

図の外化によるプログラム内容理解における学習者評価

Learner Evaluation in Program Comprehension by Diagram Externalization

疋田 将之^{*1}, 仲林 清^{*2}

Masashi HIKITA^{*1}, Kiyoshi NAKABAYASHI^{*2}

^{*1}千葉工業大学大学院

^{*1}Graduate School of Chiba Institute of Technology

^{*2}千葉工業大学

^{*2}University of Chiba Institute of Technology

Email: s1432126ah@s.chibakoudai.jp

あらまし：プログラミングの理解力向上を目的として学習者がプログラム全体の流れやイメージを図として外化する方法を考えた。使用したプログラム課題は雑誌の購読サイトをモデルしたものであり、記事によって閲覧できるかどうかを判断するものである。いきなり図を描くことは困難なことから構文の意味や役割、必要性和クラス間の関係、繋がりを意識させる支援課題を作成した。図を描く実験群と描かない統制群の合計8人で比較を行った。両群の理解度に大きな違いはなかった。実験群は、図を外化できた学習者とできなかった学習者に分かれ、外化できた学習者には理解度の向上が見られた。

キーワード：プログラミング, 外化, 図, 理解力向上, 学習手法

1. はじめに

プログラミング学習をする上で重要なのは、その内容を理解することである。しかし、学習者の中にはプログラムの内容を理解したつもりとなっていることがある。この原因として構文1つ1つを単体でしか理解できていなく、全体の流れを把握できていないこと、それによって構文の役割をプログラム全体の動作の流れと結び付け考えられていないことが挙げられる。

これを解消する方法として外化に着目した。外化とは認知プロセスの途中で生み出される処理結果を何らかの形で表すことを示す⁽¹⁾。外化を行うことによって自身が考えていることが明確になり、問題を解いていく上でヒントとなる。本研究では、学習者にプログラム全体の動作の流れを図として外化してもらうことで内容理解の向上を図る。

2. 学習目標

本研究では、オブジェクト指向プログラミング(OOP)のjavaを学習対象とする。OOPのプログラムを理解するためには、個々の構文だけではなく、クラス間の関係性を理解することが必要となる。OOPの理解に関して表1のようなプログラム内容理解レベルを設定した。レベル1は、構文1つ1つの意味や役割、必要性をプログラム全体のイメージや動作の流れと結び付けて考えることができない状態であり、レベル4は、構文の意味や役割、必要性和クラス間のやり取りと繋がりが、プログラム全体のイメージや動作の流れと結び付けて考えることができ、実行結果がある程度予測できている状態である。本研究では、学習者の理解がレベル4に到達することを目標とする。

表1 プログラム内容理解レベル

レベル1	一つ一つの構文が何をしているかが分かる
レベル2	プログラム全体から見た構文の意味、役割、必要性が分かる
レベル3	クラス間、構文同士の繋がりが、関係性を理解し、正しく繋げることができる
レベル4	全体としての構文の集合体を正しく繋げて動かすことができる

3. 学習手法

3.1 内容理解レベルを上げる手法

本研究では2章で述べた表1のレベルに沿って、学習者の理解レベルを向上させることを目標とする。そのために以下3つの手法を適用する。

1つ目は学習者が考えているプログラムの全体のイメージや動作の流れを図として外化してもらうことである。外化は2回行うことを想定している。外化に際しては、特に指示を出さず、学習者のプログラムのイメージ図を描かせる。2つ目は理解レベルを上げる支援課題である。レベルごとに段階を踏んで出題し、学習者が2回目のときに図として外化できる状態にするための支援を行う。内容は構文の意味や役割、クラス間の関係を問う問題である。3つ目は学習者の理解度を確認する理解度チェック問題である。内容は実行結果の予測やソースコードの一部を改変したプログラムの実行結果の予測、クラス間でやり取りされているインスタンスや変数の中身を問う問題となっている。

学習手法を行う手順としては最初にソースコードを配布し、1回目の外化を行う。その後、支援課題でプログラムの内容を意識させ、2回目の外化を行う。最後に理解度チェック問題を行い学習者の内容理解

度を確認する。

3.2 問題の分類

理解度チェック問題の各問について表 1 のレベルと比較し、2 種類の問題を作成した。

1 つ目は、表 1 のレベル 2~レベル 3 相当の問題で、部分的なプログラムの動作の流れが分かっているならば答えが予測できる問題である。これを分類 1 とする。2 つ目は表 1 のレベル 4 相当の問題で、プログラムの動作の流れや内容を理解できていないと解くのが難しい問題である。これを分類 2 とする。

3.3 評価基準

2 回目以外化した図と、理解度チェック問題の解答を評価した。

図の評価基準は以下の 2 点である。

(1) メインクラスを中心として見たときの動作の流れやロジックが正しいかどうか

(2) メインクラスとサブクラスのやり取りや繋がりが正しいかどうか

理解度チェック問題は分類 1 と分類 2 の正答数を比較した。

4. 実験の構成

図を描く実験群と描かない統制群で比較実験を行った。統制群の学習を行う手順は図を外化することを除いたものとなる。事前問題を作成し、実験群と統制群の能力が均等になるようにした。対象者は情報系大学の学部 4 年生 8 人である。

5. 実験結果

実験を行った結果、実験群は図として外化できた学習者 2 名とできなかった学習者 2 名に分かれ、正答数に大きな差が出た。実験群の理解度チェック問題の結果を表 2 に、学習者 C、D が外化した図を図 1、図 2 に示す。

C、D は図を外化できた学習者である。C は分類 1 の問題と比べて分類 2 の正答数が多かった。これは C の外化した図がプログラムのメインとなるロジックを中心とした内容であり、このような図を描くことによって理解度が向上し、レベル 4 に到達したのではないかと考える。

一方、D は分類 1 の正答数が多かった。これは D の図が C と比べてロジックに関わる内容が描かれていなかったため、理解度が向上せず、レベル 4 に到達しなかったと考える。

図を外化できなかった A、B はアンケート結果から C、D との大きな違いとしてクラス間や構文同士などのオブジェクト間の関係を意識していなかったことが分かった。

A は図を外化するときプログラムの大まかな流れしか意識できていないことが分かった。これは支援課題で意図したことが意識できず、プログラムに対するイメージがぼやけたままだったのではないかと考えられる。

一方、B は図を外化するとき何も意識していなかった。これはプログラムのイメージが持っておらず、支援課題で意図したことを意識しようがなかったのではないかと考えられる。

表 2 実験群の理解度チェック問題の正答数

	実験群			
	A	B	C	D
学習者				
分類 1	1	1	5	6
分類 2	1	1	6	1
合計	2	2	11	7

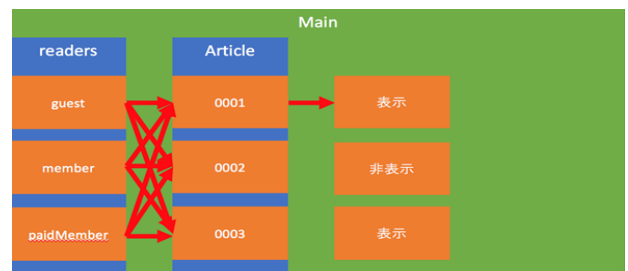


図 1 学習者 C が外化した図の例

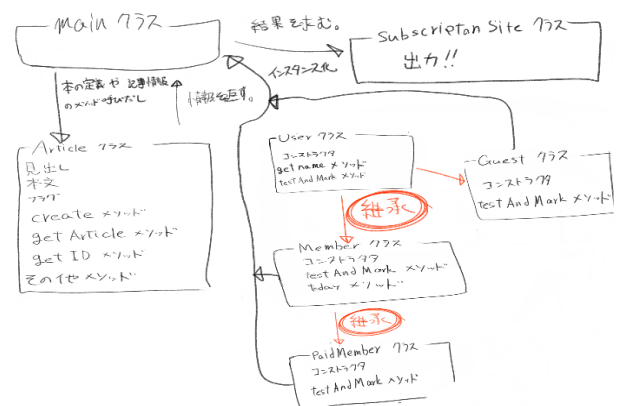


図 2 学習者 D が外化した図の例

6. 今後の課題

今後の課題として、図を外化できなかった学習者ができたこと、図を外化できたが本研究で想定していたレベルまで理解度の向上が見られなかった学習者が出たことからこれらのことについて解決策を考えていく。

参考文献

- (1) 鈴木宏明 教養としての認知科学 東京大学出版会 (2009)
- (2) 宮田仁: “状況論理的アプローチによるプログラミングの指導と認知的徒弟制理論を取り入れたアプローチ”, 教育情報研究, 15 巻, 1 号, P21-31(1999)
- (3) 独立行政法人情報処理推進機構: “問題冊子・配点割合・解答例・採点講評(2014 平成 26 年)”
https://www.jitec.ipa.go.jp/1_sukiru/mondai_kaitou_2014h26.html#26aki