

Processing を使ったプログラミング入門教育の実践

- モチベーションの向上を目指して -

土肥紳一*

Email: dohi@mail.dendai.ac.jp

*1: 東京電機大学 システムデザイン工学部 デザイン工学科

プログラミング入門教育, モチベーション, Processing

1. はじめに

2017年4月に、システムデザイン工学部が東京千住キャンパスに開設した。この学部は、情報システム工学科(定員130名)とデザイン工学科(定員110名)の2学科で構成されている。筆者はデザイン工学科に所属し、1年生の後期に「コンピュータプログラミングⅠ」を、2年生の前期に同Ⅱを担当する。「コンピュータプログラミングⅠ」は、手続き型のプログラミングを学び、同Ⅱはオブジェクト指向のプログラミングを学ぶ。プログラミング言語は、Processingを採用した。ProcessingはJavaで記述されており、情報環境工学科で長年Javaを教えたノウハウを継承できる。Processingを使ったプログラミング入門教育について、これを採用するに至った経緯、教授の工夫、授業の理解度調査と本取組の評価等について述べる。

2. Processingを採用するに至った経緯

2.1 初学者におけるプログラミング入門教育の関門

一般的にプログラミングには「おまじない」が付きものである。Javaでは大文字小文字の区別、複数種類の括弧、プログラムを保存する時にファイル名とクラス名を合わせる等、たくさん注意を払う必要がある。コマンドプロンプトを利用する場合、コンパイルはjavacコマンドで行う。javacの後にはソースファイル名を指定するが、拡張子の「.java」を付けなければならない。誤りを修正しながら、コンパイルエラーを消す作業を繰り返す。実行はjavaコマンドを利用する。javaの後には、コンパイルされたファイル名を指定するが、拡張子は指定しない等、初学者にとって操作が煩雑である。実行結果は、文字列等が表示されるだけで、受講者の興味を喚起しない。Processingは、初学者をこれらの煩雑な操作(関門)から解放できる。

2.2 高大連携型教育の体験

2013年~2016年まで、高大連携型教育で「プログラミング入門体験」の講義を担当した。例年5月の土曜日に1回実施し、10:00~12:00までの2時間で完結する内容である。当時、適切な言語を探していたところ、Processingの存在を知った[1,2]。スケッチブックに絵を描く感覚でプログラミングができる開発思想が、この講義に適合した。しかも、タイピング量は少ないにも関わらず、結果をグラフィカルに表示できることに感銘を受けた。実行結果を見ながら、少しパラメータを変更するだけで実行結果が変化し、受講者の興味を喚起することがわかった[3,4]。高校生を対象としたプログラミングの初学者が、試行錯誤を繰り返しながら容易に取り組めることが分かり、Processingを導入する決め手の一つとなった。

2.3 デザイン工学科との親和性

デザイン工学科の学生さんは、プログラミングの専門家を目指すわけではないが、イメージを具現化する目的でキャンパスに絵を描くことを体験する。Processingは、基本図形や線を駆使しながら絵を描く感覚でプログラミングができる思想が、これにぴったり当てはまった。さらに、デザイン工学科のいくつかの研究室ではProcessingを研究で活用しており、容易に賛同を得ることができた。先行して未来科学部の情報メディア学科がProcessingを導入していること、理工系の大学等への導入も進み始めたことも大きな後押しとなった[5]。

3. 「コンピュータプログラミングⅠ」について

3.1 授業内容

「コンピュータプログラミングⅠ」は、1年生の後期に開講する必修科目である。単位数は2単位、1コマ90分/週、15週/半期である。授業内容は表1に示す。なお、2018年度から1コマ100分/週、14週/半期となった。

表1 授業内容

授業回数	授業内容
1回	ガイダンス: プログラムの入力と実行
2回	プログラムの姿: 基本的な図形(線、三角形、四角形、多角形、円、弧)を描画する関数について学び、これらの関数を組み合わせることによって、複雑な図形を描画する。プログラムにおける処理の順序を理解する。
3回	変数: 変数を使った基本的な計算を学習し、その活用によって、図形の描画に役立てる。
4回	繰り返し構造: for文を使った繰り返しを学習し、この活用によって、複数の図形の描画が簡潔に記述できることを学習する。
5回	マウスによる入力: マウスを用いて、座標やボタンの状態を入力するプログラムを学習し、その動きに応じて、図形を描画できることを学習する。
6回	条件分岐: if文を使い、プログラムの流れを条件によって分岐する方法を学習し、マウスのボタンを押した時の色を変更する。
7回	キーボードからの入力: if文と組み合わせることによって、入力したキーに応じて、図形の描画が変更できることを学習する。
8回	中間試験
9回	画像データの扱い: ラスタ画像、ベクタファイル等を取り入れ、写真やグラフが表示できることを学習する。
10回	画像データの応用: パラパラ漫画の原理を活用して、アニメーション表示を学習する。さらに乱数やタイマーの概念を学習する。
11回	図形データの扱い: 画像や図形の代表的な処理(移動、回転、拡大、縮小)を学び、基本的な機能を組み合わせることにより、複雑な動きを実現する。
12回	関数: 関数の定義方法、引数、戻り値の関係について学習します。関数を適切に定義することにより、プログラムが簡潔に記述できることを学習する。
13回	配列: 1次元配列の活用について学習する。活用の一例として、1次元配列の要素をシフトする例を取り上げ、マウスを追跡する描画の例を学習する。
14回	プログラムコンテスト: これまで学習してきた例題をもとに、プログラムコンテストを実施する。
15回	期末試験

3.2 教科書

教科書は、オライリーの「Processingをはじめよう 第2版」を使っている。巻末にはクイックリファレンスも付いており、重宝している。この書籍は、「コンピュータプログラミング I」では、1章～9章、11章配列(オブジェクトを除く)までを学ぶ。残りの章は、同IIで学習する。授業内容は web サイトで公開している[6]。

3.3 授業の実施形態

授業は2クラスに分割し、2名の教授者が同時開講する。1クラスの受講者数は約60名で女子学生は約4割である。大学院生のTA(Teaching Assistant)は、各クラスに2名付く。教室の教卓側にはスクリーンが2面あり、書画カメラと教授者のノート PC の画面を表示しながら授業を行う。座席は、自由席である。教室は縦長になっており、3つの島に分かれている。各島は3人掛けのテーブルが17列並び、計153名収容できる。なお、中央の島にはACコンセントが無いので、両側の島に着席する受講者が多い。教室後方から撮影した教室の様子を図1に示す。



図1 教室の様子

3.4 授業で取り上げたプログラムの紹介(抜粋)

授業で取り上げたプログラムの実行結果の一部を紹介する。図2は、ベクトルを使った描画でマウスの操作によって画像の大きさを変更できる。拡大しても、画質が劣化しない。図3は、ロボット(Robot 7)を描画した結果である。教科書では、各章末に Robot 1 から Robot 10 が紹介されており、新しく学んだことをロボットの表示に取り入れている。図4は、フクロウを表示する owl 関数を定義したものである。学生さんから、「可愛い」と評判が良い。

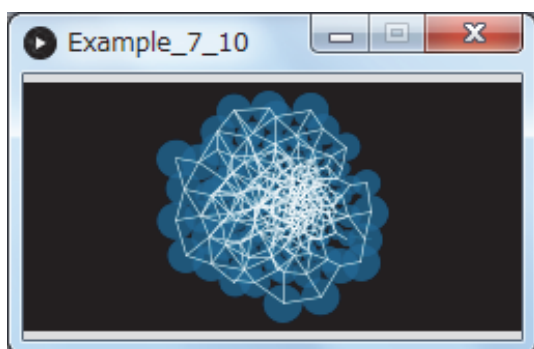


図2 ベクトルを使った描画

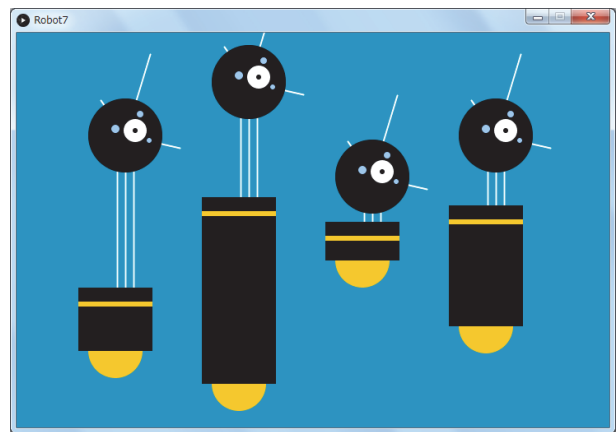


図3 ロボットの描画

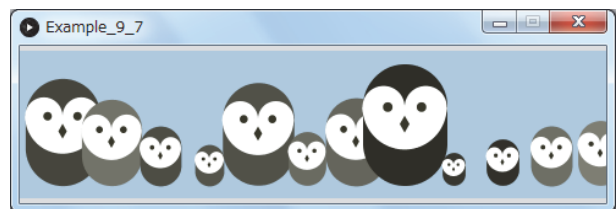


図4 フクロウの描画

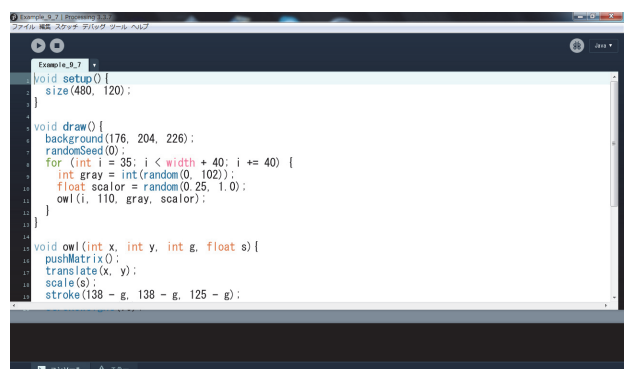


図5 Processingの統合環境

4. 教授の工夫

4.1 白紙の状態からプログラム作成

授業では、新規作成で白紙の状態からブロック構造に注意しながらプログラムを入力する。教授者は入力する部分を読み上げ、また何をしている部分なのかを補足する。この工夫は、Javaで教えていた時も効果的であった。受講者の様子を観察しながら入力できるため、授業の微妙な進行速度の調整が可能である。図5は、Processingの統合環境を示した。白い部分がテキストエディタで、入力と同時に文法チェック等の機能が働く。エラーが検出されると、テキストエディタの下にあるコンソールに表示され、直ぐに誤りを見付けられ大変ありがたい。

4.2 タイピングの苦手な受講者への対応

スケッチブックに絵を描く感覚でプログラミングはできるが、タイピングは遅い。タイピングが苦手な受講者が多いため、授業の前半は、様子を見ながらゆっくり進めた。逆に、タイピングが早い受講者、予習を行ってきた受講者は、教科書を見ながら先に進むように指導し、受講者間のバラツキを吸収する工夫を取り入れた。

4.3 プログラムコンテストの実施

受講者の興味を喚起させる目的で、プログラムコンテ

ストを導入した。テーマは自由とし、発表時間は1分/人である。作品の提出は、総合メディアセンターのファイルサーバへ提出する。提出されたファイルは、教授者のノートPCに取り出す。発表をスムーズに行うため、教授者のノートPCにワイヤレスマウスとキーボードを接続する工夫を行った。発表者用の机の上に、ワイヤレスマウスとキーボードを置き、受講者がこれを使って発表する。教卓で教授者のノートPCも操作できるため、発表者が入れ替わる間に、次の人の作品を表示する準備ができる。受講者のノートPCを繋ぎ変えながら発表すると切替えに時間がかかるが、このささいな工夫は、コンテストをスムーズに進行することに大変役立った。

5. 授業の理解度調査の実施

5.1 調査項目について

情報環境工学科でJavaを教えていた時の工夫を活用し、授業の理解度調査を目的としたアンケート調査を導入した。理解度調査項目は、表2に示す90項目である。項番は授業回数と設問の番号で構成した。例えば「2-1」は、2回目の設問1を示している。紙面の都合で、「は、理解できましたか。」の文言を省略した。

5.2 受講者へのフィードバック

アンケート調査結果は次の授業の冒頭で紹介し、フィードバックをかける。アンケート調査は、UNIPAで実施した。受講者はノートPCのwebブラウザを使って回答する。集計結果はCSV形式で取り出すことができる。さらに集計結果はグラフも表示できる。この集計結果はHTMLのファイルで作られているため、グラフ部分を残し、CSVで取り出した自由記述を加え、授業用のwebサーバで公開している。毎回のアンケート調査は手間が掛かるが、次の授業でフィードバックすることで、受講者相互のフィードバック、教授者へのフィードバックを行え、教授者の一方的な授業にならないために効果を発揮する。

5.3 理解度の推移

授業毎の理解度調査結果は、図6~図8に示す。横軸は表2に示した項番、縦軸は理解調査項目に対して「はい」と回答した割合(%)である。なお7回と8回は理解度調査を行っていない。図6は授業の前半部分である。2回は80%以上の理解が得られているが、3回は70%程度まで低下し、段々、難易度が上がっていることが窺える。

図7は授業の中盤部分である。「5-5」「6-1」「6-9」「9-4」の理解度が大きく低下していた。5回は「マウスによる入力」で、「5-5」はイーザングの原理である。イーザングは、滑らかに移動を行うテクニックであるが、その原理の理解とプログラムの理解は難しかった。6回は「条件分岐」で、「6-1」はboolean型である。int型やfloat型は、授業中の例で何度も出てくるが、boolean型は馴染みが少ないため理解が難しかった。「6-9」はmap関数である。この関数は、ある範囲から別の範囲に値を変換する働きがあるが、難しいことが分かった。9回は「画像データの扱い」で、「9-4」はpushMatrix関数である。平行移動や回転などの座標変換を行う場合、変換前の座標系を保存し、座標変換によって描画を行った後、再び、元の座標系に戻すことで、複雑な図形の描画を容易にする。座標変換等の前提知識が不足もしくは慣れていないため、理解が難しいと考えられる。これらの対策として、次年度は図に

よる補足資料を準備したい。

図8は授業の後半部分である。10回は70%、11回は50%、13回は40%を切る理解度となり、授業後半は初学者にとってかなり難しいことが示された。「11-3」「11-6」「13-5」の理解度が大きく低下している。11回は「図形データの扱い」である。「11-3」はトゥイーニングで、2点間の移動をスムーズに見せる手法であるが、その仕組みを理解し、プログラムを理解することが難しい。「11-6」

表2 授業の理解度調査内容

項番	質問項目	項番	質問項目	項番	質問項目
2-1	size 関数	3-7	float 型	10-4	PFont 型
2-2	point 関数	3-8	for 文	10-5	createFont 関数
2-3	座標系	3-9	関係演算子	10-6	textFont 関数
2-4	line 関数	4-1	二重の for 文(Example 4-10 ループにループを埋め込む)	10-7	testSize 関数
2-5	quad 関数	4-2	一重の for 文(Example 4-11 横と縦の列)	10-8	text 関数
2-6	triangle 関数	4-3	二重の for(Example 4-12 ビンと線)	10-9	String 型
2-7	rect 関数	4-4	二重の for 文(Example 4-13 網点)	10-10	PShape 型
2-8	ellipse 関数	5-1	setup 関数	10-11	loadShape 関数
2-9	arc 関数	5-2	draw 関数	10-12	createShape 関数
2-10	radians 関数	5-3	システム変数 mouseX, mouseY	10-13	shape 関数
2-11	描画の順序	5-4	システム変数 pmouseX, pmouseY	11-1	システム変数 frameRate
2-12	strokeWeight 関数	5-5	イーザングの原理	11-2	frameRate 関数
2-13	strokeCap 関数	5-6	if 文の働き	11-3	トゥイーニング
2-14	strokeJoin 関数	6-1	boolean 型	11-4	random 関数
2-15	ellipseMode 関数	6-2	円の内外側の判定方法	11-5	randomSeed 関数
2-16	background 関数	6-3	長方形の内外側の判定方法	11-6	constrain 関数
2-17	fill 関数	6-4	システム変数 keyPressed	11-7	millis 関数
2-18	stroke 関数	6-5	char 型	11-8	sin 関数
2-19	透明度の設定	6-6	システム変数 key	11-9	cos 関数
2-20	beginShape 関数	6-7	押されたキー検出する仕組み	12-1	サイコロを振る関数 rollDice
2-21	endShape 関数	6-8	カーソルキーで図形を動かす仕組み	12-2	int 関数
2-22	vertex 関数	6-9	map 関数	12-3	owl 関数
2-23	隙間を閉じる方法	9-1	translate 関数	12-4	パラメータ(引数: ひきすう)
2-24	コメント(注釈)	9-2	rotate 関数	12-5	値を返す関数
3-1	変数	9-3	scale 関数	12-6	return 文
3-2	代入	9-4	pushMatrix 関数	13-1	変数と配列の違い
3-3	四則演算	9-5	popMatrix 関数	13-2	インデックス(添字)
3-4	int 型の変数に浮動小数点数を代入できないこと	10-1	PImage 型	13-3	length
3-5	float 型の変数に整数を代入できること	10-2	loadImage 関数	13-4	配列の値のシフト
3-6	int 型	10-3	image 関数	13-5	nf 関数

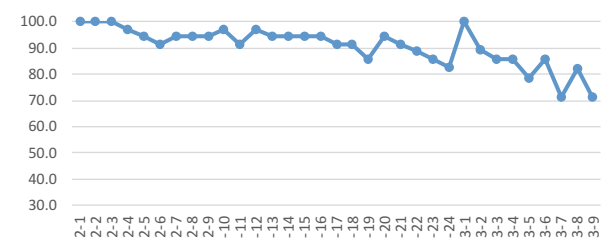


図6 理解度調査結果(2-1 から 3-9)

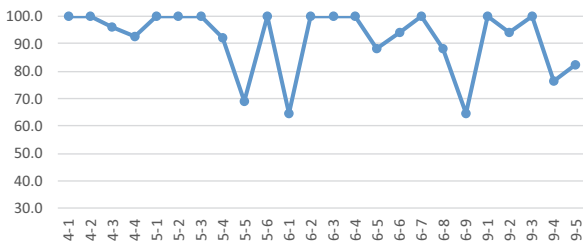


図 7 理解度調査結果(4-1 から 9-5)

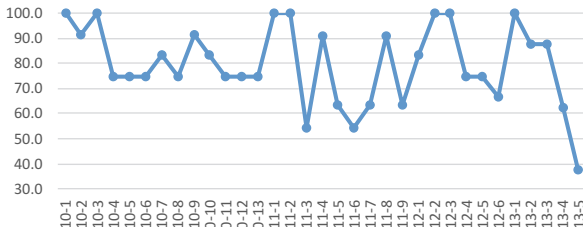


図 8 理解度調査結果(10-1 から 11-5)

表 3 アンケート調査項目

	設問内容
問1	あなたのこの授業への出席率ほどの程度でしたか。
問2	この授業について、予習・復習や課題など、今まで授業時間外に取り組んだ学習時間は、1週間あたり平均してどの程度でしたか。
問3	あなたは、シラバスを参照し、科目選択や予習・復習に役立てましたか。
問4	授業内容は、おぼろげにシラバスに書かれていた通りでしたか。
問5	授業内容の難易度はあなたに合っていましたか。
問6	授業の進捗はあなたに合っていましたか。
問7	話し方(声量やスピードなど)は聞き取りやすかったですか。
問8	専門知識や用語の説明は分かりやすかったですか。
問9	板書やスライド、視聴覚教材などは、授業の理解を助けるものでしたか。
問10	教科書や配布資料(デジタルコンテンツを含む)は役に立ちましたか。
問11	教員は授業に意欲的に取り組み、指導は熱心でしたか。
問12	教員は、教室の学びやすい雰囲気を作るように努めていましたか。
問13	この科目の内容について、興味と関心が深まりましたか。

表 4 問1の集計結果

回答	人数	割合
100%	26	60.5
90%以上100%未満	10	23.3
70%以上90%未満	6	14.0
50%以上70%未満	0	0.0
50%未満	1	2.3
未回答	0	0.0

表 5 問2の集計結果

回答	人数	割合
2時間以上	3	7.0
1時間以上2時間未満	8	18.6
30分以上1時間未満	13	30.2
30分未満	14	32.6
全くしなかった	5	11.6
未回答	0	0.0

表 6 問5の集計結果

回答	人数	割合
難しかった	4	9.3
やや難しかった	24	55.8
ちょうどよかった	14	32.6
やや易しかった	0	0.0
易しすぎた	1	2.3
未回答	0	0.0

表 7 問6の集計結果

回答	人数	割合
速すぎた	5	11.6
やや速かった	20	46.5
ちょうどよかった	17	39.5
やや遅かった	0	0.0
遅すぎた	1	2.3
未回答	0	0.0

表 8 問3～問4, 問7～問13の集計結果(%)

	問3	問4	問7	問8	問9	問10	問11	問12	問13
そう思う	11.6	20.9	46.5	30.2	34.9	34.9	55.8	41.9	30.2
ややそう思う	23.3	11.6	25.6	41.9	46.5	37.2	27.9	32.6	41.9
どちらともいえない	20.9	20.9	25.6	20.9	16.3	23.3	14.0	23.3	23.3
あまりそう思わない	2.3	0.0	0.0	4.7	0.0	2.3	0.0	0.0	0.0
全くそう思わない	7.0	2.3	2.3	2.3	2.3	2.3	2.3	2.3	4.7
未回答	34.9	44.2	0.0	0.0	0.0	0.0	0.0	0.0	0.0

は constrain 関数で、指定した最小値と最大値を超えないように値を設定する関数であるが、初学者にとって難し

いことが分かった。13回は「配列」である。「13-5」はnf関数で、数値を指定したフォーマットに変換する。教科書では、「frame-0000.png」から「frame-0011.png」の画像ファイルを順に読み込む時に、「0000」から「0011」の文字列を生成する例があり、不慣れた文字列の演算が難しい。

6. 本取組の評価

本取組の評価は、学期末に事務局が実施した授業アンケートの調査結果で示す。アンケート調査項目は、表3に示す13問である。回答項目が共通しない問1～問2, 問5～問6は、表4～表7に回答者数と割合を示した。回答項目が共通している問3～問4, 問7～問13は、表8に割合のみ示した。

表4から「90%以上の出席」は83.8%と高かった。表5から授業外の学習時間は、「1時間以上」は25.6%であり、全体的に少ない結果であった。表6から「やや難しい」以上の割合は65.1%であり、難しいと感じている人が半数を超えた。表7から「やや速かった」以上は58.1%であり、タイピングが遅い受講者の影響も窺える。

表8から、問3と問4はシラバスに対する回答である。問7は教授者の話し方で、問8は専門用語の説明の分かりやすさで、「ややそう思う」以上は共に72.1%であった。問9は視聴覚教材を尋ねたもので、「ややそう思う」以上は81.4%と高かった。問10は教科書と配布資料で、「ややそう思う」以上は72.1%であった。問11は教授者の指導の熱心さで、「ややそう思う」以上は83.7%であった。問12は学びやすい雰囲気で、「ややそう思う」以上は74.5%であった。問13は興味や関心が喚起されたかで、「ややそう思う」以上は72.1%であった。

7. おわりに

プログラミングを専門としないデザイン工学科に Processing を導入し、初学者がプログラミングを学ぶ時の敷居を低くでき、キャンバスに絵を描く感覚でプログラムを学べるようにできたことは大きな成果である。Processing を導入した初めての授業であったが、70%以上の受講者が興味や関心を持ったことに対して満足している。今後は、理解度が低かった部分の改善に努めたい。

参考文献

- (1) Processing を は じ め よ う <http://www.oreilly.co.jp/books/9784873115153/> 2018年6月15日閲覧
- (2) Processing <https://processing.org/> 2018年6月15日閲覧
- (3) 土肥紳一, 今野紀子: Processing による高校生を対象としたプログラミング入門体験2, 情報処理学会, 情報教育シンポジウム論文集, Vol.2014, no.2, pp.119-126, 2014.
- (4) 土肥紳一, 今野紀子: Processing による高校生を対象としたプログラミング入門体験, 情報処理学会, 情報教育シンポジウム論文集, Vol.2013, no.2, pp.217-224, 2013.
- (5) 三浦元喜: Processing Web IDE を用いたプログラミング基礎教育の試み, 情報処理学会, 情報教育シンポジウム SSS2013, Vol.2013, no.2, pp.225-231, 2013.
- (6) コンピュータプログラミング I, web サイト, <http://dohi.chiba.dendai.ac.jp/~dohi/computer-programming-1/ad/> 2018年6月15日閲覧