

内部コード系と識別子の違いに着目した日本語プログラムの可読性の評価

4K-5

平田篤志, 早川栄一, 並木美太郎, 高橋延匡

(東京農工大学工学部電子情報工学科)

1. はじめに

プログラミングの対象が地域的文化的な問題に広がると、対象や処理を記述するのに日本語が使えないと不便である。このようなことから、日本語で制限なくプログラミングできる環境が必要である。

その実現には、計算機システムに次の拡張をしなければならない。

- (1) 計算機の内部コード系をフル 2 バイトコード系とする
- (2) プログラム中の識別子名の記述に日本語文字の使用を可能にする

プログラムの識別子名の記述に日本語ができれば、記述しやすさの面で次のような効果が期待できる。

- ・英語で識別子名を考える必要がなくなるので、プログラミングの際に思考の中断がなされなくなる
- ・仕様書中のキーワードをそのまま識別子名として記述できるため、仕様書とプログラムとの対応がよくなり保守がしやすくなる

一方、読みやすさについての効果は、過去に紙上での読解実験により、プログラムの識別子に用いる言語として、英語よりも日本語を用いた方が可読性が高いことが確認されている。

これに対し本稿では、日本語識別子を用いることの有効性について、実際のプログラミング行動のなかでの作業能率を評価する必要性から、デバッグ作業効率・機能拡張作業効率からみた評価実験をマシン上で行なった。

また、従来の計算機システムにおける 1,2 バイト混在コード系とのコード系の違いによる、プログラムの読みやすさの違いについての評価も行なった。

混在コード系では文字列のバイト数が文字数に比例しないため、文字の内容をチェックしながら文字列処理を行なう必要がある。したがって、2 バイト固定長コード系におけるプログラムの方が読みやすいと考え、実験を行なった。

2. 実験の流れ

実験の流れの概要は次の通りである。

実験用プログラムの日英識別子で記述し分けたものを、二組にグループ分けした被験者に図 1 のように振分けて実験をさせ、その作業全体にかかった時間を評価項目として計測する。

3. 実験方法の設計方針

3.1. 識別子の言語の違い

デバッグ効率を評価する試みは、過去に紙上実験が行なわれたが、正しくデバッグできた被験者が少なかったために、統計をとれていない。そこで、実際に試行錯誤や実行確認などができるマシン上の作業環境で、デバッグ・機能拡張実験を行なう。

3.2. 内部コード系の違い

3.1. と同様にマシン上で実験を行なうことを考えると、内部コード系の異なる 2 台のマシン上で作業を行わねばならない。マシンが異なれば、当然マシン間でのツール環境や作業スタイルの違いが伴う恐れがある。したがって、紙上での読解実験で行なうことにする。

4. デバッグ・機能拡張効率を測る方法

デバッグ実験は、プログラムにあらかじめバグを混入しておき、そのデバッグを被験者に行わせる。一方、機能拡張実験では与えられたプログラムに拡張すべき機能を追加させる。この作業時間を実験中に被験者に計測させるのは困難であるので、マシン上のツールの実行開始・終了時刻の履歴が、実験中に自動的に記録される仕掛を組み込んでおく。

5. 読解実験方法

紙上での読解実験は、プログラムを被験者に読解させた後に、プログラムの理解を確認するための内

容についての設問に回答させる。その作業全体の時間を読解時間として、被験者に計測してもらう。

6. 評価実験

6.1. 被験者

当研究室内の学部 4 年生と大学院生計 18 人を被験者の対象として、評価実験を試みた。

実験用プログラム課題として次の四つを用意した。

- (1) 文書中の漢字の割合を出力する
- (2) 文書中の文の長さの分布を出力する
- (3) 電話番号の数字列にハイフンを挿入する
- (4) 文書中から検索文字列の存在位置を出力する

6.2. 実験結果

各プログラムにおける日英版の平均読解時間と、各プログラムについて英語プログラムより日本語プログラムの方が読みやすいことを調べる目的で t 検定を行った結果を表 1 に示す。

表から結論づけると、プログラム (3) 以外のプログラムについて日本語識別子のプログラムの方が読みやすいという結果を得ることができた。

過去の実験結果は紙上読解実験によるものであったが、本稿におけるマシン上実験においても同様に、日本語識別子の方が読みやすいという結果を得ることができた。デバッグ実験をマシン上で行なったことにより、各被験者が本来のスタイルでデバッグ作業を行なうことができたと考えられる。

また、文字列処理を行なうプログラムにおいて実験を行なった結果、フル 2 バイトコード系におけるプログラムの方が読みやすいことを紙上での読解実験により示すことができた。

6.3. 結果考察

プログラム (2) は他のプログラムと比較して、はるかに日本語識別子の有効性が高い。これは、他のプログラムと比較して、プログラム内容のキーワードとなる用語の意味が分りにくかった、知らなかったせいであると考えられる。

その一方で、マシン上の実験による結果は、識別子の違いによる差だけでなく、被験者のデバッグスタイルの差の要素も含んでしまっている。

じっくり読解を行なってからデバッグ、機能拡張を行なうスタイルと、バグ位置や機能拡張位置にあ

たりをつけてそこを中心に読解を行なうスタイルの差により、いくつかのプログラムで読解時間の分布が 2 分してしまっている。

プログラム (3) だけが英語プログラムの方がよいのは、英語版を読解した被験者の大半が後者のスタイルで行なったためであると考えられる。

7. おわりに

本稿における評価実験において、プログラムの識別子に用いる言語として、英語よりも日本語を用いた方がプログラムの可読性が高いことをデバッグ実験により示すことができた。

また、文字列処理を行なうプログラムにおいては、フル 2 バイトコード系の方が可読性が高いことを読解実験により示すことができた。

今後の課題は、デバッグ実験で読みやすさを測る場合に、被験者間のデバッグ能力差やスタイルの差を排除した上で実験を行なう必要があることである。

参考文献

- [1] 中川正樹 他：“母語プログラミングの理念、実現、実践とその効果”，電子情報通信処理学会論文誌，Vol. J77-D-I, No.5, PP.364-374
- [2] 並木美太郎 他：“言語 C コンパイラのマルチバイト化の実現方式”，情報処理学会論文誌，Vol.33 No.11, PP.1331-1340

表 1・被験者グループとプログラムの振り分け

被験者グループ	プログラム	プログラム 1	プログラム 2
グループ A	プログラム 1 (英語識別子)	プログラム 2 (日本語識別子)	
グループ B	プログラム 1 (日本語識別子)	プログラム 2 (英語識別子)	

表 2・各プログラムの平均作業時間と検定結果

実験方法	プログラム番号	平均読解時間		t 検定結果
		日本語版	英語版	
デバッグ実験	1	19.4	24.7	自由解 1 で 1=2 の検定を 4% で日本語の方が読みやすい
	2	14.0	24.3	自由解 2 で 1=5 の検定を 0.01% で日本語の方が読みやすい
機能拡張実験	3	18.0	14.4	自由解 1 で 1=0 の検定を 2% で日本語の方が読みやすい
	4	20.1	24.2	自由解 1 で 1=0 の検定を 2% で日本語の方が読みやすい

(読解時間の単位は分)

表 3・平均読解時間と検定結果

	フル 2 バイトコード系	混在コード系	t 検定結果
読解時間	21.3	24.3	自由解 1 で 1=0 の検定を 32% でフル 2 バイトコードの方が読みやすい

(時間の単位は分)