On lines of code and programming productivity

Dear Editor:

Those of your readers who were interested in the Walston-Felix article in the *IBM Systems Journal* (Vol. 16, No. 1), entitled "A method of programming measurement and estimation," might also be interested in the scientific explanation for their data.

References 1 and 2 show that the number of elementary mental discriminations (E) required to implement a program is related to the program's volume (V), the language level (λ) the Stroud Number (S), and the time (T) according to

$$E = ST = V^{1.5}\lambda^{-0.5},$$

where $S = 18$ discriminations per second, and λ, although variable from one program to another, has a mean value for PL/I of 1.53 and for FORTRAN of 1.14. The volume (V) is related to the vocabulary (η) and the length (N) through

$$V = N \log_2 \eta \simeq \eta \log_2 (\eta/2) \log_2 \eta$$

Forum

Consequently, for an average value of λ, it is possible to obtain the length (N) from the time (T). Having the length (N), it is possible to estimate the number of source statements (P) rather roughly, by noting that the average executable FORTRAN statement contains 7.5 operators and operands ($N = 7.5$), and that executable statements constitute approximately half of the total, or $P = (2/7.5)N$.

On page 57, Walston and Felix mention the extremes of their data base, 4000 lines of code in 12 man-months as the minimum, and 467 000 lines of code in 11 758 man-months for the maximum project. Applying the preceding relationships to these values, we obtain the following results:

	Minimum	Maximum
T(Man-months)	12	11 758
$E = ST = 18 \times 60 \times 60 \times \dfrac{2000}{12} T$	1.30×10^6	1.27×10^{11}
$V = \lambda^{\frac{1}{3}} E^{\frac{2}{3}}$ ($\lambda = 1.34$)	2.83×10^5	2.79×10^7
η [from $V = \eta \log_2 (\eta/2) \log_2 \eta$]	2.45×10^3	1.06×10^5
$N = \eta \log_2 (\eta/2)$	2.51×10^4	1.66×10^6
$P = (2/7.5)N$	6700	444000

Now on page 62, Walston and Felix present their statistically obtained relation $E = 5.2\ L^{0.91}$, where E is effort in man-months, and L is lines of code in thousands. Using this relation, we may compare its results with the theory above, and with the observed values as follows:

	Observed	Walston-Felix	Software Science
Maximum Project	467000	4854000	444000
Minimum Project	4000	2510	6700

The result is clear. The data of Walston and Felix confirm the software relations, but not the Walston-Felix relation.

CITED REFERENCES

1. M. H. Halsted, *Software Physics: Basic Principles*, IBM Research Report RJ 1582, IBM Research Laboratory, San Jose, California 95193 (May 1975).
2. M. H. Halstead, *Elements of Software Science*, Elsevier North-Holland Incorporated (1977).

M. H. Halstead
Professor of Computer Sciences
Purdue University
Lafayette, Indiana 47907

Authors' response

Responding to the letter from M. H. Halstead, we note that the data used to develop his results are taken from the statement on page 57 of our article. This statement, unfortunately, is in error and does not accurately reflect the ranges of the data base. We refer to Figure 1, page 62, which shows that there are two projects with delivered source lines of code below 1000 and three projects with delivered source line of code greater than 400000. These projects and their associated efforts in man-months are as follows:

Project	Delivered Source Lines of Code	Total Effort in Man-Months
A	910	6
B	940	3
C	401099	1022
D	486834	11758
E	712362	2178

The relationship between Effort and Delivered Code, $E = 5.2L^{0.91}$, given on page 62, was obtained through a least squares fit to the data, as shown in Figure 1. If a relationship is desired for expressing delivered source lines of code as a function of total man-months of effort, then the least squares fit to the data yields the expression

$$L = 0.925\ E^{0.70},$$

where L is delivered source code in K lines and E is the total effort in man-months. Interestingly, this relationship is close to Halstead's expression, which is approximated as follows:

$$L = 1.27 E^{0.67}.$$