

オープンソースを用いた C 言語記述スタイルの統計分析の試み

宮島 明寛, 松本章代, Martin J. Dürst

青山学院大学 理工学部 情報テクノロジー学科

1 はじめに

大学のプログラミング教育において、重要視されているのは意図どおり稼働するかという点であり、可読性は軽視されがちである。しかし、プログラムを作成する場合、後に追加や書き換えなど編集する作業が多数生じる。また、既存のプログラムの保守や、アップグレード、デバッグにかかる時間は新規プログラムの作成にかかる時間よりも多くなってしまうことがある。そのため、これらをいかに効率よく解決するかが、プログラミングにおいて非常に重要である。すなわち、プログラミング教育においても文法やアルゴリズムの習得だけでなく、可読性の高いソースコードの記述を習慣づける必要がある。

我々はこれまで、学生のプログラムの可読性チェックを目的として、スペースの空け方や改行の入れ方、インデントなどについて解析を行うシステムを開発してきた [1, 2]。今回我々は、オープンソースと学生が記述したソースファイルのプロジェクトを対象として、コメント文や空白類が実際にどのように扱われているのかについて実態調査を行う。その結果から相違点を学習者に示し、文法やアルゴリズムだけでなく記述スタイルの重要性について認識させることを目的とする。

2 関連研究

2.1 プログラミングスタイルの診断

関本らは、C 言語のプログラム中からプログラミングスタイルに違反する箇所を検出する診断システムを提案している [3]。例えば、if 文の条件式に代入の '=' を入れてしまうケースや、case 文の break 忘れなど、初心者が犯しがちな間違いを指摘することが可能である。本研究では、構文に関する可読性よりも、空白類等による見た目に重点を置いている。

2.2 コメント・空白・変数名の割合の調査

Spinellis は様々なプログラムの構成について調査を行っている [4]。コメント、空白、変数名の長さは C 言語においてコンパイラに通す際、結果に影響を与えないが、Spinellis の調査によるとこれらがプログラムの

An Attempt to Statistically Analyse C Programming Language Formatting Style based on Open Source
Akihiro MIYAJIMA, Akiyo MATSUMOTO and Martin J. DÜRST
Department of Integrated Information Technology, College of Science and Engineering, Aoyama Gakuin University
5-10-1 Fuchinobe, Sagamihara, Kanagawa 229-8558, Japan
akihiro@sw.it.aoyama.ac.jp, {akiyo, duerst}@it.aoyama.ac.jp

K&R スタイル	GNU スタイル
<pre>void style (...) { if (a > b) { c = a; } else { c = b; } }</pre>	<pre>void style (...) { if (a > b) { c = a; } else { c = b; } }</pre>

図 1: C 言語の一般的なスタイル例

中で占める割合はほとんどのプロジェクトで 50% を超えている。本研究では、空白類やコメントの量だけではなく、プロジェクト内で記述スタイルが統一されているかどうかの視点を持って調査を行う。

3 プログラミングスタイル

3.1 インデント

インデントは空白を用いてプログラムの構造を見やすく伝えるために行い、特に条件分岐やループなどの制御構造を明示するために使用される。一般的な字下げスタイルには K&R スタイルや GNU スタイル等がある (図 1 参照)。C 言語では特に定まった字下げスタイルは存在しないが、1つのプロジェクト内ではスタイルを統一することが望ましい。

3.2 コメント

プログラムを記述する場合には、何をしたいのかが明確に把握できていなければならない。コメントを用いて、わかりやすく読みやすい言葉でその考えを書き記すことにより、可読性が高まる [5]。

4 実態調査

4.1 調査対象

- オープンソース
プログラム言語の実装や web ブラウザ、サーバなどのオープンソースプロジェクトをソフトウェア開発のためのリポジトリサイト¹などからダウンロードする。計 124 のプロジェクトにおける 65,794 本のソースファイルを用いる。
- 学生が記述したソースプログラム
本学の授業で 2 年生 75 名が作成したソースプロ

¹sourceforge.jp: <http://sourceforge.jp/>

表 1: #include 後のスペース量

	0 個	1 個	2 個以上
オープンソース	0.9%	98.7%	0.4%
学生のソース	33.6%	65.6%	0.8%

表 2: if, for, while, swich 後のスペース量

	0 個	1 個	2 個以上
オープンソース	8.2%	91.5%	0.4%
学生のソース	92.1%	7.9%	0.0%

グラムを使用する。内容は 10~180 行程度のもの
で、計 4,697 本のソースファイルを用いる。

上記のソースファイルを対象に、オープンソースは
プロジェクトごとに、学生は個人ごとに集計を行う。

4.2 調査項目

調査項目は次の通りである。

- #include と<の間のスペース量
- if, for, while, swich と (の間のスペース量
- K&R もしくは GNU スタイルで統一されているか
- インデントがスペースまたはタブで統一されているか
- コメントが占める割合

5 結果・考察

- #include, if, for, while, swich 後のスペース量 (表 1, 表 2)
オープンソースではスペースが 1 個以上挿入されている場合が 90% 以上であるのに対して、学生のソースファイルでは#include の場合は 3 割程度、if, for, while, swich の場合は 9 割以上でスペースが挿入されていない。日本語は文章中にスペースを入れないため、日本人が書くプログラムはスペースが不足しがちとなる傾向にあると考えられる。
- K&R と GNU スタイルの比較 (図 2)
学生は比較的記述スタイルを統一していることがわかる。これは、担当教員や教科書が K&R を使用しているためだと考えられる。一方オープンソースでは統一されていないプロジェクトが多数存在する。これは、オープンソースでは多人数によって開発されることが多いためであると考えられる。実際、オープンソースではプロジェクトごとではなくファイルごとに統一されていることが多かった。
- インデントにおける空白類の比較 (図 3)
学生はほとんどタブを用いていることがわかる。一方オープンソースではスペースを使用している割合が高い。これは多人数で開発を行うため、様々な環境で扱いやすいスペースが多く用いられていると考えられる。
- コメントが占める割合 (表 3)

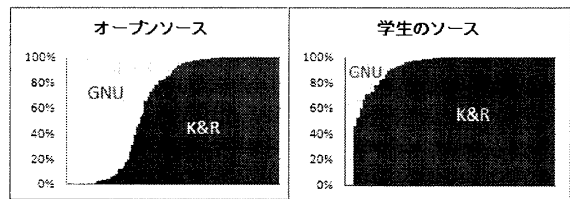


図 2: K&R と GNU スタイルの比較

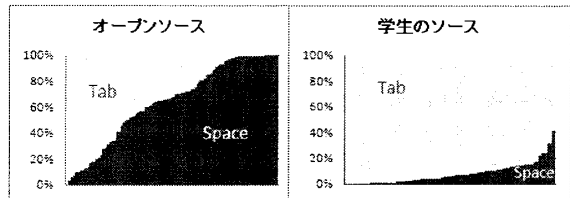


図 3: インデントにおける空白類の比較

表 3: コメントが占める割合

	割合
オープンソース	22.8%
学生のソース	2.7%

学生はほとんどコメントを挿入しないことがわかる。授業内容が基礎的な内容であるため、コメントを用いて内容を把握する必要性はないかもしれないが、学習する段階で、コメントを挿入するように促す必要がある。

6 まとめ

本研究ではオープンソースと本学の学生が記述したソースファイルを用いて空白類やコメントを用いた可読性についての調査を行った。今後はこの結果に基づき学生の可読性向上を支援するシステムの構築を目指す。

参考文献

- [1] 中島正登, 伊藤一成, Martin J. Dürst. C 言語プログラミングにおけるフォーマット特製の抽出. 情報処理学会第 69 回全国大会, 2007.
- [2] 池谷武, 伊藤一成, Martin J. Dürst. フォーマットの診断によるプログラミングスタイル支援システムの提案. 情報処理学会第 68 回全国大会, 2006.
- [3] 関本理佳, 海尻賢二. プログラミングスタイルの診断システムの構築. 教育システム情報学会誌, Vol. 17, No. 1, pp. 21-29, 2000.
- [4] Diomidis Spinellis. The way we program. *IEEE Software*, Vol. 25, No. 4, pp. 89-91, 2008.
- [5] Steve Oualline. C 実践プログラミング. オライリー・ジャパン, 1998.