Tabulation of R and H for $\mu = (a + b\sqrt{5})/2$—*Continued*

a	b	R	H	a	b	R	H	a	b	R	H	a	b	R	H
97	13	600	5	98	6	432	36	98	18	360	30	99	13	288	24
97	15	432	36	98	8	336	28	99	1	192	16	99	15	192	16
97	17	288	24	98	10	336	28	99	3	192	16	99	17	240	20
97	19	360	3	98	12	840	7	99	5	288	24	99	19	288	24
98	2	288	24	98	14	288	24	99	7	264	22				
98	4	840	7	98	16	240	20	99	11	192	16				

The computation was monitored for about the first 200 tally operations to make sure the score-keeping was correct in all possible cases. The tallying was, as before, basically a question of seeing that every permutation and change in sign in the triple (ξ_1, ξ_2, ξ_3) counted as a unit. The total running time was roughly 2.5 hours. One might remark that the human time involved in computing these class numbers H from basic *algebraic* concepts would have to be measured in "life-times," not "man-hours."

The computation was completed 18 April 1958 and was sponsored in part by the National Science Foundation Grant G-4222.

Department of Mathematics
University of Arizona
Tucson, Arizona

1. L. E. Dickson, *History of the Theory of Numbers*, v. III, G. E. Stechert, New York, 1934 (for references in the first paragraph).
2. F. Götzky, "Über eine zahlentheoretische anwendung von modulfunktionen zweier veränd-licher," *Mathematische Annalen*, v. 100, 1928, p. 411–437.
3. H. Maass, "Uber die darstellung total positiver zahlen des körpers $R(\sqrt{5})$ als summe von drei quadraten," *Abhandlungen aus dem Mathematischen Seminar der Hansischen Universität*, v. 14, 1941, p. 185–192.

Multiplication Time on The IBM 709

By D. D. Wall

Average multiply time is useful for roughly estimating problem running time for various problems, as well as for roughly comparing different computing machines. Determining average multiply time for the 709 is complicated, however, due to its zero-skipping feature, and requires an investigation of runs of zeros in binary sequences. The particular problem we solve is that of evaluating $R(n, l)$ = total number of runs of length l in all the 2^n words of n bits each, and $S(n, l) = \sum_{x=l}^{n} R(n, x)$ = number of runs of length $\geq l$ in the 2^n words of n bits each. The resulting 709 average multiply time is 193 microseconds fixed point, or 170 microseconds normalized floating point, and the purpose of this note is to derive these two numbers.

We make use of a device which we call "differencing modulo 2," which obtains an $n - 1$ bit number from a given n bit number by writing 1 or 0 according as the successive bits in the given number exhibit a change or no change. For example, each of the (complementary) 8 bit numbers 11010001 and 00101110 gives the same result 0111001 as its 7 bit difference modulo 2.

By differencing modulo 2, each run of length l in a given word of n bits gives rise to a run of 0's of length $l - 1$ in the $n - 1$ bit difference, and, conversely, each $n - 1$ bit number with a run of $l - 1$ 0's may be obtained by differencing either of two unique (complementary) n bit numbers with a run of length l. Finally, among all the 2^n numbers of n bits each, the number of runs of 0's of length l is the same as the number of runs of 1's of length l, and from these considerations we conclude that

$$R(n, l) = R(n - 1, l - 1) = \cdots = R(n - l + 1, 1),$$

where, again, $R(n, l)$ = number of runs of length l in the 2^n words of n bits each.

We evaluate $R(n, 1)$ by counting the occurrences of an isolated 0 in each of the n possible positions of an n bit number: 2^{n-2} of these numbers begin 01, 2^{n-3} of these numbers begin 101, etc., and 2^{n-2} of these numbers end 10. Since the total number of runs is twice the number of runs of 0's, we have

$$R(n, 1) = 2[2^{n-2} + (n - 2)2^{n-3} + 2^{n-2}] = (n + 2)2^{n-2}, n > 1, \text{ and } R(1, 1) = 2.$$

Therefore, $R(n, l) = (n - l + 3)2^{n-l-1}$, $n > l$, and $R(n, n) = 2$.

To find the number of runs of length $\geq l$ we use the formula for summation by parts, $\sum_a^b u(x)\Delta v(x) = u(b + 1)v(b + 1) - u(a)v(a) - \sum_a^b v(x + 1)\Delta u(x)$, and obtain

$$S(n, l) = \sum_{x=l}^{n} R(n, x) = (n - l + 2)2^{n-l}.$$

Now, 709 multiplication requires 6μ for each 1 bit in the multiplier plus 6μ for each run of 0's, plus 6μ additional for each of these runs of 0's of length 7–12, plus 6μ more for each of these runs of length 13–18, etc., in addition to certain time charges independent of the bit structure. The average number of 1's in an n bit number, all n bit numbers equally likely, is $n/2$, and the average number of runs of 0's is $(n + 1)/4$. This last quantity may be found by again differencing modulo 2, since the number of runs in an n bit number is 1 plus the number of 1's in the $n - 1$ bit difference.

For fixed point multiplication, 12μ are required for instruction time, 12μ for data transfer, 6μ are reserved for other purposes, and 3μ must be added as an average waiting time at the end of the operation in order to total a multiple of 12μ basic cycles. Therefore, with $n = 35$ and all n bit numbers assumed equally likely, average fixed point multiply time is given by $T = 33 + 6(35/2) + 6(36/4) + d = 192 + d$, where $d = 6 \cdot \frac{1}{2} \cdot 2^{-35}[S(35, 7) + S(35, 13) + \cdots]$ is the delay occasioned by runs of 0's of length >6. On evaluating this expression we find that $d < 1$ and so obtain the 193μ result. Although the 709 disposes of a zero multiplier in a different and quicker manner, this exception does not modify the 193μ answer because of the assumption that all 2^{35} numbers are equally likely.

For normalized floating point multiplication, the mantissa has 27 bits of which the first is a 1, and 12μ are required to add the exponents. The resulting average time is $T = 45 + 6(1 + 26/2) + 6(27/4) + d = 169.5 + d$, and $d = 6 \cdot \frac{1}{2} \cdot 2^{-26}[S(26, 7) + S(26, 13) + \cdots]$ is $< \frac{1}{2}$, thereby giving the 170μ result.

International Business Machines Corporation
112 East Post Road
White Plains, New York