ソースコード読解教材におけるリファクタリング原則の適用に関する研究

A Study on Utilizing Refactoring Principles to Source Code Reading Materials

前田 暉正[*1], 松本 慎平 [1]
Terumasa MAETA[*1], Shimpei MATSUMOTO[*1]
[*1] 広島工業大学情報学部
[*1] Faculty of Applied Information Science, Hiroshima Institute of Technology
Email: {bl18106, s.matsumoto.gk}@cc.it-hiroshima.ac.jp

あらまし: 大学など教育現場においては，リファクタリングの原則を十分に教授できていない．事実，java などオブジェクト指向の基礎を学ぶ講義の中では，できる限りメソッド分割を行わないソースコードが教材の中で用いられている．これは，学習者に余計な負荷をかけないように配慮された結果である．一方，このような配慮は，ソースコードの設計論やデザインパターンに意識を向ける機会を奪っていることも事実である．この点について，メソッドに関して，学習者が許容可能な水準まで設計(分割)可能ならば，そのようなソースコードを教材に採用することは理想的だといえる．そこで本研究では，リファクタリング原則の中からメソッド分割に着目し(本稿では，メソッド分割のみに焦点を当てたソースコード記述の改善のことを「リファクタリング」と記述する)，メトリクスや認知負荷の観点から，プログラミング学習中に教材として適切だと考えられる論理的構造の水準を明らかにすることを目的とする．
キーワード: プログラミング，リファクタリング，メソッド分割，Java

1. はじめに

大学など教育現場においては，リファクタリングの原則を十分に教授できていない．その理由としていくつかの要因が考えられる．ひとつは，時間的制約上，リファクタリング原則を教授する段階まで到達できない点である．つぎに，リファクタリング技法を適用したソースコードは高度な読解力が必要だと考えられているため，学習教材として不適切だと見なされている点である．事実，java などオブジェクト指向の基礎を学ぶ講義の中では，できる限りメソッド分割を行わないソースコードが教材の中で用いられている．これは，学習者に余計な負荷をかけないように配慮された結果である．一方，このような配慮は，ソースコードの設計論やデザインパターンに意識を向ける機会を奪っていることも事実である．この点について，メソッドに関して，学習者が許容可能な水準まで設計(分割)可能ならば，そのようなソースコードを教材に採用することは理想的だといえる．そこで本研究では，リファクタリング原則の中からメソッド分割に着目し(本稿では，メソッド分割のみに焦点を当てたソースコード記述の改善のことを「リファクタリング」と記述する)，メトリクスや認知負荷の観点から，プログラミング学習中に教材として適切だと考えられる論理的構造の水準を明らかにすることを目的とする．

2. リファクタリング原則に着目する意義

ソフトウェアライフサイクルにおいて，保守作業量が占める割合は非常に高い．保守の全行程の中でも特に，ソースコードの内容理解(読解)は最も時間的コストの高い作業と言われている(1)．よって，ソースコードの内容理解に関するコストを少しでも低減するため，開発現場ではプログラムの振舞いを変えずに内部構造を改善するリファクタリングという方法論が適用されている．リファクタリングは，ソースコードの保守性，すなわち可読性の向上に有効とされている．その結果として，保守作業割合の効率化に繋がると言われている．よって，プログラミング学習段階からリファクタリングの原則を教授することが理想だと考えられるが，その妥当性については十分に議論されていない．

3. 提案

本研究では，複数段階にリファクタリングされた教材を学習者に提示し，正答率や読解速度といった量的データと，ソースコードの困難度との関連性を明らかにすることが主な課題となる．本提案の概要を図 1 に示す．ソースコードの困難度は，ソースコードの理解性・可読性から算出する．なお，ソースコードの理解性・可読性に厳密な定義は存在しない．Buse と Weimer は，テキストの理解のしやすさを可読性（Readability）と定義した上で，ソースコードの可読性を測定するため，識別子，特定の記号，インデントや空行などのフォーマット，コメントといったソースコード上の様々な特徴との相関について調査を行った(2)．Buse と Weimer の調査結果では，変数が定義されてから参照されるまでの間に多くの処理を含む場合，理解するためのコストが増大するという報告があった(3)．このコストを評価する方法として，石黒らは仮想メンタルシミュレーションモデル(以降，VMSM)を用いてテキスト理解のしやすさを評価した(4)．本研究では VMSM を用いて，理解速度と記憶能力コストの ASSIGN と RCL，再計算コス

トの BT_CONST，SUM_UPD，VAR_UPD の 5 つを理解容易性尺度として可読性を量的に定義する．

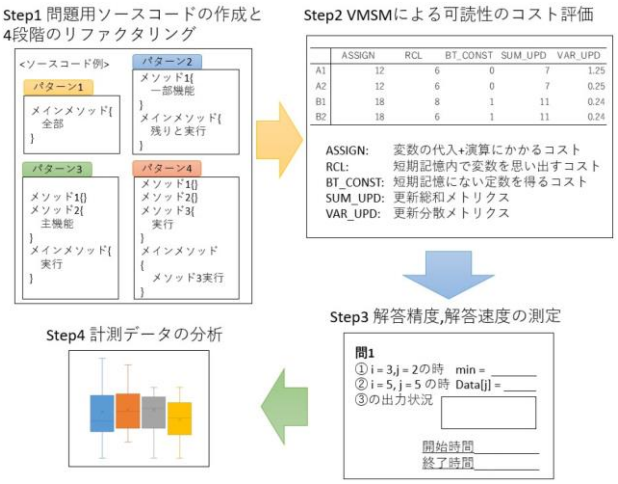

図 1 本研究の取り組みの概要

4. 実験方法

実験方法は次の通りとする．まず，外的振舞いが同一のソースコードを 4 種類 4 段階で作成(図 1 の Step1 参照)し，それぞれ VMSM でコストを算出する．本研究では，リファクタリングが特に重要となるオブジェクト指向型言語を対象とし，被験者が習得済みの Java を実験に用いる．問題種別として，並び替え，探索，演算 2 種の 4 種類の問題を以下 4 段階のソースコードで用意する．

- [第 1 段階]　リファクタリングなし
- [第 2 段階]　出力・初期化処理メソッド追加
- [第 3 段階]　メインメソッドが実行処理のみ
- [第 4 段階]　実行メソッド追加

次に，各ソースコードの内容理解に要する時間と解答精度を調査するための実験を行う．情報学を専攻するプログラミング，オブジェクト指向の基礎を習得済みの大学 3，4 年生を対象とし，ソースコードの特定条件下における一部の変数の値や最終結果の出力状況などを解く記述形式の実験を行う(図 1 の Step3 参照)．なお，問一つごとに 20 分の解答時間を目安とし，4 問出題する．実験後に行う認知負荷量や VMSM による可読性との関係を分析することで，教材として適切なリファクタリング水準を明らかにする．

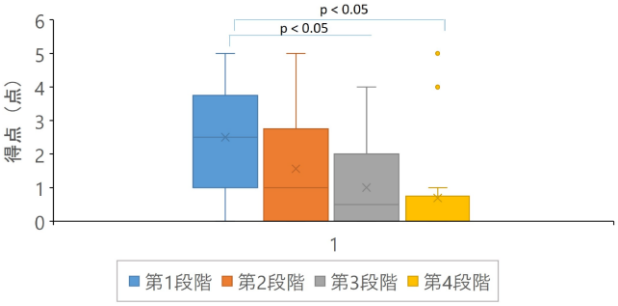

図 2 実験結果

表 1　理解容易性尺度

	ASSIGN	RCL	BT_CONST	SUM_UPD	VAR_UPD
Lv1	398	140	85	173	.0062
Lv2	454	141	112	201	.0076
Lv3	584	156	163	265	.0089
Lv4	609	156	175	279	.0092

5. 実験及び評価

図 2 に得られた結果を示す．これは段階別の得点の分布を示している．図 2 の結果からは，第 1 段階が最も正答が多く，メソッド分割を多く行ったソースコードほど正答が少なくなるという結果が得られた．また，Bonferroni 法を用いた多重比較検定より，第 1 段階と第 3 段階，第 1 段階と第 4 段階において有意差が見られた．第 2 段階のみ他の段階と有意差が見られなかったことについては，第 2 段階で扱われたメソッドが初期化と出力処理であり，変数の更新に直接関わらなかったことにあると考えられる．なお，表 1 は VMSM によって理解容易性尺度を計算した結果である．表 1 の結果から，リファクタリングの段階が進むにつれて理解のためのコストが大きくなっていることを確認できる．

6. おわりに

本研究では，VMSM によるコスト評価を用いてリファクタリング水準を明確にした上で，メソッド分割されたプログラミング教育向け教材の適切な水準を明らかにした．実験の結果では，現行の教育現場においてはメソッド分割されていないソースコードが最も教材として適していることが明らかになった．

謝辞

本研究は，独立行政法人日本学術振興会科学研究費助成事業(基盤研究(C)20K0319，22K02815)の助成を受けて実施した成果の一部である．

参考文献

(1) Ko, A. J., Myers, B. A., Coblenz, M. J., & Aung, H. H. (2006). An exploratory study of how developers seek, relate, and collect relevant information during software maintenance tasks. IEEE Transactions on software engineering, 32(12), 971-987.
(2) Buse, R. P., & Weimer, W. R. (2009). Learning a metric for code readability. IEEE Transactions on software engineering, 36(4), 546-558.
(3) Nakamura, M., Monden, A., Itoh, T., Matsumoto, K. I., Kanzaki, Y., & Satoh, H. (2004, September). Queue-based cost evaluation of mental simulation process in program comprehension. In Proceedings. 5th International Workshop on Enterprise Networking and Computing in Healthcare Industry (IEEE Cat. No. 03EX717) (pp. 351-360). IEEE.
(4) 石黒誉久，井垣宏，中村匡秀，門田暁人，& 松本健一. (2004). 変数更新の回数と分散に基づくプログラムのメンタルシミュレーションコスト評価. 電子情報通信学会技術研究報告; 信学技報, 104(466), 37-42.