

7E-6

誤りの傾向からみた初級プログラマの
プログラミング理解の状況 COBOL言語の場合

立花 靖弘

慶応義塾大学

1. はじめに

プログラミングの過程は、デバッキングの過程でもあるといえるほど、エラーとのつきあいが深く長いのが普通である。それらのエラーはプログラミングの理解の状況を反映してさまざまである。

ところで、プログラミングの理解の状況の個人差は、どのような種類のエラーに顕著に反映するであろうか。もし、エラーの個人差からプログラマの適性を推定するとすれば、どのようなエラーを考察すべきであろうか。

2. 目的

初級プログラマが、プログラミングの過程でおかしたエラーを、次の3つの段階に分類し、個人差の所在と、その分布を推定する。

第1段階 (Mレベルとする) ; コンパイラの助けを借ずに机上デバックで発見したエラー

第2段階 (Cレベルとする) ; コンパイラからの情報を得て見出したエラー

第3段階 (Rレベルとする) ; コンパイラによってエラーなし(構文エラーなし)とされた後の実行(RUN)時のエラー

ただし、今回は個人差をプログラマの適性と相関づけることを当面の目的にするのではなく、より効果的なデバックツールの開発や教育のための資料にすることを目的としている。

3. 方法

19~20歳の40名の初級プログラマがCOBOL言語による6つの課題を完成させる。入力はマークカード方式である。計算機使用回数は、MCRの各レベルに応じた重みづけ点数の合計点が、指定された制限点数以内におさまるような回数

である。期間は約6週間である。

4. 結果

2888個のプログラムリストから得たエラーを、表1の分類法にしたがって分類した。分類された結果エラー総数は、10341個である。各エラーカテゴリーに於けるRの占める割合を図1に示す。個人差の最も大きいとみられるのは、"人力エラー"である。人力エラーの頻度は、1課題当たり約0.3と小さいので、個人のおかすこの種のエラー頻度を、ポアソン分布にしたがうものと仮定する。そうすると、全データはPE分布(ポリア・エゲンベルガー分布)へのあてはめがよい(図3)ことから、個人差の分布は Γ 分布となることが知られる。エラーの相対頻度と順位の関係は、図4の如くである、Mandelbrotの法則のあてはめがよい。

表1 Error Categories
(case of Cobol language)

| | |
|------------------------|-------------------------|
| Misspellings | ワードの部分的エラー |
| Structural words error | ワードの追加・変更・削除・交換など |
| Sentence error | 行単位の挿入・削除、演算ミスなど |
| Period error | ピリオドの追加・削除など |
| Column error | 欄区切り、カラムズレ |
| Input error | JCLエラー、データ取り扱いエラー、行・桁ズレ |

5. 検討

エラーの分類法は、例えば E. A. Youngs¹⁾ のように、Syntax/Semantic/Logic/Clerical/Other といった分類法の方が一般的かも知れない。しかし、これらの分類法では、認知心理学的な意味で厳密な手法をとらないかぎり、しばしばあいまいさを生じやすい。多量のエラーを、統計処理で第一近似的に問題の所在をつかむには、むしろ、あいまいさの入る余地の少ない分類を行なうのが妥当と考える。本稿では、意味的分類よりむしろ階層的分類をとった。

Mandelbrotの法則を、Zipfの法則の拡張と考え、Zipf流に "principle of least effort" と解釈するなら、図4は、努力の効果がまだ現われていない（すなわち、エラーの立場からみると努力していないとみなされるような）プログラムの初期状態と解釈されよう。もし、J. Q. Stewart が意味づけたように、競争状態にある要素間の均衡²⁾とみるなら、図4はまた、初級プログラムの、プログラムの中にLogicalな整合性をうちたてようとする力が、それをあいまいにしようとする種々の力 (Syntax/Semantic/Clerical/Other errors) に拮抗している様相と解釈されよう。したがって、熟達化に伴いこれらの均衡は、解消していくのではないか。参考までに C. R. Litecky らの論文³⁾からのデータを表1の分類にしたがって再分類して比較すると、米国のプログラマの方が日本の初級プログラマより熟達しているらしいことが推定される。

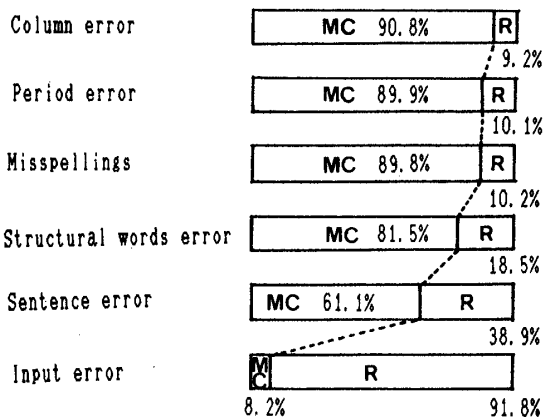


図1 エラーカテゴリー別MC・R比

M: 机上デバック C: コンパイル時エラー
R: 実行時エラー

参考文献

- 1) E. A. Youngs; Human Errors in Programming; Int. J. Man-Machine Studies 6(1974)
- 2) 館稔; 人口分析の方法; 古今書院(1963)
- 3) C. R. Litecky and G. B. Davis; A Study of errors, error-proneness, and error diagnosis in COBOL; Comm. ACM, 19(1976)

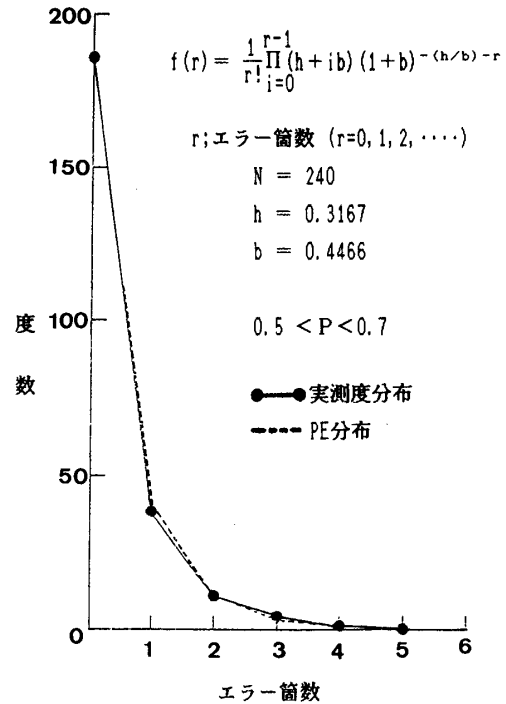


図3 全入力エラーの分布 (PE分布)

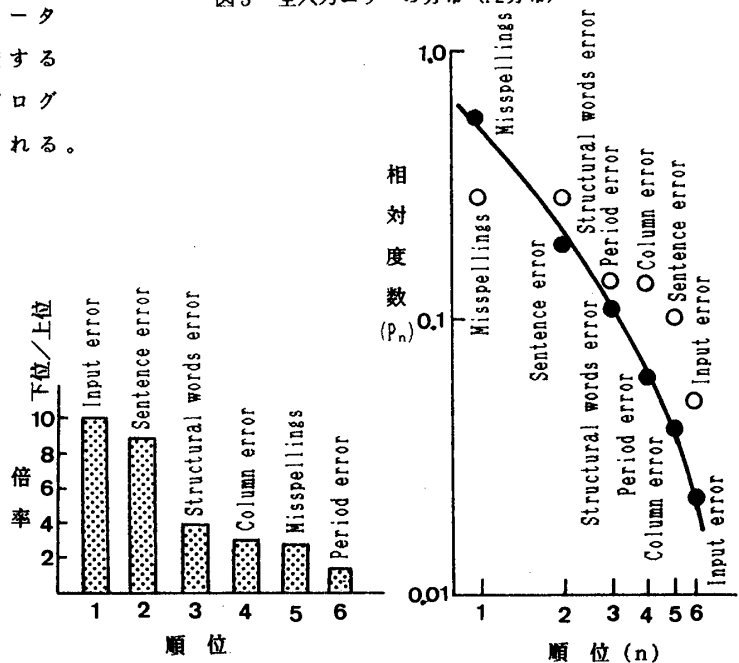


図4 回帰式 (Mandelbrotの法則)

$$\log P_n = 0.952 - 2.911 \log (1.60 + n)$$

●日本のプログラマ ○米国のプログラマ

図2 エラー頻度 (下位/上位) 比
エラー頻度上位下位各8人