











方法などの解説を記載することが必要であると考えられる。

- (2) 授業中にエラーの読み方を解説する  
テーマによって発生する可能性のあるエラーに偏りがあることが分かったため、必要に応じて、対象テーマの中で発生し易いエラーに関する注意を実施すると効果が高いことが予想される。

## 6.2 エラーのフィードバック方法

エラー解析の活用方法として、上記以外に直接学生にフィードバックする方法が考えられる。例えば、学生ごとのエラーの発生数を分析し、エラーの多少や種別などを通知することも技術的に可能である。

しかし、エラーの多少は学生がプログラミングを理解しているかどうかに関係があるかどうか、明らかではない。安易なフィードバックは学生を萎縮させる可能性がある。むしろ、積極的にエラーを出してプログラムのコツを掴むようなことを推奨したい。

また、明らかに課題の一部のセルでエラーを出し続け、進捗が止まっているようなケースを検出し、支援を行うような方法が考えられる。

## 7. 関連研究

近年 C/C++ や Java に代わり Python を初等プログラミング教育に利用する事例が報告されている。Alzahrani ら<sup>7)</sup>によると従来の C++ のコースと Python のそれとの比較では、Python の方が顕著に「苦労が多い」との報告がある。Python の方が学習が容易であると考えられていたこともあり、事前の予想とは異なる結果である。このため、更に分析が必要であると考えられる。

Liu らは参考文献 8 において Python のエラーメッセージを改良し、統計的な情報を用いて共通的なエラーの解説をつけることで、学習を助けるシステムを提案している。本取り組みでも学生の演習のエラーに関する多量のデータが得られており、それらを活用することで、有益な情報を学生に提供できる可能性があると考えられる。

## 8. まとめと今後の課題

本稿では、Jupyter Notebook 環境を活用してプログラミングの授業を実施した結果を報告した。600 人規模の授業を実施し、有益なデータを収集することができたが、本稿では、Python 言語処理系の出す

エラーに注目し、その分析結果を報告した。

よくあるエラーについては、本分析で得られた結果に基づき、教科書を改定しその中で解説する予定である。また、授業でもエラーの読み方について説明する予定である。

ここで述べたデータの解析は、授業実施中に得られるため、その場で解析することで学生のフィードバックすることが可能である。今後の課題として、授業中にエラーの傾向を解析し、タイムリーに学生にフィードバックする方法を検討したい。

本研究は JSPS 科研費 (JP18K11561) の「クラウドを活用したプログラミング演習環境に関する研究」の助成を受けたものである。

## A. 参考文献

1. Project Jupyter, Project Jupyter Homepage, <http://jupyter.org/> (2020/2/25 参照)
2. 桑田喜隆, 石坂 徹, 政谷好伸, 長久勝, 横山重俊, 浜元信州, Jupyter Notebook の実行履歴を活用したプログラミング演習の状況把握, 第 24 回人工知能学会 知識流通ネットワーク研究会, 2019 年 3 月 8 日
3. 桑田喜隆, 石坂 徹, 小川祐紀雄, 政谷好伸, 長久勝, 横山重俊, 浜元信州, Jupyter Notebook を使ったプログラミング模擬演習の評価, 第 25 回人工知能学会 知識流通ネットワーク研究会, 2019 年 9 月 13 日
4. 長久勝, 政谷好伸, 合田, 憲人, Notebook による講義・演習環境の開発, 情報処理学会 教育学習支援情報システム研究会報告 2019-CLE-27, 2019 年 3 月 13 日
5. Literate Computing for Reproducible Infrastructure, <https://literate-computing.github.io/> (2020/2/25 参照)
6. 室蘭工業大学情報教育研究会編, Jupyter Nortenotebook で始めるプログラミング 2019, 学術図書出版社, 2019 年 9 月
7. Nabeel Alzahrani, Frank Vahid, Alex Edgcomb, Kevin Nguyen and Roman Lysecky, "Python Versus C++: An Analysis of Student Struggle on Small Coding Exercises in Introductory Programming Courses", ACM SIGCSE'18, February 21-24, 2018
8. David Liu and Andrew Petersen, "Static Analyses in Python Programming Courses", ACM SIGCSE '19, February 27-March 2, 2019
9. Python Software Foundation, Python 3.7 組み込み例外, <https://docs.python.org/ja/3.7/library/exceptions.html> (2020/2/25 参照)

※ 記載されている会社名, 商品名, 又はサービス名は, 各社の商標又は登録商標です。