

## 段階的アニメーションによるJava学習支援システム

高津陽平      田部井俊彦      伊藤小琴      前川仁孝      伊與田光宏  
Yohei Takatsu    Toshihiko Tabei    Ogoto Ito    Yoshitaka Maekawa    Mitsuhiro Iyoda

千葉工業大学  
Chiba Institute of Technology

### 1. はじめに

本学情報工学科では、1年生180人を対象にJava言語のプログラミング演習が行われている。学習はWebテキストの教材データを分析、改良、考察する自習形式で進められる。演習には指導補助員が参加しており、学生からの質問を受け付けている。上記の環境下において、言語の知識を基にした問題解決への考え方を学生に学ばせることを最も重要な目的としている。しかし、現状として、サンプルコードから処理内容を理解できないなど、アルゴリズムの考案に悩む以前の問題を抱える学生が多く見られた。

そこで、本研究では演習での利用を想定し、文法を調査可能にさせ、利用結果を自動記録する機能、そして、サンプルコードを分割、段階的に実行し、実行結果の違いを調査させる機能を有する学習支援システムを構築する。そして、学生にサンプルコード内の処理への理解を支援することを目的とする。

### 2. アンケートによる学生の現状と分析

1年生140人にどのような点が難しいかについて、複数の回答をして貰った。結果を以下に示す。

- ・ サンプルコード内の処理への理解 (93人)
- ・ 動作を可能にするアルゴリズムの考案 (31人)
- ・ ソースコードの記述及びコンパイル (24人)

多くの学生がアルゴリズムを考案する段階に到達し難い状況に陥っている。は学生の自習不足と共に、サンプルコードと実行動作の関係が起因している。サンプルコード内の処理は解説文を伴い、数行毎に学習するのに対し、動作確認は一度に行う。そのため、各行のコードが実行動作に対し、どのように反映されているのかが他コードの処理による実行動作と組み合わせたり、確認し難い。は数学的な空間認識に慣れていないことが起因している。教材では図形をJavaApplet上に描画するため、図形分野における計算が必要となる。しかし、この点は支援すべきではない。教育的観点から考えれば、アルゴリズムの考案に対しては大いに悩むべきである。指導補助員から学生への一方的なアルゴリズムの教授は支援と言えない。そのため、は問題点として捉えない。はが元で文法エラーが検出されること、また、が元で論理エラーが検出されることが起因している。については指導補助員が対処している。

### 3. システム提案

本システムでは、サンプルコード内の処理への理解を促進させる支援に特化し、以下の特徴を有するシステムを提案する。

文法を調査可能にし、利用結果を記録する機能

自習不足である学生に対し、自主的に文法を調査可能となる環境を提供。また、学生が教材の解説文に説明不足を感じる箇所があるため、サンプルコードにおいて、どの行の文法を調査しているのか、教材管理を行う指導者側に伝達

段階的なアニメーションによる用語の理解促進  
解説文において理解できない用語がある場合には、解説文と連動して段階的に動作するアニメーションを表示し、変化に注目させて理解を促進  
サンプルコードを分割、段階的に実行し、実行結果の違いから処理を調査させる機能

サンプルコードと実行動作の提示方法のずれを解決し、ソースコードの差分と実行結果の変化を段階的に閲覧、操作を通して確認

教材と同様の動作を持つJavaAppletを使用

学習支援システムから提示されたソースコードや文法を理解し、得た内容を教材のサンプルコード、文法に当てはめ、再度、理解し直すといった行動を短縮させる。これにより、次段階となるアルゴリズムの考案へ移行させることを狙う

### 4. 支援の段階

演習中、Webテキストによる教材データから、文法への理解及びサンプルコード内の処理への理解が困難であると学生が判断した際、本システムでは以下の3段階の流れで支援を行う。

#### 1. 文法の調査

ソースコードの行単位でどのような処理が記述されているのかを明確にさせる

#### 2. 用語の調査

Javaにおいて、言葉のみで説明されても理解しにくいクラス、継承などの概念はアニメーション表示し、視覚的な効果から明確にさせる

#### 3. ソースコード内の処理の調査

ソースコードの複数行単位で、どのような実行動作を実現しているのかを明確にさせる

## 5. システム構成

図1にシステム構成を示す。演習を行う学生のみがシステムを利用できるよう、利用開始時にHTMLページでJavaScriptによる団体認証を行う。その後、文法の調査と段階的な実行動作からの処理の調査を選択するページが取得される。文法の調査を選択した場合は、サンプルコードが書かれたJPEG画像が取得される。画像はClickableMapとなっており、各座標は、コード名を示すコードネームIDとCGIの呼び出し命令を持つ。学生が画像中の調査したいコードをクリックすると、Pealにより、CGIが実行される。CGIは呼び出された座標のコードネームIDの情報と、datファイルに予め書かれたコードネームIDを照合し、datファイルをクリック数をカウント、上書きしながら対応するLink先URLを割り出す。そして、文法の解説が行われる。解説文中の用語が理解できない場合にはFlashによって段階的なアニメーションが表示される。一方、処理の調査を選択した場合はソースコードとJavaAppletによる実行結果が複数取得される。表示する順序は、ある1つ処理を実行するソースコードに対して、別の処理を行うコードを段階的に追加し、その度に実行結果を表示する形式をとる。

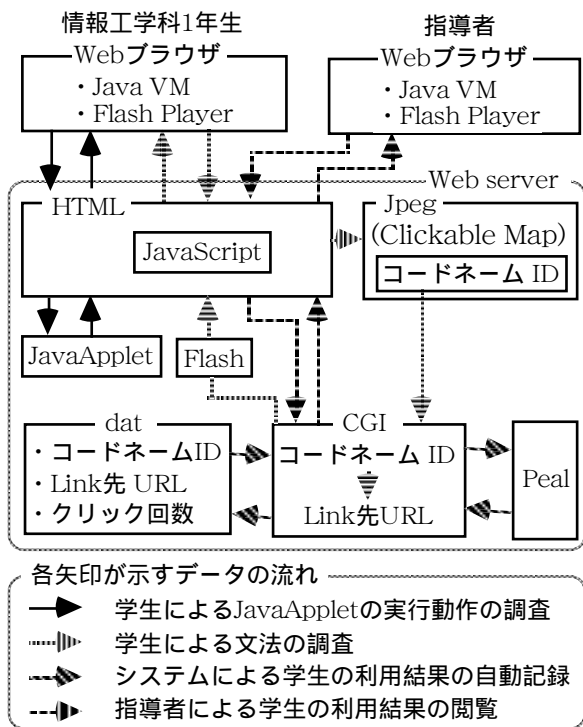


図1 システム構成図

## 6. 実行例

段階的なアニメーションによる用語の理解促進画面例を図2に示す。アニメーションは処理の実行順序、過程を学ぶ上で重要となるクラス、フィールド、メソッドを単語や抽象図で表現し、HTML内に設けた専用のフレーム内に表示していく。解説文の進行に合わせ、アニメーションが変化していく様子を見せ

ることにより、学生の視点をその変更点に注目させ、用語の意味、処理の順序、過程を理解させる。

継承という用語のアニメーション例を以下に示す。

1. クラスの概念をコンピュータの設計図に見立てて Computerクラスという抽象図(図2- )を表示
2. クラスが持つコンストラクタやメソッドの解説文に応じて、コンストラクタ名(図2- )、メソッド名(図2- と )を表示
3. 別のコンピュータの設計図に見立てたTowerクラスの抽象図(図2- )を表示
4. 別のクラスが持つコンストラクタやメソッドの解説文に応じて、コンストラクタ名(図2- )、メソッド名(図2- )を表示
5. TowerクラスがComputerクラスを継承し、Computerクラスのフィールド、メソッドを継ぐ
6. Towerクラスのインスタンスred(図2- )というコンピュータはComputerクラスとTowerクラス、両方のフィールド、メソッドを利用可能
7. Sampleクラスの変数red(図2- )で管理

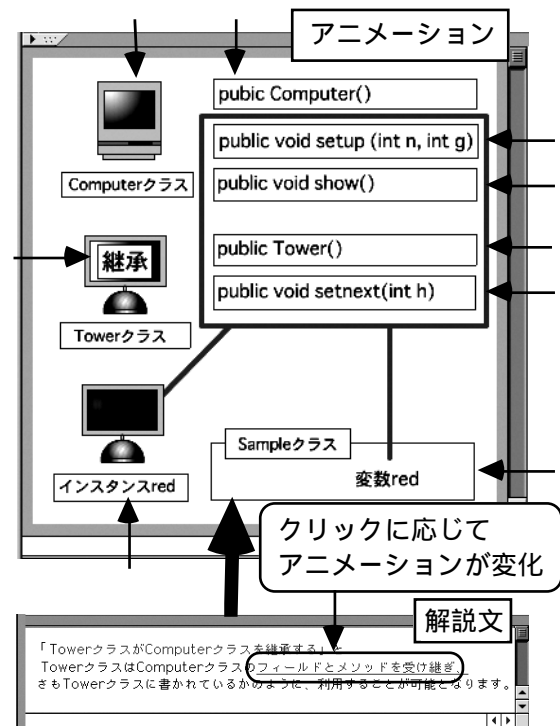


図2 段階的なアニメーションによる用語の理解促進画面例

## 7. おわりに

本システムを演習に導入することで、限られた演習時間内での文法、用語への理解が促進され、ソースコード内の複数行で実行される処理を段階的な実行動作から確認し、実際に触れることにより、理解や把握が容易になると考えられる。また、指導者側に学生が理解し難い文法を数値で伝達する機能は、今後、演習を続ける上で、指導者が解説文の量を考える際、有用なデータとなると考えられる。