Journal of Research of the National Bureau of Standards Vol. 59, No. 3, September 1957 Research Paper 2783

A Computation of Cyclic Cubic Units[1]

Harvey Cohn and Saul Gorn[2]

The paper is a report on a tabulation of units performed on the EDVAC at the U. S. Army Proving Grounds in Aberdeen, Maryland. The algebraic number fields involved were 45 cyclic cubic fields of discriminant l^2, where l is one of the primes of form $3m+1$ from 7 through 499. The object of the search was the discovery of units through a specific method, an algorithm based on Minkowski's geometric number theory but particularly amenable to a stored-program computer. In the computation, combined use was made of integral arithmetic and decimal arithmetic but with careful error analysis.

1. Introduction

An essential part of the lure of algebraic number theory consists of tabulations of almost unpredictable irregularity. The advanced theory has somehow surpassed the scope of numerical data without perceptibly improving the power to predict such data. As a result, for a small calculation an algebraic number theorist would not forego the personal satisfaction that results from the many ingenious, fortuitous, and deeply meaningful combinations of integers that he would encounter. Yet to really appreciate these vital irregularities he needs longer tabulations, with the inevitable requirements of uniformity and efficiency. Here the modern electronic digital computer can be used profitably.

The discussion of the problem that follows will be primarily from the point of view of the computer program rather than from its theoretical development. Thus, for instance, the field will have to be regarded not as a set of elements satisfying certain axioms, etc., in the manner of Dedekind and Steinitz, but as an algebra with a definite basis and definite "structure constants" for multiplications. As a matter of historical fact the exhibiting of a basis was generally far from trivial and lagged considerably behind existence theorems [*1*, *10*] [3]

2. Description of Fields

For any prime $l(=3m+1)$ a cyclic cubic field will be defined. Our interest centers around *algebraic integers* of such a field, written as follows (owing to the presence of the so-called normal basis):

$$\xi=a\omega_1+b\omega_2+c\omega_3=(a,b,c). \tag{1}$$

Here a, b, c are rational integers (or coordinates), and ω_1, ω_2, ω_3 (or the normal basis) are the three roots of the defining (normal) equation [*8*].

$$F(\omega)=\omega^3+\omega^2-m\omega-n=0. \tag{2}$$

Here n is defined as follows: First, $4l$ is decomposed (uniquely) into the combination

$$4l=A^2+27g^2, \tag{3}$$

where the signs of A and g are specified by the conditions (in integral k)

$$A=3k+1, \qquad g>0. \tag{4a}$$

---

[1] The computing time was provided under contract DA-028-ORD-12332 with the United States Army Office of Ordnance Research. The first named author was then at Wayne University, Detroit, Mich., and the second named author was at the Ballistics Research Laboratory, Aberdeen, Md. The first named author is currently at Washington University, St. Louis, Mo.

The preliminary report (reference [*5*]), was cleared for publication by the Ballistics Research Laboratory on April 26, 1956. The background description, the illustrative examples, and the final tabulation were prepared at the National Bureau of Standards. This work has formed the basis for a further calculation currently on the SEAC.

[2] Now at Moore School of Electrical Engineering, Philadelphia, Pa.

[3] Italicized figures in brackets indicate the literature references at the end of this paper.

Then, finally an integer n always exists, for which

$$27n=(A+3)l-1. \tag{4b}$$

All algebraic integers under discussion are *real*.

The most general *basis* of the field would be given by an integral unimodular combination $\zeta_i=\sum a_{ij}\omega_j \quad i,j=1,2,3$, $\det|a_{ij}|=\pm 1$. There is essentially no other basis [9].

The particular designation of roots is standardized as

$$\omega_1>\omega_2>\omega_3. \tag{5}$$

Thus we can distinguish among conjugates ξ (given above), and ξ', ξ'' given by

$$\xi'=c\omega_1+a\omega_2+b\omega_3=(c,a,b) \tag{6a}$$

$$\xi''=b\omega_1+c\omega_2+a\omega_3=(b,c,a) \tag{6b}$$

(formed by cyclic permutation of ω_1, ω_2, ω_3), so that $\xi'''=\xi$, etc. As a matter of notation, the subscripts denote conjugates only in reference to the basis ω_i. Later on ξ_i shall denote (unrelated) integers of the field.

It was mentioned earlier that the discriminant of the field or the determinant

$$\begin{vmatrix} \omega_1 & \omega_2 & \omega_3 \\ \omega_2 & \omega_3 & \omega_1 \\ \omega_3 & \omega_1 & \omega_2 \end{vmatrix}^2 \tag{7}$$

has the value l^2. This fact (which can be verified directly from eq (1)) is not used directly in the machine computation but uniquely identifies the field in question [6].

The multiplication of any two numbers in the field can be expressed in integral arithmetic; e. g.,

$$\xi_1=(a_1,b_1,c_1), \qquad \xi_2=(a_2,b_2,c_2), \tag{8}$$

then one can find the explicit representation of $\xi_1\xi_2$ in terms of the normal basis by means of the structure constants α_1, β_1, γ_1, α_2, β_2, γ_2, defined by

$$\omega_1^2=\alpha_1\omega_1+\beta_1\omega_2+\gamma_1\omega_3$$

and

$$\omega_2\omega_3=\alpha_2\omega_1+\beta_2\omega_2+\gamma_2\omega_3.$$

The explicit formula is, then, e. g.,

$$\xi_1\xi_2=(A,B,C), \tag{9a}$$

where

$$\left.\begin{aligned} A&=\alpha_1a_1a_2+\gamma_1b_1b_2+\beta_1c_1c_2+\beta_2(a_1b_2+a_2b_1)+\alpha_2(b_1c_2+b_2c_1)+\gamma_2(c_1a_2+c_2a_1),\\ B&=\beta_1a_1a_2+\alpha_1b_1b_2+\gamma_1c_1c_2+\gamma_2(a_1b_2+a_2b_1)+\beta_2(b_1c_2+b_2c_1)+\alpha_2(c_1a_2+c_2a_1),\\ C&=\gamma_1a_1a_2+\beta_1b_1b_2+\alpha_1c_1c_2+\alpha_2(a_1b_2+a_2b_1)+\gamma_2(b_1c_2+b_2c_1)+\beta_2(c_1a_2+c_2a_1). \end{aligned}\right\} \tag{9b}$$

This product can therefore be computed in the following tabular form:

a_1a_2	α_1	β_1	γ_1
b_1b_2	γ_1	α_1	β_1
c_1c_2	β_1	γ_1	α_1
$a_1b_2+a_2b_1$	β_2	γ_2	α_2
$b_1c_2+b_2c_1$	α_2	β_2	γ_2
$c_1a_2+a_1c_2$	γ_2	α_2	β_2
	$(A$	B	$C).$

These structure constants are given explicitly by the formulas:

$$\left.\begin{aligned} \alpha_2&=\frac{m+k+1}{3}, & \beta_2&=\frac{g+m-\alpha_2}{2}, & \gamma_2&=\frac{-g+m-\alpha_2}{2},\\ \alpha_1&=\alpha_2-m+1, & \beta_1&=\frac{g-m-\alpha_2}{2}, & \gamma_1&=\frac{-g-m-\alpha_2}{2}. \end{aligned}\right\} \tag{9c}$$

They are listed in table 2 (see p. 166).

In particular, the norm of ξ is defined for (1) by the integer

$$N(\xi)=\xi\xi'\xi''=-n(a+b+c)^3-l\{(k-g+1)\Sigma a^2b/2+(k+g+1)\Sigma ab^2/2+abc\}, \tag{10a}$$

and the trace is defined by

$$t(\xi)=\xi+\xi'+\xi''=-(a+b+c), \tag{10b}$$

so that the defining equation of ξ, ξ', or ξ'' is

$$\xi^3-t(\xi)\xi^2+t(\xi\xi')\xi-N(\xi)=0, \tag{10}$$

with the further designation

$$\bar{t}(\xi)=t(\xi\xi')=-m(a+b+c)^2+l\Sigma ab. \tag{10c}$$

Note that if $a+b+c=0$, then $N(\xi)$ is divisible by l. Thus, applying this information to

$$\xi-\xi'=((a-c),(b-c),(c-a)),$$

we find $N(\xi-\xi')=(\xi-\xi')(\xi'-\xi'')(\xi''-\xi)$ is an integer divisible by l.

3. Units

Units are algebraic integers of norm equal to ± 1. They are written as

$$\eta=u_1\omega_1+u_2\omega_2+u_3\omega_3=(u_1,u_2,u_3), \tag{11}$$

where, as before, u_1, u_2, u_3 are integers. The most general unit has the form $\pm\eta_o^a\eta_o'^b$, where a and b are (positive, negative, or zero) integers, whereas η_0 is a specially designated *fundamental* unit [*6*, p. 19]. The only *fundamental* units are $\pm\eta_o^{\pm 1}$, $\pm(\eta_o')^{\pm 1}$, $\pm(\eta_o'')^{\pm 1}$. In what follows, units will be normalized (by a change of sign) to have, conveniently, only norm $+1$.

If we take eq (10a) and set $N(\xi)=1$ and $(a,b,c)=(u_1,u_2,u_3)$, we obtain a cubic equation in these three latter unknowns, which we are in effect solving. Now such an equation (unlike, say, the Pell equation), has very little intrinsic interest. The interest in units is derived entirely from their role in *unique factorization*.

To give an example, when $l=19$, one finds (as a byproduct of the main calculation) that

$$N(\omega_1)=N(2\omega_1-\omega_2)=7. \tag{12}$$

This does *not* mean that 7 has two different factorizations, although seemingly

$$7=\omega_1\omega_2\omega_3=(2\omega_1-\omega_2)(2\omega_2-\omega_3)(2\omega_3-\omega_1). \tag{12a}$$

The reason, of course, is that the factors can pair off as *associates*, or

$$(2\omega_2-\omega_3)/\omega_1=(-\omega_1-2\omega_2), \tag{12b}$$

a unit of the field. To see this, in (12a) calling one set of factors ξ, ξ_1', ξ_1'' and the other ξ_2, ξ_2', ξ_2'', respectively, we must verify that one of the ratios ξ_2/ξ_1, ξ_2'/ξ_1, ξ_2''/ξ_1 is a unit. Thus for example, in accordance with the rules of section 2, we compute

$$\frac{\xi_2'}{\xi_1}=\frac{\xi_2\xi_1'\xi_1''}{\xi_1\xi_1'\xi_1''}=\frac{(A^*,B^*,C^*)}{N(\xi_1)} \tag{13}$$

and find that A^*, B^*, and C^* are now divisible by $N(\xi_1)$. In practice, the whole computation would be performed modulo $N(\xi_1)(=7$ here), using the structure constants of table 2.

Thus the preliminary step in understanding factorization in fields becomes the recognition of units. Traditionally, one creates, by trial and error, combinations such as (12a), and one hunts for units, usually with better luck than one can completely explain [4].

The method used here will be very systematic. It is an algorithm for creating a surfeit of algebraic integers with small norms (including norm unity). In the process, a unit, indeed a fundamental unit, is inevitably produced. (Unfortunately, machine limitations forced the weakening of the method, as is explained in section 6.)

4. The Sign-Discrimination Algorithm

One operates with so-called *reduced* 3×4 matrices of algebraic integers [2, section 4]:

$$\phi=\begin{pmatrix}\xi_1 & \xi_2 & \xi_3 & \xi_4\\ \xi_1' & \xi_2' & \xi_3' & \xi_4'\\ \xi_1'' & \xi_2'' & \xi_3'' & \xi_4''\end{pmatrix}. \tag{14a}$$

The columns represent conjugates of algebraic integers of sum zero

$$\xi_1+\xi_2+\xi_3+\xi_4=0 \tag{14b}$$

and such that any three of the four ξ_i form a basis. The further, and most vital, condition is that the sign pattern be

$$\operatorname{sgn}\phi=\begin{pmatrix}+ & - & - & +\\ - & + & - & +\\ - & - & + & +\end{pmatrix} \tag{14c}$$

either as ϕ now stands or under some rearrangement of columns.

The algorithm is a method of generating such matrices in chains. A matrix ϕ_1 is said to be a *neighbor* of ϕ if it is formed by adding one column of ϕ to another column of ϕ and at the same time subtracting it from a third column of ϕ (the three columns being different). Thus there are 4! possibilities to consider corresponding to the distinct triples $(j_1j_2j_3)$, where $1\leq j\leq4$. For instance, the operation

$$\phi[341]\phi_1 \tag{15a}$$

can be understood to mean that the "third column of ϕ is increased by the fourth column, which in turn diminishes the first," or, writing just the first rows,

$$\left.\begin{aligned}\phi &=(\xi_1,\ \xi_2,\ \xi_3,\ \xi_4)\\ \phi_1&=(\xi_1-\xi_4,\ \xi_2,\ \xi_3+\xi_4,\ \xi_4).\end{aligned}\right\} \tag{15b}$$

(The relation (15a) can be read *backward* as an operation on ϕ_1.) Of the 4! possible neighbors, either 3, 4, 5, or 6 will be reduced, depending on inequalities satisfied by the elements [2, section 10].

We next define the conjugate ϕ' of a matrix ϕ to be the matrix formed by replacing the first row ξ_i by its conjugate ξ_i' and forming the remaining rows from ξ_i' instead of ξ_i. The conjugate ϕ' will not be reduced as it stands, but a rearrangement of the columns (in this case the interchange of first and second) will render it reduced. Likewise, we define ϕ''. Note $\phi'''=\phi$ again. Finally, two matrices ϕ_1 and ϕ_2 are said to be *proportional* to one another if, for $\phi_1^*=\phi_1$, ϕ_1', or ϕ_1'', and numbers η, η', η'',

$$\phi_2 = \begin{pmatrix} \eta & 0 & 0 \\ 0 & \eta' & 0 \\ 0 & 0 & \eta'' \end{pmatrix} \phi_1^* . \tag{16}$$

It is easily seen that η, η', η'' are necessarily the conjugates of a unit, possibly 1, having norm $+1$. (In some cases ϕ_1 is proportional to itself, leading to interesting possibilities related to ramified primes but beyond the scope of the present discussion [2, section 22].) In all cases, proportional matrices will have the same set of norms (possibly under rearrangement).

The algorithm operates as follows: The initial matrix is taken for convenience (writing only the first row) as

$$\phi_0 = (\omega_1, \omega_3, \omega_2, 1), \qquad n > 0, \tag{17a}$$

with norms respectively

$$(n, n, n, 1), \tag{17b}$$

or as

$$\phi_0 = (\omega_1 + \omega_2, \omega_3 + \omega_1, -\omega_1, 1) \qquad n < 0 \tag{17c}$$

with norms respectively

$$(-n+m, -n+m, -n, 1). \tag{17d}$$

Branching out from ϕ_0, one forms all reduced neighbors of matrices present, terminating a branch when the matrix is proportional to one already present. Indeed, the number of reduced matrices is finite to within proportionalities [2, section 8]. The factors of proportionality η are units among which can always be found the fundamental unit [2, section 18].

5. Modified Algorithm

The original algorithm, as just described, is a branching algorithm in which each ϕ may have more than one successor, and each ϕ is compared with all those preceding. Because each ϕ contains 12 components and there could be more than 50 produced, the internal storage of the EDVAC would be taxed before the algorithm had gone very far. The branching algorithm was therefore modified to a form in which only one successor is chosen for each ϕ, and the comparison (16) is always between ϕ_0 and the most recent ϕ.

The modification consisted in trying only one-third of the 4! possible neighbors, namely,

$$[123], [132], [423], [432], [214], [241], [314], [341] \tag{18}$$

characterized by the fact that the second and third indices are either 2,3 or 1,4 in some order. Either one or two neighbors of this type occur, but the EDVAC takes only the first that occurs in the program, thus we have a chain-type algorithm rather than a branching algorithm, with the further property that $\sum_1^4 |\xi_i|$ decreases at each step [2, section 17], whereas the corresponding sums for the second and third row (conjugates) increase or remain the same. This assures us that when the inevitable proportionality occurs, the unit will not be 1 (but it need not be a fundamental unit either).

As an illustration (see table 1) [4] the matrix sequence ϕ_i is reproduced for $l = 19$. It is

$\phi_0[241]$ $\phi_1[132]$ $\phi_2[314]$ $\phi_3[314]$ $\phi_4[314]$ $\phi_5[423]$ $\phi_6[341]$ $\phi_7[123]$ $\phi_8[123]$ $\phi_9[123]$ $\phi_{10}[432]$ $\phi_{11}[214]$ ϕ_{12}.

The EDVAC discovered that ϕ_0 is proportional to (some rearrangement of) ϕ_{12}. The specific numerical values are of some interest. We therefore calculate

[4] A full explanation of table 1 is given in section 7.

$$\omega_1 = \ \ 2.507\ 018\ 643$$
$$\omega_2 = -1.221\ 876\ 162$$
$$\omega_3 = -2.285\ 142\ 481,$$

which (except for minor modifications in section 7) is the accuracy presented by the EDVAC. Thus

$$\phi_0 = \quad ((1,0,0) \qquad (0,1,0) \qquad (0,0,1) \qquad (-1,-1,-1))$$

or

$$\phi_0 = \begin{pmatrix} 2.507\ 018\ 643 & -2.285\ 142\ 481 & -1.221\ 876\ 162 & 1 \\ -1.221\ 876\ 162 & 2.507\ 018\ 643 & -2.285\ 142\ 481 & 1 \\ -2.285\ 142\ 481 & -1.221\ 876\ 162 & 2.507\ 018\ 643 & 1 \end{pmatrix}$$

$$\phi_{12} = \quad ((7,5,5) \qquad (4,-3,6) \qquad (2,6,-1) \qquad (-13,-8,-10))$$

or

$$\phi_{12} = \begin{pmatrix} 0.014\ 037\ 286 & -0.017\ 151\ 828 & -0.032\ 077\ 205 & 0.035\ 191\ 747 \\ -7.443\ 752\ 324 & 17.010\ 034\ 653 & -18.661\ 625\ 853 & 9.095\ 343\ 524 \\ -9.570\ 284\ 962 & -23.992\ 882\ 825 & 11.693\ 703\ 058 & 21.869\ 464\ 729 \end{pmatrix}$$

The EDVAC then "recognized" that the first column would serve as a proportionality factor, i. e., if we rearrange the columns of ϕ_{12} to form

$$\phi_{12}^* = ((-13,-8,-10),\ (2,6,-1),\ (4,-3,6),\ (7,5,5)),$$

then

$$\phi_{12}^* = \begin{pmatrix} \eta & 0 & 0 \\ 0 & \eta' & 0 \\ 0 & 0 & \eta'' \end{pmatrix} \phi_0,$$

where $\eta = 0.014\ 037\ 286$, $\eta' = -7.443\ 752\ 34$, and $\eta'' = -9.570\ 284\ 962$. In this case the decimal accuracy would be easily sufficient for "recognition," but integral arithmetic was used (see section 8).

The unit produced, $\eta = (7,5,5)$, is not fundamental, but, early in the course of the computation, for ϕ_1, ξ_2 happened to come out as $\eta_0 = (-1,-1,0)$, a fundamental unit (see section 10). Actually,

$$\eta = (\eta_0'')^2/\eta_0'.$$

This can be easily verified from the multiplication scheme (9), e. g., $(\eta_0'')^2 = (-6,-3,-5)$, $(\eta_0')^{-1} = (2,2,1)$.

6. Numerical Accuracy

As the algorithm was first conceived and applied in experiment [2, section 11], decimal accuracy seemed sufficient because the symmetric functions could be correctly evaluated by the nearest integer when necessary. In a long computation, however, the algorithm can go astray in many ways:

(a) The algorithm may incorrectly discriminate the reduced neighbors from the others. This is bound to happen because the subset of the reduced neighbors chosen by the modified algorithm just mentioned has $\sum_1^4 |\xi_i|$ approaching zero monotonically. The error in the quantities ξ_i can soon be bigger than the quantities themselves, making a simple discrimination in sign untrustworthy.

(b) It may make uncertain the decision as to whether ϕ and ϕ^* are proportional; for example, it may have to be decided that (using an obvious notation), $\xi_1/\xi_1^* = \xi_2/\xi_2^* = \xi_3/\xi_3^* = \xi_4/\xi_4^*$, in which all the quantities involved are in error.

(c) It may of course make the approximations to the all-important resulting units untrustworthy; for these are the ratios discussed in (b).

A vital and desirable modification, to control the accuracy of the computations, consisted in carrying out all arithmetic operations exactly by using integral arithmetic; (explicit formulas for the integral arithmetic operations on two such integers are given in section 2). Furthermore, all quotients were removed from the algorithm (see section 8).

It would also have been quite easy in principle to have handled the sign discriminations in integral arithmetic. For instance, to tell if two algebraic numbers ξ_1 and ξ_2 agree in sign, all we need have done is test to see if the three conjugates $\xi_1\xi_2$, $\xi_1'\xi_2'$, and $\xi_1''\xi_2''$ are each positive (total positiveness), which, by Descartes' law of signs, is a matter of seeing that the defining eq (10) for $(\xi_1\xi_2)$ has (integral) coefficients with alternating sign. At the time, however, the EDVAC internal storage was not sufficient to handle the enormous integers, so that an automatic error analysis was used instead. This was the only departure from integral arithmetic.

7. Input-Output and Error Controls

With the exception of the following operations, the problem was run completely internally, with no human intervention:

(a) For each l, the structure constants in table 2 were computed in advance by hand (merely to save space in the memory) through the α_2 column. (The entire table is given for convenience of hand checking.)

(b) For each l, the zeros (and maximum errors) of the polynomial (2) were computed in advance by machine. The polynomial was scaled by dividing its zeros by m and carrying through the computation in fixed point. The root-finding method for the scaled polynomial,

$$f(\omega)=\left(\frac{\omega}{m}\right)^3+\frac{1}{m}\left(\frac{\omega}{m}\right)^2-\frac{1}{m}\left(\frac{\omega}{m}\right)-\frac{n}{m^3}, \tag{19}$$

is the bisection method (Horner's method in binary), for appropriate isolating intervals. The maximum error in the coefficients of the scaled polynomial, for EDVAC, is 2^{-44}, whence the maximum error in evaluating the scaled polynomial is 7×2^{-44}; if, then, we use $\epsilon=2^{-41}$, the machine will provide an automatic error analysis in solving for ω_i by finding the zeros r_i^+ of $f(\omega)+2\epsilon$, r_i^- of $f(\omega)-2\epsilon$, and setting

$$\frac{\omega_i}{m}=\frac{r_i^+ + r_i^-}{2}, \qquad \epsilon_1=\frac{r_i^+ - r_i^-}{2}.$$

Because the numerical values of the zeros ω_i are only used in the main algorithm to discriminate the signs of linear homogeneous expressions, $a\omega_1+b\omega_2+c\omega_3$, the scaled values ω_i/m and their error bounds ϵ_i were used directly.

(c) The input and output of the algorithm itself was by means of IBM cards, in decimal. (This required internal conversion on input and output because the EDVAC arithmetic is binary.) For each l two IBM cards were read in. One contained the integers l, m, g, α_2, n. The other contained the integer l together with the (10-digit) decimals ω_1/m, ω_2/m, ω_3/m, and the errors in ω_1, ω_2, ω_3.

The output cards were as follows: First, there were copies of the two input cards; second, for each ϕ_i, as soon as it was produced, four cards were punched with $j=1,2,3,4$, respectively, indicating the column of ϕ_i (prior to possible rearrangement) being described. The algebraic integer ξ_j was described by the integers l, i, j, a_j, b_j, c_j, $N(\xi)$, $t(\xi)$, $\bar{t}(\xi)$, where N, t, $\bar{t}$ refer to the defining eq (10) and

$$\xi_j=a_j\omega_1+b_j\omega_2+c_j\omega_3.$$

Third, there would be a card containing the integers (see (11)) l, i, $j(=5)$, u_1, u_2, u_j, $N(\xi)(=1)$, $t(\xi)$, $\bar{t}(\xi)$ if the algorithm runs to completion to produce a ϕ_i proportional to ϕ_0 with factor η.

TABLE 1. *Input-output cards*

The material in this table is presented in the form in which it is printed out by EDVAC.

INPUT CARDS

Structure constants	l				m	g	k	α_2	n
	19				6	1	2	3	7
Roots and errors ($\times 10^{-10}$)	l			ω_1/m	ω_2/m	ω_3/m	Error ω_1/m	Error ω_2/m	Error ω_3/m
	19			4178364406	−2036460270	−3808570802	1	1	1

OUTPUT CARDS

Matrix cards	l	Matrix number	Column number	$(\xi_i = a_i\omega_1 + b_i\omega_2 + c_i\omega_3)$			$N(\xi_i)$	$t(\xi_i)$	$\bar{7}(\xi_i)$
				a_i	b_i	c_i			
(ξ_1)	19	0	1	1	0	0	7	−1	−6
(ξ_2)	19	0	2	0	0	1	7	−1	−6
(ξ_3)	19	0	3	0	1	0	7	−1	−6
(ξ_4)	19	0	4	−1	−1	−1	1	3	3
(ξ_1)	19	1	1	2	1	1	11	−4	−1
(ξ_2)	19	1	2	−1	−1	0	1	2	−5
(ξ_3)	19	1	3	0	1	0	7	−1	−6
(ξ_4)	19	1	4	−1	−1	−1	1	3	3
(ξ_1)	19	2	1	2	2	1	1	−5	2
(ξ_2)	19	2	2	−1	−2	0	1	3	−16
(ξ_3)	19	2	3	0	1	0	7	−1	−6
(ξ_4)	19	2	4	−1	−1	−1	1	3	3
(ξ_1)	19	3	1	2	2	1	1	−5	2
(ξ_2)	19	3	2	−1	−2	0	1	3	−16
(ξ_3)	19	3	3	2	3	1	11	−6	−7
(ξ_4)	19	3	4	−3	−3	−2	7	8	15
(ξ_1)	19	4	1	2	2	1	1	−5	2
(ξ_2)	19	4	2	−1	−2	0	1	3	−16
(ξ_3)	19	4	3	4	5	2	7	−11	−4
(ξ_4)	19	4	4	−5	−5	−3	11	13	31
(ξ_1)	19	5	1	2	2	1	1	−5	2
(ξ_2)	19	5	2	−1	−2	0	1	3	−16
(ξ_3)	19	5	3	6	7	3	1	−16	3
(ξ_4)	19	5	4	−7	−7	−4	7	18	51
(ξ_1)	19	6	1	2	2	1	1	−5	2
(ξ_2)	19	6	2	−1	−2	0	1	3	−16
(ξ_3)	19	6	3	7	9	3	19	−19	−57
(ξ_4)	19	6	4	−8	−9	−4	1	21	14

TABLE 1. *Input-output cards*—Continued

OUTPUT CARDS—continued									
Matrix cards	l	Matrix number	Column number	$(\xi_i = a_i\omega_1 + b_i\omega_2 + c_i\omega_3)$			$N(\xi_i)$	$t(\xi_i)$	$\bar{t}(\xi_i)$
				a_i	b_i	c_i			
(ξ_1)	19	7	1	10	11	5	7	−26	29
(ξ_2)	19	7	2	−1	−2	0	1	3	−16
(ξ_3)	19	7	3	−1	0	−1	1	2	−5
(ξ_4)	19	7	4	−8	−9	−4	1	21	14
(ξ_1)	19	8	1	9	9	5	11	−23	75
(ξ_2)	19	8	2	−1	−2	0	1	3	−16
(ξ_3)	19	8	3	0	2	−1	7	−1	−44
(ξ_4)	19	8	4	−8	−9	−4	1	21	14
(ξ_1)	19	9	1	8	7	5	7	−20	89
(ξ_2)	19	9	2	−1	−2	0	1	3	−16
(ξ_3)	19	9	3	1	4	−1	11	−4	−115
(ξ_4)	19	9	4	−8	−9	−4	1	21	14
(ξ_1)	19	10	1	7	5	5	1	−17	71
(ξ_2)	19	10	2	−1	−2	0	1	3	−16
(ξ_3)	19	10	3	2	6	−1	7	−7	−218
(ξ_4)	19	10	4	−8	−9	−4	1	21	14
(ξ_1)	19	11	1	7	5	5	1	−17	71
(ξ_2)	19	11	2	−3	−8	1	11	10	−353
(ξ_3)	19	11	3	2	6	−1	7	−7	−218
(ξ_4)	19	11	4	−6	−3	−5	1	14	21
(ξ_1)	19	12	1	7	5	5	1	−17	71
(ξ_2)	19	12	2	4	−3	6	7	−7	−408
(ξ_3)	19	12	3	2	6	−1	7	−7	−218
(ξ_4)	19	12	4	−13	−8	−10	7	31	200
Unit card	l	Matrix	Indicator	$(\eta = u_1\omega_1 + u_2\omega_2 + u_3\omega_3)$			$N(\eta)$	$t(\eta)$	$\bar{t}(\eta)$
				u_1	u_2	u_3			
	19	12	5	5	7	5	1	−17	71

The algorithm is then applied to the next case automatically. A complete set of input-output cards is listed in table 1 for $l=19$.

On the other hand, *error controls* could stop a case (or the current value of l) before it has run to completion in the following ways:

(a) The first error control computes the numerical approximations to ξ_1, ξ_2, ξ_3, ξ_4 from their components a_j, b_j, c_j and values of ω_i. Thus the maximum error in $\xi_j = a_j\omega_1 + b_j\omega_2 + c_j\omega_3$ is $\epsilon = |a_j|\epsilon_1 + |b_j|\epsilon_2 + |c_j|\epsilon_3$, so that the (scaled) condition $\epsilon_i/m < \epsilon$ would mark the sign discrimination on ξ_i as untrustworthy. In this case an extra card was printed containing l, i, 5, ξ_1^*, ξ_2^*, ξ^*, ξ_4^*, with ξ_j^* being the numerical approximation. The computation proceeded to the next l automatically.

(b) The second error control checked each computed norm to stop the machine if any negative norms appeared or if any ϕ_i failed to have one of the eight neighbors (18). (None did.)

(c) The third error control checked the imminence of overflow by testing to see if the largest term appearing in the norm calculation, $n(a+b+c)^3$, becomes too large ($>2^{44}$) to fit in the registers. In case of this eventuality, the machine proceeded to the next case, printing no signal card. These controls required very little storage.

8. Neighbor Formation and Comparison

What remains is now to describe the portion of the program that is strictly internal, and which, by itself, would not come to the attention of the machine operator. The two main steps of the algorithm are neighbor formation and comparison.

The neighbor formation consists of the formation of ϕ_{i+1} from ϕ_i and, ultimately, the replacement of ϕ_i by ϕ_{i+1}. Here, essentially, each neighbor operation of type (18) was tried in succession on ϕ_i, and the sign discriminations were made to test if the pattern (14c) were valid. Here the numerical estimates were kept in double precision, the integral and fractional parts being stored separately (each consisting of at most about 13 decimal digits).

The comparison of ϕ_0 and ϕ_i is more complex. First of all, the cases $l=7$ and 13 were excluded by virtue of the fact that $|n|=1$ (so that $+\omega_1$ or $-\omega_1$ is a unit anyway). In the other cases ($l\geq 19$), the set of norms in ω_0 (see eq (17b,d)) contains exactly one norm of value 1. Thus a ϕ_i is not conceivably proportional to ϕ_1 unless some column in ϕ_i is a unit (say) η. A further necessary condition is that all the norms in ϕ_i match those of ϕ_0. When this was satisfied the machine multiplied the top row of ϕ_0 by η, η', and η'', in turn (using (9a,b)) testing at each stage to see if ϕ_i or a conjugate was obtained. (In this last step, the possible column rearrangements are fewer if there is only one norm in ϕ_0 equal to 1.)

It is seen that the machine part of the algorithm at no time involved division (even such minor calculations as division by 2 were performed by shifting the binary numbers).

9. Supplementary Hand Computations

The restriction that the whole process be contained in a memory of 1,024 cells proved to be very great. As a result, in the EDVAC program provision was made only for double precision numerical estimates (the integral part and the fractional part being stored separately) and single precision integers. Thus only a few cases ran to completion, but only 18 cases failed to produce at least one unit in some ϕ_i (with or without giving rise to proportionality). In these cases a hand computation produced units from the algebraic numbers automatically computed. The general scheme for these hand computations was the following: Suppose the machine has produced two nonconjugate numbers ξ_1, ξ_2 with same norm, preferably a small prime p. If p is uniquely factorable in the field, then ξ_1 must be associated with a conjugate of ξ_2, as we can test in the manner described earlier in section 3, completely in integral arithmetic. If the norm is not prime, more complicated combinations must be tried, but they are no different in principle.

At least half the hand-computed units were obtained quickly in this fashion. The cases that did not yield to this method directly did so when more than two integers were used in proper combination.

10. Fundamental Units

In each case those units η were noted for which

$$\max t(\eta),\bar{t}(\eta) \tag{20}$$

over all units *produced in the algorithm* was a minimum. In accordance with arguments given by Hasse [6], this would be the fundamental unit if one were present in the list.

By a remarkable coincidence, almost simultaneously with the completion of the runs (Aug. 1954) Peter Swinnerton-Dyer kindly communicated an independent table giving the

traces t, $\bar{t}$ of the fundamental units for $7 \leq l < 300$. A comparison of the tables showed perfect agreement except in the three cases, $l = 373, 379, 463$. In just these cases the (hand) computed units had positive traces for t and $\bar{t}$, which should have suggested that the units computed from the EDVAC output are squares.

In the listing in table 2, the components and traces of both a unit and its reciprocal are shown in each case (here $t(\eta) = \bar{t}(1/\eta)$). In those cases where the units were computed by hand, (*) is shown. In general, regardless of how the units were produced or derived, the symbol "f" indicates the fundamental unit (and reciprocal) and "ff" is reserved for cases where both fundamental unit and reciprocal were produced by machine. In the three exceptional cases, the units, denoted by "s", are squares of fundamental units.

In order, however, to give some kind of reasonable impression of the amount of labor, particularly in view of 13-digit coordinates (!), we should point out that the number of trials in testing fundamental units is easily

$$O\{\log[\max \bar{t}(\eta), t(\eta)]/\log l\}^2. \tag{21}$$

To see this very quickly, note that if η is *not* a fundamental unit, we can write (recalling $\eta\eta'\eta'' = 1$),

$$\left.\begin{aligned} \eta &= \eta_0^a (\eta_0')^b \\ \eta' &= (\eta_0')^a (\eta_0'')^b = \eta_0^{-b} (\eta_0')^{a-b} \\ \eta'' &= (\eta_0'')^a \eta_0^b = \eta_0^{-b-a} (\eta_0')^{-a} \end{aligned}\right\}, \tag{22}$$

where η_0 is some (as yet unknown) fundamental unit, and a and b are rational integers. By taking reciprocals, or conjugates if necessary, we can assure ourselves that $a \geq b \geq 0$. Then

$$\eta^b (\eta')^a = \eta_0^M, \tag{22a}$$

where

$$M = a^2 - ab + b^2. \tag{22b}$$

Letting $\max(|\eta|, |\eta'|, |\eta''|) = \mathrm{H}$, $\max(|\eta_0|, |\eta_0'|, |\eta_0''|) = \mathrm{H}_0$, we find

$$\mathrm{H}^{a+b} > \mathrm{H}_0^M, \tag{23a}$$

or since $(a+b)/M \leq 2a/M \leq 2a/(4a^2/3) = 3/(2a)$,

$$\mathrm{H}_0 \leq \mathrm{H}^{(3/2a)}. \tag{23b}$$

But, by the concluding remarks of section 2,

$$l \leq |\eta_0 - \eta_0'||\eta_0' - \eta_0''||\eta_0'' - \eta_0| < 8\mathrm{H}^3$$

and

$$\mathrm{H}^2 < \eta^2 + (\eta')^2 + (\eta'')^2 = t(\eta)^2 - 2\bar{t}(\eta);$$

thus

$$0 \leq b \leq a \leq \frac{3 \log \mathrm{H}}{2 \log \mathrm{H}_0} \leq 18 \frac{\log 2\,[\max |t(\eta)|, |\bar{t}(\eta)|]}{\log [l/8]}.$$

This verifies the order of magnitude (21).[5]

The problem of showing the units η to be fundamental now resolves itself to checking on whether any of the units $\eta^b(\eta')^a$ is a perfect M power, where a and b come from the finite set (24). But this check is somewhat laborious, even to program for the EDVAC, and owing to the convenience of Swinnerton-Dyer's table, it was never carried out.

---

[5] Much stronger inequalities are given by Hasse [6] using more specialized notation and information.

TABLE 2. *Structure constants and units*

Prime	Structure Constants										Result of algorithm		Components of unit produced: $\eta,^{-1} = u_1\omega_1 + u_2\omega_2 + u_3\omega_3$			Trace of units, $t(\eta)$, $t(\eta^{-1})$
l	m	k	g	n	α_2	β_2	γ_2	α_1	β_1	γ_1	No. of matrices	No. of units	u_1	u_2	u_3	
7	2	0	1	1	1	1	0	−2	−1	−2	special	special	1	0	0	−1
													1	1	0	−2
13	4	−2	1	−1	1	2	1	−4	−2	−3	special	special	−1	0	0	1
													1	2	1	−4
19	6	2	1	7	3	2	1	−4	−4	−5	12 comp	7ff	−1	−1	0	2
													2	1	2	−5
31	10	1	2	8	4	4	2	−7	−6	−8	30 comp	3f	−5	−7	−3	15
													11	17	21	−49
37	12	−4	1	−11	3	5	4	−10	−7	−8	13 comp	3ff	−1	−2	−1	4
													2	2	3	−7
43	14	−3	2	−8	4	6	4	−11	−8	−10	32 over	1f	9	9	7	−25
													−39	−47	−65	151
61	20	0	3	9	7	8	5	−14	−12	−15	32 over	2f	−13	−16	−10	39
													11	14	17	−42
67	22	−2	3	−5	7	9	6	−16	−13	−16	19 comp	1f	−6	−9	−5	20
													52	51	54	−157
73	24	2	3	27	9	9	6	−16	−15	−18	34 over	2f	17	19	13	−49
													−31	−41	−47	119
79	26	−6	1	−41	7	10	9	−20	−16	−17	39 over	9f	−2	−3	−2	7
													3	3	4	−10
97	32	6	1	79	13	10	9	−20	−22	−23	50 over	13f	−3	−3	−2	8
													4	3	4	−11
103	34	4	3	61	13	12	9	−22	−22	−25	46 disc	3f	−12	−14	−9	35
													66	85	93	−244
109	36	−1	4	4	12	14	10	−25	−22	−26	37 disc	1f	521	373	443	−1337
													449	533	627	−1609
127	42	−7	2	−80	12	16	14	−31	−26	−28	43 over	1f	−131	−81	−99	311
													14565	15367	19277	−49209
139	46	−8	1	−103	13	17	16	−34	−29	−30	36 over	8f	−3	−4	−3	10
													4	4	5	−13
151	50	6	3	123	19	17	14	−32	−33	−36	45 disc	1f	20	17	18	−55
													−184	−229	−243	656
157	52	−5	4	−64	16	20	16	−37	−32	−36	36 over	1f*	−10557	−8071	−8883	27511
													2683	2953	3509	−9145
163	54	8	1	169	21	17	16	−34	−37	−38	46 disc	8f	−4	−4	−3	11
													5	4	5	−14
181	60	2	5	67	21	22	17	−40	−38	−43	22 over	1f	12	15	10	−37
													199	233	258	−690
193	64	−8	3	−143	19	24	21	−46	−40	−43	43 disc	1f	142	171	135	−448
													1030	1086	1299	−3415

199	66	−4	5	−59	21	25	20	−46	−41	−46	33 over	2f	−45	−35	−39	119
													95	105	121	−321
211	70	4	5	125	25	25	20	−46	−45	−50	63 over	1f	17	12	16	−45
													233	273	296	−802
223	74	9	2	256	28	24	22	−47	−50	−52	35 over	1f*	−117735	−120801	−96959	335495
													61785	49591	60217	−171593
229	76	7	4	212	28	26	22	−49	−50	−54	23 over	1f*	5673	6737	7111	−19521
													162609	129785	154103	−446497
241	80	−6	5	−125	25	30	25	−56	−50	−55	27 over	1f	−107	−86	−93	286
													87	94	108	−289
271	90	−10	3	−261	27	33	30	−64	−57	−60	30 over	1f*	−37	−19	−28	84
													6029	6254	7316	−19599
277	92	−9	4	−236	28	34	30	−65	−58	−62	34 over	1f	−27	−21	−23	71
													277	291	337	−905
283	94	−11	2	−304	28	34	32	−67	−60	−62	33 over	1f*	14945	17561	14607	−47113
													−38735	−39631	−46569	124935
307	102	5	6	216	36	36	30	−67	−66	−72	30 disc	1f*	−3988957	−4265923	−3500815	11755695
													1233880499	1405925741	1503532433	−4143338673
313	104	−12	1	−371	31	37	36	−74	−67	−68	39 over	4ff	−5	−6	−5	16
													6	6	7	−19
331	110	0	7	49	37	40	33	−74	−70	−77	30 disc	1f	−31	−34	−28	93
													29	32	35	−96
337	112	−2	7	−25	37	41	34	−76	−71	−78	39 over	1f*	127	144	116	−387
													5189	5685	6266	−17140
349	116	12	1	517	43	37	36	−74	−79	−80	57 over	7ff	−6	−6	−5	17
													7	6	7	−20
367	122	−12	3	−435	37	44	41	−86	−78	−81	43 over	1f*	10967	12600	10682	−34249
													654389	671994	771641	−2098024
373	124	4	7	221	43	44	37	−82	−80	−87	35 over	1s*	−863424	−924319	−772262	2560005
													−771297	−862345	−923164	2556806
379	126	−10	5	−365	39	46	41	−88	−80	−85	38 over	1s*	−269376	−304892	−257481	831749
													−258046	−269967	−305561	833574
397	132	11	4	544	48	44	40	−85	−88	−92	60 disc	1f	37	33	35	−105
													−627	−719	−741	2087
409	136	10	5	515	49	46	41	−88	−90	−95	31 disc	1f*	5133	5330	4511	−14974
													495	418	476	−1389
421	140	6	7	343	49	49	42	−92	−91	−98	31 disc	1f*	122	130	109	−361
													1167	1304	1381	−3852
433	144	−1	8	16	48	52	44	−97	−92	−100	50 disc	1f*	−111055	−120569	−102055	333679
													91329	99383	107897	−298609
439	146	9	6	504	52	50	44	−95	−96	−102	49 over	1f*	958722876509	814793743515	917779916345	−2691296536369
													−2609272963	−2939073045	−3070187671	8618533679
457	152	3	8	220	52	54	46	−101	−98	−106	47 over	1f*	−4710533	−5036695	−4281243	14028471
													2763599779	3040707889	3251216741	−9055524409
463	154	−8	7	−343	49	56	49	−106	−98	−105	44 over	1s*	−57726	−49333	−52147	159206
													−49086	−51886	−57437	158409
487	162	8	7	505	57	56	49	−106	−106	−113	33 over	1f*	10649	11170	9561	−31380
													2607	2904	3046	−8557
499	166	−11	6	−536	52	60	54	−115	−106	−112	56 over	1f*	4533791744955	5045136467209	4350456032733	−13929384244897
													−3447237	−3665999	−4104363	11217599

11. Conclusion

The EDVAC produced 176 pages of tabulations, too long to be reproduced here, including some 3,500 algebraic numbers with small norms, in less than 6 hours. With the precision provided for in the coding, only four cases ran to completion, but 25 of the 43 cases run produced at least one unit in the process (for $l=97$, 13 units were produced). The remaining 18 cases provided enough material to permit the hand computation of at least one unit. (For $l=499$, the unit required 43 decimal places to check by hand).

In every case where $g=1$ the explicitly known fundamental unit $(-(k-g+1)/2, -(k-g+1)/2, -(k-g-1)/2$ was among those produced by machine, as well as the (non-fundamental) unit $(-1,-2,0)$, except when $l=37$ and 313.

In addition to units, mentioned earlier, the number of matrices produced after ϕ_0 is shown in table 2, with the following symbols denoting the outcome of the computation (see section 7): "comp" meaning completed, "disc" meaning incomplete (by failure of sign discrimination), "over" meaning incomplete (by overflow in norm formation).

It is clear, however, that to obtain the fullest benefit of this algorithm one would require considerably greater storage, both to perform the unmodified algorithm (guaranteeing a fundamental unit) and to permit sufficient decimal (or preferably integral) precision.

12. References

[1] A. A. Albert, Normalized integral bases of algebraic number fields, Ann. of Math. **38,** 923 (1937).

[2] H. Cohn, A periodic algorithm for cubic forms, I, II, Amer. J. Math. **74,** 821–833 (1952), **76,** 904 (1954).

[3] H. Cohn, Numerical study of signature rank of cubic cyclotomic units, Math. Tables and other Aids to Comput. **47,** 186 (1954).

[4] R. Dedekind, Uber die Anzahl der Idealklassen in reinen kubischen Zahlkörpern, J. Reine Angew. Math. **121,** 123 (1900).

[5] S. Gorn, A computation with algebraic numbers, B. R. L. Report No. 963, Aberdeen Proving Ground, Maryland (1955).

[6] H. Hasse, Arithmetische Bestimmung von Grundeinheit und Klassenzahl in zyklischen kubischer und biquadratischen Zahlkörpern, Abh. Deutsch. Akad. Wiss. Berlin (Berlin, 1950).

[7] E. Hecke, Theorie der algebraischen Zahlen Akad. Verlag, Leipzig, 1923).

[8] G. B. Mathews, Theory of numbers, pt. I, p. 222 (Deighton Bell & Co., Cambridge, 1892).

[9] M. Newman and O. Taussky, On a generalization of the normal bases in abelian algebraic number fields, Comm. Pure Appl. Math. **9,** 85 (1956).

[10] L. Tornheim, Minimal basis and inessential discriminant divisors for a cubic field, Pacific J. Math. **5,** 623 (1955).

WASHINGTON, September 12, 1956.