問題解決スクリプトの試作と運用

## - 初学者のプログラミングを例に

土屋孝文\*1

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

\*1: 中京大学情報理工学部メディア工学科

◎Key Words プログラミング, 学習支援

### 1. はじめに

基本プログラムを段階的に作成していく問題解決スクリプトを仮定し、各ステップで学習者に簡単な問いかけを行い、学習者に説明や推論結果の応答を求める簡単な Web ページを作成した。ページに回答を評価せず、正答と背景知識を提示して次のステップへ進む。学習者が定型的な対話を適切に継続させられるかという観点から運用例を報告する。

### 2. プログラミングの学習の難しさ

入門的なプログラミングの授業では、例題プログラム(問題と解答例)の理解と、課題の問題解決(プログラムコード生成)が繰り返される。理解はプログラムから、その表現の裏に隠れたさまざまな意味を読み取っていく過程であり、問題解決は多様な知識を適切に運用してプログラム表現を生成する過程である。

プログラミング学習の難しさの一つに、このような場面で初学者が利用する暗黙的な学習方略があげられる。例題プログラムの理解では、プログラム中の記号列や構造の解釈に偏り、例題プログラムの生成過程や、生成を支える関連知識には注意が向かない。そこで我々は、仲間に例題プログラムを解説する協調活動の中に、プログラム説明の枠組み(説明テンプレート)と説明の具体例を提供し、多様な読みを自然に引き起こす支援を検討してきた<sup>(2)</sup>。実践では説明者が具体例の説明レベルまで、説明テンプレートの各項目を詳述し、適切な説明の誘発に一定の効果がみられた。

一方、課題のプログラム作成では、例題プログラムの表面的な表現を課題プログラムに向けて記号的に操作/変換する方略に偏り、対象問題の手続きを分析してからトップダウン的にコードの生成へ向かう規範的な問題解決方略<sup>(2)</sup>は、あまりみられない。実際、このような問題解決方略や解決過程は、例題プログラム表現からは直接読み取れない情報で、初学者には、作成方略とその具体的な適用例に関する学習素材が必要と考えられる。

本研究は、基本的な再帰的プログラムの作成を例に、規範的な問題解決方略(問題解決スクリプト)を仮定し、課題に応じた運用例を段階的に示すことを検討する。スクリプトは、問題の手続き分析フェーズとコーディング(C言語)フェーズに大別され、学習者自身の説明や推論を促すため、適当な段階で、簡単な問いかけを繰り返す形式(対話テンプレート)で進行する。以下では、学習者が利用する基本的な再帰問題構造と、この構造を用いた記述活動を報告したあと、問題ス

クリプトと対話テンプレートの作成例と運用結果を報告する。

### 3. 再帰的手続き表現の難しさ - ユークリッドの互除法

再帰的プログラムの解釈と生成を支える知識として、比較的単純な再帰的解法の適用可能な問題の分析<sup>(2)</sup>を基に、図1に示す再帰問題構造(再帰スキーマ)を作成し、教授した。

(A) 停止条件	: 解の定義
(B) 再帰条件	:
(B-1)	N 次の問題を N-1 次の問題へ還元する記述
(B-2)	N-1 次の問題の解を N 次の問題の解と関係づける記述(解の合成に関する記述)

図1 再帰スキーマ

ユークリッドの互除法を例に、再帰スキーマを利用して再帰的手続きの表現を行った実践を報告する。69名の参加者には、再帰的な階乗計算の例とその手続きに関する再帰スキーマ記述例が与えられた。あわせて、4つの例題を用い、互除法手続きの獲得を確認した。続いて、参加者は自分の手続きを停止条件と再帰条件とに整理して記述した。記述は Web ページで共有された。

表1に、記述事例が各条件のどのレベルの表現にあたるかを分類した結果を示す。全員が手続きを知っているはずだが、適切なレベルで表現できた参加者は少ない。2つの条件を判別する式(2つの自然数の剰余が0かどうか)のみで記述が終わってしまい、停止条件や再帰条件の詳細を記述していない参加者(32名、46.3%)が多く、停止条件における解の記述および再帰呼び出し(還元)とその解の合成を記述できた学習者は6名(8.7%)だった。

表1 再帰的手続き構造の表現分類 (N=69)

		再帰条件		
		還元 解合成	還元 —	— —
停止 条件	解の 定義	6	10	5
	—	4	12	32

正しい再帰手続きを実行できていても、再帰表現の枠組みや表現例が与えられるだけでは、その手続きを

うまく表現（言語化）できない。プログラム生成の学習支援には、表現の枠組みのほかに、どのような方略によって、どのように手続きを言語化していくのかを例示する必要があるだろう。特に、再帰条件の同定や表現については工夫が求められる。

#### 4. 対話テンプレート

##### 4.1 問題解決スクリプト

図 2 に示すように、再帰的問題の手続き分析の問題解決スクリプトは、順に再帰スキーマを具体化しようとするトップダウンな方略として、また C 言語へのコーディングの問題解決スクリプトは、関数の構造にそって記述を進める方略として構成される。この一般的スクリプトは、問題に応じて具体的なものになる。たとえば階乗のプログラム作成の手続き分析スクリプト 4 は、「N 次の階乗問題で N が具体的に 4 の場合、N-1 次はどんな問題にあたるか」を探索する。

- |  |
|--|
| <p>(A) 手続き分析スクリプト</p> <ol style="list-style-type: none"> <li>1. 再帰の特徴（再帰呼び出し、停止/再帰の構造）</li> <li>2. 停止条件と再帰条件の一般的定義</li> <li>3. 停止条件の表現</li> <li>4. 再帰条件、還元の方法：具体例による探索</li> <li>5. 再帰条件、合成の方法：具体例による探索</li> <li>6. 再帰条件、還元と合成：一般化表現</li> <li>7. 停止条件、再帰条件の判別条件</li> </ol> <p>(B) コーディングスクリプト</p> <ol style="list-style-type: none"> <li>1. 関数テンプレート</li> <li>2. 停止条件コーディング</li> <li>3. 再帰条件コーディング</li> </ol> |
|--|

図 2 一般的問題解決スクリプト

##### 4.2 対話テンプレート

教授場面における基本的な会話形式には、一般的に (I)教師やチューターの問いかけ、(II)生徒の応答、(III) 応答に対する評価という 3 つの発話単位からなる連鎖が分析されている (IRE 連鎖<sup>2)</sup>)。教師からみれば、正答を返答できない生徒に問いかけて、その応答への評価を効果的にフィードバックし、次の話題へ移行する前に必要な補習をする状況として特徴づけられる。

一方、生徒は、このような会話形式を、誤っているかもしれない自己説明を行い、次に自分自身で説明を修正するのに重要な情報が与えられる状況として認識できる。このような立場から本研究は、問いかけへの正答と関連知識の解説を定型的に提示するだけの単純な対話テンプレートが学習の支援になるかを検討する。ITS 研究が目標とする学習者の応答に対する適応的なフィードバックは行っていない。

対話テンプレートは、問題解決スクリプトに沿った各ステップに対応する。階乗の問題には、手続き分析に 7 ターン、コーディングに 3 ターンの対話テンプレートを構成し、Web ページを実装した。図 3 に階乗の問題に関する手続き分析スクリプト 5 の問いかけと、正答と解説からスクリプト 6 へ向かう対話テンプレートの例を示す。

【N-1 の場合の解と、N の場合の解を結びつける方法がわからないときは具体例を】 4 の階乗を、3 の階乗結果を使って述べなさい。

(ユーザ応答タイプ B 事例：3 の階乗の計算式は  $3 \cdot 2 \cdot 1 = 6$  これに 4 をかけるだけ)

4 の階乗は、3 の階乗の計算結果に 4 をかけたものである。5 の階乗や 6 の階乗も同じようにとらえられる。

【一般化しよう】 i の階乗を、i-1 の階乗の計算結果を使って一般的に述べなさい

図 3 対話テンプレート例

#### 5. 運用結果

階乗の手続き分析スクリプトは、21 名の学習者によって利用され、7 つの対話テンプレート実行に平均約 20 分を要した。

対話テンプレート上の応答の様子をとらえるために、147 の応答内容を (A) ほぼ正解、(B) 詳述を要するもの（不十分なもの、部分的な誤りを含むもの）、(C) 誤っているものに大別した。それぞれの応答数は 101, 17, 29 であった。タイプ C を、応答を修正しなければならない場合と考えると、82.0% の応答では、学習者が自分の推論によって対話テンプレートに適切に応答し、対話を継続させていたと考えられる。タイプ B やタイプ C の応答から解説提示の後にタイプ A で応答できた連鎖数と、タイプ B やタイプ C が連続した連鎖数は、それぞれ 23 と 15 だった。タイプ A への修復 (60.5%) は学習者が対話テンプレート中の正答提示や関連知識の解説を利用し、自分の説明を修正した結果と考えられる。

一方、7 つのステップから応答をみると、表 1 にも見られたように、スクリプト 6,7 にあたる再帰条件の一般的な表現について他のスクリプト項目に比べて誤答数が多い（どちらも 8）。また、タイプ A からタイプ C へ移行する連鎖数も 16 (10.9%) みられており、知識運用が不安定な様子もわかる。

#### 6. おわりに

対話テンプレートを用いたプログラミング作成支援の実践を報告した。今回の実践では、問題解決方略が内化され、他のプログラミング課題で利用されるようになるかどうかの検証を行っておらず、今後の課題である。

#### 参考文献

- (1) 土屋孝文, 各務弘基, 塚本久美, 栗山准佳: “仲間に向けた例題解説を作成することからの学習”, 2010PC カンファレンス論文集, pp.20-21 (2010).
- (2) 神宮英夫: “スキルの認知心理学”, pp.163-168, 川島書店 (1993).
- (3) Roberts, E. R. : “Thinking Recursively”, John Wiley & Sons (1986). (邦訳 有川ら訳 “再帰的思考法”, pp. 9-10, オーム社 (1993)).
- (4) Mehan, H. : “The Structure of classroom discourse”, In T.A. van Dijk (ed.) Handbook of discourse analysis. Vol. 3, pp. 119-131, London: Academic Press (1985).