

## オブジェクト指向の継承の概念を学習する手法と学習環境の提案

本多 佑希                      岸本 有生                      漆原 宏丞                      兼宗 進  
 四天王寺大学                  大阪電気通信大学              大阪電気通信大学              大阪電気通信大学  
 honda@shitennoji.ac.jp                  高等学校

小中高でプログラミング教育の比重が高まっている。小学校段階から構造化プログラミングの学習が重要視されており、教育用に設計された多くの教育用プログラミング言語が「逐次・分岐・反復、サブルーチン」といった処理の構造化や、「配列」といったデータ構造の学習を実現している。一方で、関連する変数や処理を構造的に扱うようなオブジェクトベース、オブジェクト指向を採用している教育用言語は多くなく、初学者を対象とした教育事例も見受けられない。本稿では、対話的なプログラムを開発する学習の中で、定義した概念に共通する性質を抽象的な概念に定義する、オブジェクト指向の継承の考え方を学習する手法と、それを実現する環境を提案する。また、オブジェクト指向の教育を受けていない大学生を対象とした実証授業を行ったので、内容と結果を報告する。

### 1. はじめに

プログラミングにおいて、構造化は重要な考え方である。小中高のプログラミング教育では「逐次・分岐・反復、サブルーチン」といった処理・データの構造化が扱われることから、プログラミングの要素として構造化が重要であることは広く認知されており、多くの教育用言語によって、初学者がそれらを学習できる環境が実現されている。関連する処理やデータを纏めて扱うオブジェクトベースな考え方の学習も Scratch やドリトル<sup>①</sup>が実現している。本研究では、それらの学習の次のステップとして、各オブジェクトに共通する性質を抽象的なオブジェクトに纏めるようなオブジェクト指向の考え方を初学者に学習させる手法と、それを実現する環境を提案する。

### 2. 現状のプログラミング教育

プログラミングにおいて構造化は重要であり、プログラミング教育の場でも初学者を対象とした学習環境・教育事例は多く存在する。しかし、同じように重要な考え方である抽象化については、プログラミングに習熟した大学生が学習する、高度な概念であるという認識は共通している。情報処理学会が策定・公開している大学を対象としたカリキュラム標準 J17<sup>②</sup>では、クラス設計やオブジェクト指向は講義形式の授業の中で扱われている。また、高井らは大学3年生を対象とした講義の中で、提示されたクラス図を基に Java のプログラムを記述する課題を扱う取り組みを行っている<sup>③</sup>。

手続き型パラダイムでは、関連するはずの変数や処理が独立しているため、モデルの理解や拡張は難しい。Scratch はオブジェクトベースを採用しており、オブジェクトに関連するイベントが各

スプライトに定義するためモデルの理解もしやすく拡張もしやすい。ドリトルはオブジェクト指向を取り入れているため、ある機能が纏まったオブジェクトから作られたオブジェクトを生成し、それを操作する活動が行われ、Scratch と同じくモデルの理解や拡張が行いやすい。

本研究で目的とする、初学者を対象とした継承の概念を学習とする学習は、これらに続く流れとして、重要なものである。

### 3. 検討

#### 3.1 題材・ベース言語の検討

オブジェクト指向の継承という概念を説明する際には、「吠える」という処理、「声」という情報を定義した「犬」「猫」というクラスを抽象化した「動物」というクラスを定義するようなモデルがよく扱われる。犬と猫や動物といった、現実存在する概念をモデルとして扱うことは分かりやすい一方で、こういったモデルを扱いながら、意味のあるプログラムにすることは難しい。

今回は、プログラムではこうした「現実に存在する概念をモデル化し、概念ごとの性質を定義する」活動を行い、そのプログラムに対して性質を問いかけると応答してくれる、チャットボットのような実行モデルを採用することにした。

また、ベースとするプログラミング言語については、オブジェクト指向を採用したプログラミングであることは必須の要件である。今回は、教育用に設計されており、識別子に2バイト文字を用いることができるドリトルを採用することにした。

#### 3.2 プログラムの検討

「すずめは飛ぶ」という問いかけを「すずめ」

鳥＝クラス！作る。  
 鳥：飛ぶ？＝「クローバー！」飛ぶ”話す」。  
 すずめ＝鳥！作る。  
 ハト＝鳥！作る。  
 トキ＝鳥！作る。  
 ペンギン：飛ぶ？＝「  
     クローバー！」飛ばない”話す  
 」。

図 1- 鳥の性質を定義するプログラム例

オブジェクトの「飛ぶ？」命令の実行と解釈することにした。図 1 に、「鳥」とそれを具体化した「すずめ、ハト、ペンギン」のモデルを定義するプログラム例を示す。これらは鳥の性質を引き継いでいるため「すずめは飛ぶ？」という問いかけに対し、「飛ぶ」と返答する。ペンギンは鳥の性質を引き継ぎつつ飛ばないという例外的な性質を持つため、「飛ぶ？」という命令を上書きしている。

#### 4. 実行環境

今回提案する内容を実際にプログラミングし、実行する環境を開発した。Web ブラウザ上でプログラミングできる。プログラムは、チャット形式で問いかけて実行できる他、スマートスピーカーである LINE Clova のアプリケーションとして実行することもできる。

#### 5. 実証実験

今回提案する手法・環境で、初学者がオブジェクトの継承関係を用いたプログラミングが可能なことを確認するために、実証授業を行った。対象は工学系大学生 2 年生 72 名で、C のプログラミングを事前に学んでいるが、オブジェクト指向言語は授業内で扱っていない。

図 1 のプログラムとともに、図 2 のプログラムを演習として扱った。図 1、図 2 のモデルをクラス図で表現した例を図 3、図 4 にそれぞれ示す。これらのモデルを演習で扱うとともに、これらのモデルと同様の関係の概念をそれぞれ考えてプログラムとして表現する課題を出した。

課題 1 では、演習で扱ったプログラムの規模を大きく超えるプログラムを提出する学生が見受けられた。一方で、課題 2 では、明らかに的外れのプログラムを提出した学生はいなかったものの、ほとんどの学生が演習と同じ程度の規模のプログラムだった。良い題材が思いつかなかったのかモデルが難しかったのかについては、プログラムを観察しつつ、詳細に分析を進めたい。

これらの結果から、オブジェクト指向に初めて触れる初学者でも、十分に継承を活用したモデル

動物＝クラス！作る。  
 動物：鳴き声？＝「クローバー！」(自分：声) 話す」。  
 犬＝動物！作る。  
 犬：声＝”ワンワン”。  
 猫＝動物！作る。  
 猫：声＝”ニャーニャー”。

図 2- 動物の性質を定義するプログラム例

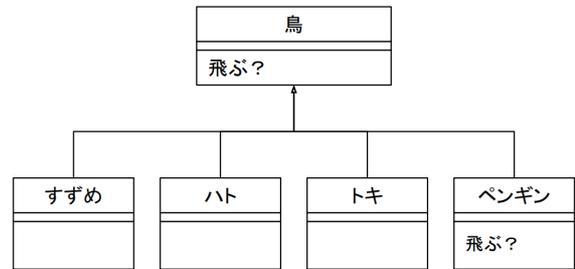


図 3- 図 1 のモデルをクラス図で表現した例

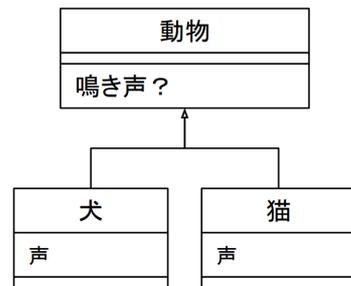


図 4- 図 2 のモデルをクラス図で表現した例

を考えてプログラミングできることが分かった。

#### 6. まとめ

初学者でもオブジェクト指向の継承の考え方をを用いたプログラミングが可能な手法と、それを実現する環境を開発し、提案した。実証授業から、十分に初学者でも継承を用いたモデルの設計と実装が行えることがわかった。

#### 参考文献

- (1) 大阪電気通信大学 兼宗研究室: プログラミング言語「ドリトル」, 入手先 <<http://dolittle.eplang.jp/>> (参照 2021-5-15).
- (2) 情報処理学会: カリキュラム標準 J17, 入手先 <[https://www.ipsj.or.jp/annai/committee/education/j07/curriculum\\_j17.html](https://www.ipsj.or.jp/annai/committee/education/j07/curriculum_j17.html)> (参照 2020-10-15).
- (3) 高井久美子, 渡辺博芳, 佐々木茂, 鎌田一雄: 個別学習と協調学習を組み合わせた授業例-オブジェクト指向モデリング導入教育における設計と実践-, 教育システム情報学会誌, Vol.28, No.3, pp.210-222, 2011.