A SUBALGOL PROGRAM FOR CALCULATION OF MOLECULAR COMPOSITIONAL FORMULAS FROM MASS SPECTRAL DATA

Introduction.

Mass spectrometry is a rapidly growing technique of structural analysis and identification of organic molecules. Its potential usefulness in biochemical analysis is especially exciting, and it is a leading contender for automated analysis of planetary surfaces by landing missions.

The principal datum returned by a mass spectrometer is the precise molecular weight of a molecule or molecular fragment. The computational problem considered here is the establishment of the molecular formulas consistent with this datum. Since the precise atomic weight of each nuclide, except for ^{12}C defined as 12.00000, differs slightly from an exact integral value, a precise mass number is often diagnostic of one or a limited number of formula, depending on the precision with which it has been measured.

Tables to assist this calculation have been computed and published[1,2]. However, they are limited to compounds of C, H, N, and O, and within the most common ranges these elements still require a modest amount of manual computation. As a further step towards the real-time computer-operated control and reduction of mass spectral data a program has been written for the IBM 7090 to accomplish the following extension of the tables:

1. Consideration of all arithmetically admissible values of C, H, N and O.

[redacted]

2. Admission of other elements within specified ranges.

3. Selective printout of acceptable solutions.

Since these extensions potentially admit a very large span of possible combinations, special attention was given to recalculations of scan limits in the nested iterations for values of each element. The program is therefore substantially larger, in coded instructions, than is needed for generation of complete tables; however, it is substantially shorter in computation time since fruitless loops are generally avoided.

The program has been extensively run and tested and is now routinely available for job shop runs (with punch card input and output). It is currently being translated for operation on a small "LINC" computer for on-line use.

The program listing and examples of its output are given following a general description and flow sheet.

Copies of the 7090 program are available on request, either in SUBALGOL or absolute binary decks (available for use under compatible versions of IBSYS-FORTRAN).

FORMULA GENERATOR

Purpose.

This program generates combinations of specified chemical components which will have a given molecular weight.

General Description.

The molecular weight, tolerance, and ranges for the number of carbon, hydrogen, oxygen, nitrogen, sulphur, phosphorus, chlorine, bromine, carbon 13 isotope and deuterium atoms to be considered are provided, and from this information additional chemical and arithmetical restrictions are imposed by the program. Using nested iterative steps, all combinations within these limits are produced, tested against the required weight, and, if found within the tolerance boundaries, recorded as possible solutions.

Because the solutions are accepted primarily on their arithmetic qualifications, many are apt to be chemical impossibilities and must be discarded by the program user. No attempt is made in this program to determine the spatial arrangement or linear sequence of the elements or groups of the proposed molecule.*

Background Chemistry.

A molecular weight is considered to have two distinct parts, the integral and the fractional (INTWT and FRWT). Both these partial weights are the sums of the corresponding parts of the component atomic weights and either may be used independently in deriving a solution. The atomic weights are based on 12.000000 as the weight of one carbon atom.

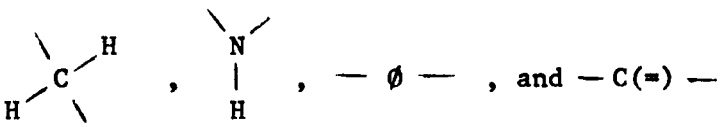
---

* The DENDRAL system (dendritic algorithm) for denoting and generating molecular structures is a subject of further reports.

The main routine of the program considers a molecule composed of only C, H, Ø* and N atoms. To account for other atoms which may be present, the molecular weight is reduced appropriately for each additional substituent in the current trial. This reduced molecule must also be intact (i.e., bonding complete or no free bonding electrons). To maintain this state, each Cl, Br or D atom removed is replaced by an H, and ^{13}C is replaced by ^{12}C. It is not necessary to replace a bivalent atom or group since, for these calculations, an electron pair or new bond will occupy the vacated position. Thus an S atom is simply removed.

The maximum number of H's possible for any weight (MAXH) is extremely useful for limiting the calculations in the main routine and is obtained through the following arguments.

A molecule containing only C, H, Ø and N atoms may be considered a concatenation of bivalent primitives of four types



plus two terminal H atoms. These primitives have weights of 14, 15, 16 and 12; among these the CH_2 group has the highest concentration of H per unit of weight. A saturated hydrocarbon will give the highest proportion of hydrogens: 2(WT/14) (two H for each CH_2 unit), plus 2 (two terminant H atoms).

---

* Ø will be used as a symbol for oxygen, to distinguish it from 0 (zero).

A residue modulo 14* (remainder after division by 14) other than 2 can be obtained by incorporating other primitives. Table 2 presents replacements of CH_2 which will produce the required residue with the least possible reduction of H atoms. This reduction, a function of the residue modulo 14 (R), is given as HVAL(R).

If we take Q = WT/14, the maximum number of H's for a weight WT can now be calculated:

MAXH(WT) = 2Q + HVAL(R) or

MAXH(WT) = 2(WT/14) + HVAL(MOD(WT,14)) ..... (1)

Of the four primitives, only NH can contribute odd numbered values to the total molecular weight; this is the familiar rule that an odd INTWT implies an odd number of NH groups in the solution. The minimum number of nitrogen atoms (NMIN) is made odd or even depending on this condition, and successive N trial values are in increments of 2.

Other maximum and minimum values are established through the use of the following three equations. C, H, Ø and N represent the number of those atoms present:

INTWT = 12C + H + 14N + 16Ø ..... (2)

FRWT = 7825H + 3074N - 5085Ø ..... (3)

MAXH = 2C + N + 2 ..... (4)

Since the largest contribution per atom to the FRWT is made by hydrogen, and finding MAXH does not involve oxygen, the maximum number of Ø atoms (ØMAX) can be determined by restating equation 3 in the following form:

7825MAXH(INTWT - 16Ø) >/(FRWT + 5085Ø - 3074NVAL)

---

* The integers which share a given remainder after division by an integer m are said to belong to a given "congruency class modulo m". For example: 9 is congruent to 149 modulo 14. In computer notation this equivalence is expressed as: R = MOD(I,m), i.e., 9 = MOD(149, 14).

The highest value of Ø which satisfies the condition is accepted as ØMAX.

The Program Description.

Execution of the program begins by reading a data card on which is recorded the molecular weight, tolerance (TOL), whether the molecule may be radical, protonated or intact (RAD), the maximum and minimum values for C, H, Ø and N, and maximum values for S, P, Cl, Br, ^{13}C and D. The experimental weight and corresponding tolerance are recorded as six place decimal numbers with numbers with trailing zeros if necessary. Maximum and minimum numbers may be presented as percentage weights.

The third parameter (RAD) indicates which of the three categories has been described or instructs the program to iterate through the other choices. If a radical or extra proton is to be considered, the weight read is corrected by that of 1 H atom to give an intact molecule.

So that all possible solutions for each molecular weight will be found, the program attempts using INTWT's of 1 unit separation from that given, with the FRWT's adjusted accordingly. This assumes that the tolerance and true fractional weight together will not contribute more than ±1, to the integral weight. Note that $H_{128} \equiv 129.0016$; H has the largest proportional mass fraction.

The range of possible fractional weights for any INTWT is reduced if its lower level is less than -5085(INTWT/16), the maximum Ø contribution, or if its upper level is greater than 7825INTWT, the maximum H contribution.

At the first entry into the main routine, which treats only compositions of C, H, Ø and N, the values of S, P, Cl, Br, D and ^{13}C are set to zero.

With the weights of the reduced molecule established, NMIN, ØMIN and ØMAX are evaluated. Ø is set to ØMIN and the INTWT and FRWT of an oxygen-free molecule are calculated.

The MAXH of INTWT is determined and if found less than (FRWT - TOL) /7825, the program returns to set Ø to its next higher value.

NMAX is evaluated, N is set to NMIN and the integral weight of the remaining C and H atoms is found. Division of this weight by 12 gives the first value of C to be tried as the quotient and the first value of H as the remainder. If C and H are within their respective ranges, and if FRTOT, the total fractional weight of C, H, Ø and N, differs from the FRWT of the reference molecule no more than the given tolerance, the answer is printed. Subsequent tests of C, H and FRTOT are made with one C replaced by 12 H atoms. The replacements continue until

a) C is less than its lower limit, or

b) H exceeds its upper limit or the value (2C + N + 2), or

c) FRTOT exceeds FRWT + TOL.

If N + 2 is not greater than NMAX, N is incremented by 2 and the program following the first setting of N is repeated.

When N has reached NMAX, a similiar comparison is made between Ø + 1 and ØMAX and a larger portion of the program may be repeated with the new value of Ø.

After using ØMAX, increasing values of P, S, Cl, Br, ^{13}C and D are tried. As any value is changed, preceding list members become zero, and for successive tests are incremented in turn to produce all possible combinations.

The printed answers are also output on series of punched cards. Both records will show the given molecular weight, the solution weight, and the

atoms of each type in the calculated composition. Included in the printed results, only, are

a) the molecular percentage weights of C, H, Ø and N based on atomic weights corrected for expected isotopic contributions,

b) the expected abundances of ^{13}C, D, $^{18}Ø$, ^{15}N and ^{34}S, and

c) the sum of ^{13}C, D and ^{15}N abundances giving the expectation at MASS WT + 1, and the sum of ^{34}S and ^{18}O giving the expectation at MASS WT + 2.

The program is written in SUBALGOL, a computer language which is the Stanford University extension of the Burroughs Algebraic Compiler. The sample results shown were obtained after 22 seconds (15 seconds for program compilation and 7 seconds for execution and printing) on the IBM 7090 computer at the Stanford University Computation Center.

REFERENCES

1. Lederberg, J., "Computation of Molecular Formulas for Mass Spectrometry", Holden-Day, Inc., San Francisco, 1964.
2. Lederberg, J., "Tables and an Algorithm for Calculating Functional Groups of Organic Molecules in High Resolution Mass Spectrometry", NASA Scientific and Technical Aerospace Report No. N64-21426, 1964.
3. Kendrick, E., Anal. Chem. 35:2146 (1963).

Table 1

WEIGHTS USED IN PROGRAM CALCULATIONS

	C=12.0 Based Atomic Wts.	Int.Wts.	10^6 x Fr.Wts.	Isotope Exp'tn. Corrected Wts.
H	1.007825	1	7825	1.00797
N	14.003074	14	3074	14.0067
O	15.994915	16	- 5085	15.9994
S	31.972074	32	-27926	32.064
P	30.973763	31	-26237	30.9738
Cl	34.968855	35	-31145	35.453
Br	78.918348	79	-81652	79.909
D	2.014102	2	14102	
^{13}C	13.003355	13	3355	
^{12}C	12.00000	12	0	12.01115

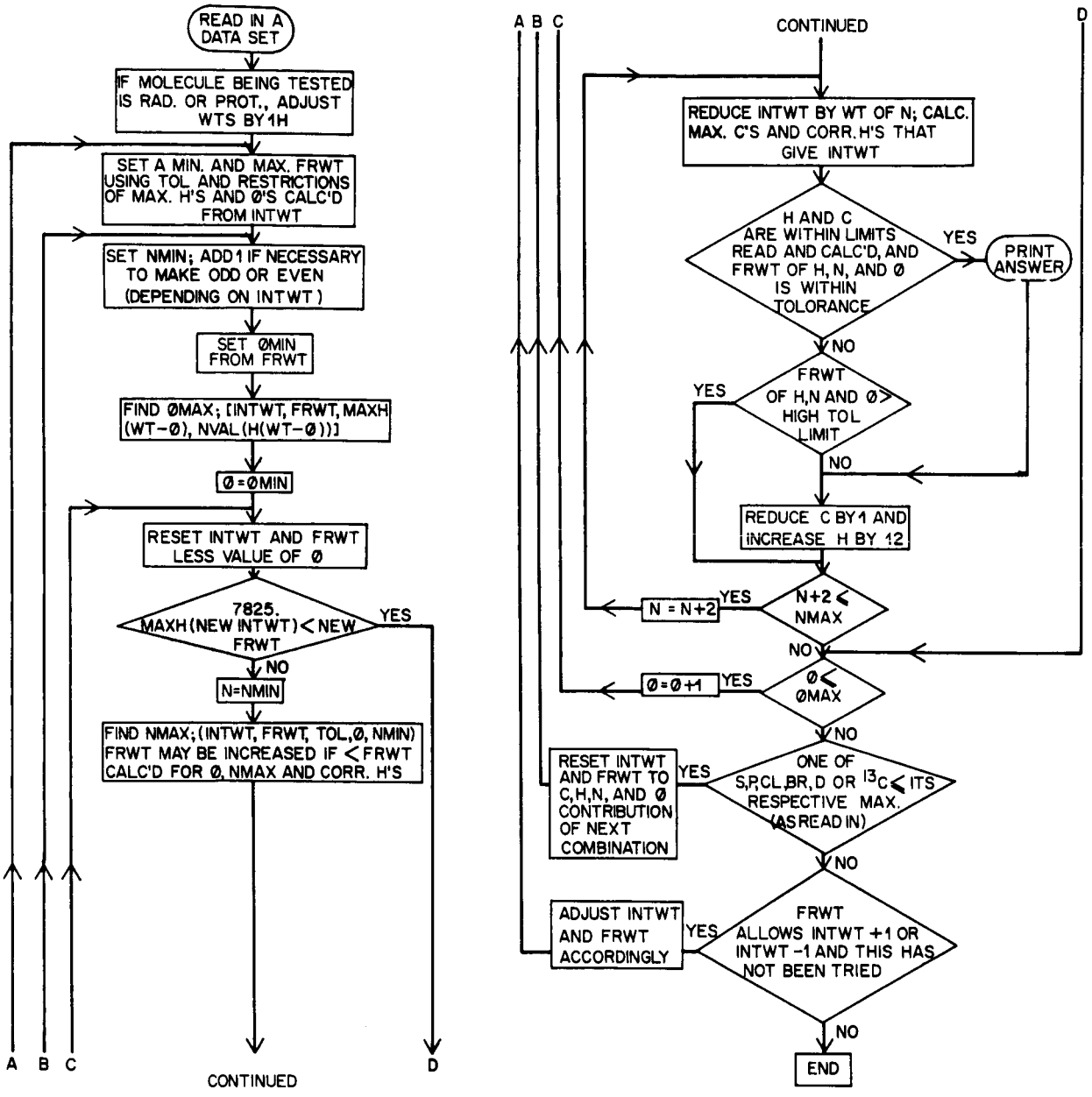
Table 2

FORMULAS CONTAINING THE LARGEST NUMBER OF HYDROGEN ATOMS

MOD(WT,14) R	HVAL (R)	Formula: CH_2 Groups: WT/14-	+ Further Substituents	
0	0	1	C=, 2H	
1	-1	2	C=, NH, 2H	--- to satisfy valency
2	2	0	2H	C=, 2H for another CH_2
3	1	1	NH, 2H	
4	0	2	2NH, 2H	
5	-1	3	3NH, 2H	
6	-2	4	4NH, 2H	
7	-3	5	5NH, 2H	
8	-4	6	6NH, 2H	
9	-5	4	4C=, NH, 2H	" " " "
10	-2	2	3C=, 2H	" " " "
11	-3	3	3C=, NH, 2H	" " " "
12	0	1	2C=, 2H	" " " "
13	-1	2	2C=, NH, 2H	" " " "

Some Alternatives

7	-7	5	5C=, NH, 2H	
8	-4	2	4C=, 2H	" " " "
9	-5	7	7NH, 2H	
10	-6	5	4C=, 2NH, 2H	


READ IN A DATA SET
IF MOLECULE BEING TESTED IS RAD. OR PROT., ADJUST WTS BY 1H
SET A MIN. AND MAX. FRWT USING TOL AND RESTRICTIONS OF MAX. H'S AND Ø'S CALC'D FROM INTWT
SET NMIN; ADD 1 IF NECESSARY TO MAKE ODD OR EVEN (DEPENDING ON INTWT)
SET ØMIN FROM FRWT
FIND ØMAX; [INTWT, FRWT, MAXH (WT−Ø), NVAL (H(WT−Ø))]
Ø=ØMIN
RESET INTWT AND FRWT LESS VALUE OF Ø
7825. MAXH (NEW INTWT) < NEW FRWT
YES
NO
N=NMIN
FIND NMAX; (INTWT, FRWT, TOL, Ø, NMIN) FRWT MAY BE INCREASED IF < FRWT CALC'D FOR Ø, NMAX AND CORR. H'S
A
B
C
CONTINUED
D
A B C
CONTINUED
D
REDUCE INTWT BY WT OF N; CALC. MAX. C'S AND CORR. H'S THAT GIVE INTWT
H AND C ARE WITHIN LIMITS READ AND CALC'D, AND FRWT OF H, N, AND Ø IS WITHIN TOLORANCE
YES
PRINT ANSWER
NO
FRWT OF H,N AND Ø > HIGH TOL LIMIT
YES
NO
REDUCE C BY 1 AND INCREASE H BY 12
N+2 < NMAX
YES
N = N+2
NO
Ø < ØMAX
YES
Ø=Ø+1
NO
ONE OF S,P,CL,BR,D OR ^{13}C < ITS RESPECTIVE MAX. (AS READ IN)
YES
RESET INTWT AND FRWT TO C,H,N, AND Ø CONTRIBUTION OF NEXT COMBINATION
NO
FRWT ALLOWS INTWT +1 OR INTWT −1 AND THIS HAS NOT BEEN TRIED
YES
ADJUST INTWT AND FRWT ACCORDINGLY
NO
END

DATA READ IN

INT WT	MIN FR WT	TOL RANGE	C	H	O	N	D	C13	S	P	CL	BR
EXAMPLE 1												
718	0.368300	12000	0 - 0	0 - 0	8 - 0	5 - 4	0	0	0	0	0	0
EXAMPLE 2												
718	0.368300	12000	0 - 0	0 - 0	0 - 0	0 - 0	0	0	0	0	0	0

ANSWERS PRINTED

MOL WT	PROPOSED FORMULA RD	C	H	O	N	S	P	CL	BR	D	C13	WEIGHT PERCENTAGE C	H	O	N	100 X ABUNDANCE C13	D	O18	N15	S34	ISO. SUM M+1	M+2
EXAMPLE 1																						
718.375715		46	48	3	5							76.95	6.74	6.68	9.75	49.7	.8	6.0	1.9	.00	52.4	6.014
EXAMPLE 2																						
718.378394		49	46		6							81.97	6.46	.00	11.70	52.9	.7	.00	2.3	.00	56.0	.0000
718.371686		40	42		14							66.91	5.90	.00	27.31	43.2	.7	.00	5.3	.00	49.2	.0000
718.376714		15	34		36							25.09	4.77	.00	70.23	16.2	.5	.00	14.	.00	30.5	.0000
718.370006		6	30		44							10.04	4.21	.00	85.83	6.48	.5	.00	17.	.00	23.7	.0000
718.378897		34	42	1	18							56.88	5.90	2.23	35.11	36.7	.7	2.0	6.9	.00	44.3	2.005
718.372189		25	38	1	26							41.82	5.33	2.23	50.72	27.0	.6	2.0	9.9	.00	37.5	2.005
718.377217			30	1	48							.00	4.21	2.23	93.64	.000	.5	2.0	18.	.00	18.8	2.005
718.374372		44	46	2	8							73.61	6.46	4.46	15.61	47.5	.7	4.0	3.1	.00	51.3	4.010
718.379400		19	38	2	30							31.78	5.33	4.46	58.52	20.5	.6	4.0	11.	.00	32.6	4.010
718.372692		10	34	2	38							16.73	4.77	4.46	74.13	10.8	.5	4.0	14.	.00	25.8	4.010
718.374875		29	42	3	20							48.51	5.90	6.68	39.02	31.3	.7	6.0	7.6	.00	39.6	6.014
718.379903		4	34	3	42							6.69	4.77	6.68	81.93	4.32	.5	6.0	16.	.00	20.9	6.014
718.377058		48	50	4	2							80.30	7.02	8.91	3.90	51.9	.8	8.0	.76	.00	53.4	8.019
718.370350		39	46	4	10							65.24	6.46	8.91	19.51	42.1	.7	8.0	3.8	.00	46.7	8.019
718.375378		14	38	4	32							23.42	5.33	8.91	62.43	15.1	.6	8.0	12.	.00	27.9	8.019
718.368670		5	34	4	40							8.36	4.77	8.91	78.03	5.40	.5	8.0	15.	.00	21.2	8.019
718.377561		33	46	5	14							55.20	6.46	11.14	27.31	35.7	.7	10.	5.3	.00	41.7	10.02
718.370853		24	42	5	22							40.15	5.90	11.14	42.92	25.9	.7	10.	8.4	.00	35.0	10.02
718.373036		43	50	6	4							71.93	7.02	13.37	7.80	46.5	.8	12.	1.5	.00	48.8	12.03
718.378064		18	42	6	26							30.11	5.90	13.37	50.72	19.5	.7	12.	9.9	.00	30.0	12.03
718.371356		9	38	6	34							15.06	5.33	13.37	66.33	9.73	.6	12.	13.	.00	23.3	12.03
718.380247		37	50	7	8							61.90	7.02	15.60	15.61	40.0	.8	14.	3.1	.00	43.8	14.03
718.373539		28	46	7	16							46.84	6.46	15.60	31.21	30.3	.7	14.	6.1	.00	37.1	14.03
718.378567		3	38	7	38							5.02	5.33	15.60	74.13	3.24	.6	14.	14.	.00	18.3	14.03
718.369014		38	50	8	6							63.57	7.02	17.83	11.70	41.1	.8	16.	2.3	.00	44.2	16.04
718.374042		13	42	8	28							21.75	5.90	17.83	54.62	14.0	.7	16.	11.	.00	25.4	16.04
718.376225		32	50	9	10							53.53	7.02	20.05	19.51	34.6	.8	18.	3.8	.00	39.2	18.04
718.369517		23	46	9	18							38.48	6.46	20.05	35.11	24.9	.7	18.	6.9	.00	32.5	18.04
718.371700		42	54	10								70.26	7.58	22.28	.00	45.3	.9	20.	.00	.00	46.2	20.05
718.376728		17	46	10	22							28.44	6.46	22.28	42.92	18.4	.7	20.	8.4	.00	27.5	20.05
718.370020		8	42	10	30							13.38	5.90	22.28	58.52	8.64	.7	20.	11.	.00	20.8	20.05
718.378911		36	54	11	4							60.22	7.58	24.51	7.80	38.9	.9	22.	1.5	.00	41.3	22.05
718.372203		27	50	11	12							45.17	7.02	24.51	23.41	29.2	.8	22.	4.6	.00	34.6	22.05
718.379414		21	50	12	16							35.13	7.02	26.74	31.21	22.7	.8	24.	6.1	.00	29.6	24.06
718.372706		12	46	12	24							20.07	6.46	26.74	46.82	13.0	.7	24.	9.2	.00	22.9	24.06
718.374889		31	54	13	6							51.86	7.58	28.97	11.70	33.5	.9	26.	2.3	.00	36.7	26.06
718.375392		16	50	14	18							26.77	7.02	31.20	35.11	17.3	.8	28.	6.9	.00	25.0	28.07
718.377575		35	58	15								58.55	8.14	33.42	.00	37.8	.9	30.	.00	.00	38.7	30.07
718.370867		26	54	15	8							43.49	7.58	33.42	15.61	28.1	.9	30.	3.1	.00	32.0	30.07
718.378078		20	54	16	12							33.46	7.58	35.65	23.41	21.6	.9	32.	4.6	.00	27.1	32.08
718.373553		30	58	17	2							50.19	8.14	37.88	3.90	32.4	.9	34.	.76	.00	34.1	34.08

```
S744364BALGOL   1       400         WIGHTMAN          FORMULA GENERATOR
                                STANFORD UNIVERSITY COMPILER -- VERSION OF  1/27/64

      *-SPACE                                                                    .
    144...    INTEGER OTHERWISE $                                                .
    144...    RESERVEDWORD EXTR,SHRT,SHLT $                                      .
    144...    ARRAY FRAGPKS(0..600)$                                             .
    144...    ARRAY COR(10), FRKP(0..50) $                                       .
    144...    ARRAY HVAL(0..13) = (0,-1,2,1,0,-1,-2,-3,-4,-5,-2,-3,0,-1 )$       .
    144...    ARRAY NVAL(0..13) = (0,1,0,1,2,3,4,5,6,7,0,1,0,1 )$                .
    144...    REAL ARRAY FLNOS(0..10)$                                           .
    144... PROCEDURE SPACE(N)$ BEGIN INTEGER OTHERWISE$                          .
    155...    IF NOT N$ (LINECOUNT=0$ WRITE($$ PAGE)$)$                          .
    162...    IF N$ (LINECOUNT=LINECOUNT+N$ WRITE($$ SPACES)$)$                  .
    171...    FORMAT PAGE(W3), SPACES($ABS(N)$W)$ RETURN$ END SPACE()$           .
    206... INTEGER FUNCTION MAXH(WT) = 2(WT/14) + HVAL(MOD(WT,14)) $             .
    234... SUBROUTINE FINDOMAX $  BEGIN    OMAX = OMIN $                        .
    242...    OOMAX = INTWT/16 $ IF HIO NEQ 0 $ OOMAX = MIN(HIO,OOMAX)$          .
    254...    OM1..  TESTWT = INTWT - 16(OMAX + 1 )$                             .
    262...       IF (OMAX +1) GTR OOMAX $ RETURN $                               .
    270...       IF 7825MAXH(TESTWT) GEQ (FRWT+5085(OMAX+1)-3074(NVAL(MOD(TESTWT, .
    316...          14)) ))$ (OMAX = OMAX+1$ GO OM1 )$  RETURN END FINDOMAX $    .
    325... SUBROUTINE FINDNMAX $ BEGIN  NMAX =  NMIN $                           .
    333...    FOR I = (49,-1,0)$ FRKP(I+1) = 0 $                                 .
    352...    NMAXO = II/14$ IF HN NEQ 0$ NMAXO = MIN(HN,NMAXO)$                 .
    366...    NM1..      IF(NMAX + 2) GTR NMAXO$ RETURN$                         .
    374...       HMAXNFR=(  FRWT -3074(NMAX+2)+508500)$                          .
    410...       HMAXN = HMAXNFR /7825$                                          .
    415...       IF HMAXNFR LSS 0 $ (IF (HMAXNFR + TOL) LSS 0$ RETURN $          .
    425...           HMAXN = 0)$                                                 .
    426...       IF 14(NMAX+2) LEQ (INTWT  -1600 -HMAXN -12(HMAXN                .
    452...          -(NMAX+2) -2)/2 ) $   ( NMAX = NMAX +2 $                     .
    457...          IF HMAXNFR LSS 0 $   FRKP(NMAX) = -HMAXNFR $  GO NM1 )$      .
    466...    RETURN END FINDNMAX $                                              .
    467... SUBROUTINE PRINTANSWER $  BEGIN                                       .
    500...       OO = OO + 3(PP) $       CC = CC - C13 $                         .
    503...       HH = HH - DU + PP - CLL - BRR + RAD $                           .
    512...       INTOUT = INTOT +1(RAD EQL 1) -1(RAD EQL -1) +32SS +80PP +34CLL+ .
    555...          78BRR +DU +C13 $                                             .
    557...       FROUT = FRTOT +7825(RAD EQL 1) -7825(RAD EQL -1) +27926SS+      .
    635...          33667PP -38970CLL -89477BRR + 6277DU +3354C13 $              .
    643...    CORCUT = 12.01115(CC+C13)+1.00797(HH+RAD+DU)+15.999400 + 14.0067NN .
    716...          + 32.064SS + 30.9738PP + 35.453CLL + 79.909BRR $             .
    726...    FLNOS(0) = 100.0/CORCUT $  FLNOS(1) = 12.01115CC.FLNOS(0) $        .
    742...    FLNOS(2) = 1.00797(HH + RAD).FLNOS(0)$FLNOS(3) = 15.9994(OO).FLNOS .
    761...          (0)$  FLNOS(4) = 14.0067NN.FLNOS(0)$  FLNOS(5) = 1.0806(CC)$ .
    775...    FLNOS(6) = 0.0160(HH) $    FLNOS(7) = 2.0048(OO) $                 .
   1007...    FLNOS(8) = 0.3815(NN) $    FLNOS(9) = 4.4  (SS) $                  .
   1021...    WRITE($$ ANAN,SWER )$                                             .
   1024...    IF PUNCHCARDS $ WRITE($$CARDR,ECORD)$                             .
   1031...    HH = HH + DU - PP + CLL + BRR - RAD $                             .
   1040...    OC = OC - 3PP $   CC = CC + C13 $                                 .
   1051...    OUTPUT CARDR(INTOUT-1(FROUT LSS 0),FROUT +1000000(FROUT LSS 0),   .
   1127...       CC,HH,OO,NN,SS,PP,CLL,BRR,C13,DU,INWT,FRTN,TL)$                .
   1132...    FORMAT ECORD(I4,*.*,L6,B3, $CC GTR 0$I2,$CC LEQ 0$(Z,B2), $HH GTR 0$.
   1251...       I3,$HH LEQ 0$(Z,B3),$OO GTR 0$I3,$OO LEQ 0$(Z,B3),$NN GTR 0$I3,  .
   1323...       $NN LEQ 0$(Z,B3), $SS GTR 0$I3,$SS LEQ 0$(Z,B3), $PP GTR 0$I3,   .
   1375...       $PP LEQ 0$(Z,B3), $CLL GTR 0$I3,$CLL LEQ 0$(Z,B3), $BRR GTR 0$I3,.
   1447...       $BRR LEQ 0$(Z,B3),$C13 GTR 0$I4,$C13 LEQ 0$(Z,B4),$DU GTR 0$I3,  .
   1511...       $DU LEQ 0$(Z,B3), $RAD EQL 1$(* PRO*),  $RAD EQL 0$(B4),         .
```

```
1545...       $RAD EQL -1$(* RAD*),B2,$FRAGMENT EQL O$(I4,*.*,L6,B3,I6),        .
1566...       $FRAGMENT GTR O$(Z,B11,Z,B9,Z),P)$                                .
1566...    OUTPUT ANAN(INTOUT -1(FROUT LSS O), FROUT +1COOOOO(FROUT LSS O),     .
1651...       CC,HH,OO,NN,SS,PP,CLL,BRR,DU,C13,FOR I=(1,1,9)$FLNOS(I),          .
1665...       FLNLS(5) + FLNOS(6) + FLNUS(8) , FLNOS(7) + FLNOS(9) )$           .
1667...    FORMAT SWER(I4,*.*,L6,B1,$RAD EQL 1$('*'),$RAD EQL -1$('/'),$RAD EQL.
1775...       C$B1,B3,                $CC GTR O$I4,$CC LEQ O$(Z,B4),$HH GTR O$I3,.
2050...       $HH LEQ C$(Z,B3),$OO GTR O$I3,$OO LEQ O$(Z,B3),$NN GTR O$I3,$NN   .
2122...       LEQ O$(Z,B3),$SS GTR O$I3,$SS LEQ C$(Z,B3),$PP GTR O$I3,$PP LEQ   .
2174...       C$(Z,B3),$CLL GTR O$I3,$CLL LEQ C$(Z,B3),$BRR GTR O$I3,$BRR LEQ   .
2246...       O$(Z,B3),$DU GTR O$I3,$DU LEQ O$(Z,B3),$C13 GTR O$ I3,$C13 LEQ    .
2300...       O$(Z,B3),B4,4(X6.2),B10,S4.3,B2,S2.1,B2,S3.2,B2,S3.2,B2,S3.2,B2,  .
2305...       S4.3,B1,S5.4, W )$                                                .
2305...     RETURN END  PRINTANSWER $                                           .
2306... UNTIL SNTL $  BEGIN                                                     .
2313...    READ($SNTL$CONTROLS)$  RCARD($SNTL$MAS,SWT)$      SPACE(0)$          .
2321...    IF SNTL $  GO NEXT $                                                 .
2323...    WRITE($$ECH,OREAD)$                                                  .
2326...    IF PUNCHCARDS$  WRITE($$TITLECD)$                                    .
2332...    IF CP EQL 39$  BEGIN                                                 .
2342...       HC = (HC.INWT)/1200 + 1 $                                         .
2345...       IF LC NEQ O$  LC = (LC.INWT)/1200 $   END $                       .
2355...    IF HP EQL 39$  BEGIN                                                 .
2365...       HIH = (HIH.INWT)/100 + 1 $                                        .
2370...       IF LH $ LH = (LH.INWT)/100 $ END $                                .
2400...    IF CP EQL 39$  BEGIN                                                 .
2410...       HIO = (HIO.INWT)/1600 + 1 $                                       .
2413...       IF LO $ LO = (LO.INWT)/1600 $  END $                              .
2423...    IF NP EQL 39$  BEGIN                                                 .
2433...       HN = (HN.INWT)/1400 + 1 $                                         .
2436...       IF LN $ LN = (LN.INWT)/1400 $  END $                              .
2446...    IF TL  EQL -1$  TL  = 10INWT $                                       .
2454...    ISC =(C13 NEQ 0) OR(DEU NEQ 0)$                                      .
2464...    HET = (SUL NEQ 0) OR (PHOS NEQ 0) OR(CL NEQ C) OR (BR NEQ 0) $       .
2500...    RAD = RD $                                                           .
2502...    IF RD EQL 3 $(RAD = 0$ GO SETRAD)$                                   .
2507...    IF RD EQL 2 $(RAD = 1$ GO SETRAD)$                                   .
2515...    COR(1) = INWT $  COR(2) = FRTN $                                     .
2521... SETRAD.. IF (RAD EQL -1)$ (COR(1) = INWT + 1 $ COR(2)= FRTN +7825 )$    .
2532...          IF (RAD EQL  1)$ (COR(1) = INWT - 1 $ COR(2)= FRTN -7825 )$    .
2543...    COR(3) = COR(1)$ COR(4) = COR(2)$                                    .
2547...    MINFR = COR(4) - TL  $   MAXFR = COR(4) + TL  $                      .
2555... SETWTS..                                                                .
2565...    IF 7825(COR(3) -1) GEQ (1000000 +MINFR)$   BEGIN                     .
2572...       COR(3) = COR(3) -1 $  COR(4) = MINFR + 1000000 $                  .
2575...       TOL = MAX(7825COR(3),1000000 + MAXFR) - COR(4) $                  .
2610...       ENTER MOLCOMP  $ COR(3) = COR(1) $  COR(4) = COR(2) $      END $  .
2615...    COR(4) = MAX(MINFR, -5085(COR(3)/16) )$                              .
2630...    TOL = MIN(MAXFR,7825COR(3))-COR( 4) $                                .
2640...    ENTER MOLCOMP $                                                      .
2641...    COR(3) = COR(1) $ COR(4) = COR(2) $                                  .
2645...    IF (1000000 -5085(COR(3)+1)/16) LEQ MAXFR $  BEGIN                   .
2662...       COR(3) = COR(3) + 1 $  COR(4) = MAX(-5085(COR(3)/16),-1000000 +   .
2677...       MINFR) $  TOL = (-1000000 + MAXFR) - COR(4) $                     .
2703...       ENTER MOLCOMP $  END $                                            .
2704...    GO CHECKRAD $                                                        .
2705... SUBROUTINE MOLCOMP $ BEGIN                                              .
2711...    DU = C13 = SS = PP = CLL = BRR = O$                                  .
2717...    COR(5) = COR(3) $  COR(6) = COR(4) $                                 .
2723...    SPACE(0) $                                                           .
```

```
2725...     WRITE($$DAT,AREAD )$                                                .
2730...     WRITE($$HEADING )$                                                  .
2732... SETISOTOPES..    SS = PP = CLL = BRR = 0 $                              .
2736... CALC..    INTWT = COR(5) $ FRWT = COR(6) $                              .
2742...     OMIN = MAX( -(FRWT+TOL)/5085,LO)$                                   .
2757...     NMIN = MAX(0,LN) + (MOD(NMIN,2) NEQ MOD(INTWT,2)) $                 .
3004...     OC = OMIN $  ENTER FINDOMAX $                                       .
3007... SETO..BEGIN                                                             .
3013...         II = INTWT - 1600 $  DD = FRWT + 508500 $                       .
3020...         HHMAX = MAXH(II) $  IF HIH NEQ 0 $ HHMAX = MIN(HHMAX,HIH )$     .
3032...         HHMAXFR = 7825HHMAX $  LODD = DD - 3074(NVAL(MOD(II,14)) )$     .
3050...         IF LODD GTR HHMAXFR $ GO NEXTO$                                 .
3054...         NN =  NMIN $  ENTER FINDNMAX $                                  .
3057... SETN..    BEGIN                                                         .
3071...           CHWT = INTWT - (1600 + 14NN)$IF CHWT LSS 0$ GO ENDCH$         .
3073...           CC = CHWT/12$ HH = MOD(CHWT,12)$                              .
3105... TRIAL..   IF CC LSS LC$ GO ENDCH$                                       .
3111...           IF(HH GTR HHMAX) OR                                           .
3122...             (HH GTR (2CC + NN+2))$ GO ENDCH$                            .
3124...           IF HC NEQ 0$ IF CC GTR HC$ GO ALTERCH $                       .
3132...           IF HH LSS LH $  GO ALTERCH $                                  .
3136...           INTOT = 12CC + HH + 14NN + 1600$                              .
3154...           FRTOT = 7825HH + 3074NN - 508500 $                            .
3172...           EITHER IF (FRTOT LEQ (FRWT + TOL)) AND (FRTOT GEQ FRWT)$      .
3201...               ( ENTER PRINTANSWER $  GO ALTERCH )$                      .
3205...           OR IF FRTOT GTR FRWT$       GO ENDCH $                        .
3213...           OTHERWISE$ GO ALTERCH $                                       .
3215... ALTERCH..       (CC = CC-1$ HH = HH+12$ GO TRIAL)$                      .
3224... ENDCH..                                                                 .
3226...         IF FRKP(NN) NEQ 0 $FRWT = FRKP(NN) $                            .
3231...         NN = NN+2 $  IF NN LEQ NMAX $  ( IF FRKP(NN) NEQ 0 $ (          .
3245...           I = FRKP(NN) $ FRKP(NN) = FRWT$ FRWT = FRWT +I )$ GO SETN )$  .
3253... ENDN..    END SETN $    NEXTO..                                         .
3256...     OC = OC+1 $ IF OC LEQ OMAX $  GO SETO $                             .
3262... ENDO..    END SETO $                                                    .
3262...     IF NOT HET $ GO ENDHETATOMS $                                       .
3264...     IF SS LSS SUL $(SS=SS+1 $ COR(5) =COR(5) -32 $ COR(6) = COR(6)+27926.
3301...       $  GO CALC )$                                                     .
3302...     IF PP LSS PHOS$(PP=PP+1$COR(5)=COR(5) -80 +32SS $ COR(6) =COR(6)    .
3331...       +33667 -27926SS $ SS = 0 $ GO CALC )$                             .
3333...     IF CLL LSS CL $ (CLL =CLL+1 $ COR(5)= COR(5) -34 +80PP +32SS $      .
3356...       COR(6) =COR(6) +38970 -33667PP -27926SS $ PP=SS=0$ GO CALC )$     .
3377...     IF BRR LSS BR $( BRR=BRR+1 $  COR(5)=COR(5) -78 +34CLL +80PP +32SS $.
3427...       COR(6)=COR(6) +89477 -38970CLL -33667PP -27926SS $ CLL=PP=SS=0 $  .
3455...       GO CALC )$                                                        .
3456... ENDHETATOMS..                                                           .
3457...     IF NOT ISO $ GO ENDSETISO $                                         .
3460...       IF DU LSS DEU $ (DU=DU+1 $ COR(5) = COR(3) - DU -C13 $            .
3473...         COR(6) = COR(4) - 6277DU -3354C13 $ GO SETISOTOPES )$           .
3507...       IF C13 LSS C13 $ (C13 = C13 +1 $ COR(5) = COR(3) -C13 $           .
3521...         COR(6) = COR(4) -3354C13 $ DU = 0 $ GO SETISOTOPES )$           .
3531... ENDSETISO..                                                             .
3532...     RETURN END MOLCOMP $                                                .
3532... CHECKRAD..                                                              .
3537...     IF (RD EQL 2) AND (RAD EQL 1) $ ( RAD = -1 $ GO SETRAD )$           .
3543...     IF((RD EQL 3) AND (RAD EQL 0))$                                     .
3547...       (RAD = 1$ GO SETRAD)$                                             .
3553...     IF ((RD EQL 3) AND (RAD EQL 1))$                                    .
3560...       (RAD = -1$ GO SETRAD)$                                            .
3564...     END $                                                               .
```

```
3565...     INPUT CONTROLS(PUNCHCARDS,TIMECHECK,DEBUG,FRAGMENT )$                .
3600...     INPUT MAS(NAME1,NAME2,INWT,FRTN,TL,RD,HC,LC,CP,HIH,LH,HP,HIO,LO,OP, .
3660...        HN,LN,NP,DEU,C13,SUL,PHOS,CL,BR )$                                .
3663...     FORMAT SWT(R, B1, 2A6, B2,I4,B, I6,B2, I6,B, I2,B3, I2,B1,I2,A1.5,   .
3725...        I2,B1,I2,   A1.5,I2,B1,I2,   A1.5,I2,B1,I2,   A1.5,I3,B1,I2,B2,   .
3736...        I1,B1, I1,B1, I1,B1, I1,B1 )$                                     .
3736...     OUTPUT ECH( NAME1,NAME2,INWT,FRTN,TL,RD,HC,LC,HIH,LH,HIO,LO,HN,LN,   .
4006...        DEU,C13,SUL,PHOS,CL,BR)$                                          .
4011...     FORMAT OREAD(B25,*DATA  READ  IN  FOR   *,2A6,W,*INT WT   FR WT   *,.
4043...        * TCL    RAD      C      H      O      N      S   P   CL   BR  *,.
4076...        * D  C13*,W,I5,B3,L6,I8,I6,B3,I3,*-*,I2,$CP EQL 39$('P'),$CP NEQ .
4144...        39$B1,I3,*-*,I2,$HP EQL 39$('P'),$HP NEQ 39$B1,I3,*-*,I2,$OP EQL .
4221...        39$('P'),$OP NEQ 39$B1,I3,*-*,I2,$NP EQL 39$('P'),$NP NEQ 39$B1,  .
4226...        2I4,2I5,2I4, W)$                                                  .
4226...     FORMAT TITLECD(' FORMULA WT    C  H  O  N  S  P CL BR  C*  D',       .
4250...        B8 ,*WT READ IN    TOL*, P)$                                      .
4250...     OUTPUT DAT(NAME1,NAME2,COR(5)-1(COR(6) LSS 0), COR(6)-1000000(COR(6).
4336...        GEQ 1000000)+1000000(COR(6) LSS 0), TOL,HC,LC,HIH,LH,HIO,LO,HN,LN.
4352...        ,DU,C13,SUL,PHOS,CL,BR)$                                          .
4355...     FORMAT AREAD(2A6,B5,*DATA CONSIDERED*,W2,* INT WT   FR WT*,B7,*TOL*,.
4411...        B7,*C       H       O       N       D   C13    S    P   CL   BR*,.
4447...        W, B2,I4,B2,$COR(6) GEQ 1000000$(*1.*),$COR(6) LSS 1000000$(*0.*).
4476...        ,L6,B3,I7,B3,I2,B1,*-*,2I3,B1,*-*,2I3,B1,*-*,2I3,B1,*-*,I3,I5,I5,.
4503...        B4,I2,3I5, W2)$                                                   .
4503...     FORMAT HEADING(W2,B13,*PROPOSED   FORMULA*,B23,*WEIGHT  PERCENTAGE*,.
4537...        B16,*100 X  ABUNDANCE*,B7,*ISO.  SUM*,W,*  MOL WT   RD      C  H*.
4552...       ,*  C  N  S  P CL BR  D C13      C     H     O     N*,B12 ,          .
4563...        *C13   D   O18  N15  S34   M+1  M+2*, W2 )$                       .
4563...     OUTPUT ARES(INTOT -1(FRTOT LSS 0), FRTOT +10C0000(FRTOT LSS 0),      .
4625...        OO,NN,HH,CC,OMAX,NMAX,NMAXO )$                                    .
4630...     FORMAT  ULT (B10,I5,*.*,L6,B4,I2,3I4,B3,3I6,W)$                      .
4646... NEXT..                                                                   .
4646... FINISH $                                                                 .
COMPILED PROGRAM ENDS AT     4647
LIBRARY PROGRAMS END AT     15374
PROGRAM VARIABLES BEGIN AT  75752
```