

# プログラムの理解状況に応じた図の外化による学習手法

疋田 将之<sup>\*1</sup>, 仲林 清<sup>\*2</sup>

<sup>\*1</sup> 千葉工業大学大学院, <sup>\*2</sup> 千葉工業大学

## Learning Method by Externalization Using Diagrams to Depending on the Understanding of the Program

Masashi Hikita<sup>\*1</sup>, Kiyoshi Nakabayashi<sup>\*2</sup>

<sup>\*1</sup> Graduate School of Chiba Institute of Technology, <sup>\*2</sup> Chiba Institute of Technology

プログラムの理解度向上を目的としてプログラムのイメージや動作の流れを図として外化させる学習手法を検討した。使用したプログラム課題は雑誌の購読サイトをモデルとしたプログラムである。学習者の理解状況を踏まえ、アクティビティ図・シーケンス図などをもとに考案した複数種類の図を使い分けて提示し、外化の支援を行った。外化する前と後で学習者のプログラムの理解度を調査・分析した。実験の結果、大きな理解度の変化は見られなかったが、一部に理解度の向上が見られた学習者がいた。

キーワード: 図, 外化, プログラミング, 学習手法, 理解度向上

### 1. はじめに

プログラミング学習者にとって重要なことは、プログラム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理解が不十分な学習者もいる。この原因として学習者自身がデータ構造をイメージできていないこと、構文を1つ1つ単体でしか把握しておらず、全体の流れが不明瞭になっていること、ソースコードの処理とプログラムの機能を結びつけて考えられないことが挙げられる。

これら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<sup>(1)</sup>。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、問題を解いていくうえでのヒントとなる。また、頭の中での考えを何らかの形で表すことにより、思考に割けるリソースが増え、物事を考えやすくなる。

そこで、本研究では外化を利用し、学習者にプログラムの内容を図として外化してもらうことによってプログラムの内容理解度の向上を図る。

前回の実験結果<sup>(2)</sup>では、図として外化できなかった学習者がでたことや、図の外化の内容を学習者に任せ

てしまったことから効果が薄かった。そこで、今回の報告では、学習者の理解状況を調査し、アクティビティ図やシーケンス図などをもとに考案した複数種類の図を使い分け、学習者の理解状況に応じて提示することにより、外化の支援を行った。また、理解状況を踏まえることにより、その学習者に適した図を外化してもらうことによって更なる理解度の向上を狙った。以下、第2章では本研究の目的について述べる。第3章では学習目標について述べ、第4章では学習目標を達成するための学習手法について述べる。第5章では実験結果を述べ、第6章では分析と考察を行い、第7章ではまとめと今後の課題について述べる。

### 2. 前回の結果と今回の目的

#### 2.1 前回の学習目標と学習手法

前回の実験<sup>(2)</sup>では、オブジェクト指向プログラミング(OOP)のjavaを学習対象とした。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、学習者の理解がレベル4に到達することを目標とした。

表1のレベルを上げるために以下の3つの手法を適応することで内容理解度の向上を図った。

1 つ目は学習者が考えているプログラム全体のイメ

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。この時、外化する図に関しては特に指示を出さず、学習者が持っているイメージをそのまま描かせた。

2 つ目は理解度を向上させ、図として外化できる状態にするための支援課題である。レベルごとに段階を踏んで構文の意味や役割、必要性やクラス間の関係性を意識させるような問題を出題することでイメージ形成の補助を行い、学習者が図として外化できるレベル3の状態まで支援する目的で行った。

3 つ目は理解度チェック問題である。実行結果の予測や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を出し、学習者の理解度を計った。

表 1 プログラム内容理解レベル

レベル1	1つ1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、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

## 2.2 前回の実験結果と考察、課題

図を描く実験群と描かない統制群の合計8人で比較を行った結果、両群の理解度に大きな違いはなかった。実験群は図として外化できた学習者とできなかった学習者に分かれ、図として外化学習者の中でも理解度の向上が見られた学習者とあまり見られなかった学習者に分かれた。図として外化できなかった学習者はプログラムを外枠の流れのみでとらえていたことや、プログラムの内容について全く意識できていなかったことから、支援課題によるプログラムのイメージ形成補助がうまく機能せず、図として外化できるほど明確ではなかったと考えられる。一方、図として外化でき、内容理解の向上が見られた学習者は表1のレベル4の正答数が多く、図の内容がプログラムのメインとなるロジックを中心とした図であった。理解度の向上が見られなかった学習者は表1のレベル2~3の問題の正答数が多く、図の内容はプログラムの全体のやり取りや繋がりを表す図であり、プログラムのロジックの部分や細かいやり取りは描かれていなかった。このことから学習者によって理解度の向上に適した図と適さない図があることが想定される。また、学習者の理解度の

状況を考慮せずに行ったため、理解状況に応じて効果的な図の外化と、効果が薄い図があるのではないかと考えた。そのため、学習者と学習者の理解状況に応じて適切な図を外化させる必要があると考える。

## 2.3 今回の目的

前回の実験結果を踏まえて、本研究では、プログラムの内容理解が不十分な学習者を対象として、外化してもらう図を複数種類考案し、学習者の理解状況に応じて提示することにより、内容理解度を向上させることを目的とする。本研究に使用した言語はオブジェクト指向プログラミング(OOP)のjavaである。学習者の理解状況を調査するために、プログラムに関するスキル<sup>(3)(4)</sup>とプログラムの構造の観点をもとにプログラムの内容理解の過程を考案した。また、外化してもらう図はアクティビティ図やシーケンス図などもとに作成した。プログラムの内容理解の過程については第3章で説明し、外化してもらう図については第4章で説明する。

## 3. 学習目標

### 3.1 方針

前回の実験結果を踏まえ、本研究では、学習者の理解状況に応じて、外化してもらう図を複数種類使い分けることにより、内容用理解度の向上を図る。そのため、学習者の理解状況を調査するための、理解過程のモデルを考案する。

### 3.2 理解過程のモデルを構成する要素

理解過程のモデルを考案するにあたってプログラムの内容を構成する要素が必要であると考えた。そこでプログラムに関するスキル<sup>(3)(4)</sup>に着目した。プログラムに関するスキル<sup>(3)(4)</sup>とはプログラミングを行う上で必要となる能力のことである。たとえば、コードを書くスキルであれば、機能の説明をもとに、その機能の処理を実行するためのコードを書くことができるスキルである。プログラムを説明するスキルは、与えられたプログラムのソースコードを見て、その機能について説明できるスキルである。トレーススキルであれば、プログラムの動作に従って、変数などの値を正確に把握できるスキルである。このようなスキルの

中から、プログラムを説明するスキルとトレーススキルに着目した。

また、理解過程のモデルを構成する要素としてクラスの構造を理解するスキルも必要だと考えた。トレーススキルとプログラムを説明するスキルのみの場合、プログラムのデータ構造や処理構造の要素がなく、プログラムの内容を構成する要素として不十分だと考えたからである。

プログラムに関する2つのスキルとプログラムの構造理解のスキルをそれぞれのスキルを構成する要素に分解した。

### 3.2.1 プログラムを説明するスキルの要素

プログラムを説明するスキルについては与えられたプログラムのソースコードを見て、その機能について説明できるスキルである。このスキルはソースコードの処理内容を理解していること、プログラムがどのような機能を持っている理解していること、プログラムの機能とソースコードの処理内容を正しく結び付けられることの3つ要素に分解できると考えられる。

### 3.2.2 トレーススキルの要素

トレーススキルは、プログラムの動作に従って、変数などの値を正確に把握できるスキルである。このスキルはプログラムの構文が1つ1つの処理が何をやっているか理解していること、メソッドやコンストラクタなど構文を塊として捉え、どのような処理が行われているか理解していること、変数にどのような値が格納されているか理解していること、プログラムの動作によって変数のやり取りや値の変化を追えることの4つの要素に分解できると考えられる。

### 3.2.3 プログラムの構造理解のスキルの要素

プログラムの構造の理解を理解するスキルはプログラムを見て、プログラム全体の構造を把握できるスキルである。このスキルはプログラムが持っているクラス構造が理解できていること、プログラムのデータ構造が理解できていること、プログラムの制御構造が理解できていることの3つの要素に分解できると考えられる。

## 3.3 理解過程のモデル

3.2節の3つのスキルを構成する要素にもとにプログラムの理解過程をモデル化した。図1に示す。

プログラムを理解していくための前準備段階では実際にプログラムを理解するために必要な知識を理解していく過程であり、3つの項目から構成される。「プログラムの概要理解」はそのプログラムが何をしているプログラムなのかを理解することである。「クラスの繋がり」ではクラスの繋がりを理解することでプログラム全体のデータ構造を大まかにつかむことである。「クラスが持つ構造」ではそのプログラムのクラス構造を理解することである。

プログラムの内容を理解していく段階では4つの項目から構成されている。「処理を追っていく段階」では構文1つ1つの処理をクラスごとに目で追っていき、どのような処理を行っているか理解し、各クラスの制御構造を確認していく段階である。「処理を理解していく段階」ではメソッドなどの構文を集合体として捉え、どのような役割を持っているか理解し、クラス間の繋がりやプログラム全体の動作を意識していく過程である。

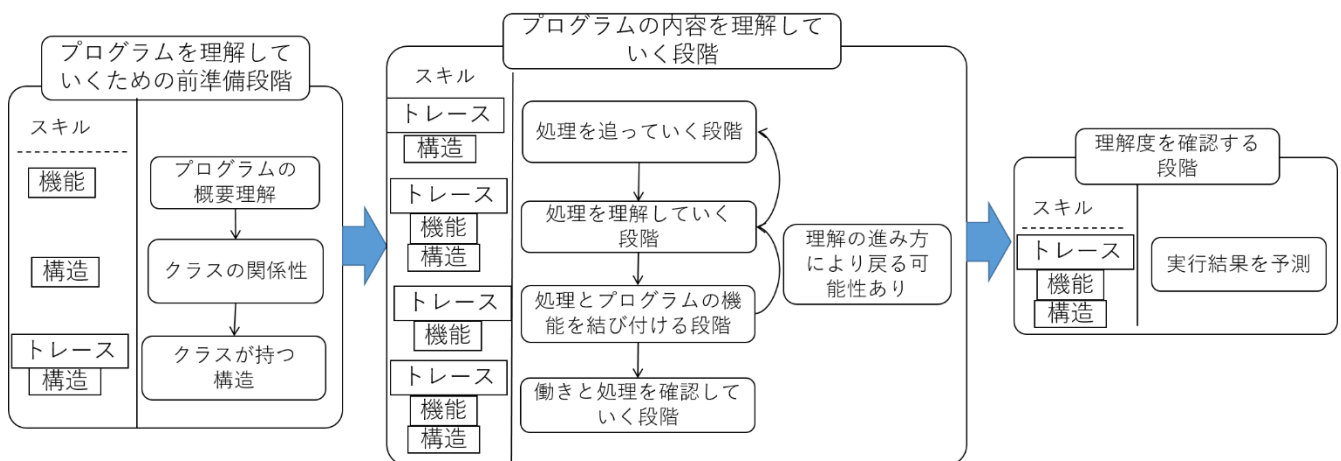


図1 理解過程のモデル

る。「処理とプログラムの機能を結び付ける段階」では、各クラスのメソッドや意味を持つ構文の集合体の機能とプログラムから見た機能を結び付けていく過程である。「働きと処理を確認していく段階」ではプログラム全体のやり取りや動作を、変数の中身の値の変化を確認しつつ、動作の流れに従って確認していく段階である。

理解度を確認する段階では最後にプログラムの動作に従って処理を理解でき、実行結果を予測できることを示す。

### 3.4 学習目標

本研究では、3.3 節の理解過程のモデルを使い、学習者の理解状況を調査し、理解度を現段階から向上させることを目標とする。理解度を調査するために、図 1 をもとに理解度調査問題を作成する。また理解度を向上させる手法として、理解状況に合った図を外化させる手法をとる。それぞれの詳しい説明は第 4 章で説明する。

## 4. 学習手法

### 4.1 学習者に提示する図

本研究では、学習者のプログラムの内容理解度を図 1 に沿って向上させるために、それぞれの理解状況に合った図を学習者に提示することで支援を行い、外化させることによって理解度の向上を狙う。そのためにプログラムの内容を理解していく段階の項目に対応した図を複数種類考案した。提示する図は問題に使用するプログラムではなく、別のプログラムを図で表したものを提示した。なお、プログラムを理解していくための前準備段階と理解度を確認する段階についてはそれぞれ学習者の初期状態と学習後の状態を想定しているため、対応する図は考案しなかった。

#### 4.1.1 処理を追っていく段階の図

処理を追っていく段階の図を図 2 に示す。

この段階では 1 つ 1 つの構文の処理が何をやっているか理解し、クラスごとの処理の流れがどのようなものになっているか確認していく状態である。そのため、アクティビティ図をもとに、各クラスの処理をメソッド単位に区切り、処理の流れに沿って書き表わした図を考案した。この図の狙いは 2 つある。1 つ目はクラスごとの処理の流れと 1 つ 1 つの構文の処理が何をや

っているか明確させることである。2 つ目は構文 1 つ 1 つの処理内容とクラスの制御構造の理解させることである。

#### 4.1.2 処理を理解していく段階の図

処理を理解していく段階の図を図 3 に示す。この段階はメソッドなど構文を 1 つの塊として意味を持つ処理の機能を理解していき、クラス間同士のやり取りやプログラム全体の動作を意識していく状態である。そのため、シーケンス図をもとに、各メソッドのやり取りと処理内容をプログラムの動作の流れに沿って書き表わした図を考案した。この図の狙いは 2 つある。1 つ目はメソッドの処理がどのような働きをしているかを明確にさせることである。2 つ目はメソッドや意味を持つ構文の塊としての機能とクラス間の繋がりを理解させることである。

#### 4.1.3 処理とプログラムの機能を結びつける段階の図

処理とプログラムの機能を結びつける段階を図 4 に示す。この段階はソースコードの処理をプログラムの機能と結びつけていく段階である。そのため、各クラスにあるメソッドと意味を持つ構文の塊の処理に対してプログラムから見た機能の説明を書き、メソッドや意味を持つ構文の塊と各クラスのやり取りを描き表わした図を考案した。この図の狙いは 2 つある。1 つ目はメソッドや意味を持つ構文の塊とプログラムから見た機能を結び付けさせることである。2 つ目は各クラスとやり取りとプログラムの機能との結びつきを明確にさせ、ソースコードの処理とプログラムの機能の関係性を理解させることである。

#### 4.1.4 働きと処理を確認していく段階の図

働きと処理を確認していく段階の図を図 5 に示す。この段階は、プログラム全体の働きや、変数の中身の値を確認しつつ、動作の流れに従って確認していく段階である。そのため、各クラスの処理の説明と、処理によって発生した変数のやり取りや中身をプログラム全体の動作の流れに沿って書き表わした図を考案した。この図の狙いは、プログラムの処理によってどのように変数の値が変化し、最終的にどのような結果になるかを意識させ、プログラム全体流れと変数のやり取りや処理によって発生する変数の中身の値の変化を理解させることである。

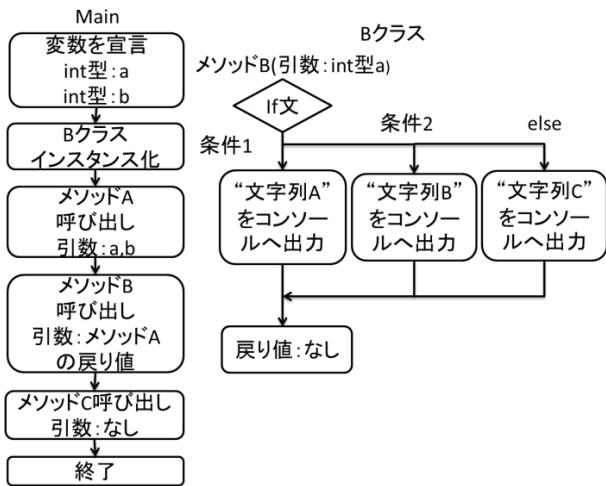


図 2 処理を追っていく段階の図

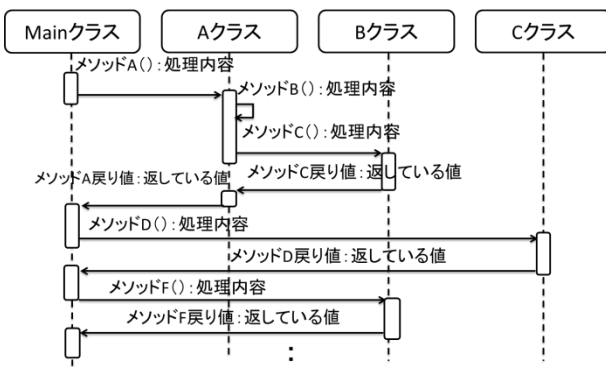


図 3 処理を理解していく段階の図

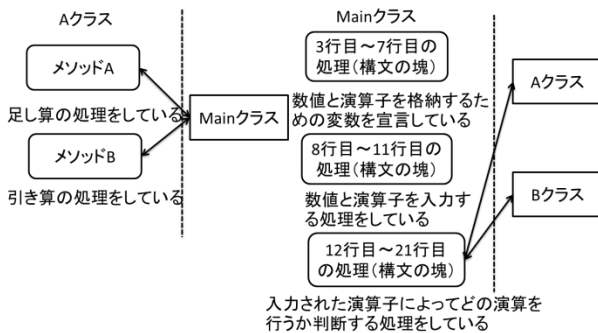


図 4 処理とプログラムの機能を結びつける段階の図

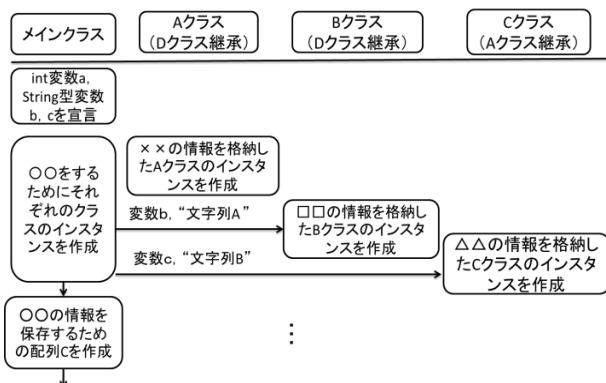


図 5 働きと処理を確認していく段階の図

## 4.2 図の提示方法

本研究では、4.1 節にある複数種類の図を学習者の理解状況に応じて提示することにより、理解度の向上を狙う。学習者に指示する方法は文章による図の内容説明と 4.1 節で取り上げた図、その図の説明文を提供する。図の内容説明ではどのような図を描いてほしいか文章による説明を提示する。これにより、どのような図を描くかイメージさせる。その後、図の例と図の説明文を参照させることによりプログラムのイメージを明確にさせ、図の外化を行ってもらう。図だけでなく、説明文を提示することで、単なる図の模倣を防ぐことを狙った。

## 4.3 学習課題

本研究で使用したプログラムは H26 基本情報技術者試験春季の java の問題である。このプログラムは雑誌の購読サイトをモデルとしたプログラムである。記事と閲覧者には種類があり、種類によって閲覧者が記事を閲覧できるかできないか判断する内容である。このプログラムをもとに学習者の理解状況を計る問題を 2 つ作成した。1 つ目は学習者の理解状況を調査する問題 1 である。問 1 から問 5 までの 5 種類あり、それぞれ理解過程のモデルの「クラスが持つ構造」から「働きと処理を確認していく段階」に対応している。

「クラスが持つ構造」の問題はクラスの継承関係やクラスのやり取りを問うものとなっている。「処理を追っていく段階」の問題は処理内容を出題し、その処理がソースコードのどこにあるのか答えるものやクラスの制御構造を問うものとなっている。「処理を理解していく段階」の問題はメソッドや意味を持つ構文の塊の処理をソースコードの中から探すものとなっている。「処理とプログラムの機能を結び付ける段階」の問題はプログラムの機能を実装している構文を答えるものとなっている。「働きと処理を理解していく段階」の問題は変数の中身を問うものとなっている。問題 1 は図として外化する前と後の 2 回行うことを想定しており、それぞれの問題の点数の差によって理解度が向上したかどうか調査する。外化する前に行ったものは問題 1 とし、外化後に行ったものを問題 1 (2 回目) とする。

2 つ目は問題 1 の類題である問題 2 である。問題 2 は外化後に回答させることを想定している。外化後の

問題 1 の点数と問題 2 の点数を見ることでより詳しく理解度を調査する。例として問題 1 の問 2 を表 2 に示す。この問題は処理を追っていく段階に対応していて、処理内容に対応するソースコードを抜き出す問題と、クラスの制御構造を問う問題となっている。構文 1 つ 1 つの処理とそのクラスの制御構造を理解して解答する必要がある。

表 2 問題 1 の問 2 の内容

番号	問題文
(1)	Article クラスの中で Article クラスをインスタンス化する処理をしている構文はどれですか？ソースコードから抜き出してください。
(2)	Main クラスの中で ARTICLES 内の指定された位置にある要素を返す処理をしている構文はどれですか？ソースコードから抜き出してください。
(3)	Article クラスと Member クラスの中にはどのような制御構造が含まれていますか？下の三つの中から該当するものを選んでください。 Article クラス 選択肢：順次構造 選択構造 反復構造 Member クラス 選択肢：順次構造 選択構造 反復構造

#### 4.4 学習の流れ

学習の流れを表 3 に示す。問題 1 の解答結果から学習者の理解状況を判断し、それぞれの理解状況にあった図を外化してもらうように指示を出す。

表 3 学習の流れと目的

学習の流れ	目的
1. 事前アンケート	プログラムに対する意識調査
2. ソースコード配布	プログラムの簡単な概要説明の資料とプログラム内で使われる API の説明も配布
3. 問題 1 配布	学習者の理解状況を調査 (全 11 問, 11 点満点)
4. 図として外化	理解度の向上を狙う
5. 問題 1 (2 回目) 配布	外化前後での理解状況の差異を調査 (全 11 問, 11 点満点)
6. 問題 2 配布	問題 1 と同じ傾向で違う問題を使うことにより、さらなる理解状況の調査する (全 6 問, 6 点満点)
7. 事後アンケート配布	図として外化したことが問題を解くのに役に立ったか、またプログラムに対する意識の変化の調査

#### 4.5 評価基準

図として外化する前の問題 1 と後の問題 1 (2 回目) の解答結果の差異と問題 2 の解答結果、学習者が外化した図を評価した。

問題 1, 問題 2 の評価は正解 (1 点), 一部正解 (0.5 点), 不正解 (0 点) の 3 段階で行った。

学習者が外化した図はこちらが指示した通りの図の意図に沿っているかどうか評価した。

### 5. 実験結果

#### 5.1 実験概要

表 3 をもとに実験を行った。対象とした学生はプログラミングの講義を受けたことのある情報系学科 3 年生 5 人である。実験期間は 2 週間であり、その中で他人と相談することや調べることはせずに各自で問題に取り組んでもらった。

#### 5.2 学習者に指示した図

図として外化前の問題 1 の点数と解答結果から学習者 C, E には処理を追っていく段階に対応した図 2, 学習者 A には処理を理解していく段階に対応した図 3, 学習者 B には処理とプログラムの機能を結び付ける段階の図 4, 学習者 D には働きと処理を確認していく段階の図 5 を描くように指示した。

#### 5.3 問題の解答結果

図として外化した前と後の問題 1 の点数結果と問題 2 の点数結果を表 4 に示す。図として外化する前と後の問題 1 の点数に大きな差はなく、図として外化する問題 1 より外化後の問題 1 (2 回目) の点数が低い学習者がいた。問題 1 (2 回目) の点数が上がった学習者は E だけであり、E は問題 1 (2 回目) の点数が全体的に上がっていた。学習者 A は問題 1 と問題 1 (2 回目) 点数に差がなく、解答にも変化が見られなかった。点数が下がっていた学習者は B, C, D であった。学習者 B は外化した図に対応した項目の間の点数は変わらず、それ以外の問の点数が下がっていた。学習者 C は外化した図に対応した問の点数は上がっていたが、それ以外の問の点数は下がっていた。学習者 D は全体的に点数が下がっていた。問題 2 の点数は問題 1 に比べて全体的に点数率が高い結果となった。

表 4 問題 1 と問題 2 の点数結果 (問題 1: 11 点満点, 問題数 11 問 問題 2: 6 点満点, 問題数 6 問)

	問題 1	問題 1 (2 回目)	問題 2
学習者 A	3.5 点	3.5 点	5 点
学習者 B	8.5 点	6 点	5 点
学習者 C	4.5 点	4 点	2.5 点
学習者 D	6 点	2.5 点	4 点
学習者 E	3 点	6 点	5 点

#### 5.4 図の結果と評価

図として外化した後の点数が上がった学習者 E と点数が下がった学習者 B, D の図の例の一部をそれぞれ図 6, 図 7, 図 8 に示す。

学習者 E の図は指示した図とは形が違うが、内容はこちらが指示した通りとなっており、各クラスの構文 1 つ 1 つの処理とメソッドごとの処理の流れを網羅したものとなっている。

学習者 B の図はこちらが指示した図の通りとなっており、プログラムに含まれるメソッドをすべて書き表したものとなっている。しかし、一部、メソッドの処理とプログラムの機能の結び付けを間違えていた部分があった。

学習者 D の図はこちらが指示した通りの図となっているがプログラムの内容が簡略されすぎて Main クラスに関する処理以外は描かれておらず、変数のやり取りや中身が書かれていなかった。

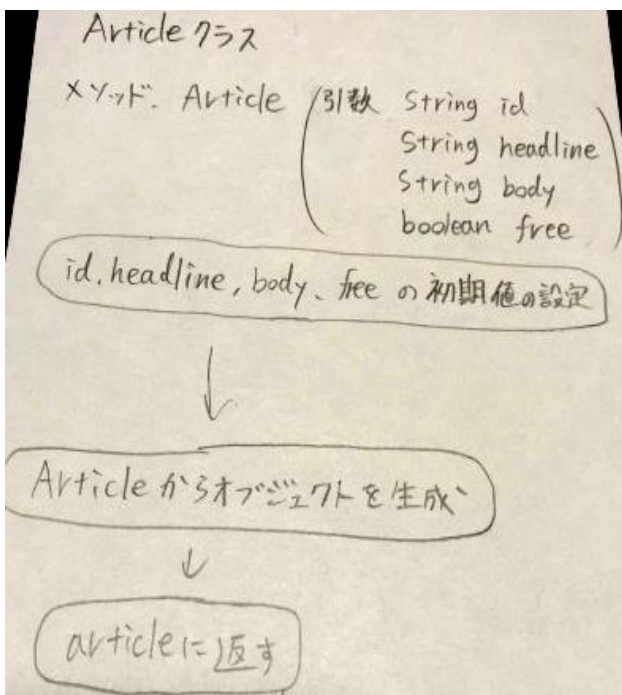


図 6 学習者 E の外化した図 (一部)

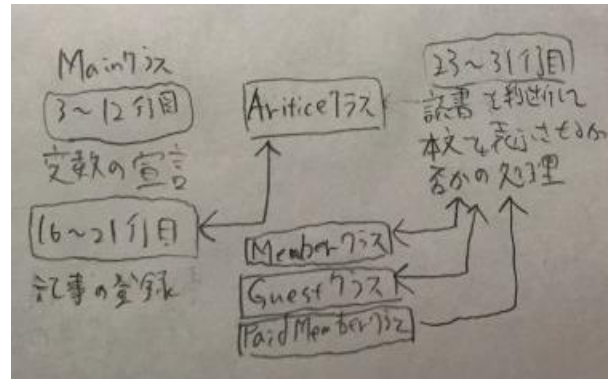


図 7 学習者 B の外化した図 (一部)

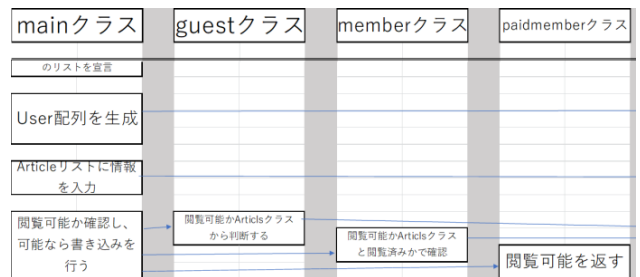


図 8 学習者 D の外化した図 (一部)

## 6. 分析と考察

本研究では、外化してもらう図を複数種類考案し、学習者の理解状況を踏まえ、使い分けて提示することにより、プログラムの内容理解度の向上を狙った。この章では実験結果を踏まえ、問題 1 と問題 2 の解答結果の分析を行い、図と問題の点数の関係の考察を行う。

### 6.1 解答結果の分析

解答結果を詳しく見ると全体的にプログラム全体から答えを探す問題の点数が低くなっていた。

例えば、問題 1 の問 3 にある「ソースコードの中で一般会員が記事閲覧できるかどうかを判断する機能を持っている部分はどこか」という問題は、プログラム全体の中から一般会員に関する処理を特定し、その処理を行っているソースコードを抜き出す問題となっている。そのため、一般会員に関するクラスに答えがあるが、ほとんどの学習者は Main クラスにあるすべての会員に共通する処理を抜き出して回答していた。

このことから、一部のクラスの処理しか理解していなかったと考えられる。また、他のプログラム全体から答えを探す問題の解答にも同じような傾向がみられたことから、ほとんどの学習者は、プログラムを全体のイメージはできておらず、一部のクラスのみイメ

ージのみしかなかったと考えられる。

## 6.2 図と問題の点数の関係の考察

外化後の問題 1 (2 回目) の点数が上がった学習者は E のみであった。

学習者 E の図はこちらが指示した図とは形式が違うが、内容はこちらが指示した通りとなっていた。このことから、提示した図をイメージ形成の補助として使い、自分が持っているイメージを図として外化できたことにより、理解度が向上したと考えられる。一方、問題 1 (2 回目) の解答の変化がなかった学習者は A のみであった。学習者 A はこちらが指示した通りの図となっているがこちらが提示した図に当てはめたような形となっていた。また、図の内容に間違いが多く見られたことから、プログラムのイメージが不明瞭な状態で図として外化したことが考えられる、そのため、理解度の向上がみられなかったと考えられる。

外化後の問題 1 (2 回目) の点数に学習者と下がっていた学習者は B, C, D である。学習者 B はこちらが指示し通りの図となっていたが、一部、メソッドの処理とプログラムの機能の結び付けを間違えていた。また、実験後のヒアリングの結果から図を描いたことで自分の間違いに気づき答えを訂正したと答えていることから、図として外化したことでプログラムの内容を誤解してしまい、問題 1 (2 回目) の点数が下がり、理解度の向上がみられなかったと考える。学習者 C も描いた図の内容がところどころ間違えており、学習者 B と同じ傾向が見られた。学習者 D はこちらが指示した通りの図となっているがプログラムの内容が簡略されすぎており、こちらが指示した図と形がほぼ同じになっていた、また、アンケート結果から問題 1 (2 回目) はソースコードを見ずに図のみで解答したと答えていることから点数が下がり、正確な理解状況を調査することが出来なかったと考える。

## 7. まとめと今後の課題

本研究ではプログラムの内容理解が不十分な学習者を対象として外化してもらった図を複数種類考案し、学習者の理解状況を踏まえ、使い分けて提示することにより、内容理解度を向上させることを目的とした。理解状況を調査する問題を作成し、図として外化する

前と後の点数の差異によって理解度の向上を調査した。実験を行った結果、理解状況を調査する問題の点数に大きな差は見られなかったが、一人理解度が向上した学習者がいた。理解度が向上した学習者 E はこちらが提示した図とは形式が違ったが、図の本質はとらえており、正しくプログラムの内容を図として外化していたことから、理解度が向上したと考えられる。一方、理解度の向上がみられなかった学習者は図として外化したことでプログラムの内容を誤解してしまったことや、プログラムのイメージ形成が十分でなく、プログラムの内容をあまり理解せずに図として外化してしまったことから理解度の向上がみられなかったと考える。

このことから今後の課題としてプログラムのイメージ形成補助の追加と学習者に提示する図の見直しが考えられる。今回は学習者に提示する図と理解度の向上の関係の考察が十分できておらず、提示した図と理解度の関係性のさらなる考察のもと、図の見直し行っていく。

## 参考文献

- (1) 鈴木弘明: “教養としての認知科学”, 東京大学出版会, (2009)
- (2) 疋田将之, 仲林清: “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”, 教育システム情報学会第 43 回全国大会, P1-20, pp39-40 (2018)
- (3) 山本三雄, 関谷貴之, 山口和紀: “プログラミングのスキル階層に関する研究”, 情報処理学会研究報告, Vol20 10-CE-104, No3, pp.1-25 (2010)
- (4) Lopez, M., Whalley, J., Robbins, P. and Lister, R: “Relationships between reading, tracing and writing skills in introductory programming”, ICER '08: Proceedings of the fourth international workshop on Computing education research, pp101-112(1972)