

Processingを用いたプログラミング導入教育の取り組み

関 和広，北村 達也，永田 亮，和田 昌浩，梅谷 智弘，前田 多章

甲南大学 知能情報学部 知能情報学科
神戸市灘区岡本 8-9-1, 658-8501

概要

知能情報学部では、「プログラミングに関する検討ワーキンググループ」での議論を通し、プログラミング教育に関するカリキュラムの改善を図ってきた。その結果、プログラミング関連科目を大幅に増やすようカリキュラムの改訂を行い、2014年度より新しいカリキュラムのもとでプログラミング教育を施している。本稿では、プログラミング関連科目の導入教育として1年次前期に開講している「プログラミング演習Ⅰ」を取り上げ、その取り組みと成果について報告する。具体的には、プログラミング関連科目におけるプログラミング演習Ⅰの位置付け、実際の進行スケジュールなどの授業の構成や運用・サポート体制、受講者を対象にしたアンケート結果から見た演習の効果などについて報告する。

キーワード: プログラミング演習, 導入教育, Processing

1 はじめに

知能情報学部では、2012年度から「プログラミングに関する検討ワーキンググループ」(以降WG)を組織し、プログラミング教育に関するカリキュラムの改善のための議論を行った。当時の知能情報学部では、プログラミング科目として1年次対象の「プログラミング実習(通年)」と2年次対象の「アドバンストプログラミング(前期)」が開講されていた。「プログラミング実習」ではC言語でプログラミングを教えており、数値や文字を処理するプログラムの作成が主であった。そのため、GUIになじんだ受講生の意欲を維持することが難しいという問題があった。また、プログラミング科目がこれら2つだけであったため、2年次後期以降の講義・演習でプログラミングを行う機会が十分ではないという問題もあった。

このような状況を改善するため、WGにおける議論を経て、プログラミング関連科目を大幅に増やすようカリキュラムの改訂を行った。2014年度より、新しいカリキュラムのもとでプログラミング教育を施しており、新カリキュラム移行後の入学者は、本年4月に3年次になるところである。本稿では、導入教育として1年次前期に開講している「プログラミング演習Ⅰ」について取り上げ、その取り組みと成果について報告する。

以下、2節でプログラミング関連科目におけるプログラミング演習Ⅰの位置付けやねらいについて述べ、3節で進行スケジュールや各回の概要などについて紹介する。続いて、4節でプログラミング演習Ⅰを円滑・効果的に実施するための運用体制について述べ、5節で受講者を対象にしたアンケート結果から演習の効果や問題点を考察する。最後に、6節で本稿のまとめと今後の展望について述べる。

2 プログラミング演習Ⅰの位置付け

甲南大学の唯一の情報系の学部として、知能情報学部におけるプログラミング教育充実は、在学生からの要望や社会的な要請が高い。また、配属研究室や研究テーマによるものの、3年次に履修するセミナーや4年次に行う卒業研究においてもプログラミングの知識が求められることが多く、その関連科目群は、当学部のカリキュラムの土台の一つを成している。特に、プログラミング演習Ⅰは1年次前期に開講されていることから、そのようなプログラミング教育の入り口としてきわめて重要である。プログラミング演習Ⅰでプログラミング全般に苦手意識を持ってしまうと、直接的・間接的にプログラミングに関わる後続の科目群の履修の障害となり、不可の単位が積み上がりやすい。そこでプログラミング演習Ⅰでは、プログラミングに親しみをもち、到達目標をシラバス内で次のように明示している。

プログラミングの基礎科目として、以下の事項の達成を目指す。

1. 計算機の基礎操作を習得する。
2. プログラミングに親しみをもち、簡単なプログラムを作成することができる。

この到達目標を達成するため、プログラミング演習Ⅰでは、プログラミング言語として画像や動画の描画が容易な Processing¹ を採用し、プログラムの動作を視覚的に確認させることでプログラミングの効果的な習得を図っている。また、少人数クラス、プログラミング相談室の設置、オリジナルプログラムの作成・発表などの取り組みを行っている。これらの取り組みについての詳細は、次節以降で述べる。

3 授業の構成

2015年度のプログラミング演習Ⅰの大まかな進行スケジュールを表1に示す。プログラミング演習Ⅰは、学生が入学直後に受講する科目であるため、最初の3回は大学の計算機システムの使い方やタイピング、UNIXコマンドの使い方、テキストエディタの使い方など、ITリテラシ的な内容を行っている。実際にプログラミングの演習を行うのは4～12回目(計9回)であり、最後の3回はオリジナルプログラムの作成と成果発表会に当てられている。

4回目以降では、プログラミング言語として前述のように Processing を採用し、演習を行っている。Processing は、当時 MIT メディアラボに在籍していた Reas と Fry が開発した Java ベースのプログラミング言語であり、アートやデザインの学生の使用を念頭におき、グラフィックスとインタラクションに焦点が当てられている [1]。そのような成り立ちから、Processing では画像の描画やアニメーションがきわめて容易に扱え、プログラムの修正や変更に対する視覚的なフィードバックが即座に得られるという特長がある。結果的に、アート等の分野だけでなく、Processing はプログラミング初心者の学習に適していると言われている [2]。

Processing がプログラミング初心者に適しているもう一つの理由は、シンプルな開発環境 (Processing Development Environment; PDE) が一体化しており、プログラムの作成から実行までの流れが大変分

¹<http://processing.org>

表 1: 2015 年度プログラミング演習 I の進行スケジュール .

週	内容
1	概要, 計算機システムの使い方など
2	タイピング, UNIX 基礎
3	UNIX 基礎, テキストエディタの使い方など
4	Processing によるプログラミングの基礎, 図形の描画など
5	様々な図形の描画, 色の指定, コメント文など
6	変数, 計算, for 文による繰り返し
7	for 文による繰り返し (応用)
8	アニメーションの描画, マウス入力
9	if 文による条件分岐
10	if 文による条件分岐 (応用)
11	配列と繰り返し
12	配列と繰り返し (応用)
13-14	オリジナルプログラム作成
15	成果発表会

かり易いことにある。図 1 に PDE のスクリーンショットと IDE に記述した楕円を描くプログラム (`ellipse(50, 50, 80, 60);`) の実行結果を示す。

学習者は PDE 中央のエディタ部にプログラムを書き、左上の実行ボタン (▶) をクリックするだけで、右側の図のような実行結果を即座に確認することができる。プログラム自体も、C や Java などのプログラミング言語では必要となるライブラリの読み込みや関数の宣言が必須ではなく、例のようにわずか一行のプログラムから実行できる。上述のように、コマンドによるコンパイル (`cc` や `javac`) など、初学者には理解の障害となりやすい要素に触れることもない。なお、実行にはプログラムの保存すら必要ない (もちろん保存することも簡単にできる)。これらにより、学習者はプログラム実行以前の様々な決まりごとに煩わされることなく、プログラミングのみに集中できるという大きなメリットがある。

このような特長を持つ Processing を教材として、4~12 回目の演習では、Processing に固有の図形描画に関する命令 (関数) を学んだのち、for 文による繰り返しや if 文による条件分岐、配列など、他の言語にも共通する基本的な事項を学習する。ただし、他の言語とは異なり、プログラムの実行結果を即座に視覚的に確認できるという Processing の特長を活かし、for 文で図形を繰り返し描画したり、if 文でアニメーションに変化を与えたりといった方法を通して、プログラムの基本制御構造を直感的に理解しつつ学習していく。また、実際に手を動かし、自ら試行錯誤することでプログラミングを習得できるよう、練習問題を随所に取り入れている。

13~14 回目には、演習の総仕上げとして、オリジナルプログラムの作成を行っている。この課題では、各自が自由な発想でプログラムの作成を行う。ただし、いきなりプログラムを書き始めるのではなく、作品の質がある程度以上の水準になるよう、最初に方眼紙に設計図を手書きで作成し、教員の承認後にプログラミングを開始することになっている。設計図の一例を図 2 に示す。なお、この設計図を基に完成した作品は、後述のウェブページ (図 4 参照) で紹介されている。

設計図が承認されたのち、最終回の成果発表会に向けて学生はプログラムの作成を行い、発表準備を整えていく。2015年度の発表会では、学生の作品をプロジェクタで教室前方のスクリーンに投影し、学生自身が、(1)何のプログラムなのか、(2)実装に苦労した点、(3)独創的な点を2分間で説明し、続く2分間で質疑を行うという形式をとった。発表会の様子を図3に示す。そして、発表内容に基づき、難易度(実装の困難さ)、独創性(自分で考えた優れたアイデアの有無)、描画の美しさ(構図や色のセンス、アニメーションの効果など)の観点から複数教員で採点を行った。また、学生の相互評価の仕組みとして、最も優れていると思われる作品3件を各学生が推薦できるようにした。推薦多数の作品は、教員からの評価と合わせて優秀作品選定の基礎資料として用いた。なお、選定された優秀作品はウェブ²で公開(図4)し、学生のモチベーション向上を図るとともに、高校生向けの学部紹介などにも役立てている。



図 3: 最終成果発表会の様子。

4 運用体制

プログラミング演習Ⅰを受講する多くの学生は、この演習で初めてプログラミングに触れる。そのため、指導補助・質問対応などに通常の演習よりもきめ細かいサポートが必要である。そこで2015年度のプログラミング演習Ⅰでは、受講者を50名程度の少人数クラスに分けて演習を行った。なお、当該科目は1年次前期開講であり、学生のプログラミング能力を見極めることが難しいことから、習熟度等によるクラス編成は行っていない。これに対し、1年次後期開講のプログラミング演習Ⅱでは、演習Ⅰで学生の習熟度をある程度把握できているため、2015年度より習熟度別のクラス編成としている。

受講者に対するサポート体制を充実させるため、少人数クラスに加えて、各クラス4~5名のスタッフで学生への指導(補助)を行っている。人数の内訳は、教員2名、大学院生のティーチングアシス

²<http://www.konan-u.ac.jp/faculty/ii/public/prog1/showcase>



図 4: 優秀作品を紹介する学部のウェブページ。

タント（TA）が2～3名である。また、最初の数回は操作上のトラブル・質問が多く、より多くの人手が必要であるため、各クラスにさらに1名の教員がサポートに加わった。

さらに、授業時間内だけでは十分なサポートができない可能性があるため、2015年度には授業時間外に定期的に時間を設け、プログラミング相談室を計7回開室した。相談室では、授業の内容や課題、自宅での演習環境構築など、プログラミングに関する質問・相談であれば何でも受け付けるものとし、教員とTAを合わせて1～3名程度で対応した。相談室の利用者は1～15名程度であり、その週の課題の難しさ等によって増減するようであった。なお、プログラミング相談室が主にターゲットとしていた学生は、習熟度が低い層であったものの、実際の利用者は習熟度が中程度以上の学生が多かった。次年度以降、よりサポートが必要な層の学生を呼び込むための何らかの方策を講じたい。

5 演習の効果

5.1 終了後の自由記述アンケートから

2015年度のプログラミング演習Ⅰを受講した学生を対象に、前期終了後、MyKonan（学内専用のポータルサイト）を通して自由記述のアンケートを行った。再履修を除く受講生135名中116名の回答があり、全体的には「大変だったが楽しかった」という意見が多数であった。具体的には、「全体を通してグラフィカルで楽しかった...」、「...processingを触ってみて、とても楽しかったです。なにより実行結果がすぐわかるので...」、「数行のプログラムで簡単に絵を描くことができ、プログラミングに対して最初に感じていたハードルのようなものが無くなった...」等の意見があり、Processingを中心に演習を構成した効果が如実に現れていた。

難しかった点についての具体的記述としては、後半の配列（表1参照）以降という意見が複数見られた。一方、前半は簡単だったという意見が多かったため、来年度以降は、後半に時間を多めに配分することで、進度と理解度のバランスが向上するものと考えられる。

5.2 後続科目へのつながり

Processingは非常に強力なプログラミング言語であるものの、プログラミング関連科目群のカリキュラムにおけるその位置付けは、プログラミングに親しみつつ、繰り返しや条件分岐といった基礎制御構造を理解し、後続の科目で扱うC言語やJavaの習得を助けることにある。よって、プログラミング演習Ⅰの効果は、後続科目であるプログラミング演習Ⅱ（1年次後期開講）やアドバンストプログラミング（2年次後期開講）における学生の理解度などからも間接的に測ることができる。

特に、2014年度以降の新カリキュラムでは、プログラミング演習ⅠでProcessingを採用したことがそれ以前のカリキュラムからの大きな変化であるため、演習ⅠでのProcessingの学習が、続く演習ⅡでのC言語の学習に役立っているか否かについての検証が欠かせない。そこで、2015年度プログラミング演習Ⅱの受講生132名（再履修を除く）に対して期末試験後にアンケートを実施し、以下の設問によってプログラミング演習Ⅰの効果の評価を試みた。

前期の Processing から後期の C へ言語が変わったことについて（一つだけ選択）

- 言語の違いに戸惑い，混乱した
- 言語の違いはあったものの，共通する部分もあり理解の助けになった
- 言語の違いはあったものの，内容に重複が多く無駄に感じた
- その他（具体的に記入：_____）

なお，本演習のねらいとしては，「言語の違いはあったものの，共通する部分もあり理解の助けになった」という回答を得られることが望ましい．125 人から得られた回答を集計した結果，この選択肢を選んだ受講者は 83 人（66.4%）であった．一方「言語の違いに戸惑い，混乱した」を選択した受講者も 32 人（25.6%）おり，演習 I と II の接続性をさらに高める努力が求められる．なお「言語の違いは気にならなかったが，内容に重複が多く無駄に感じた」を選択した受講者は 8 人（6.4%）であった．これらの学生は理解度・習熟度が高い学生であると考えられ，余裕がある学生向けの発展問題の追加や，後述の KONAN スーパー IT 人材育成プロジェクトの枠組みの中でさらにその能力を伸ばす手助けをしていきたい．

6 おわりに

本稿では，知能情報学部におけるプログラミング導入教育として位置付けられる「プログラミング演習 I」について，その取り組みと成果を報告した．カリキュラム改訂後，当該科目は 2014 年度と 2015 年度の 2 度開講しており，徐々に教員側でもノウハウが蓄積されつつある．また，1 年次後期，2 年次のプログラミング関連科目担当者からも，学生の能力向上についての好意的な意見を聞いており，手応えを感じているところである．一方，アンケート結果を見ると，時間配分や C 言語への接続性に一部課題も残されており，次年度以降も漸次的に内容・運営体制を改善していきたい．

なお，プログラミング演習 I が，導入教育として学部全体のプログラミング能力の底上げをねらっているのに対し，意欲と能力のある学生の力をさらに伸ばすことを目指す「KONAN スーパー IT 人材育成プロジェクト」が，2016 年度プレミアプロジェクトの一環として始まる予定である．これらの取り組みを通して，プログラミング教育における知能情報学部の学内外でのプレゼンスを向上させるとともに，優秀な IT 人材を社会に送り出していきたい．

参考文献

- [1] Casey Reas and Ben Fry. Processing をはじめよう. オライリージャパン, 2011. 船田巧 訳.
- [2] 菊池誠. Processing によるプログラミング教育. 情報処理, Vol. 52, No. 2, pp. 213–215, 2011.