

プログラミングの入門科目と専門科目の授業設計

久保 守†

金沢大学†

はじめに：

近年、数理およびデータサイエンスの重要性が急速に高まっていることに対して、大学においては、その知識と技術を習得する人材育成が求められており、Python 言語を使用した演習が各地の大学で実施されている。

一方、金沢大学では、2008 年度に学域・学類制が導入され、2018 年度には機械・電子情報・化学工学の分野を融合したフロンティア工学類が創設された。また、2006 年度からノート PC の全学生必携化が実施された。2019 年 4 月からフロンティア工学類 2 年生の専門科目を始めるに当たり、必携 PC を活用したプログラミングの演習科目を授業設計することとなった。今年度は分野共通の基礎科目としてプログラミング演習、電子情報分野の専門科目として電子情報基礎実験、パターン認識を実施した。以前の電子情報学類では、それぞれの授業で、C 言語、MATLAB、R 言語など複数のプログラミング言語で実施されていたが、今回は、プログラミング言語の中でも汎用性が高い Python 言語を共通に使用することとし、3 科目の授業設計を行った。

また、学生が主体的に授業に取り組む、アクティブラーニングの反転学習やグループワークが盛んに導入されている。事前に内容を予習し、授業中は知識の定着や応用力の育成を重視して行うのだが、分かりやすいスライド資料、スマホや e ラーニングの普及の影響で、ノートを取らずに理解しようとする「見て覚える」学生が多く、「書いて考える」過程が失われており、読解力の低下にもつながっている。この授業設計では、プログラミングの入門科目と専門科目を通じて「書いて考える」ことで理解を深め読解力を高めることを重視する。

入門科目：

プログラミング演習の目標はプログラムの作成と理解であり、全 16 回の内容は以下の通りである。

- 変数とデータ型
- 演算子
- if, else
- for, while
- list, dict
- 関数、引数、戻り値
- tkinter
- ゲームプログラミング
- 期末試験

プログラミング教育は、職業としてのソフトウェア開発者を目指すだけでなく、その過程で試行錯誤することにより、論理的思考力や問題解決能力を養う点も注目されている。しかし、「習ったのに書けない」ということをよく耳にする。これは、説明を受けた例題プログラムをそのまま打ち込んで動かす、理解していない状態で課題を提出する、いわゆる写経の弊害が大きい。

そこで、苦手な学生は最低限の作成と理解度、できる学生は高度な作成と理解度を達成できるように授業設計した。まず、1 回の題材の文法や機能をできる限り絞り込んで、簡単な例題を説明し、それを動かして動作を確認する。使用する統合開発環境にはステップ実行機能があり、コードおよび変数の追跡が頭の中で十分できない学生に推奨した。

†Instructional design of introductory and advanced course for programming

†Kubo Mamoru · Kanazawa University

課題プログラムは、例題の一部を変更する簡単なものから難しいものまで、難易度が高いほど高得点として3種類提示した。学生は簡単な課題から順に取り組み、1つを選んで提出する。また、最終課題の題材をゲームとして、苦手な学生の動機づけとした。提出するプログラムは、理解度を確認するために、各コードの意味をコメントとして記載することとした。期末試験は、文法などの知識問題やプログラムの穴埋め問題ではなく、コードや変数を追跡する演習プリントを事前に実施し、類似問題を出題した。成績評価は、提出課題と期末試験との合計で行い、前者でプログラム作成能力、後方で理解度および読解力を評価する。

専門科目：

前期のプログラミング演習に続いて、後期に電子情報基礎実験とパターン認識の専門科目があり、Python言語と必携ノートPCを使用する。

実験ではRaspberry Piを使って以下のPythonプログラミングを行う。

- LED出力・スイッチ入力、
- 距離センサ入力、モータ出力制御
- カメラ画像処理

一般的な実験では、提示された手順書を見ながら取り組むが、本実際では理解度を深めるために、実際に取り組む前に手順書から各自の実験ノートに内容を整理して手書きで写し、実験中は結果と気づきを記載して、終了時にノートを点検する。これによって実験内容の理解と読解力を育成し評価する。

パターン認識では以下のPythonモジュールと題材を扱う。

- pandas/データ解析
- matplotlib/グラフ可視化
- scikit-learn/機械学習
- keras(tensorflow)/深層学習

毎回、パターン認識の手法を一つ取り上げ、その原理と例題プログラムを説明し、学生は例題の動作確認後、課題に取り組み、プログラムを提出する。理解度の確認のために、手計算の例題も実施し、類似問題を期末試験で出題する。

まとめ：

今年度に新たに開講した3つのプログラミングの入門科目と専門科目について授業設計した。授業中、学生同士で教え合って理解を深めているが、質問の対応などは十分でなく、今後の課題である。次年度は、画像処理と制御工学の3年生の専門科目でPython言語を使ったプログラミング演習が予定されている。また、全学の1年生導入科目としてデータサイエンス基礎が計画されており、プログラミング演習科目の系統的な授業設計および実践と改善を継続して実施していきたい。

参考文献：

特集：プログラミング入門をどうするか、情報処理 Vol. 57 No. 4 (2016).

特集：フレッシュマンに向けたプログラミングのススメ、情報処理 Vol. 60 No. 6 (2019).