

LMS を活用した Processing 言語によるプログラミングの 学力考查の試行

土肥 紳一†

東京電機大学 システムデザイン工学部†

1. はじめに

システムデザイン工学部デザイン工学科は、プログラミング言語に Processing を採用した。この言語は、実行結果をグラフィックスで表示でき、お絵描き感覚でプログラミングを学べるように設計されている。さらにインタープリター方式のため、入力した瞬間に文法チェックが始まり、スペルミスなど初学者が陥りやすい誤りを素早く発見できる特徴がある。これまで学力考查は問題を PDF で閲覧し、できあがったプログラムをファイルサーバで回収し、人海戦術で採点を行って来た。本学の LMS は WebClass が導入され、授業毎のアンケート調査や、事前学習、事後学習等に活用してきた。2019 年度の後期科目である「コンピュータプログラミング I」の筆者のクラスで、試験の機能を使い、中間の学力考查を試行した。その結果について述べる。

2. LMS の活用

プログラミングの授業では、白紙の状態からプログラムを完成できることを目指している。完成したプログラムはファイルサーバに提出するが、提出先のフォルダを間違える、提出物はプログラムを保存したフォルダを求めているが、フォルダ中のファイルを提出するなど人的な誤りが生じていた。提出された答案は、人海戦術で採点を行うため採点に時間がかかる、採点のブレが生じる等の問題が発生していた。先行研究では穴埋め問題を用いたプログラミング教育支援ツール pgtracer がある [1]。

本学の LMS は WebClass が導入されている。先行研究の様な独自のツール等の開発は避け、WebClass の機能でどの程度実用的な試験ができるのかを探った。WebClass は、授業毎のアンケート調査や、事前学習、事後学習等に活用してきた経験がある。事前学習は、教科書を読んでおけば解ける程度の内容で、主に選択式の問題である。事後学習は、穴埋め式の問題になっており、解答欄に該当する文字列を入力する。受講者と教授者が WebClass に慣れて来たことから、テスト機能を使って中間の学力考查に活用した。

3. 問題の作成

WebClass では HTML または PDF で問題を作成できる。Processing の実行結果はグラフィックで表示することが一般的である。問作を効率よく行うために、WORD で作成し PDF に変換して公開するこ

Trial of an attainment test of computer programming in the processing language by using LMS

†Shinichi Dohi, School of System Design and Technology, Tokyo Denki University

とにした。中間の学力考查は 2019 年 10 月 25 日に、試験時間は 60 分で実施した。各自の教科書、ノート PC 等は参照可とした。座席は、3 人がけの机に、2 人以内で中央は空けて着席するように指示し、途中退室不可である。中間の学力考查は 3 問出題した。ここでは問題 1 を取り上げ、問作について述べる。この問題は図の幅(変数 w)と高さ(変数 h)に基づいて、図 1 に示すように図が変化するプログラムである。

問題 1 絵の表示

図 1 の絵を表示するプログラムを、条件を満足するように作成しなさい。なお、set up 関数と draw 関数は使いません。(ファイル名 Prog2019_Middle_Q1) 2.0 点満点

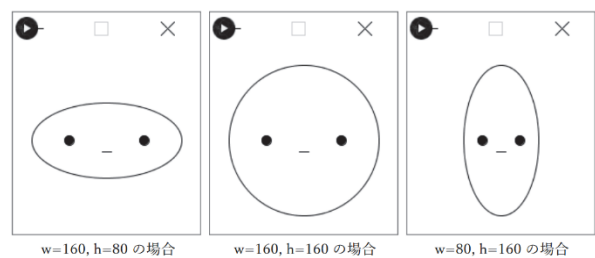


図 1 絵の表示例

図 1 図の変化

図 2 は図形を表示するための条件と図形の位置関係を示したものである。ここに示された仕様を満足するプログラムを完成する。

(条件)

- ① ウィンドウの大きさは、幅 200 ピクセル、高さ 200 ピクセルです。
- ② 大きな楕円の中心の X 座標は変数 x 、Y 座標は変数 y 、幅は変数 w 、高さは変数 h です。 w と h の初期値は、160 です。
- ③ 各々の楕円と線の位置関係は、図 2 に示す通りです。
- ④ 線は、最後に表示します。

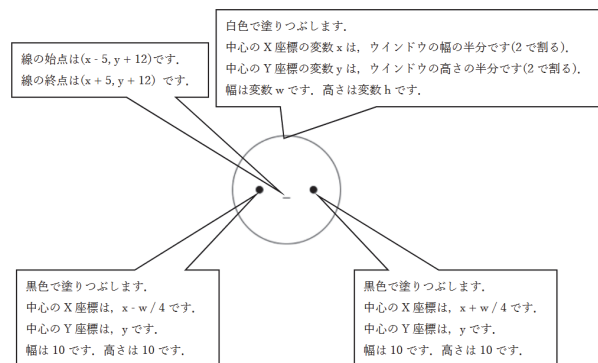


図 2 位置関係

図 2 図形の位置関係

図 3 は、穴埋めの問題部分を示した。紙面の都合

から2段組で表示したが、WebClassでは1段で表示される。図2の仕様を満足するように、(1)~(10)に適切な文字列をWebClassの解答欄に入力する。

(1)から(10)を解答欄に入力しなさい。	x = (4)
size((1));	y = (5)
int (2) = 160;	background(255);
int h = 160;	(6)
(3)	ellipse((7));
int y;	(8)
	(9)
	ellipse(x + w / 4, y, 10, 10);
	(10)

図3 穴埋めの問題部分

図4は、採点結果を表示したものである。教授者は学籍番号で個人を選択でき、右上に問題文、右下に回答リストと正解リストが表示される。回答リストには、正解したものは○が、不正解は赤色で×が表示され、正解リストと比較することにより、何を間違えたかが分かるようになっている。試験終了後、教授者がWebClassの設定を行うことで、受講者は各自の採点結果を閲覧できるようになる。

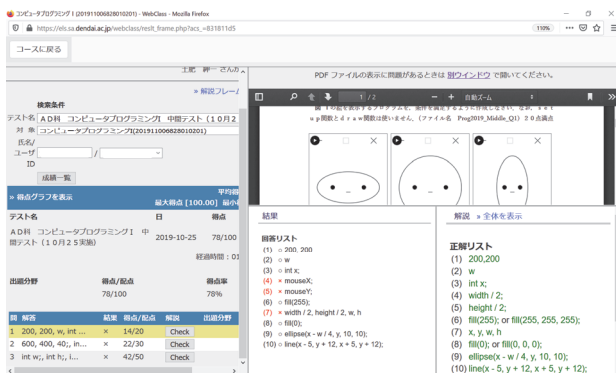


図4 採点結果の表示

4. 正解リスト作成時の問題点

穴埋め式の問題を出題し、問題点も見えてきた。問作時に想定した正解を文字列で入力するが、別解の存在がある。一度、想定した解答で採点を行った後、全ての解答を確認し、別解の存在を調べる。別解が発見された場合は、正解リストに追加し、再採点を行う。再採点は瞬時に完了するが、別解の確認は人海戦術となり、この時間が新たに必要になることが分かった。

別解の例であるが、穴埋め(6)は fill(255); 以外に fill(255, 255, 255); と記述することもできる。正解リストには fill(255); or fill(255, 255, 255); のように or で複数の文字列を入力して対応する。さらに実引数は16進数でも指定でき、これらの組み合わせを考えると、複雑な正解リストになる。事前にすべての組み合わせを入力することは難しく、採点后、別解の存在を発見した段階で、正解を追加し、再採点を行う必要がある。

別な例として、図3の(9)は ellipse(x - w / 4, y, 10, 10); が正解である。しかし、実引数の部分に着目すると x - w / 4 は -w / 4 + x と記述することもできる。つまり、交換法則が成立する式の場合は、これらを考慮して、複数の正解を登録しておかなければならない。同様に(10)は line(x - 5, y + 12, x + 5, y + 12); が正解であるが、実引数の x - 5 や y + 12 や x + 5 や y + 12 は、各々、交換法則が成立する。すべての組み合わせを正解リストに入れることは、非現実的である。変数は左側に書くなどの条件を設ける等、問題自体を工夫する必要がある。

5. 考察

本試みの最大のメリットは、採点が瞬時に完了することである。さらに採点結果は、試験終了後、受講者自身で確認でき、何を間違えたかが分かる。また採点のブレも生じない。

デメリットは、別解の存在への対応である。一例であるが int x;; のようにセミコロンを余分に付けることができる。無意味な ; は付けられないことが理想であるが、文法的には誤りではない。問題文に意味のない記述は減点になることを明記し、正解を絞る等の対策が必要である。

字下げ等の作法は重要であるが、半角と全角のスペースを無視する設定が前提となり、採点できない。その他、穴埋め式の解答は、受講者の転記ミスが発生した。問題の仕様を満足するプログラムを Processing で完成し、正しい実行結果を確認した後、WebClass の解答欄に入力する。この時 ; の付け忘れ等の転記ミスが見受けられた。受講者がこのような試験環境に慣れることも必要である。さらにプログラムの実行テストを行った後、解答欄に転記する必要があり、時間のロスが発生する。慣れてくると、プログラムの実行テストを行わなくても、前後関係から正解を推測できるようになる。

授業毎に実施している理解度調査の中で、授業に対する要望感想をアンケート調査している。受講者からは以下の指摘があった。

- ・(1)の表記は、どの部分が穴埋めなのか、プログラムに埋もれて分かりにくい。
- ・解答欄に貼り付けるとき、文字の欠落が生じることがある。
- ・解答欄に入力したプログラムを実行できない。
- ・慣れないと、解答しにくい。

6. まとめ

プログラミングは、与えられた問題に対して、自由な発想でプログラムを作ることが理想である。白紙の状態から作成したプログラムを自動採点できることは理想であるが、WebClass にこのような機能は無い。既存の機能で、穴埋め問題を作成することにより、解答の制約は生じるが、実用的なレベルで中間の学力考査を行えた。今後は、LMS の制約を考慮しながら、良問を出題したい。

参考文献

- [1]柳田峻, 太田康介, 大槻美佳, 掛下哲郎: 穴埋め問題を用いたプログラミング教育支援ツール pptracer の運用実験, 情報処理学会, SSS2014, pp.135-142