

## 「プログラミングコンテストの解答を利用した同一アルゴリズムの別実装の比較」

私は情報科学研究科に進学し、同一アルゴリズムを複数人がプログラムで実装した際の、それぞれのコードの優位性を評価し定量化する方法を研究したいと考えています。私は会社のインターンや学外のプロジェクトを通して、何度か多人数によるシステム開発の経験を経てきましたが、他人のコードが動かなかったり、読みにくかったりする問題がありました。多人数で開発する際にプログラマが協調して開発を進められるようなGitなどのメソッドやツールが普及し、広く利用されていますが、そのコード自体に何らかの欠陥が存在すれば、開発を妨げる要因になり得ます。

先行研究の堤さんはプログラミングコンテストの初心者と上級者の間の違いに着目し調査を行っていました[1]。しかし、同一の問題を解くアルゴリズム実装間の調査は行えていません。ある同一アルゴリズムにおけるプログラムの実行時間や可読性などの優位性を一律に評価出来たら、開発における局所的な場面でのコードの評価にも応用できると考えています。開発のリファクタリングの工程において、コードの合理性や可読性は非常に重要な要素になります。言うまでもなく、読みやすく無駄がないコードは他人からも理解されやすく保守性が高くなり、修正の工数が削減できると考えます。

具体的なアプローチとしては、競技プログラミングの問題に対する回答のソースコードのデータセットを元に、同一アルゴリズム(出題された同一の問題)における各コードの比較を行い、優位性のあるコードの特徴量はどこで抽出できるかの研究を行います。ある問題を解くために書かれたコード間において、実行時間や使用メモリ、プログラムの構造等で比較して優位性を持つコードの特徴をそれぞれのプログラミングコンテストの問題毎に調査し、一律に評価できる特徴量の特定を行います。また、使用するメトリクスについても調査対象とし、普遍的にコードの優位性評価に用いることが出来るメトリクスの特定も行いたいと考えています。

以上のアプローチを用いることにより、局所的なコードの優位性の評価が可能になります。研究対象が競技プログラミングのコードになるので、システム全体を加味したコードの評価はできませんが、コードの評価の第一の方法としてリファクタリングなどが容易になり、コードの質の向上が期待できます。そのために、特徴量と使用メトリクスについて検討と比較を重ねて、手法を確立させたいと考えています。

[1] 堤 祥吾, プログラミングコンテスト初級者・上級者間におけるソースコード特徴量の比較

