

# SIEM を活用したオブジェクト指向プログラミング入門教育における教授の工夫

土肥 紳一<sup>†</sup> 宮川 治<sup>†</sup> 今野 紀子<sup>‡</sup>

東京電機大学 情報環境学部<sup>‡</sup>

## 1. はじめに

初学者を対象としたオブジェクト指向プログラミング入門教育は、「コンピュータプログラミング B」(以下「コンプロ B」と略)を開講している。この授業では受講者のモチベーションの向上を目指す教授法(SIEM:ジーム)を実践している[1]。SIEM では授業毎に理解度調査を実施し、この結果を追跡する中で、理解度が低下する授業内容は、コレクションクラスの ArrayList オブジェクトであることが明らかになった[2]。このオブジェクトの概念を理解させるために、実物を活用しながら説明の工夫を行うが、限界があった。この問題を解決するために、オブジェクトの参照に重点を置いた教授の工夫を取り入れることによって、理解度とモチベーションが向上していることが分かった。本論文では、試行錯誤の中から得られた教授の工夫と効果について述べる。

## 2. 授業と理解度調査結果について

「コンプロ B」は1コマ50分の授業を2コマ連続し、週2回開講している。時間帯は月水の14:30~16:20(途中10分間の休憩有)である。受講者が多いため5クラスに分割し、1クラスは50~60名の受講者である。2コマ目の後半では授業時間内で解ける程度の課題を出題し、授業の理解度調査を目的としたwebベースのアンケートと一緒に課題を提出させる。理解度は、授業毎に数個の調査項目を設け、「はい」「いいえ」の2択で回答を得る。2005年から2014年までの理解度を分析し、計113項目に対する回答について「はい」の割合を算出した。2005年当初には調査項目が存在しないものがあり、この場合は実在するデータのみで算出した。この結果を図1に示す。横軸が調査項目の番号、縦軸が理解度を示す。理解度が80%を下回る項目は、受講者にとって理解が難しい項目と判断した。図1では80%を赤い線で示し、これを下回る質問項目を抽出したものを表1に示した。noは質問項目の番号を、以下同様に「はい」の割合、質問項目、章番号、備考の順に示した。各質問項目に対して、「○○は理解できましたか」の問いを設けた。講義内容は、参考文献を参照されたい[3]。no33からno40までの項目は、4章の内容である。複数のサイコロ(Dice オブジェクト)を扱うために ArrayList オブジェクトを活用し、カップ(Cup オブジェクト)を作る内容である。Dice クラスと Cup クラスの API 仕様を表2に、クラスの間接関係を図2に示す。

表 1 理解度(はい)が 80%を下回る項目

no	はい	質問項目	章	備考
5	79.6	クラスとオブジェクトの違い	2	
30	79.8	ArrayList の get メソッド	3	ArrayList
33	77.3	Cup クラスの add メソッド	4	ArrayList を使って Cup オブジェクトを作成する。
34	73.0	Cup クラスの get メソッド		
35	76.0	Cup クラスの size メソッド		
36	72.0	Cup クラスの cast メソッド		
37	73.3	Cup クラスの getSum メソッド		
38	68.8	Cup クラスの getValue メソッド		
39	74.5	引数		
40	75.4	返却値	5	インタフェースを実装したオブジェクトを ArrayList で扱う。貯金箱、本箱、財布を取り上げる。
53	77.2	DiceGame クラスの judge メソッド		
57	75.2	インタフェースの抽出		
58	77.5	インタフェースの実装方法		
62	78.0	小切手クラス(Check)	6	インタフェースを実装したオブジェクトを ArrayList で扱う。貯金箱、本箱、財布を取り上げる。
63	76.1	小切手クラス(Check)に Value インタフェースを実装する方法		
64	79.9	プログラム B6_4 の ArrayList オブジェクトに Coin50, Coin100, Check オブジェクトが入る理由		
69	68.1	CoinBox クラスの remove メソッド		
72	75.9	CoinBox クラスの print メソッド	4,6	
74	67.3	Bookshelf クラスの get メソッド		
77	72.6	Wallet クラスの remove メソッド		
79	79.1	Wallet クラスの getSum メソッド	7	UML 参照の理解を深める。
85	74.0	シーケンス図		
88	77.2	EducatteeList クラス	8	インタフェースを実装したオブジェクトを ArrayList で扱いビンゴへ応用する。
89	76.7	EducatteeList の API 仕様		
96	78.4	ConcreteBox クラス		
97	76.5	演習 8-6 のオブジェクト図		
98	76.9	プログラム B8_2	8	インタフェースを実装したオブジェクトを ArrayList で扱いビンゴへ応用する。
104	76.0	Ball インタフェースの実装によって、ビー玉も容易に扱えること		



図 1 理解度の推移

Devises of teaching method in Object Oriented Programming Education for novice programmer by using the SIEM  
<sup>†</sup>Shinichi Dohi, Osamu Miyakawa, Noriko Konno  
<sup>‡</sup>The School of Information Environment, Tokyo Denki University

目の番号を、以下同様に「はい」の割合、質問項目、章番号、備考の順に示した。各質問項目に対して、「○○は理解できましたか」の問いを設けた。講義内容は、参考文献を参照されたい[3]。no33 から no40 までの項目は、4 章の内容である。複数のサイコロ(Dice オブジェクト)を扱うために ArrayList オブジェクトを活用し、カップ(Cup オブジェクト)を作る内容である。Dice クラスと Cup クラスの API 仕様を表 2 に、クラスの間接関係を図 2 に示す。

表 2 Dice クラスと Cup クラスの API 仕様

Cup	API仕様
arrayList.ArrayList<Dice>()= new ArrayList<Dice>()	
Cup() add(dice: Dice):void get(number:int):Dice size():int cast(): getSum():int getValue(number:int):int	コンストラクターです。 引数でDiceオブジェクトを受け取り、ArrayListにaddします。 引数numberで指定したオブジェクトの参照を返却します。 Cupオブジェクトの中のDiceオブジェクトの個数を返却します。 Cupオブジェクトの中のDiceオブジェクトを全て振ります。 Cupオブジェクトの中のDiceオブジェクトの目の合計を返却します。 引数numberで指定したDiceオブジェクトの値を返却します。
Dice	API仕様
random:Random = new Random() Value:int	
Dice() cast():void getValue():int	コンストラクターです。 1から6の乱数を生成し、状態のvalueに格納します。 状態のvalueを返却します。



図 2 クラスの関係

no53, no57, no58 の項目は、5 章の内容である。サイコロの目の合計が偶数か奇数かを当てるゲーム (DiceGame オブジェクト) を取り上げる。no53 は勝敗判定の理解である。勝敗の判定は ArrayList オブジェクトが管理している Dice オブジェクトを順に参照し、その目の合計を求める必要があるが、初学者にとって難解である。no57, no58 はインタフェースの概念を教わり、偶数しか出ないサイコロ (EvenDice), 奇数しか出ないサイコロ (OddDice) を同一視して扱うことを学ぶ。no62 から no64, no69, no72, no74, no77, no79 は、6 章の内容である。Value インタフェースを定義し、これを実装する貨幣 (Coin オブジェクト), 小切手 (Check オブジェクト) を取り上げ、貯金箱 (CoinBox オブジェクト) で扱う。最後は、ビンゴゲームへと続く。

### 3. 教授の工夫

表 1 の備考欄に示した通り、ArrayList オブジェクトが随所に出現する。この概念を理解させるために、筆者は蓋の付いた箱を使って説明している。この様子を図 3 に示す。左図は箱の中にサイコロが 3 つ入っており、右図は蓋を閉めた箱がカップの中に入り、図 2 の関係を例示している。ArrayList オブジェクトの add メソッドはサイコロを箱の中に入れること、size メソッドは箱の中にサイコロが何個入っているかを知ることとしてイメージさせ易い。一方、get メソッドは箱の中からサイコロが取り出すことがイメージさせ易いが、ここに実物を使った説明の限界がある。get メソッドはオブジェクトへの参照を返却するだけであり、サイコロを取り出さない。実物を使った入れ子の関係は直感的であるが、操作性が悪く煩雑な説明になりがちである。この問題を解決するために、オブジェクトへの参照の概念に重点を置いた教授の工夫を加えるようになった。



図 3 実物を使った教授の工夫

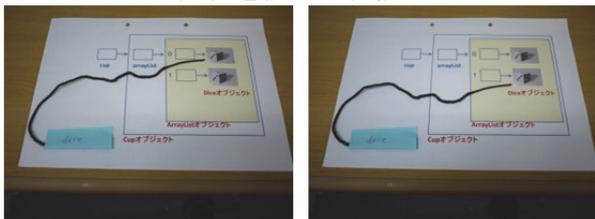


図 4 ポストイットと組紐を使った教授の工夫

具体的には、煩雑な説明を回避するためにオブジェクトの入れ子の関係を印刷した用紙、ポストイット、組紐を活用するものである。ポストイットには変数名を記載し、これに組紐をセロテープで取り付け、組紐がサイコロを指すように配置し、参照を表現する。この様子を図 4 に示す。左図は get メソッドによって変数 dice が最初に入れたサイコロを、右図は 2 つ目のサイコロを参照している様子を示しており、サイコロを取り出しているわけではない。参照の概念が、容易に説明可能となった。

### 4. 教授の工夫の効果

教授の工夫の効果を比較するために、2010 年と 2014 年の理解度調査結果を比較した。この様子を図 5 に示す。2010 年は、実物を活用しながら理解を深める工夫を行っていた頃である。これに加えて 2014 年は、オブジェクトへの参照の概念をポストイットと組紐で教授の工夫を行った。授業全体を通じて、2014 年は理解度が向上していることが窺える。また、モチベーションの推移を図 6 に示す。2010 年のモチベーションは前期(18.8), 中期(18.1), 後期(17.4)であったが、2014 年は前期(21.3), 中期(19.2), 後期(18.0)のすべてにおいて向上した。

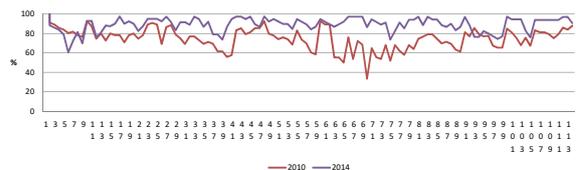


図 5 理解度の推移(2010 年と 2014 年の比較)

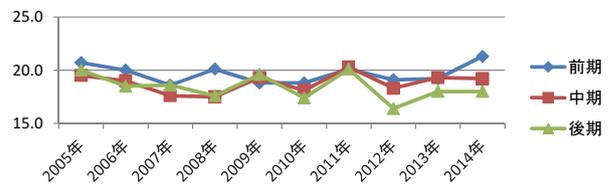


図 6 モチベーションの推移

### 5. まとめ

ArrayList オブジェクトの仕組みを理解させるために、オブジェクトの入れ子の関係を印刷した用紙、ポストイット、組紐を活用した教授の工夫によって、理解度とモチベーションの向上が示された。今後も受講者の理解度の向上、モチベーションの向上に取り組む計画である。本研究の一部は、科学研究費補助金(基盤研究(C) 課題番号 21500957), 東京電機大学総合研究所一般研究(Q12J-02)として行っている。

#### 参考文献

- 1) 土肥紳一, 宮川 治, 今野紀子: SIEM によるプログラミング教育の客観的評価, 情報科学技術フォーラム, 情報科学技術レターズ, Vol.3, no.3, pp.347-350(2004).
- 2) 土肥紳一, 宮川 治, 今野紀子: 初学者におけるオブジェクト指向プログラミングの難しさの分析と教授の工夫, 情報処理学会, 情報教育シンポジウム論文集, Vol.2012, no.4, pp.21-28(2012).
- 3) 「コンピュータプログラミング B」講義ノート, <http://www2.dcl.sie.dendai.ac.jp/dohi/2014/proB/>