

404653

633-4



NRL Report 5885

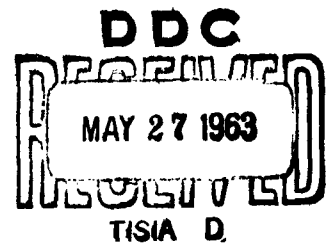
A COLLECTION OF FORTRAN PROGRAMS FOR CRYSTAL STRUCTURE ANALYSIS

H. G. Norment

Diffraction Branch
Optics Division

April 29, 1963

404 653



U. S. NAVAL RESEARCH LABORATORY
Washington, D.C.

CONTENTS

Preface	iv
Problem Status	iv
Authorization	iv
 1. UNIT CELL PARAMETERS AND ERRORS BY LEAST SQUARES	 1 1
1.1 Purpose	1
1.2 Input	1
1.3 Method of Calculation	2
1.4 Output	3
1.5 Limitations	3
1.6 Special Subroutines Called	3
1.7 Special Tape Requirements	3
 2. QUASI NORMALIZATION OF STRUCTURE FACTORS	 3
2.1 Purpose	3
2.2 Input	3
2.3 Method of Calculation	4
2.4 Output	4
2.5 Limitations	5
2.6 Special Subroutines Called	5
2.7 Special Tape Requirements	5
 3. RATIONAL DEPENDENCE	 5
3.1 Purpose	5
3.2 Input	5
3.3 Method of Calculation	6
3.4 Output	7
3.5 Limitations	7
3.6 Special Subroutines Called	7
3.7 Special Tape Requirements	7
 4. SIGMA-2 LISTINGS	 7
4.1 Purpose	7
4.2 Input	7
4.3 Method of Calculation	8
4.4 Limitations	8
4.5 Special Subroutines Called	8
4.6 Special Tape Requirements	9
 5. TRIPLE PRODUCT SUMMATION FOR ORTHORHOMBIC CRYSTALS	 9 9
5.1 Purpose	9
5.2 Input	9
5.3 Method of Calculation	10

5.4 Output	11
5.5 Limitations	11
5.6 Special Subroutines Called	11
5.7 Special Tape Requirements	11
6. STRUCTURE FACTOR CALCULATION	11
6.1 Purpose	11
6.2 Input	11
6.3 Data Deck	15
6.4 Method of Calculation	15
6.5 Printed Output	16
6.6 Tape Output	17
6.7 Special Subroutines Called	17
6.8 Special Tape Requirements	17
7. INTERATOMIC DISTANCES AND ANGLES WITH INTERPOLATION FOR PEAK CENTER LOCATION	17
7.1 Purpose	17
7.2 Input	18
7.3 Data Deck	20
7.4 Method of Calculation	20
7.5 Output	20
7.6 Limitations	21
7.7 Special Subroutines Called	21
7.8 Special Tape Requirements	21
8. LEAST-SQUARES PLANE AND LINE FITTER	21
8.1 Purpose	21
8.2 Input	21
8.3 Data Deck	23
8.4 Method of Calculation	23
8.5 Output	23
8.6 Limitations	24
8.7 Special Subroutines Called	24
8.8 Special Tape Requirements	24
9. POINT TO PEAK DISTANCE CALCULATION	24
9.1 Purpose	24
9.2 Input	24
9.3 Data Deck	25
9.4 Method of Calculation	25
9.5 Output	26
9.6 Limitations	26
9.7 Special Subroutines Called	26
9.8 Special Tape Requirements	26
10. FORM FACTORS FOR THE BUSING LEAST-SQUARES REFINEMENT PROGRAM	26
10.1 Purpose	26
10.2 Input	26

10.3 Data Deck	26
10.4 Method of Calculation	27
10.5 Output	27
10.6 Special Subroutines Called	27
10.7 Special Tape Requirements	27
 11. VARIANCE-COVARIANCE MATRIX AND ATOMIC COORDINATE INPUT FOR THE BUSING FUNCTION AND ERROR PROGRAM	 27
11.1 Purpose	27
11.2 Input	27
11.3 Data Deck	28
11.4 Output	28
11.5 Limitations	29
11.6 Special Subroutines Called	29
11.7 Special Tape Requirements	29
 12. OTHER FUNCTIONS AND SUBROUTINES	 29
12.1 Function ARCSIN(X)	29
12.2 Subroutine RECIP (AR, BR, CR, ALR, BER, GAR, AA, BB, CC, COSAL, COSBE, COSGA)	29
12.3 Function DOTPRD (U, V, W, X, Y, Z, B1, B2, B3, COSA, COSB, COSC)	30
12.4 Function SCAFAC (A, A1, B, B1, C, S)	30
12.5 Subroutine MTXMUL (L, M, N, A, B, C)	30
12.6 Function INTG(A)	31
 REFERENCES	 31
 APPENDIX A - FORTRAN Listings of General Utility Programs Applicable to Crystal Structure Analysis	 32

PREFACE

This report contains descriptions and operating instructions for a collection of general utility programs applicable to various phases of crystal structure analysis with computers. All coding is done in FORTRAN II language for 32K core IBM 704 and 7090 machines; the FORTRAN-language programs are completely listed in the Appendix.

The data input to many of the programs has been designed to be compatible with the output of the automatic data reduction program XRDDR (H.G. Norment, "An X-Ray Diffraction Data Reduction Program for the IBM 704 and 7090," NRL Report 5739, Feb. 1962).

Where data input from special tape is required, the logical tape numbers are assigned symbolic designations near the beginnings of the FORTRAN programs. Thus, if it is required that these tape numbers be changed, it is sufficient to change only the one FORTRAN statement card and then recompile.

In the form presented in this report, the programs are written for use with the IBM 7090 IB Monitor system.

PROBLEM STATUS

This is a final report on one phase of the problem; work on other phases continues.

AUTHORIZATION

NRL Problem C07-03
Project RR 001-02-43-4805

Manuscript submitted November 21, 1962.

A COLLECTION OF FORTRAN PROGRAMS FOR CRYSTAL STRUCTURE ANALYSIS

1. UNIT CELL PARAMETERS AND ERRORS BY LEAST SQUARES

1.1 Purpose

Given a set of reflection data consisting of Miller indices and $\sin \theta$ or $\sin \theta/\lambda$ values, compute the least-squares reciprocal and real-space unit cell parameters and uncertainties for any crystal system.

1.2 Input

The data input is taken from cards immediately following the program deck. It consists of one control card and a deck of reflection cards.

A. Control Card:

columns	1-2	3-4	5-12	13-72
data	N1	LIST	YAM	Hollerith information
formats	I2	I2	E8.5	10A6

1. N1 specifies the crystal as follows:

Crystal System	N1
triclinic	1
monoclinic*	2
orthorhombic	3
tetragonal	4
hexagonal	5
cubic	6

2. If LIST = 0 or blank, the complete reflection data with observed and calculated $[(\sin \theta)/\lambda]^2$ are printed out. If LIST $\neq$ 0, only the cell parameter and errors are printed.

3. YAM is the wavelength of x radiation used, or else YAM is 0. If YAM = 0 or blank, the program assumes that $(\sin \theta)/\lambda$ values have been loaded. If YAM = λ , the program assumes that $\sin \theta$ values have been loaded.

B. Reflection Card:

columns	1-4	5-8	9-12	13-22	23-32
data	h	k	l	S	W
formats	I4	I4	I4	E10.5	E10.5

1. h, k, l are the Miller indices

2. S is either $\sin \theta$ or $(\sin \theta)/\lambda$ (see above)

3. W is a least-squares weighting factor. If W is not punched, the program assumes $W = (\sin \theta)/\lambda$.

*It is always assumed by the program that c is the unique axis; thus the first setting is used for the monoclinic system.

The reflection card deck is terminated with a blank card.

A second calculation may follow. The reflection deck of the last calculation is terminated with two blank cards.

1.3 Method of Calculation

$$\text{Let } S^2 = 4(\sin^2 \theta) / \lambda^2$$

$$H_{ij} = h_i h_j, \quad i, j = 1, 2, 3$$

$$g^{ij} = a^i a^j, \quad i, j = 1, 2, 3$$

where h_i and a^i are respectively a Miller index and a reciprocal cell edge vector.

Then

$$S^2 = H_{ij} g^{ij}$$

where the repetition of a subscript as a superscript in the same term of an expression denotes summation over that index.

The least squares criterion requires that

$$F = \left[\sum (H_{ij} g^{ij} - S^2)^2 \right] = \text{minimum},$$

whence

$$\frac{1}{2} \frac{\partial F}{\partial g^{nm}} = \left[\sum (H_{ij} g^{ij} - S^2) H_{nm} \right] = 0,$$

or

$$\left[\sum H_{nm} H_{ij} \right] g^{ij} = \left[\sum S^2 H_{nm} \right], \quad n, m, i, j = 1, 2, 3,$$

where the brackets denote summation over all experimental observations, and $\sum$ is a weighting factor, different in general for each observation.

After collecting terms, this becomes

$$(2 - \delta_{nm}) (2 - \delta_{ij}) \left[\sum H_{nm} H_{ij} \right] g^{ij} = (2 - \delta_{nm}) \left[\sum S^2 H_{nm} \right]$$

where $n, m, i, j = 1, 2, 3$; $m \geq n$; $j \geq i$; and δ_{ij} is the Kroneker delta. In matrix notation this becomes

$$(h_{(nm)(ij)}) \cdot (G_{(nm)}) = (D_{(nm)})$$

where

$$h_{(nm)(ij)} = (2 - \delta_{nm}) (2 - \delta_{ij}) \left[\sum H_{nm} H_{ij} \right]$$

$$G_{(nm)} = g^{nm}$$

$$D_{(nm)} = (2 - \delta_{nm}) \left[\sum S^2 H_{nm} \right].$$

The $G_{(nm)}$ are found by the usual matrix methods. The variances likewise are found by the usual methods, i.e.,

$$\sigma^2 G_{(nm)} = \frac{F}{|h|} \cdot \frac{h_{(nm)(nm)}}{k - l}$$

where $h^{(nm)(nm)}$ is the cofactor of $h_{(nm)(nm)}$ element of $|h|$, k is the number of terms in the bracket summation, and l is the order of $|h|$.

The uncertainties in the unit cell parameters and volume are then found by error propagation methods. The equations are quite messy and will not be reproduced here.

1.4 Output

The output is self explanatory. It consists of the reciprocal cell parameters, the real cell parameters with uncertainties, the real cell volume with uncertainty, and, if desired, the observed and calculated S values are listed for each reflection.

1.5 Limitations

No more than 150 observations may be used.

1.6 Special Subroutines Called

In addition to the executive program, the following special subroutines are used:

- | | |
|-----------|------------|
| a. PARAM | c. ERRREL |
| b. ERRPRM | d. OUTPRM. |

Other nonlibrary programs used are:

- e. RECIP
- f. ARCSIN
- g. Modified versions of RW MATS and RW DET (SHARE distn. no. 635).

1.7 Special Tape Requirements

None

2. QUASI NORMALIZATION OF STRUCTURE FACTORS

2.1 Purpose

Prepare a BCD tape containing h, k, l , and ϱ^2-1 , similar to the one written by XRDDR, using for input the h, k, l , F^2 tape written by XRDDR in the Busing least-squares program format (1).

2.2 Input

The input consists of a card input, which is loaded following the program deck, and a tape input, which usually will consist of the data tape output from SUBROUTINE OUTPUT of XRDDR. The tape input is from logical tape 16.

A. Control Card:

columns	1-10	11-20	21-30	31-40	41-50	51-60
data	NOIB	NASP	NF	BT1	EX	LIB
formats	I10	I10	I10	E10.4	E10.4	I10

1. NOIB is the number of reflection records on logical tape 16.
2. NASP is the number of different atomic species in the crystal.
3. NF is the number of files on the input tape to be spaced over. This allows input to be taken from a library tape.
4. BT1 is the modified Wilson equation temperature factor B (as taken directly from the output of XRDDR).
5. EX is the exponent in the modified Wilson equation (labeled X in the output of XRDDR).
6. LIB is 1 if a library tape is used as input. It is blank otherwise. (It is assumed that the first record of a library tape is an identification record.)

Atomic Scattering Factor Cards:

columns	1-8	9-16	17-24	25-32	33-40	41-44
data	AS	AS1	BS	BS1	CS	AN
formats	F8.4	F8.4	F8.4	F8.4	F8.4	F4.0

The quantities AS, AS1, BS, BS1 and CS are the parameters A, a, B, b, and C defined and listed by Forsyth and Wells for calculation of atomic scattering factors (3). AN is the number of atoms of the atomic species in the unit cell. There is one card for every different atomic species in the crystal. The deck consists of one control card followed by NASP atomic scattering factor cards.

B. Reflection Data Tape:

columns	1-9	10-18	19-27	28-36	37-54	55-63
data	h	k	l	F ²	blank	(sin θ)/ λ
formats	F9.2	F9.2	F9.2	F9.2	18X	F9.6

2.3 Method of Calculation

F² values are converted to $\mathcal{E}^2$ values according to the following equation

$$\mathcal{E}^2 = F^2 \left(\exp \left[B \left(\frac{\sin}{\lambda} \right)^2 \right] \right) / \left(\sum_i f_i^2 \right).$$

The reflections are read from the input tape and processed in groups of 500 (or less for the last group). Thus, any number of reflections may be processed in one computer run.

2.4 Output

Logical tape 15 is rewound at the beginning of the calculation. Output is BCD on tape 15. The program writes an END FILE on 15 and rewinds it when the calculation is finished.

columns	1-4	5-8	9-12	13-32
data	h	k	l	e^2-1
formats	I4	I4	I4	F20.5

2.5 Limitations

There can be no more than twenty different atomic species in the crystal.

2.6 Special Subroutines Called

A FAP coded subroutine FLSKPD is called which forward-spaces tape 16 the number of files specified by NF (see control card).

2.7 Special Tape Requirements

Logical tape 16 is used for BCD reflection input.

3. RATIONAL DEPENDENCE*

3.1 Purpose

To determine the extent of rational dependence, as defined by Hauptman and Karle (4), in a crystal structure using the normalized structure factor magnitudes.

3.2 Input

The input consists of a card input, which is loaded following the program deck, and a tape input, which usually will consist of the data tape output from program SF NORM of XRDDR. The tape input is from logical tape 12.

A. Card Input

There are two types of cards in the card input.

- a. 72 columns of Hollerith characters (these characters are used to title the output).
- b. Control Card:

columns	1-10	11-20	21-30	31-40	41-50	51-60	61-70
data	NA	NINT	NFIN	MINN	AN	SDMIN	JPUT
formats	I10	I10	I10	I10	E10.4	E10.4	I10

1. NA is 1 for noncentrosymmetric crystals; it is 0 or blank for centrosymmetric crystals.
2. NINT is the initial modulus.

*After Block and Yannoni, National Bureau of Standard (unpublished).

3. NFIN is the final modulus.

4. MINN is the minimum number of reflections considered significant in a subset showing rational dependence effects. Results will not be output for subsets of reflections with fewer than MINN members.

5. AN is the minimum number of standard deviations which is considered a significant difference between the subset average and overall average of E^2 .

6. SDMIN is the smallest deviation between the subset average and the overall average of E^2 , where the deviation is considered to significantly show the presence of rational dependence effects.

7. JPUT is not 0 when complete listings of reflections are desired for each subset for which evidence of rational dependence has been found. JPUT is 0 when the reflection listings are not desired.

Additional control cards may follow if additional calculations are desired. The card deck is terminated with a blank card.

B. Tape Input.

This tape may be tape number 5 or 7 from program SF NORM of XRDDR:

1. The first record on the tape consists of the number of reflection records on the tape (format I7).

2. Reflection records:

columns	1-4	5-8	9-12	10-32
data	h	k	l	E^2-1
formats	I4	I4	I4	F20.5

3.3 Method of Calculation

A FORTRAN source program written by Block and Yannoni was obtained from Dr. Block of the National Bureau of Standards. The program has been modified with respect to reflection input; certain redundancies in the selection of subsets have been eliminated; a provision for specifying the minimum significant number of reflections in the subsets has been added; the criterion for accepting as significant a deviation of average E^2 from the overall value has been operationally improved by making it independent of the numbers of reflections in the subsets; and, finally, the output has been vastly expanded and improved. Otherwise the calculations are done essentially as coded by Block and Yannoni.

All reflections in a subset satisfy the relation

$$ah + bk + cl = n \pmod{m}$$

where m is a positive integer limited (by the program) to a maximum value of 14; a , b , and c may have integral values ranging from -13 to +14, depending upon the value of m .

If the value of average E^2 for a subset differs significantly from the average over all observed E^2 , then rational dependence is said to exist in the crystal structure.

The program automatically checks all possible subsets for rational dependence effects for all moduli (i.e., all m) between the limits specified on the control card.

3.4 Output

The output consists of three parts A, B, and C. A and B are always obtained, and C is or is not obtained at the option of the program user.

A. This is a leading one-page printout consisting of the control information, overall average E^2 , and the total number of reflections on the input tape.

B. For each subset for which significant rational dependence effects are found the following is printed: a, b, c, n, m , average E^2 , and the number of reflections in the subset.

C. If the control datum JPUT is not zero, data for all reflections in the subset defined by the printout of B above are listed, immediately following the B printout, in the form: h, k, l, E^2-1 , and E^2-A , where A is the overall average E^2 .

3.5 Limitations

The only limitation in addition to those already mentioned in section 3.3 is that the number of input reflections be less than 5000.

3.6 Special Subroutines Called

A subroutine OUTRAT, especially designed to write output for this program, is called.

3.7 Special Tape Requirements

Logical tape 12 is used for reflection input.

4. SIGMA-2 LISTINGS

4.1 Purpose

For each member of an ordered list of reflections, where the member is defined by the Miller index triple $\vec{h}$ and normalized structure factor magnitude E , the program lists all pairs of reflections $\vec{h}_i, E_i$, and $\vec{h}_j, E_j$ from a similar ordered list of reflections which satisfy the relation

$$\vec{h} = \vec{h}_i + \vec{h}_j.$$

A pair of reflections $\vec{h}_i$ and $\vec{h}_j$ is said to be a Σ_2 interaction pair for the reflection $\vec{h}$ if the equation above is satisfied.

4.2 Input

The data input consists of one control card, which is positioned at the end of the program deck, and a list of h and E^2-1 , which is read in either from logical tape 11 or from cards. The output tapes from program SF NORM of XRDDR may be used.

A. Control Card:

columns	1-10	11-20	21-30
data	Z	IZ	NOIB
formats	E10.4	I10	I10

1. All reflections for which $E^2-1 < Z$ are rejected from consideration.
2. If $IZ = 0$ the reflection data are read from logical tape 11. If $IZ \neq 0$ the reflection data are read from cards.
3. Lists of Σ_2 interaction pairs are tabulated for the NOIB reflections with largest E^2-1 values. If NOIB is negative, lists are prepared for the entire group of reflections for which $E^2-1 > Z$ (see method of computation below).

B. Reflection Data.

The reflection data is prefaced with one card (or record) NRT containing the number of reflections to be read. The format is I7.

The reflection records contain the following information:

columns	1-4	5-8	9-12	13-32
data	h	k	l	E^2-1
formats	I4	I4	I4	F20.5

4.3 Method of Calculation

As each reflection is read into the computer it is accepted or rejected on the basis of whether or not the criteria on h, k, l specified in REJVEC (see section 4.5) are satisfied. If the criteria are not satisfied, REJVEC sets $E^2-1 = -10$. Then, the reflection is tested on the basis of the magnitude of E^2-1 , and all reflections for which $E^2-1 < Z$ are rejected. (The minimum physically meaningful value of E^2-1 is -1.)

The selected set of reflections is arranged in decreasing order of magnitude of E^2-1 . Then lists of Σ_2 interaction pairs are calculated and tabulated for the top NOIB reflections in the selected set. If NOIB has been entered as a negative number, the lists are calculated for every member of the selected set.

4.4 Limitations

The selected set of reflections may not exceed 2400 reflections. There may not be more than 1000 Σ_2 interaction pairs for each $\bar{h}$.

4.5 Special Subroutines Called

a. REJVEC.

This subroutine imposes acceptance criteria on h, k, l of each reflection. The program user may write his own program or use the dummy program already written. The quantities N1, N2, N3, and Q in the calling sequence are h, k, l , and E^2-1 .

b. SIGVEC.

A crystal-system-dependent subroutine which selects $\bar{h}_i$ and $\bar{h}_j$ for each $\bar{h}$. If SIGVEC for the triclinic case is used, and the reflection decks are always "blownup" so as to include all symmetry mates of all reflections, then this subroutine becomes noncrystal-system dependent. SIGVEC versions for triclinic, monoclinic, and orthorhombic crystals are available.

c. OUTSIG.

This subroutine prints out the list of $\bar{h}_i$, E_i and $\bar{h}_j$, E_j and the products $E_i E_j E$ for each $\bar{h}$, E .

4.6 Special Tape Requirements

Logical tape 11 is used for reflection input.

5. TRIPLE PRODUCT SUMMATION FOR ORTHORHOMBIC CRYSTALS

5.1 Purpose

These programs (one for centrosymmetric and one for noncentrosymmetric crystals) are designed to calculate the quantity $E_1 E_2 E_3$, as defined by Hauptman and Karle (5), or the quantity $|E_1 E_2 E_3| \cos(\phi_1 + \phi_2 + \phi_3)$, as defined by Karle and Hauptman (6).

5.2 Input

The input consists of a card input, which is loaded following the program deck, and a tape input, which usually will consist of an output tape from program SF NORM of XRDDR.

A. Control Card:

columns	1-10	11-20	21-30	31-40
data	NA	IM	JM	KM
formats	I10	I10	I10	I10

1. NA is the number of atoms (exclusive of hydrogen atoms) in the unit cell.
2. IM is the maximum value of Miller index h .
3. JM is the maximum value of Miller index k .
4. KM is the maximum value of Miller index l .

Triple Product Selection Card:

columns	1-10	11-20	21-30	31-40	41-50	51-60
data	h_1	k_1	l_1	h_2	k_2	l_2
formats	I10	I10	I10	I10	I10	I10

1. h_1, k_1, l_1 are the Miller indices of the reflection with normalized structure factor E_1 (see section 5.1).

2. h_2, k_2, l_2 are the Miller indices of the reflection with normalized structure factor E_2 (see section 5.1).

The deck consists of one control card followed by any number of triple product selection cards. Calculation is terminated by the second blank card read (the first blank card causes calculation of the triple for $\bar{h}_1, \bar{h}_2 = 0$).

B. Reflection Data.

The first record on the reflection data input tape contains the number of reflection records to follow (format I7).

Each reflection record consists of:

columns	1-4	5-8	9-12	13-32
data	h	k	l	E^2-1
formats	I4	I4	I4	F20.5

Tape input is from logical tape 9.

For many problems it may be necessary to recompile the source program in order that the dimension assignments on the triply indexed variable E be changed (see section 5.3).

5.3 Method of Calculation

The program stores each value of $(E^2-1)_{h,k,l}$ as the triply indexed FORTRAN variable $E(h,k,l)$. The program then runs through the list of reflections in a triple nest of DO loops in order to select summands.

The orthorhombic symmetry operations on h,k,l are coded into the program.

Since the program uses h,k,l as indices, these quantities always must be positive. The program automatically rejects one- and two-dimensional data. Core size limits the maximum values of h,k,l rather severely. As listed in this report, the program accepts maximum values of h,k,l as large as 30, 10, 30. E is the only dimensioned variable in the programs, and most of the core is available to it; however it will frequently be necessary to recompile.

The only difference between the programs for centrosymmetric and noncentrosymmetric crystals is that for the former, the average is divided by $\bar{N}A^{3/2}/8$, whereas the latter is divided by $\bar{N}A^{3/2}/2$. (Note the third statement following statement number 8 in the FORTRAN listing given in Appendix A.)

Whereas only three-dimensional data are included in the summations, $\bar{h}_1, \bar{h}_2, \bar{h}_3$ need not be three dimensional.

If none of the Miller indices of $\bar{h}_1, \bar{h}_2, \bar{h}_3$ are zero, the program includes the correction term $(E^2_{\bar{h}_1} + E^2_{\bar{h}_2} + E^2_{\bar{h}_3} - 2)/\bar{N}A^{1/2}$ in the results.

5.4 Output

The output for each reflection includes:

- a. $\bar{h}_3, \bar{h}_1, \bar{h}_2$
- b. the number of terms in the sum.
- c. the scaled triple product average without correction term.
- d. the scaled triple product average with correction term.
- e. the correction term.

5.5 Limitations

See section 5.3.

5.6 Special Subroutines Called

None.

5.7 Special Tape Requirements

Logical tape 9 is used for reflection input.

6. STRUCTURE FACTOR CALCULATION

6.1 Purpose

To provide a convenient means for calculation of structure factors and quasi-normalized structure factors for use during preliminary stages of structure determination. Reflection input is from the E^2-1 or F^2 output tapes of XRDDR. Isotropic temperature factors are used.

6.2 Input

The input may be divided into general and control information, and reflection data.

A. General and Control Information.

The cards in this part are included with the program deck.

1. 72 columns of Hollerith characters (format 12A6). These Hollerith characters are used to title the output.

2. Control Card No. 1:

columns	1-10	11-20	21-30	31-40	41-50	51-60	61-70
data	IC	IE	NR	NF	IP	LIB	LINE
formats	I10	I10	I10	I10	I10	I10	I10

IC = 0 if crystal is centric; IC $\neq$ 0 if crystal is acentric.

IE = 0 if F² reflection data input is used; IE $\neq$ 0 if E²-1 reflection data input is used.

NR is the number of input reflections (if E²-1 reflection data is used, this may be left blank).

NF is the number of files to be spaced over if the reflection data are read from a library tape.

IP = 0: E output mode is specified; IP $\neq$ 0: F output mode is specified. IP is irrelevant for centric crystals (see section 6.5.B for a description of the modes).

LIB = 0: library tape not used for reflection input; LIB $\neq$ 0: library tape is used for reflection input.

Line = 0: reflection data is read from logical tape 10; LINE $\neq$ 0: reflection data is read from cards.

3. Cell Constant Card:

columns	1-10	11-20	21-30	31-40	41-50	51-60
data	a	b	c	α	β	γ
formats	E10.4	E10.4	E10.4	E10.4	E10.4	E10.4

a, b, c, α , β , γ are the real-space unit cell parameters in units of angstroms and degrees.

4. Control Card No. 2:

columns	1-10	11-20	21-30	31-40	51-60
data	BT	XT	SFC	KKT	IOUT
formats	E10.4	E10.4	E10.4	I10	I10

BT and XT are the parameters B and X of the modified Wilson equation as taken from the output of XRDDR (if KKT = 1, these fields may be left blank).

SFC is a scale factor by which all F₀ are to be multiplied to bring them to the proper scale. It is irrelevant if E₀ data are input.

KKT has the value of 0, 1, or 2, which serves to select the desired structure factor normalization procedure as follows:

KKT = 0: apply exact normalization procedure to F_c; apply approximate normalization procedure to F₀ or E₀.

KKT = 1: apply exact normalization procedure to F_c; apply exact normalization procedure to F₀ or E₀.

KKT = 2: apply approximate normalization procedure to F_c; apply approximate normalization procedure to F₀ or E₀.

(see section 6.4 for explanations of exact and approximate normalization.)

If IOUT $\neq$ 0, a BCD output (logical tape 11) will be written (see section 6.6).

5. Scale factor cards.

If IE is 0 (i.e., F_0 reflection data are input), a deck of scale factor cards is input. Each card contains one scale factor (format E10.4). The deck is terminated with a blank card. The scale factors are indexed in the order loaded and are applied according to the procedure given in section 6.2.B under description of SCF.

6. Atomic scattering factor cards.

There is one card for each unique atomic species in the crystal:

columns	1-8	9-16	17-24	25-32	33-40
data	A	a	B	b	C
formats	F8.4	F8.4	F8.4	F8.4	F8.4

A, a, B, b, and C are the parameters used in the Vand, Eiland, Pepinsky approximation equation for calculation of atomic scattering factors as defined by Forsyth and Wells (3,7).

The deck of atomic scattering factor cards is terminated by a blank card.

The loading order of the atomic scattering factor cards is important and must be consistent with the quantities I on the coordinate cards (see section 6.2.A.7).

7. Coordinate cards.

There is one card for each atom in the asymmetric unit:

columns	1-10	11-20	21-30	31-40	41-50
data	X	Y	Z	B	I
formats	E10.4	E10.4	E10.4	E10.4	I10

X, Y, and Z are the fractional atomic coordinates of one atom.*

B is the isotropic temperature factor.

I is an integer which relates the atom to one of the atomic scattering factor cards and thus specifies the atomic type. Each atomic scattering factor card is numbered 1,2,3,... in sequence as it is read into the computer. Thus, if the second scattering factor card to be read is for carbon, then I = 2 on all carbon coordinate cards.

The deck of coordinate cards is terminated with a blank card.

8. Transformation cards.

The atoms in the asymmetric unit are transformed according to the relations given in "International Tables for X-Ray Crystallography," Vol. I, (N.F.M. Henry and K. Lonsdale, Birmingham: Kynoch Press, 1952) so as to fill out the unit cell. There is one card for each transformation.

columns	1-10	11-20	21-30	31-40	41-50	51-60
data	T1	T2	T3	U1	U2	U3
formats	E10.4	E10.4	E10.4	E10.4	E10.4	E10.4

*If an atom is at the origin, X,Y,Z should be given the values 1,1,1 instead of 0,0,0.

The transformations are done as follows:

$$X' = T1 + U1 \cdot X$$

$$Y' = T2 + U2 \cdot Y$$

$$Z' = T3 + U3 \cdot Z.$$

Example: Space group $P2_1/c$

Transformations: $\bar{x}, \bar{y}, \bar{z}; x, 1/2 - y, 1/2 + z; \bar{x}, 1/2 + y, 1/2 - z.$

Transformation cards

T1	T2	T3	U1	U2	U3
.0	.0	.0	-1.	-1.	-1.
.0	.5	.5	1.	-1.	1.
.0	.5	.5	-1.	1.	-1.

The deck of transformation cards is terminated with a blank card.

B. Reflection Data.

The reflection input may be included along with the program deck, following the general and control information cards, or be taken from logical tape 10. It may be taken directly from an XRDDR output tape or from a library tape, each file of which contains the data from an XRDDR input or output tape. In the latter case, the program assumes that each file is prefixed with one identification record.

The program calculates structure factors for each reflection in the reflection input.

If F_0^2 reflection input is used ($\sin \theta / \lambda$ values may or may not be included (they are automatically included on the XRDDR F^2 output tapes). The format is the same as that used for input to the Busing least-squares program.

If E^2-1 reflection input is used, the first record must consist of the number of reflections to be read (format I7). No provision is made for input of ($\sin \theta / \lambda$) data.

F^2 reflection data:

columns	1-9	10-18	19-27	28-36	37-45	46-54	55-64
data	h	k	l	F_0	(blank)	SFC	$(\sin \theta) / \lambda$
formats	F9.2	F9.2	F9.2	F9.2	9X	F9.2	F9.6

The quantity SCF is a floating point integer which relates the reflection to one of the scale factors described in section 6.2.A.5 above. The integer corresponds to the loading sequence number of the scale factor (i.e., the first scale factor loaded corresponds to $SCF = 1.0$, etc.). SCF may be punched on every card. Alternatively, the reflection input may be grouped into sets where all reflections in a set are multiplied by the same scale factor. In this case, it is sufficient to punch SCF in the first card only of each set. This is the same scaling procedure used in the Busing least-squares refinement program; however, in this program, the F_0 values are scaled.

E²-1 reflection data:

columns	1-4	5-8	9-12	13-32
data	h	k	l	E ₀ ² -1
formats	I4	I4	I4	F20.5

If another calculation is to follow, a card with a 1 punched in column 7 follows the last card of the main deck; otherwise, a blank card is last.

6.3 Data Deck

The composition of the data deck is as follows:

- a. Hollerith card.
- b. control card no. 1.
- c. cell constant card.
- d. control card no. 2.
- e. all scale factor cards (if IE ≠ 0, there are no scale factor cards).
- f. blank card.
- g. all atomic scattering factor cards.
- h. blank card.
- i. all coordinate cards.
- j. blank card.
- k. all transformation cards (if any).
- l. blank card.
- m. reflection cards unless reflection input is from tape.
- n. a card with 1 in column 7 or a blank card depending upon whether or not another calculation is to follow.

6.4 Method of Calculation

Structure factors are calculated from the total unit cell contents (rather than from the asymmetric unit contents) using the equations for the triclinic case. Nevertheless, coordinates may be entered for one asymmetric unit only, provided that transformations follow by which the asymmetric unit set is expanded to fill the unit cell. The transformations may be copied directly from the "International Tables for Crystallography," Vol. I (1952). In this way, the program is completely general in its applicability while requiring the most simple type of input.

The quasi normalization of structure factors (and in the case where E²-1 reflection input is used, the inverse normalization) may be done in either of two ways, designated as exact and approximate normalization.

1. Exact normalization:

$$E^2 = F^2 / \sum_{i=1}^n f_i^2.$$

2. Approximate normalization:

$$E^2 = \frac{F^2 \exp [BT(\sin^2 \theta / \lambda^2)^{2T}]}{\sum_{i=1}^N f_{0,i}^2}$$

where

$$f_i = f_{0,i} \exp [-B_i(\sin^2 \theta) / \lambda^2]$$

and N is the number of atoms in the unit cell.

6.5 Printed Output

A. Centric Crystals.

The output for each reflection is:

- a. h, k, l
- b. F_0
- c. F_c
- d. E_0
- e. E_c
- f. ΔF
- g. ΔE .

B. Acentric Crystals.

Either one of two modes of output may be selected:

1. F mode

- a. h, k, l
- b. E_0, E_c
- c. F_0, E_c
- d. A_0, B_0
- e. A_c, B_c
- f. $\Delta A, \Delta B$
- g. ΔF .

2. E mode

- a. h, k, l
 - b. F_0, F_c
 - c. E_0, E_c
 - d. C_0, C_c
 - e. D_0, D_c
 - f. $\Delta C, \Delta D$
 - g. ΔE .
- where $|E|^2 = C^2 + D^2$

In addition to the reflection output, the program prints $\Sigma|\Delta F|$, $\Sigma|F_0|$, and $R = \Sigma|\Delta F| / \Sigma|F_0|$ both including and excluding unobserved data.

6.6 Tape Output

If $IOUT \neq 0$ the program writes a reflection output tape containing one of the following sets of data for each reflection (format 7F10.4):

a. Centrosymmetric crystals

$$h, k, l, F_0, E_0, \Delta F, \Delta E$$

b. Noncentrosymmetric crystals

1. $IP = 0$:

$$h, k, l, C_0, D_0, \Delta C, \Delta D$$

2. $IP \neq 0$:

$$h, k, l, A_0, B_0, \Delta A, \Delta B$$

where the observed quantities are scaled and given the signs of the corresponding calculated quantities.

6.7 Special Subroutines Called

a. OUTSFC.

An output routine designed for these calculations only.

b. RECIP.

c. The FAP coded tape utility program written by P. Gum.*

d. INTG.

6.8 Special Tape Requirements

a. Logical 10 is used for reflection input.

b. Logical 11 is used for reflection output.

7. INTERATOMIC DISTANCES AND ANGLES WITH INTERPOLATION FOR PEAK CENTER LOCATION

7.1 Purpose

Given a basic set of atomic or peak coordinates and a set of symmetry transformations on these coordinates, calculate and list all interatomic distances b , where $0.7 \leq b \leq BMX$, and calculate all angles between connected pairs of distances for all possible combinations in any crystal.

*Diffraction Branch, Optics Division, NRL.

7.2 Input

The data input is from cards which are loaded immediately following the program deck. The cards are described in loading sequence as follows:

A. 72 columns of Hollerith characters (format 12A6). These Hollerith characters are used to title the output.

B. Cell Constant Card:

columns	1-10	11-20	21-30	31-40	41-50	51-60
data	a	b	c	α	β	γ
formats	E10.4	E10.4	E10.4	E10.4	E10.4	E10.4

The quantities $a, b, c, \alpha, \beta, \gamma$ are the real-space unit cell parameters expressed in units of angstroms and degrees.

C. Bond Length Limit Card:

columns	1-10	11-20
data	BMX	AMX
formats	E10.4	E10.4

BMX.

Interatomic distances b for which $0.7 \leq b \leq \text{BMX}$ are considered acceptable and are tabulated for printing.

AMX.

Angles for all pairs of acceptable distances sharing a common atom, for which the distances of both members of the pair are less than or equal to AMX, are tabulated for printing.

[WARNING: When a large number of atoms or peaks are involved and BMX and AMX are both large (i.e., greater than about 3.0A), the number of angles printed can be very large.]

D. Grid Interval Card:

If peak maxima coordinates are to be found by interpolation between grid points of a Fourier calculation result, this card contains the fractional grid increments in the directions of X, Y, and Z.

columns	1-10	11-20	21-30
data	XINC	YINC	ZINC
formats	E10.4	E10.4	E10.4

For example, if the Fourier calculation was done in increments of 1/30, 1/60, and 1/120 along X, Y, and Z, respectively, then

$$\text{XINC} = .03333, \text{YINC} = .01666, \text{and ZINC} = .008333.$$

If interpolation is not to be done, this card must be blank.

E. Coordinate Card:

columns	1-10	11-20	21-30
data	X	Y	Z
formats	E10.4	E10.4	E10.4

X, Y, and Z are the fractional coordinates of an atom or peak. There is one card for each atom or peak in the basic (untransformed) set. If interpolation is to be done, each coordinate card is followed by an interpolation card (see section 7.2.F below). If interpolation is not to be done, the interpolation cards are absent from the deck.

The coordinate card deck is terminated by a blank card.

F. Interpolation Card.

Interpolation cards are included in the coordinate deck, one following each coordinate card, if and only if card D is not blank (see section 7.2.D). .2.4).

The fractional coordinates of the grid point closest to a peak maximum in a Fourier calculation result are given on the coordinate card. Relative to this central grid point, designate the Fourier summation value at nearby grid points by $P(i, j, k)$, where i, j , and k have integral values representing the (signed) numbers of grid interval separations of the point from the central one in the directions of X, Y, and Z, respectively.

The interpolation card contains the following:

columns	1-10	11-20	21-30	31-40	41-50	51-60	61-70
data	P(0,0,0)	P(-1,0,0)	P(1,0,0)	P(0,-1,0)	P(0,1,0)	P(0,0,-1)	P(0,0,1)
formats	E10.4	E10.4	E10.4	E10.4	E10.4	E10.4	E10.4

If the user wishes to interpolate some, but not all, points, then values of $P(0,0,0) = 100.0$ and all other $P = 1.0$ may be entered for those points that are not to be interpolated.

Note that $P(0,0,0)$ must be the largest P on the card. Also, all P must be greater than zero. Negative or zero P can be accommodated by adding a constant increment to all P such that the results are all positive.

G. Transformation Card:

columns	1-10	11-20	21-30	31-40	41-50	51-60
data	T1	T2	T3	U1	U2	U3
formats	E10.4	E10.4	E10.4	E10.4	E10.4	E10.4

There is one transformation card for every symmetry transformation to be applied to the basic set of coordinates. The transformations are applied according to the equations:

$$X^1 = T1 + U1 \cdot X$$

$$Y^1 = T2 + U2 \cdot Y$$

$$Z^1 = T3 + U3 \cdot Z.$$

The transformation card is terminated by a blank card.

7.3 Data Deck

The composition of the data deck is as follows:

- a. Hollerith card.
- b. cell constant card.
- c. bond length limit card.
- d. grid interval card, or blank if interpolation is not desired.
- e. all coordinate and interpolation cards.
- f. blank card.
- g. all transformation cards (if any).
- h. blank card.

7.4 Method of Calculation

The basic set of coordinates are loaded, interpolated (if desired), and then each transformation is applied to the basic set, generating a transformed set for each transformation. Thus, if there are N atoms in the basic set and there are M transformations, the complete set contains N(M + 1) atoms.

Interatomic distances less than or equal to BMX are tabulated for all pairs of atoms in the basic set and for all distances ($\leq$ BMX) which cross the transformed set boundaries for the complete set.

Atom pairs closer together than 0.7A are listed in a separate output and are rejected from further consideration.

Next, the list of accepted interatomic distances is scanned to find pairs of interatomic distances which share a common atom. Angles for all such pairs of distances are tabulated if both distances are less than or equal to AMX. Logical tape 9 is used for intermediate storage of angles.

Interpolation is done by fitting the seven Fourier grid points closest to the peak maximum to a Gaussian function and then finding the maximum of the function. The corrections, ΔX , ΔY , and ΔZ to be added to X, Y, and Z are found by solving the matrix

$$\begin{pmatrix} \bar{a} \cdot \bar{a} & \bar{a} \cdot \bar{b} & \bar{a} \cdot \bar{c} \\ \bar{a} \cdot \bar{b} & \bar{b} \cdot \bar{b} & \bar{b} \cdot \bar{c} \\ \bar{a} \cdot \bar{c} & \bar{b} \cdot \bar{c} & \bar{c} \cdot \bar{c} \end{pmatrix} \begin{pmatrix} \Delta x \\ \Delta y \\ \Delta z \end{pmatrix} = \begin{pmatrix} D_x \\ D_y \\ D_z \end{pmatrix}$$

where

$$D_x = \frac{a^2}{2} \left[\log \frac{P(1,0,0)}{P(-1,0,0)} \right] / \left[\log \frac{P(0,0,0)}{P(1,0,0)} + \log \frac{P(0,0,0)}{P(-1,0,0)} \right],$$

etc.

7.5 Output

In the output, each atom is represented by two integers separated by a hyphen. The first integer specifies the transformation (numbered according to loading order) which has been applied, and the second integer specifies the atom in the basic set (numbered according to loading order) from which it was derived.

The output is in the following order:

- a. Atomic coordinates for the complete set.
- b. Interatomic distances.
- c. Angles.
- d. Interatomic distances less than 0.7A.

7.6 Limitations

- a. The complete set of atoms or peaks is limited to 500 in number.
- b. If more than 2900 acceptable interatomic distances are found, BMX is decremented by 0.25A and the calculation is begun again. This process is repeated until a total of 2900, or less, acceptable distances are found, or until $BMX \leq 2.0$. In the latter case, distances and angles are calculated for the basic set of atoms only. If more than 2900 acceptable distances are found in the basic set, the calculation is terminated without output of results.
- c. No more than 200 distances less than 0.7A can be accommodated.
- d. There is no limit on the number of bond angles.

7.7 Special Subroutines Called

- a. subroutine INTERP. This subroutine does the interpolation by Gaussian curve fitting.
- b. subroutine MATS (SHARE distr. no. 635).
- c. subroutine OUTBND: (distance and angle output).
- d. function DOTPRD.
- e. function ARCSIN.

7.8 Special Tape Requirements

Logical Tape 9 is used for intermediate storage.

8. LEAST-SQUARES PLANE AND LINE FITTER

8.1 Purpose

This program fits planes and/or lines to sets of points and calculates angles between planes and lines which have been fitted.

8.2 Input

- a. 72 columns of Hollerith characters (format 12A6). These Hollerith characters are used to title the output.

b. Unit Cell Card:

columns	1-10	11-20	21-30	31-40	41-50	51-60
data	a	b	c	α	β	γ
formats	E10.4	E10.4	E10.4	E10.4	E10.4	E10.4

The quantities $a, b, c, \alpha, \beta, \gamma$ are the real-space unit cell parameters expressed in units of angstroms and degrees.

c. Plane or Line Specification Card:

columns	10-3	4-6	7-12	13-72
data	N	NO	L	Hollerith
formats	I3	I3	I6	10A6

N is the number of points to be fitted.

NO is the plane or line number.

L = 0 or blank if the points are to be fitted to a plane.

L = 1 if the points are to be fitted to a line.

The Hollerith characters in columns 13-72 are used as a page heading for the line or plane in the output.

d. Coordinate Card:

columns	1-10	11-20	21-30	31-40
data	X	Y	Z	W
formats	E10.4	E10.4	E10.4	E10.4

X, Y, and Z are fractional coordinates of a point to be fitted to a line or plane.

W is a least-squares weighing factor. If W is not punched, it is assumed to have a value of one by the program.

e. Angle specification card.

This card specified a plane-plane, line-line, plane-line, or line-plane pair for which the dihedral angle is to be calculated.

columns	1-3	4-6	7-9	10-12
data	NO1	L1	NO2	L2
formats	I3	I3	I3	I3

NO1 is the number of the first plane or line.

L1 = 0 or blank if NO1 refers to a plane.

L1 = 1 if NO1 refers to a line.

NO2 is the number of the second plane or line.

L2 = 0 or blank if NO2 refers to a plane.

L2 = 1 if NO2 refers to a line.

The plane and line numbers NO are used to identify the different planes and lines both internally and in the output. The numbering may be different from the order of input. No two planes may have the same number, and no two lines may have the same number, but the numbering of lines and planes is independent. The number of any plane or line must not be greater than 50.

8.3 Data Deck

The composition of the data deck is as follows:

- a. Hollerith card.
- b. unit cell card.
- c. plane or line specification card.
- d. all coordinate cards for this plane or line (the number of coordinate cards must be the same as the N punched on the preceding plane or line specification card).
- e. plane or line specification card.
- f. all coordinate cards for this plane or line.
- g. additional sets of specification and coordinate cards for as many planes and lines as are to be fitted.
- h. blank card.
- i. all angle specification cards, if any.
- j. blank card.

8.4 Method of Calculation

The method described in detail by Shomaker, Waser, Marsh, and Bergman (8) is used.

The program is written in the fullest generality so as to be able to handle points defined in any three-dimensional coordinate system. However, no provision is made for symmetry transformation of point positions, so all points must be properly transformed before input.

For a given coordinate system, the program sequentially fits all desired planes and lines. It stores the appropriate unit vectors for the planes and lines by plane and line number. The program can then calculate dihedral angles between all pairs of planes, lines, or plane-line combinations that are specified.

8.5 Output

The output is largely self explanatory. Specifically, the following information is given:

- a. equations of the planes and lines.
- b. for planes, the coordinates of the centroid.
- c. perpendicular distances of each point from the least-squares planes and lines.
- d. average deviations, standard deviations, etc.

- e. cosines of angles between plane-plane and line-line pairs.
- f. sines of angles between line-plane pairs.

Each point is numbered in the output according to loading order. Thus, the Hollerith section of the plane or line specification card (see section 8.2.c) may be used, in part, to relate the points to atom numbers or other external designations.

8.6 Limitations

- a. No more than 50 points may be fitted to any plane or line.
- b. No less than three points may be fitted to a plane.
- c. No less than two points may be fitted to a line.
- d. There may be no more than 50 planes and no more than 50 lines.
- e. The number of a line or plane may not be larger than 50.

8.7 Special Subroutines Called

- a. subroutine RECIP.
- b. subroutine MTXMUL.
- c. function DOTPRD.

8.8 Special Tape Requirements

None.

9. POINT TO PEAK DISTANCE CALCULATION

9.1 Purpose

This program provides a means for calculating distances from one or several points to a set of points in any three-dimensional coordinate system.

9.2 Input

a. 72 columns of Hollerith characters (format 12A6). These characters are used to title the output.

b. Unit Cell Card:

columns	1-10	11-20	21-30	31-40	41-50	51-60
data	a	b	c	α	β	γ
formats	E10.4	E10.4	E10.4	E10.4	E10.4	E10.4

The quantities $a, b, c, \alpha, \beta, \gamma$ are the real-space unit cell parameters in units of angstroms and degrees.

c. Maximum distance card.

This card contains the single quantity BMX. All distances d are tabulated for which $d \leq \text{BMX}$ (format E10.4).

d. Point Card:

columns	1-10	11-20	21-30
data	XC	YC	ZC
formats	E10.4	E10.4	E10.4

XC, YC, and ZC are the fractional coordinates of the point for which distances to the set of points, specified by the coordinate cards and their transformations, are to be calculated.

e. Coordinate card

These cards contain the fractional coordinates X, Y, and Z of the basic set of points, which gives rise to a complete set via the transformations (see card below), whose distances to point XC, YC, ZC are desired. The card layout is the same as that already given for the point card above.

f. Transformation Cards:

columns	1-10	11-30	21-30	31-40	41-50	51-60
data	T1	T2	T3	U1	U2	U3
formats	E10.4	E10.4	E10.4	E10.4	E10.4	E10.4

The function of these quantities is explained in section 7.2.G.

9.3 Data Deck

- a. Hollerith card.
- b. unit cell card.
- c. maximum distance card.
- d. point card.
- e. all coordinate cards.
- f. blank card.
- g. all transformation cards.
- h. blank card.
- i. additional point cards for as many calculations as desired.
- j. blank card.

9.4 Method of Calculation

This program is an abbreviation of the Interatomic Distance and Angle Program (section 7). It does not provide for interpolation or angle calculation.

9.5 Output

The output of distances is essentially the same as that described for the Interatomic Distance and Angle Program (section 7.5).

9.6 Limitations

- a. The complete set of points must not exceed 500 in number.
- b. A maximum of 5000 distances can be accommodated.

9.7 Special Subroutines Called

- a. DOTPRD.

9.8 Special Tape Requirements

None.

10. FORM FACTORS FOR THE BUSING LEAST-SQUARES REFINEMENT PROGRAM

10.1 Purpose

Provide an automatic process for preparing form factor cards used for input to the Busing least-squares refinement program ORXLS (1).

10.2 Input

- a. One blank card.
- b. Form Factor Parameter Card:

columns	1-8	9-16	17-24	25-32	33-40	41-46
data	A	a	B	b	C	Hollerith
formats	F8.5	F8.5	F8.5	F8.5	F8.5	A6

The quantities A, a, B, b, C are the parameters, as defined by Forsyth and Wells (3), used in calculating the form factors. The six columns of Hollerith characters are used as identification on the output cards.

10.3 Data Deck

The composition of the data deck is as follows:

- a. blank card.
- b. as many form factor cards as desired.
- c. blank card.

10.4 Method of Calculation

The analytical approximation of Vand, Eiland, and Pepinsky (7), is used as extended by the work of Forsyth and Wells (3).

The approximations are good only as far as $(\sin \theta)/\lambda \approx 1.40$ (the range of $M_0K\alpha$ radiation).

These approximations are not recommended for highly accurate work.

10.5 Output

The form factor card images are written on logical tape 10 for peripheral punching. For each atomic species, five cards are punched as follows:

- a. a blank separator card.
- b. four cards containing the form factor values in intervals of 0.05 in $(\sin \theta)/\lambda$ from 1.55 to 0, in that order.

Each card has an F in column 72. Columns 73-78 contain the Hollerith characters specified in the form factor parameter cards. Column 80 contains the sequence number of the card, i.e., 1,2,3, or 4.

Essentially the same information also is printed.

10.6 Special Subroutines Called

None.

10.7 Special Tape Requirements

- a. Logical tape 10 for peripheral punching.

11. VARIANCE-COVARIANCE MATRIX AND ATOMIC COORDINATE INPUT FOR THE BUSING FUNCTION AND ERROR PROGRAM

11.1 Purpose

This program produces two BCD card decks containing, respectively:

- a. The variance-covariance matrix for a set of atomic position coordinates, with all covariance elements given a value of zero.
- b. Atomic coordinates.

These cards are ready for input to the Busing Function and Error Program ORXFE (2).

11.2 Input

- a. Preliminary Card:

columns	1-9	10-19
data	N	SF
formats	I9	E10.5

N is the order of the variance-covariance matrix, i.e., $N = 3 \times$ (number of atoms).

SF is a scale factor by which the matrix is to be multiplied. Specifically,
 $SF = \sum w (F_o - F_c)^2 / (m - n)$, or unity.

b. Unit Cell Card:

columns	1-10	11-20	21-30
data	a	b	c
formats	E10.5	E10.5	E10.5

If the standard deviations on the cards described below are in fractional (dimensionless) form, a, b, and c all have values of one.

If the standard deviations are in units of angstroms, then a, b, and c are the unit cell edge lengths in units of angstroms.

c. Standard Deviation Card:

columns	1-10	11-20	21-30
data	$\sigma(X)$	$\sigma(Y)$	$\sigma(Z)$
formats	E10.5	E10.5	E10.5

The quantities $\sigma(X)$, $\sigma(Y)$, and $\sigma(Z)$ are standard deviations or uncertainties of the atomic coordinates X, Y, and Z. These quantities may be dimensionless or may have units of angstroms (see section 11.2.b).

d. Coordinate card.

The layout of the coordinate cards is the same as for the standard deviation cards. Each coordinate card contains the fractional coordinates X, Y, and Z of an atom whose standard deviations are given in the corresponding card of the standard deviation deck.

11.3 Data Deck

- preliminary card.
- unit cell card.
- all standard deviation cards.
- all coordinate cards.

11.4 Output

Output may be on logical tape 9 for peripheral punching or, if sense switch 1 is down, the cards are punched on-line.

Each matrix card is numbered sequentially and identified with the letters SD (format 8E9.4, I4, 3H SD).

Each atomic coordinate card is numbered sequentially and identified with the letter P (format 8F9.6, I4, 24 P).

The two decks are separated by a blank card.

11.5 Limitations

The program will accommodate no more than 64 atoms.

11.6 Special Subroutines Called

None.

11.7 Special Tape Requirements

Logical tape 9 is used for peripheral punching.

12. OTHER FUNCTIONS AND SUBROUTINES

12.1 Function ARCSIN(X)

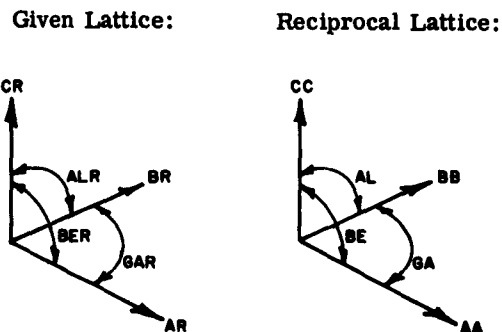
This program uses Hasting's Chebyshev approximation (sheet 39 of Ref. 9).

If $|X| > 1$, the program calls a subroutine ENDJOB which may be constructed by the user to suit his purposes.

The program was coded by B. A. Schoomer.*

12.2 Subroutine RECIP (AR, BR, CR, ALR, BER, GAR, AA, BB, CC, COSAL, COSBE, COSGA)

Given the sets of three translational and three angular parameters for a three-dimensional space lattice, this subroutine finds the parameters for the corresponding reciprocal space lattice.



$$AR, BR, CR, ALR, BER, GAR \rightarrow AA, BB, CC, AL, BE, GA$$

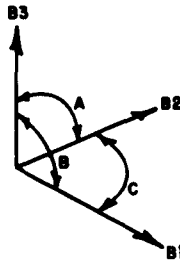
AR, BR, and CR may be given in any convenient units; ALR, BER, and GAR, are given in degrees.

*Diffraction Branch, Optics Division, NRL.

12.3 Function DOTPRD (U, V, W, X, Y, Z, B1, B2, B3, COSA, COSB, COSC)

This function calculates the scalar product of two vectors $\bar{U}$ and $\bar{V}$ where $\bar{U} = \bar{U}(U, V, W)$ and $\bar{V} = \bar{V}(X, Y, Z)$.

The vectors $\bar{U}$ and $\bar{V}$ are defined on a skew-vector basis B1, B2, B3



such that

$$\bar{U} = UB1 + VB2 + WB3$$

$$\bar{V} = XB1 + YB2 + ZB3$$

and

$$\begin{aligned} \bar{U} \cdot \bar{V} = & UX(B1)^2 + VY(B2)^2 + WZ(B3)^2 + (VZ + YW) B2 B3(\cos A) + \\ & (UZ + WX) B1 B3(\cos B) + (UY + VX) B1 B2(\cos C). \end{aligned}$$

12.4 Function SCAFAC (A, A1, B, B1, C, S)

This function calculates an atomic scattering factor using the analytic approximation of Vand, Eiland, and Pepinsky,(6).

The scattering factor $f(s)$ is

$$f(s) = A \exp(-A1 S^2) + B \exp(-B1 S^2) + C.$$

The quantities A, A1, B, B1, and C are taken from the tables of Forsyth and Wells (2). S is $(\sin \theta)/\lambda$.

12.5 Subroutine MTXMUL (L, M, N, A, B, C)

This subroutine performs matrix multiplication according to the equation

$$AB = C$$

where

the number of rows in A is L

the number of columns in A is M

the number of columns in B is N.

(As listed in this report, the dimensions of all matrices are 3×3 ; however, these may be changed to any other values needed.)

12.6 Function INTG(A)

This program converts floating point integers to fixed point integers without truncation error. This is done in a manner such that the integer value zero is not printed with a leading minus sign.

REFERENCES

1. Busing, W.R., and Levy, H.A., "A Crystallographic Least Squares Refinement Program for the IBM 704," Oak Ridge National Laboratory Central File No. 59-4-37 (1959)
2. Busing, W.R., and Levy, H.S., "A Crystallographic Function and Error Program for the IBM 704," Oak Ridge National Laboratory Central File No. 59-12-3 (1959)
3. Forsyth, J.B., and Wells, M., Acta Cryst. 12:412 (1959)
4. Hauptman, H., and Karle, J., Acta Cryst. 12:846 (1959)
5. Hauptman, H., and Karle, J., Acta Cryst. 10:267 (1957)
6. Karle, J., and Hauptman, H., Acta Cryst. 10:515 (1957)
7. Vand, V., Eiland, P.F., and Pepinsky, R., Acta Cryst. 10:303 (1957)
8. Schomaker, V., Waser, J., Marsh, R.E., and Bergman, G., Acta Cryst. 12:600 (1959)
9. Hastings, C., "Approximations for Digital Computers," Princeton:Princeton University Press (1955)

APPENDIX A

FORTRAN LISTINGS OF GENERAL UTILITY PROGRAMS APPLICABLE TO CRYSTAL STRUCTURE ANALYSIS

In the form presented in this appendix, the FORTRAN-language programs are written for use with the IBM 7090 IB Monitor system.

<u>Program</u>	<u>Page</u>
Least-Squares Cell Parameter	33
Subroutine DET	34
Subroutine PARAM	35
Subroutine ERRPRM	36
Subroutine ERRREL	37
Subroutine OUTPRM	37
Quasi Normalization of Structure Factors	39
Rational Dependence	41
Subroutine OUTRAT	43
Sigma-2 Listings	44
Subroutine OUTSIG	45
Subroutine REJVEC	46
Subroutine SIGVEC, Triclinic	46
Subroutine SIGVEC, Monoclinic	47
Subroutine SIGVEC, Orthorhombic	48
Triple Product Summation, Centrosymmetric	48
Triple Product Summation, Noncentrosymmetric	50
Structure Factor Calculation	52
Subroutine OUTSFC	56
Interatomic Distance and Angle	57
Subroutine INTERP	60
Subroutine OUTBND	60
Least-Squares Line and Plane Fitter	63
Point to Peak Distance Calculation	67
Form Factors for Busing Least Squares	68
Variance-Covariance and Atomic Parameter Input for Busing Function and Error	69

<u>Program (Cont'd)</u>	<u>Page</u>
Function ARCSIN(X)	70
Subroutine RECIP	71
Function DOTPROD	71
Function SCAFAC	71
Subroutine MTXMUL	71
Function INTG(A)	71

Least-Squares Cell Parameter

DIMENSION	NAME(10),M(3,150),W(150),S(150),SC(150),	02691
1	H(3,3,150),G(3,3),SG(3,3),BE(3,3),AL(3,3),B(3),A(3),SA(3),SAL(3,30	02691
2	),EB(3)	02691
	ITP=5	
	JTP=6	
50	MM=1	
	READ INPUT TAPE ITP,1000,N1,LIST,YAM,(NAME(I),I=1,10)	
	IF(N1)500,500,100	
100	READ INPUT TAPE ITP,2000,M(1,MM),M(2,MM),M(3,MM),S(MM),W(MM)	
	IF(XABSF(M(1,MM))+XABSF(M(2,MM))+ XABSF(M(3,MM)))140,140,110	02691
110	S2(MM) = 4.0*(S(MM) **2)	02691
	IF(W(MM))120,120,130	02691
120	W(MM) = S(MM)	02691
130	MM = MM +1	02691
	GO TO 100	02691
140	MM=MM-1	02691
	IF(YAM)145,145,141	02691
141	YAM = YAM**2	02691
	DO 142 I =1,MM	02691
142	S2(I)= S2(I)/YAM	02691
145	NN = 1	02691
	DO 147 I=1,3	02691
	DO 147 J=1,3	02691
	DO 147 K=1,MM	02691
147	H(I,J,K)=0.0	02691
	GO TO(310,290,240,210,180,150),N1	02691
150	N = 1	02691
	DO 160 I =1,MM	02691
160	H(1,1,I)= M(1,I)**2+ M(2,I)**2+M(3,I)**2	02691
	GO TO 325	02691
180	N=2	02691
	NN=2	02691
	DO 190 I =1,MM	02691
	H(2,2,I)= M(3,I)**2	
190	H(1,1,I)=M(2,I)**2 + M(1,I)**2 + M(2,I)* M(1,I)	
	GO TO 325	02691
210	N=2	02691
	DO 220 I=1,MM	02691
	H(2,2,I)= M(3,I)**2	
220	H(1,1,I)= M(2,I)**2 + M(1,I)**2	
	GO TO 325	02691
240	N = 3	02691
	GO TO 315	02691
290	N = 4	02691
	GO TO 315	02691
310	N = 6	02691

Subroutine DET

```

C      SHARE DISTRIBUTION NO. 635
      SUBROUTINE DET(A,ALPHA,N,BET)
      DIMENSION A(6,6)
      BET=1.
      IF(N-1)400,200,300
200    BET = A(1,1)
      GO TO 400
300    A(1,1)= A(1,1)- ALPHA
6      DO 15 I=2,N
      A(I,I)=A(I,I)-ALPHA
70     II=I-1

```

7	DO 15 J=1,II	DET 0008
8	IF (A(I,J))9,15,9	DET 0009
9	IF(ABSF(A(J,J))-ABSF(A(I,I)))11,10,10	DET 0010
10	R=A(I,J)/A(J,J)	DET 0011
	GO TO 130	DET 0012
11	R=A(J,J)/A(I,J)	DET 0013
	DO 12 K=1,N	DET 0014
	B=A(J,K)	DET 0015
	A(J,K)=A(I,K)	DET 0016
12	A(I,K)=B	DET 0017
	BET=-BET	
130	JJ=J+1	DET 0019
13	DO 14L=JJ,N	DET 0020
14	A(I,L)=A(I,L)-R*A(J,L)	DET 0021
15	CONTINUE	DET 0022
16	DO 20 I=1,N	DET 0023
20	BET=BET*A(I,I)	
400	RETURN	02691
	END	

Subroutine PARAM

	SUBROUTINE PARAM(N,H,W,G,SG,F,SC,MM,S2,LL)	02691
	DIMENSION H(3,3,150),W(150),G(3,3),SG(3,3),SC(150),S2(150),D(3,3)	02691
1	A(6,6),B(6),AA(6,7),X(6,1)	02691
	IF(N -3)362,365,365	02691
362	LL = N	02691
	GO TO 330	02691
365	LL = 3	02691
330	DO 360 I = 1,LL	02691
	DO 360 J = 1,LL	02691
	IF(I-J)350,340,350	02691
340	D(I,I) = 1.0	02691
	GO TO 360	02691
350	D(I,J) = 0.0	02691
360	CONTINUE	02691
367	DO 440 I =1,LL	02691
	DO 440 J =I,LL	02691
	DO 440 K =1,LL	
	DO 440 L =K,LL	02691
	IF(I - J)380,370,380	02691
370	JJ = I	02691
	GO TO 390	02691
380	JJ =1+I+J	
390	IF(K - L)410,400,410	02691
400	KK = K	02691
	GO TO 420	02691
410	KK =1+K+L	
420	A(JJ,KK)=0.0	02691
	B(JJ) = 0.0	02691
	DO 430 II=1,MM	02691
	A(JJ,KK) = A(JJ,KK) + W(II)* H(I,J,II)*H(K,L,II)	02691
430	B(JJ) = B(JJ) + W(II)* S2(II)* H(I,J,II)	02691
	A(JJ,KK) =(2.-D(I,J)) *(2.0 - D(K,L))*A(JJ,KK)	
	B(JJ) = B(JJ)*(2.0 - D(I,J))	02691
	AA(JJ,KK)= A(JJ,KK)	02691
440	AA(JJ,N+1) = B(JJ)	02691
	IF(N-1)445,445,446	
445	G(1,1)=B(1) /A(1,1)	
	GO TO 480	
446	CALL MATS(AA,X,N,1)	02691
	DO 470 I = 1,LL	02691
	DO 470 J = 1,LL	02691
	IF(I - J) 460,450,460	02691

450	G(I,I) = X(I,I)	02691
	GO TO 470	02691
460	K = 1+I+J	
	G(I,J) = X(K,I)	02691
470	CONTINUE	02691
480	CALL ERRPRM(W,H,F,SC,SG,N,MM,G,S2,A,LL,D)	02691
	RETURN	02691
	END	02691

Subroutine ERRPRM

	SUBROUTINE ERRPRM(W,H,F,SC,SG,N,MM,G,S2,A,LL,D)	02691
	DIMENSION H(3,3,150),SC(150),SG(3,3),G(3,3),S2(150),A(6,6),AA(6,6)	02691
	1),W(150),SK(6),D(3,3)	02691
	F = 0.0	02691
	DO 490 L = 1,MM	02691
	SC(L) = 0.0	02691
	DO 480 I = 1,LL	02691
	DO 480 J = 1,LL	02691
480	SC(L) = SC(L) + H(I,J,L)*G(I,J)*(2.0-D(I,J))	02691
490	F = F + W(L)*(SC(L) - S2(L))**2	02691
	V = MM - N	02691
	IF(N - 1)500,500,505	02691
500	SG(1,1) = F/V	02691
	GO TO 680	02691
505	DO 506 I=1,N	
	DO 506 J=1,N	
506	AA(I,J) = A(I,J)	
	CALL DET(AA,0.0,N,TED)	
	N2 = N - 1	02691
	DO 640 K=1,N	02691
	DO 570 I=1,N	02691
	DO 570 J=1,N	02691
	IF(K-1)520,570,510	02691
510	II = I	02691
	GO TO 530	02691
520	II = I - 1	02691
530	IF(K - J)550,570,540	02691
540	JJ = J	02691
	GO TO 560	02691
550	JJ = J - 1	02691
560	AA(II,JJ) = A(I,J)	02691
570	CONTINUE	02691
	CALL DET(AA,0.0,N2,EDT)	02691
640	SK(K) = (F*EDT)/(TED*V)	02691
	DO 670 I = 1,LL	02691
	DO 670 J = 1,LL	02691
	IF(I - J)660,650,660	02691
650	SG(I,I) = SK(I)	02691
	GO TO 670	02691
660	K = 1+I+J	
	SG(I,J) = SK(K)	02691
670	CONTINUE	02691
680	RETURN	02691
	END	02691

Subroutine ERRREL

```

SUBROUTINE ERRREL(G,SG,BE,A,AL,N,SA,SAL,V,SVR,LL) 02691
DIMENSION G(3,3),SG(3,3),BE(3,3),A(3),AL(3,3),SA(3),SAL(3,3),U(3) 02691
V = G(1,1)* G(2,2)* G(3,3) + 2.0* G(2,3)* G(1,3)* G(1,2)-G(1,1)* 02691
1(G(2,3)**2)- G(2,2)*(G(1,3)**2)- G(3,3)*(G(1,2)**2) 02691
V=SQRTF(V)
SZ = 0.0 02691
DO 5 I=1,LL
5 U(I)=0.0
IF(N-1)10,10,20 02691
10 SA(1)=(SG(1,1)*(A(1)/G(1,1))**2)/2.0 02691
GO TO 380 02691
20 DO 310 I=1,LL 02691
DO 310 J=1,LL 02691
DO 310 K=1,LL 02691
IF(I-J)100,310,100 02691
100 IF(I-K)200,310,200 02691
200 IF(J-K)300,310,310 02691
300 U(I) = 2.0* SG(J,K)-((G(J,K)/(G(J,J)* G(K,K)))**2)*(SG(K,K)*(G(J, 02691
1 J) **2)+ SG(J,J)*(G(K,K)**2)) 02691
SZ = SZ +(G(I,I)* U(I))*(G(I,I)*(G(J,K)**2)+ G(I,J)*G(I,K)* SQRTF 02691
1 (G(J,J)*G(K,K))) 02691
310 CONTINUE 02691
SZ = SZ/(2.0 *(V**4)) 02691
D = 1.0 02691
IF(N-4)330,320,330 02691
320 D = -D 02691
330 DO 370 I = 1,LL 02691
DO 370 J = 1,LL 02691
DO 370 K = 1,LL 02691
IF(LL-2)335,360,335 02691
335 IF(I-J)340,370,340 02691
340 IF(I-K)350,370,350 02691
350 IF(J-K)360,370,370 02691
360 SA(I) =(( D*(G(J,K)/(A(I)*(V**2)))*U(I) + (A(I)**2)*( SG(I,I) 02691
1/(G(I,I)**2)+ 2.0*SZ))/2.0 02691
SAL(J,K)=(1.0 - AL(J,K)**2)*(G(I,I)* U(I) + (BE(I,J)**2 + (BE(I, 02691
1 K)**2)*(1.0- BE(I,J)**2)*(AL(J,K)**2))/(1.0 - BE(I,K)**2)) * G(J, 02691
2 J)* U(J) + (BE(I,K)**2 + ((BE(I,J)**2)*(1.0 - BE(I,K)**2)*(AL(J, 02691
3 K)**2))/(1.0 - BE(I,J)**2))* G(K,K)* U(K))/(2.0*(V**2)) 02691
370 CONTINUE 02691
380 SVR=SZ*2.0 02691
420 DO 430 I = 1,3 02691
430 SVR = SVR + SG(I,I)/( G(I,I)**2) 02691
DO 435 I =1,LL 02691
435 SA(I) = SQRTF(SA(I)) 02691
440 SVR= SQRTF(SVR/((V**2)*2.)) 02691
RETURN 02691
END 02691

```

Subroutine OUTPRM

```

SUBROUTINE OUTPRM(A,AL,NAME,B,EB,VR,SA,SAL,SVR,LIST,MM,N,LL,F,SC, 02691
1S2,W,M,N1,JTP) 02691
DIMENSION NAME(10),M(3,150),W(150),S2(150),SC(150),AL(3,3),B(3), 02691
1A(3),SA(3),SAL(3,3),EB(3),ALD(3),SALD(3) 02691
IF(N-4)95,50,50
50 DO 100 I =1,2 02691
K = I +1 02691
DO 100 J =K,3 02691
L = 6-I-J 02691
60 ALD(L)= 90.0 - 57.295780* ARCSIN(AL(I,J)) 02691

```

```

70 D = SQRTF(SAL(I,J))                                02691
100 SALD(L) = 57.295780*ABSF(ABSF(ARCSIN(AL(I,J)+D))-ABSF(ARCSIN(AL(I,J)-D))) 02691
1) - D)))                                              02691
95 WRITE OUTPUT TAPE JTP,1000,(NAME(I),I=1,10),MM
DO 170 I = 1,N                                         02691
GO TO(101,102, 96),LL
96 IF(N-4)103,97,103
97 GO TO(110,120,130,160),I
101 GO TO 110                                          02691
102 GO TO(110,125),I                                  02691
103 GO TO(110,120,130,140,150,160),I
130 WRITE OUTPUT TAPE JTP,1100,B(3),A(3),SA(3)
GO TO 170                                              02691
120 WRITE OUTPUT TAPE JTP,1200,B(2),A(2),SA(2)
GO TO 170                                              02691
125 WRITE OUTPUT TAPE JTP,1100,B(2),A(2),SA(2)
GO TO 170                                              02691
110 WRITE OUTPUT TAPE JTP,1300,B(1),A(1),SA(1)
GO TO 170                                              02691
160 WRITE OUTPUT TAPE JTP,1400,EB(3),ALD(3),SALD(3)
GO TO 170                                              02691
150 WRITE OUTPUT TAPE JTP,1500,EB(2),ALD(2),SALD(2)
GO TO 170                                              02691
140 WRITE OUTPUT TAPE JTP,1600,EB(1),ALD(1),SALD(1)
GO TO 170                                              02691
170 CONTINUE
WRITE OUTPUT TAPE JTP,1700,VR,SVR
WRITE OUTPUT TAPE JTP,1800
GO TO(500,510,520,530,540,550),N1                    02691
500 WRITE OUTPUT TAPE JTP,2400
GO TO 175 .                                           02691
510 WRITE OUTPUT TAPE JTP,2500
GO TO 175                                              02691
520 WRITE OUTPUT TAPE JTP,2600
GO TO 175                                              02691
530 WRITE OUTPUT TAPE JTP,2700
GO TO 175                                              02691
540 WRITE OUTPUT TAPE JTP,2800
GO TO 175                                              02691
550 WRITE OUTPUT TAPE JTP,2900
175 IF(LIST)260,180,260                                02691
180 R = MM - N                                         02691
SD = SQRTF(F/R)                                       02691
SD2 = SD* 2.5                                         02691
L1 = MM/20                                           02691
L2 = MM - L1*20                                       02691
L3 = 0                                                02691
185 L3 = L3+1                                          02691
IF(L1 - L3)200,190,190                                02691
190 K2 = L3*20                                         02691
K1 = K2 -19                                           02691
GO TO 220                                              02691
195 L3=L3+1                                           02691
200 IF(L2)260,260,210                                  02691
210 K1 = L1*20 +1                                     02691
K2 = K1 + L2 -1                                       02691
220 NM = 0                                             02691
WRITE OUTPUT TAPE JTP,2000,L3,SD
DO 250 I = K1,K2                                       02691
DEL =(S2(I)-SC(I))*SQRTF(W(I))                      02691
IF(SD2-ABSF(DEL))230,230,240                          02691
230 NM = 1                                             02691
WRITE OUTPUT TAPE JTP,2100,M(1,I),M(2,I),M(3,I),S2(I),SC(I),W(I),
1DEL
GO TO 250                                              02691

```

```

240 WRITE OUTPUT TAPE JTP,2200,M(1,I),M(2,I),M(3,I),S2(I),SC(I),W(I),
1DEL
250 CONTINUE                                02691
    IF(NM)256,256,255                        02691
255 WRITE OUTPUT TAPE JTP,2300
256 IF(L1-L3)260,195,185                    02691
260 RETURN                                  02691
1000 FORMAT(1H1/1H4,36X,45HUNIT CELL PARAMETERS WITH STANDARD DEVIATION02691
1S//29X,10A6//46X,23HNUMBER OF REFLECTIONS =,I6//) 02691
1100 FORMAT(1H0,26X,5H C* =,F10.7,22X,4H C =,F9.5,6H +OR-,F8.5/) 02691
1200 FORMAT(1H0,26X,5H B* =,F10.7,22X,4H B =,F9.5,6H +OR-,F8.5/) 02691
1300 FORMAT(1H0,26X,5H A* =,F10.7,22X,4H A =,F9.5,6H +OR-,F8.5/) 02691
1400 FORMAT(1H0,23X,8HGAMMA* =,F10.5,19X,7HGAMMA =,F9.4,6H +OR-,F8.4/)02691
1500 FORMAT(1H0,23X,8H BETA* =,F10.5,19X,7H BETA =,F9.4,6H +OR-,F8.4/)02691
1600 FORMAT(1H0,23X,8HALPHA* =,F10.5,19X,7HALPHA =,F9.4,6H +OR-,F8.4/)02691
1700 FORMAT(1H0,64X,3HV =,F11.4,6H +OR-,F10.4////) 02691
1800 FORMAT(33X,48HUNITS ARE 1/(ANGSTROMS), ANGSTROMS, AND DEGREES) 02691
2100 FORMAT(1H0,11X,3I6,6X,E14.7,5X,E14.7,3X,E12.5,6X,E12.5,3H *) 02691
2200 FORMAT(1H0,11X,3I6,6X,E14.7,5X,E14.7,3X,E12.5,6X,E12.5) 02691
2000 FORMAT(1H1,45X,25HOBSERVED AND CALCULATED S,25X,5HPAGE ,I3//27X,102691
18HS = (2*SIN()/L)**2,10X,21HSTANDARD DEVIATION = ,E12.5///17X,13HH02691
2      K      L,10X,6HS(OBS),13X,6HS(CAL),12X,1HW, 9X,18H(DELTA S)*(W02691
3**1/2)) 02691
2300 FORMAT(1H0/9X,58H* THIS DEVIATION IS GREATER THAN 2.5*(STANDARD D02691
1EVIATION)) 02691
2400 FORMAT(1H0,45X,27HTHE CRYSTAL IS TRICLINIC ) 02691
2500 FORMAT(1H0,45X,27HTHE CRYSTAL IS MONOCLINIC ) 02691
2600 FORMAT(1H0,45X,27HTHE CRYSTAL IS ORTHORHOMBIC) 02691
2700 FORMAT(1H0,45X,27HTHE CRYSTAL IS TETRAGONAL ) 02691
2800 FORMAT(1H0,45X,27HTHE CRYSTAL IS HEXAGONAL ) 02691
2900 FORMAT(1H0,45X,27HTHE CRYSTAL IS CUBIC ) 02691
END 02691

```

Quasi Normalization of Structure Factors

```

CSFQNR QUASI NORM. OF STRUCT. FACTORS USING F TAPE INPUT
DIMENSION NME(12), AS(20),AS1(20),BS(20),BS1(20),CS(20),AN(20), 02691
1 H1(500),H2(500),H3(500),M1(500),M2(500),M3(500),AVEQ(500),SI(500)02691
ITP=5
JTAPE = 15 02691
ITAPE = 16 02691
REWIND ITAPE 02691
REWIND JTAPE 02691
READ INPUT TAPE ITP,
1 1000,NOIB,NASP,NF,BT1,EX ,LIB
READ INPUT TAPE ITP,
1 40,(AS(I),AS1(I),BS(I),BS1(I),CS(I),AN(I),I=1,NASP)
IF(NF)5,5,1
1 DO 2 I=1,NF
2 CALL FLKPD(ITAPE)
5 IF(LIB)10,30,10
10 READ INPUT TAPE ITAPE,1200,(NME(I),I=1,12)
30 NO = 0 02691
N1 = NOIB/500 02691
WRITE OUTPUT TAPE JTAPE,740,NOIB 02691
100 IF(N1-NO)400,150,200 02691
150 N4 = NOIB - N1* 500 02691
IF(N4)400,400,210 02691
200 N4 = 500 02691

```

```

210 READ INPUT TAPE ITAPE,1100,(H1(I),H2(I),H3(I),AVEQ(I),SI(I),I= 1, 02691
    1 N4) 02691
    DO 300 I = 1,N4 02691
        M1(I)= INTG(H1(I)) 02691
        M2(I)= INTG(H2(I)) 02691
        M3(I)= INTG(H3(I)) 02691
        SI(I)= SI(I)**2 02691
        SIG = 0.0 02691
        DO 250 J = 1,NASP 02691
250 SIG=SIG+AN(J)*(AS(J)*EXPF(-AS1(J)*SI(I))+ BS(J)*EXPF(-BS1(J)*SI(I) 02691
    1)+CS(J))**2 02691
300 AVEQ(I)=(AVEQ(I)* EXPF(BT1*(SI(I)**EX)))/SIG)- 1.0 02691
    WRITE OUTPUT TAPE JTAPE,680,(M1(I),M2(I),M3(I),AVEQ(I),I =1,N4) 02691
    NO = NO +1 02691
    GO TO 100 02691
400 REWIND ITAPE 02691
    END FILE JTAPE 02691
    REWIND JTAPE 02691
    PRINT 2000 02691
    CALL EXIT
1000 FORMAT(3I10,2E10.4,I10)
    40 FORMAT(5F8.4,F4.0) 02691
1100 FORMAT(4F9.2,18X,F9.6) 02691
    680 FORMAT(3I4,F20.5) 02691
2000 FORMAT( 16H      JOB FINSHED./////////)
    740 FORMAT(17) 02691
1200 FORMAT(12A6)
    END

```

Rational Dependence

```

CRATDEP    RATIONAL DEPENDANCE (AFTER BLOCK-YANNONI)
  DIMENSION NAME(12), NF(29),KA(28),KB(28),
  1KC(28),EC(29),MK(5000),ML(5000),MH(5000),ESQM(5000),MR(5000)
  COMMON MH,MK,ML,ESQM,MR,KA,KB,KC,NF,MAD,M,IT,NAME,N,J,K,MA,JTP,
  1AVESQ,JPUT,NCTR,NO
  KA(1) = 0                                06800
  KA(2) = 1                                06800
  KA(3) = -1                               06800
  KA(4) = 2                                06800
  KA(5) = -2                               06800
  KA(6) = 3                                06800
  KA(7) = -3                               06800
  KA(8) = 4                                06800
  KA(9) = -4                               06800
  KA(10) = 5                               06800
  KA(11) = -5                              06800
  KA(12) = 6                               06800
  KA(13) = -6                              06800
  KA(14) = 7                               06800
  KA(15) = -7                              06800
  KA(16) = 8                               06800
  KA(17) = -8                              06800
  KA(18) = 9                               06800
  KA(19) = -9                              06800
  KA(20) = 10                              06800
  KA(21) = -10                             06800
  KA(22) = 11                              06800
  KA(23) = -11                             06800
  KA(24) = 12                              06800
  KA(25) = -12                             06800
  KA(26) = 13                              06800
  KA(27) = -13                             06800
  KA(28) = 14                              06800
  ITP=5
  JTP=6
  KTP=12
  REWIND KTP
50 READ INPUT TAPE ITP,8,(NAME(I),I=1,12)
  READ INPUT TAPE ITP,4,NA,NINT,NFIN,MINN,AN,SDMIN,JPUT
60 READ INPUT TAPE KTP,5,NO
61 READ INPUT TAPE KTP,6,(MH(J),MK(J),ML(J),ESQM(J),J=1,NO)
  REWIND KTP
  SUMESQ=0.0
  DO 62 I=1,NO
    ESQM(I)=ESQM(I)+1.0
62 SUMESQ=SUMESQ+ESQM(I)
  BNO=NO
  AVESQ=SUMESQ/BNO
70 MA=NINT
  NCTR=-1
  WRITE OUTPUT TAPE JTP,15,(NAME(J),J=1,12),NINT,NFIN,AVESQ,AN,MINN,
  1NO,SDMIN
80 IF(NA)83,90,90                                06800
83 AN=1.414214*AN                                06800
90 DO 95 J= 1,28
92 KB(J)=KA(J)                                06800
95 KC(J)=KA(J)                                06800
180 DO 440 N=1,MA                                06800
190 DO 440 J=1,MA                                06800
200 DO 440 K=1,MA                                06800
  N=N
  J=J
  K=K
  IF(KA(N)) 440,2000,202

```

```

2000 IF(KB(J))440,2010,202
2010 IF(KC(K))440,440,202
202 MAD=MA+1
DO 207 LD=1,MAD
NF(LD)=0
207 EC(LD)=0.
2100 DO 2400 L=1,NO
KSUM=KA(N)*MH(L)+KB(J)*MK(L)+KC(K)*ML(L)
MT=XMODF(KSUM,MA)
IF(MT)2200,2300,2300
2200 MT=MT+MA
2300 MT=MT+1
MR(L)=MT
EC(MT)=ESQM(L)+EC(MT)
2400 NF(MT)=NF(MT)+1
300 PMA=MA
301 PMA=PMA/2.
302 MAN=PMA
303 MAN=MAN+1
304 IT=MA+2
310 DO 430 M=1,MAN
311 IT=IT-1
1315 IF(IT-M) 1320,1321,1325
1320 STOP 0066
1321 DNF=NF(M)
JNF=NF(M)
1322 BEC=EC(M)
1323 GO TO 1350
1325 JNF=NF(M)+NF(IT)
1330 DNF=JNF
1340 BEC=EC(M)+EC(IT)
1350 IF(JNF-MINN)430,430,1355
1355 AVESQG=BEC/DNF
1360 CHEK=AN/SQRTF(DNF)
IF(CHEK-SDMIN)1365,1370,1370
1365 CHEK=SDMIN
1370 PHEK=AVESQ+CHEK
1380 PMHEK=AVESQ-CHEK
360 IF(AVESQG-PHEK) 370,380, 380
370 IF(AVESQG-PMHEK)380, 380, 430
380 IF (MA-3)400,400,3193
3193 IR = 0
3194 CAT=2.0
3200 XMA=MA
3210 XLA=XMA/CAT
3215 IZ=0
3220 MX=XLA
3230 YMX=MX
3240 IF (YMX-XLA) 3350,3250,3350
3250 XKE=KA(N)
3260 AAD=XKE/CAT
3270 MAC=AAD
3280 ABD=MAC
3290 IF (ABD-AAD) 3350,3300,3350
3300 IZ=IZ+1
3310 GO TO (3320,3330,3340,430),IZ
3320 XKE=KB(J)
3325 GO TO 3260
3330 XKE=KC(K)
3335 GO TO 3260
3340 XKE =M-1
3345 GO TO 3260
3350 IR=IR+1
3355 IF (MA-IR-3) 400,400,3360
3360 CAT=CAT+1.

```

06800

06800

06800

06800

06800

06800

06800

06800

06800

06800

06800

06800

06800

06800

06800

06800

06800

06800

06800

06800

06800

06800

06800

06800

```

3365 IF(CAT-14,0) 3210,3210,400
400 CALL OUTRAT
430 CONTINUE
    IF(SENSE SWITCH 2)510,440
440 CONTINUE
450 MA=MA+1
460 IF(NFIN-MA) 470,180,180
470 READ INPUT TAPE ITP,4,NA,NINT,NFIN,MINN,AN,SDMIN,JPUT
    IF(NINT)500,500,70
500 CALL EXIT
510 WRITE OUTPUT TAPE JTP,9,(NAME(L2),L2=1,12)
    WRITE OUTPUT TAPE JTP,7,KA(N),KB(J),KC(K),MA
    WRITE OUTPUT TAPE JTP,16
    GO TO 500
8  FORMAT (12A6)
4  FORMAT (4I10,2E10,4,I10)
5  FORMAT (I7)
6  FORMAT(3I4,F20.5)
7  FORMAT(6H0 A =I3,6X,3HB =I3,6X,3HC =I3,6X,6HMODE =I6)
9  FORMAT (1H1,19X,12A6)
15 FORMAT(1H2,19X,12A6///20X,16HINITIAL MODULAS =I6,10X,15HFINAL MODU
1LAS =I6///20X,19HAVERAGE E**2 =F9.5///20X, 13HDEVIATIONS OFF7
2.3, 66H STANDARD DEVIATIONS FROM AVE E**2 ARE CONSIDERED SIGN
3IFICANT///20X, 40HONLY SUBGROUPS CONTAINING NO FEWER THAN I6,
4 39H REFLECTIONS ARE CONSIDERED SIGNIFICANT///20X, 10HTHERE ARE I7
5, 30H REFLECTIONS ON THE INPUT TAPE///20X,51HTHE MINIMUM ACCEPTED
6DEVIATION FROM AVE. E**2 IS F5.3)
16 FORMAT(1H0,19X,47HCALCULATION TERMINATED BY SENSE SWITCH CONTROL.)
END

```

06800
06800

Subroutine OUTRAT

```

    DIMENSION NAME(12), NF(29),KA(28),KB(28),
1KC(28),EC(29),MK(5000),ML(5000),MH(5000),ESQM(5000),MR(5000),
2LH(40),LK(40),LL(40),ESQL(40)
    COMMON MH,MK,ML,ESQM,MR,KA,KB,KC,JNF,MAD,M,IT,NAME,N,J,K,MA,JTP,
1AVESQG,JPUT,NCTR,NO
    KET=M-1
    IF(JPUT)75,50,75
50 NCTR=NCTR+1
    IF(NCTR-(NCTR/9)*9)60,55,60
55 WRITE OUTPUT TAPE JTP,9,(NAME(L2),L2=1,12)
60 WRITE OUTPUT TAPE JTP,8,KA(N),KB(J),KC(K),MA,KET, AVESQG,JNF
    GO TO 350
75 NP=0
    L4=0
    DO 220 L9=1,NO
    L9=L9
    IF(MR(L9)-M)200,210,200
200 IF(MR(L9)-IT)220,210,220
210 L4=L4+1
    LH(L4)=MH(L9)
    LK(L4)=MK(L9)
    LL(L4)=ML(L9)
    ESQL(L4)=ESQM(L9)
    IF(40-L4)230,230,220
220 CONTINUE
230 J1=L4/40
    J2=L4-J1*40
    J3=J2-20
    IF (J3)241,241,243
241 J4=J2
    GO TO 245

```

```

243 J4=J2-2*J3
GO TO 247
245 IF (J1)300,300,246
246 J5=J1
GO TO 250
247 J5=J1+1
250 DO 290 L1=1,J5
    NP=NP+1
    WRITE OUTPUT TAPE JTP,9,(NAME(L2),L2=1,12)
    WRITE OUTPUT TAPE JTP,7,KA(N),KB(J),KC(K),MA,KET,NP,AVESQG,JNF
    L5=L1*40-39
    IF (J1-L1)260,270,270
260 L6=L5+J3-1
GO TO 280
270 L6=L5+19
280 DO 290 L2=L5,L6
    AQ=ESQL(L2)-1.0
    AQA=ESQL(L2)-AVESQG
    AQ1=ESQL(L2+20)-1.0
    AQA1=ESQL(L2+20)-AVESQG
    WRITE OUTPUT TAPE JTP,10
290 WRITE OUTPUT TAPE JTP,11,LH(L2),LK(L2),LL(L2),AQ,AQA,LH(L2+20),
    LK(L2+20),LL(L2+20),AQ1,AQA1
300 IF (J4)360,360,310
310 IF (J3)320,320,315
315 L5=L6+1
GO TO 335
320 NP=NP+1
    WRITE OUTPUT TAPE JTP,9,(NAME(L2),L2=1,12)
    WRITE OUTPUT TAPE JTP,7,KA(N),KB(J),KC(K),MA,KET,NP,AVESQG,JNF
330 L5=L6+21
335 L6=L5+J4-1
DO 340 L2=L5,L6
    AQ=ESQL(L2)-1.0
    AQA=ESQL(L2)-AVESQG
340 WRITE OUTPUT TAPE JTP,12,LH(L2),LK(L2),LL(L2),AQ,AQA
350 RETURN
360 L4=0
    IF(NO-L9)350,350,220
7 FORMAT(6H0 A =I3,6X,3HB =I3,6X,3HC =I3,6X,6HMODE =I6,8X,11HREMAIN
1DER =I6,24X,4HPAGEI5///22X,10HAVE E**2 =F10.5,8X,11HNO. REFL. =I6/
2///2(60H H K L E**2 - 1 E**2 - A
3 ) ;
8 FORMAT(6H0 A =I3,6X,3HB =I3,6X,3HC =I3,6X,6HMODE =I6,8X,11HREMAIN
1DER =I6,24X,6H ///22X,10HAVE E**2 =F10.5,8X,11HNO. REFL. =I6)
9 FORMAT (1H1,19X,12A6)
10 FORMAT (1H )
11 FORMAT (2(1X,3I4,7X,F9.5,7X,F9.5,15X))
12 FORMAT (1H0,3I4,7X,F9.5,7X,F9.5)
END

```

Sigma-2 Listings

```

CSIGEXC SIGMA-2 INTERACTION EXC
    DIMENSION M1(2400),M2(2400),M3(2400),AV(2400),K1(2400),K2(2400),K3
1(2400),QAV(2400)
    COMMON M1,M2,M3,AV
    ITP=5
    IP =11
    L=0
    READ INPUT TAPE ITP,
1    1002,Z,I2,NOIB
    IF(I2) 2, 1,2

```

```

1 MM=2                                02691
  REWIND IP
  READ INPUT TAPE IP,1001,NRT
  GO TO 3
2 MM=1                                02691
  READ INPUT TAPE ITP,1001,NRT      02691
3 DO 25 I=1,NRT                      02691
20 GO TO(5,6),MM
5 READ INPUT TAPE ITP,
  1 1000,N1,N2,N3,Q
  GO TO 7                                02691
6 READ INPUT TAPE IP,1000,N1,N2,N3,Q
7 CALL REJVEC(N1,N2,N3,Q)
  IF(Q-Z)25,10,10
10 L=L+1                              02691
  M1(L)=N1
  M2(L)=N2
  M3(L)=N3
  AV(L)=SORTF(1.0+Q)
25 CONTINUE                          02691
30 DO 160 J=1,L
40 DO 70 K=1,L
50 IF(AV(K))70,60,60
60 BAV=AV(K)
  GO TO 80
70 CONTINUE
80 DO 110 I=K,L
90 IF(AV(I)-BAV)110,100,100
100 BAV=AV(I)
  I1=I
110 CONTINUE
  K1(J)=M1(I1)
  K2(J)=M2(I1)
  K3(J)=M3(I1)
  QAV(J)=AV(I1)
  AV(I1)=-1.0
160 CONTINUE
  IF(NOIB)165,180,170
165 NOIB=L
170 CALL SIGVEC(NOIB,K1,K2,K3,QAV,L)  02693
180 CALL EXIT
1000 FORMAT(3I4,F20.5)
1001 FORMAT(I7)
1002 FORMAT(E10.4,2I10)
  END

```

Subroutine OUTSIG

```

SUBROUTINE OUTSIG(M1,M2,M3,Q,N1,N2,N3,  02693
  Q1,QQ,N,K1,K2,K3,I,QAV)             02693
  DIMENSION K1(2400),K2(2400),K3(2400),  02693
  QAV(2400),M1(1000),M2(1000),M3(1000),Q(1000),  02693
  N1(1000),N2(1000),N3(1000),Q1(1000),QQ(1000)  02693
  COMMON M1,M2,M3,Q,N1,N2,N3,Q1,QQ      02693
  JTP=6
  IF(N)25,25,50                        02693
25 K=1
  WRITE OUTPUT TAPE JTP,
  1 1000,K,K1(I),K2(I),K3(I),QAV(I)
  GO TO 130                            02693
50 J=N/20                              02693
  IF(J)120,120,100                     02693

```

02693
02693
02693

02693
02693
02693

02693
5

02693

Subroutine REJVEC

```

SUBROUTINE REJVEC(N1,N2,N3,Q)
C   DUMMY
100  RETURN
END

SUBROUTINE REJVEC(N1,N2,N3,Q)
C   REJECTS ALL REFL. FOR WHICH ANY OF H,K,L EQUAL ZERO
    IF(N1)10,30,10
10  IF(N2)20,30,20
20  IF(N3)40,30,40
30  Q=-10.
40  RETURN
END

```

Subroutine SIGVEC, Triclinic

02693
02693
02693
02693
02693
02693
02693
02693
02693

```

110 IF(K3(J)-K3(K)+K3(I))160,150,160
150 N=N+1
    M1(N)=K1(J)
    M2(N)=K2(J)
    M3(N)=K3(J)
    Q(N) = QAV(J)                                02693
    N1(N)=K1(K)
    N2(N)=K2(K)
    N3(N)=K3(K)
    Q1(N)= QAV(K)                                02693
    QQ(N)= QAV(J)* QAV(K)*QAV(I)
160 CONTINUE                                    02693
170 CALL OUTSIG(M1,M2,M3,Q,N1,N2,N3,Q1,QQ,N,
    1 K1,K2,K3,I,QAV)
    RETURN                                       02693
    END                                         02693

```

Subroutine SIGVEC, Monoclinic

```

C MONOCLINIC, SECOND SETTING
SUBROUTINE SIGVEC(NOIB,K1,K2,K3,QAV,L)          02693
DIMENSION K1(2400),K2(2400),K3(2400),        02693
1QAV(2400),M1(1000),M2(1000),M3(1000),Q(1000),
2N1(1000),N2(1000),N3(1000),Q1(1000),QQ(1000) 02693
COMMON M1,M2,M3,Q,N1,N2,N3,Q1,QQ             02693
DO 170 I=1,NOIB                               02693
N=0                                             02693
DO 160 J=1,L                                  02693
DO 160 K=J,L                                  02693
    IF(XABSF(K2(J)+K2(K))-XABSF(K2(I))) 10, 20, 10
10 IF(XABSF(K2(J)-K2(K))-XABSF(K2(I)))160, 20,160
20 IF(K1(J)+K1(K)-K1(I)) 30, 60,30
30 IF(K1(J)+K1(K)+K1(I)) 40, 70,40
40 IF(K1(J)-K1(K)-K1(I)) 50, 80,50
50 IF(K1(J)-K1(K)+K1(I))160, 90,160
60 M=1
    IF(K3(J)+K3(K)-K3(I))100,150,100
70 M=2
    IF(K3(J)+K3(K)+K3(I))100,150,100
80 IF(K3(J)-K3(K)-K3(I))146,150,146
90 IF(K3(J)-K3(K)+K3(I))160,150,160
100 IF(K1(J))120,110,120
110 GO TO(140,145),M
120 IF(K1(K))160,130,160
130 GO TO(145,140),M
140 IF(K3(J)-K3(K)+K3(I))160,150,160
145 IF(K3(J)-K3(K)-K3(I))160,150,160
146 IF(K1(I))160,90,160
150 N=N+1
    M1(N)=K1(J)
    M2(N)=K2(J)
    M3(N)=K3(J)
    Q(N) = QAV(J)                                02693
    N1(N)=K1(K)
    N2(N)=K2(K)
    N3(N)=K3(K)
    Q1(N)= QAV(K)                                02693
    QQ(N)= QAV(J)* QAV(K)*QAV(I)
160 CONTINUE                                    02693
170 CALL OUTSIG(M1,M2,M3,Q,N1,N2,N3,Q1,QQ,N,
    1 K1,K2,K3,I,QAV)
    RETURN                                       02693
    END                                         02693

```

Subroutine SIGVEC, Orthorhombic

```

      ORTHORHOMBIC
      SUBROUTINE SIGVEC(NOIB,K1,K2,K3,QAV,L)
      DIMENSION K1(2400),K2(2400),K3(2400),
1QAV(2400),M1(1000),M2(1000),M3(1000),Q(1000),
2N1(1000),N2(1000),N3(1000),Q1(1000),QQ(1000)
      COMMON M1,M2,M3,Q,N1,N2,N3,Q1,QQ
      DO 170 I=1,NOIB
      N=0
      DO 160 J=1,L
      DO 160 K=J,L
      IF(XABSF(K1(J)+K1(K))-XABSF(K1(I)))100,110,100
100 IF(XABSF(K1(J)-K1(K))-XABSF(K1(I)))160,110,160
110 IF(XABSF(K2(J)+K2(K))-XABSF(K2(I)))120,130,120
120 IF(XABSF(K2(J)-K2(K))-XABSF(K2(I)))160,130,160
130 IF(XABSF(K3(J)+K3(K))-XABSF(K3(I)))140,150,140
140 IF(XABSF(K3(J)-K3(K))-XABSF(K3(I)))160,150,160
150 N=N+1
      M1(N)=K1(J)
      M2(N)=K2(J)
      M3(N)=K3(J)
      Q(N) = QAV(J)
      N1(N)=K1(K)
      N2(N)=K2(K)
      N3(N)=K3(K)
      Q1(N)= QAV(K)
      QQ(N)= QAV(J)* QAV(K)*QAV(I)
160 CONTINUE
170 CALL OUTSIG(M1,M2,M3,Q,N1,N2,N3,Q1,QQ,N,
1 K1,K2,K3,I,QAV)
      RETURN
      END

```

02693
02693
02693
02693
02693
02693
02693
02693

02693

02693

02693

02693

02693

Triple Product Summation, Centrosymmetric

```

      CORTHP ORTHORHOMBIC TRIP. PROD. PROG. FOR CENTRO. CRYST.
      DIMENSION E(30,10,30)
      ITP=5
      JTP=6
      KTP=9
      REWIND KTP
      B=1.0E-21
      READ INPUT TAPE ITP,1000,NA,IM,JM,KM
1000 FORMAT(6I10)
      DO 75 I=1,IM
      DO 75 J=1,JM
      DO 75 K=1,KM
      75 E(I,J,K)=B
      READ INPUT TAPE KTP,20Q0,NOIB
2000 FORMAT(I7)
      DO 8 NO=1,NOIB
      READ INPUT TAPE KTP,2100,I,J,K,A
2100 FORMAT(3I4,F20.5)
      IF(I)1,8,2
      1 I=-I
      2 IF(J)3,8,4
      3 J=-J
      4 IF(K)5,8,6
      5 K=-K
      6 E(I,J,K)=A

```

```
8 CONTINUE
REWIND KTP
AN=NA
SF=SQRTF(AN**3)/8.0
JJJ=0
MN=0
C START NEW TP CALC.
9 READ INPUT TAPE ITP,1000,M1,M2,M3,N1,N2,N3
JJJ=JJJ+1
I1=XABSF(M1)
I2=XABSF(M2)
I3=XABSF(M3)
J1=XABSF(N1)
J2=XABSF(N2)
J3=XABSF(N3)
L1=M1+N1
L2=M2+N2
L3=M3+N3
K1=XABSF(L1)
K2=XABSF(L2)
K3=XABSF(L3)
IF(I1+I2+I3+J1+J2+J3)10,10,50
10 MN=MN+1
EEE=1.0
COR=0.0
GO TO(55,630),MN
50 IF(M1)11,20,11
11 IF(M2)12,20,12
12 IF(M3)13,20,13
13 IF(N1)14,20,14
14 IF(N2)15,20,15
15 IF(N3)17,20,17
17 IF(L1)18,20,18
18 IF(L2)19,20,19
19 IF(L3)25,20,25
20 EEE=1.0
COR=0.0
GO TO 55
25 EEE=SQRTF((E(I1,I2,I3)+1.0)*(E(J1,J2,J3)+1.0)*(E(K1,K2,K3)+1.0))
COR=(E(I1,I2,I3)+E(J1,J2,J3)+E(K1,K2,K3)+1.0)/SQRTF(AN)
C CALCULATE TRIPLE AVERAGES
55 RN=0.0
TP=0.0
DO 600 I=1,IM
DO 600 J=1,JM
DO 600 K=1,KM
IF(E(I,J,K)-B)90,600,90
90 L=I
M=J
N=K
NNN=0
100 IH=XABSF(L1+L)
IK=XABSF(L2+M)
IL=XABSF(L3+N)
IF(IH)110,500,110
110 IF(IM-IH)500,120,120
120 IF(IK)130,500,130
130 IF(JM-IK)500,140,140
140 IF(IL)150,500,150
150 IF(KM-IL)500,160,160
160 IF(E(IH,IK,IL)-B)165,500,165
165 KH=XABSF(M1+L)
KK=XABSF(M2+M)
KL=XABSF(M3+N)
```

```

167 IF(KH)170,500,170
170 IF(IM-KH)500,180,180
180 IF(KK)190,500,190
190 IF(JM-KK)500,200,200
200 IF(KL)210,500,210
210 IF(KM-KL)500,220,220
220 IF(E(KH,KK,KL)-B)230,500,230
230 RN=RN+1.0
    TP=TP+E(I,J,K)*E(IM,IK,IL)*E(KH,KK,KL)
500 NNN=NNN+1
    GO TO(510,520,510,530,510,540,510,600),NNN
510 L=-L
    M=-M
    N=-N
    GO TO 100
520 L=-L
    GO TO 100
530 M=-M
    GO TO 100
540 M=-M
    N=-N
    GO TO 100
600 CONTINUE
    TP=TP*SF/RN+COR
    NR=RN
    TPC=TP/EEE
    IF(XMODF(JJJ,15)-1)620,610,620
610 WRITE OUTPUT TAPE JTP,1100
620 IF(MN)625,625,621
621 WRITE OUTPUT TAPE JTP,1300,M1,M2,M3,N1,N2,N3,NR,TP,TPC,COR
    PRINT 1300,M1,M2,M3,N1,N2,N3,NR,TP,TPC,COR
    GO TO 9
625 IF(SENSE SWITCH 1)626,627
626 PRINT 1200,M1,M2,M3,N1,N2,N3,NR,TP,TPC,COR
627 L1=-L1
    L2=-L2
    L3=-L3
    WRITE OUTPUT TAPE JTP,1400,L1,L2,L3
    WRITE OUTPUT TAPE JTP,1200,M1,M2,M3,N1,N2,N3,NR,TP,TPC,COR
    GO TO 9
630 CALL EXIT
1100 FORMAT(1H1,10X,75H H K L H K L NO.CONTR. TRIP.PRO
    1D. COS(INV.) COR.TERM///)
1200 FORMAT(1H0,10X,I3,2I4,I5,2I4,3X,I8,2X,3(3X,F10.5))
1300 FORMAT(1H0,10X,I3,2I4,I5,2I4,3X,I8,2X,3(3X,E10.4))
1400 FORMAT(1H0,10X,I3,2I4)
    END

```

Triple Product Summation, Noncentrosymmetric

```

CORTHIP ORTHORHOMBIC TRIP. PROD. PROG. FOR NONCENTRO. CRYST.
DIMENSION E(30,10,30)
ITP=5
JTP=6
KTP=9
REWIND KTP
B=1.0E-21
READ INPUT TAPE ITP,1000,NA,IM,JM,KM
1000 FORMAT(6I10)
DO 75 I=1,IM
DO 75 J=1,JM
DO 75 K=1,KM
75 E(I,J,K)=B
READ INPUT TAPE KTP,2000,NOIB

```

```

2000 FORMAT(I7)
      DO 8 NO=1,NO1B
      READ INPUT TAPE KTP,2100,I,J,K,A
2100 FORMAT(3I4,F20.5)
      IF(I)1,8,2
1      I=-I
2      IF(J)3,8,4
3      J=-J
4      IF(K)5,8,6
5      K=-K
6      E(I,J,K)=A
8      CONTINUE
      REWIND KTP
      AN=NA
      SF=SQRTF(AN**3)/2.0
      JJJ=0
      MN=0
C      START NEW TP CALC.
9      READ INPUT TAPE ITP,1000,M1,M2,M3,N1,N2,N3
      JJJ=JJJ+1
      I1=XABSF(M1)
      I2=XABSF(M2)
      I3=XABSF(M3)
      J1=XABSF(N1)
      J2=XABSF(N2)
      J3=XABSF(N3)
      L1=M1+N1
      L2=M2+N2
      L3=M3+N3
      K1=XABSF(L1)
      K2=XABSF(L2)
      K3=XABSF(L3)
      IF(I1+I2+I3+J1+J2+J3)10,10,50
10     MN=MN+1
      EEE=1.0
      COR=0.0
      GO TO(55,630),MN
50     IF(M1)11,20,11
11     IF(M2)12,20,12
12     IF(M3)13,20,13
13     IF(N1)14,20,14
14     IF(N2)15,20,15
15     IF(N3)17,20,17
17     IF(L1)18,20,18
18     IF(L2)19,20,19
19     IF(L3)25,20,25
20     EEE=1.0
      COR=0.0
      GO TO 55
25     EEE=SQRTF((E(I1,I2,I3)+1.0)*(E(J1,J2,J3)+1.0)*(E(K1,K2,K3)+1.0))
      COR=(E(I1,I2,I3)+E(J1,J2,J3)+E(K1,K2,K3)+1.0)/SQRTF(AN)
C      CALCULATE TRIPLE AVERAGES
55     RN=0.0
      TP=0.0
      DO 600 I=1,IM
      DO 600 J=1,JM
      DO 600 K=1,KM
      IF(E(I,J,K)-8)90,600,90
90     L=I
      M=J
      N=K
      NNN=0
100    IH=XABSF(L1+L)
      IK=XABSF(L2+M)
      IL=XABSF(L3+N)
      IF(IH)110,900,110

```

```

110 IF(IM-IH)500,120,120
120 IF(IK)130,500,130
130 IF(JM-IK)500,140,140
140 IF(IL)150,500,150
150 IF(KM-IL)500,160,160
160 IF(E(IH,IK,IL)-B)165,500,165
165 KH=XABSF(M1+L)
    KK=XABSF(M2+M)
    KL=XABSF(M3+N)
167 IF(KH)170,500,170
170 IF(IM-KH)500,180,180
180 IF(KK)190,500,190
190 IF(JM-KK)500,200,200
200 IF(KL)210,500,210
210 IF(KM-KL)500,220,220
220 IF(E(KH,KK,KL)-B)230,500,230
230 RN=RN+1.0
    TP=TP+E(I,J,K)*E(IH,IK,IL)*E(KH,KK,KL)
500 NNN=NNN+1
    GO TO(510,520,530,540,510,540,510,600),NNN
510 L=-L
    M=-M
    N=-N
    GO TO 100
520 L=-L
    GO TO 100
530 M=-M
    GO TO 100
540 M=-M
    N=-N
    GO TO 100
600 CONTINUE
    TP=TP*SF/RN+COR
    NR=RN
    TPC=TP/EEE
    IF(XMODF(JJJ,20)-1)620,610,620
610 WRITE OUTPUT TAPE JTP,1100
620 IF(MN)625,625,621
621 WRITE OUTPUT TAPE JTP,1300,M1,M2,M3,N1,N2,N3,NR,TP,TPC,COR
    PRINT 1300,M1,M2,M3,N1,N2,N3,NR,TP,TPC,COR
    GO TO 9
625 WRITE OUTPUT TAPE JTP,1200,M1,M2,M3,N1,N2,N3,NR,TP,TPC,COR
    PRINT 1200,M1,M2,M3,N1,N2,N3,NR,TP,TPC,COR
    GO TO 9
630 CALL EXIT
1100 FORMAT(1H1,10X,75H H K L H K L NO.CONTR. TRIP.PRO
1D. COS(INV.) COR.TERM///)
1200 FORMAT(1H0,10X,I3,2I4,I5,2I4,3X,I8,2X,3(3X,F10.5))
1300 FORMAT(1H0,10X,I3,2I4,I5,2I4,3X,I8,2X,3(3X,E10.4))
END

```

Structure Factor Calculation

```

CSTRFAC      STRUCTURE FACTOR EXECUTIVE PROGRAM
  DIMENSION NAME(12),X(3,500),H(3,128), F1(20),FA1(20),FB(20),FB1(
  1 20),FC(20), AA(3,3),IA(500),TB(500),AF(500),AF1(500),BF(500
  2 ),BF1(500),CF(500),SI(128),E(128),F(128),A(128),FA(128),P(500),
  3 GYP(500),FK(128),C(128),SF(128),EN(128),B(128),D(128),DF(128),DE(
  4 128),CO(128),DO(128),DC(128),DD(128),AO(128),BO(128),DA(128),DB(I
  5 28),NME(12),FJ(128),AB(3,3),SCF2(100)

```

```
50 KT=10
   IT=5
   JT=6
   LT=11
   REWIND KT
   REWIND LT
   NASP=0
   I=0
   READ INPUT TAPE IT,1000,(NAME(I),I=1,12)
   READ INPUT TAPE IT,1100,IC,IE,NR,NF,IP,LIB,LINE
   READ INPUT TAPE IT,1200,A1,A2,A3,AL,BE,GA
   READ INPUT TAPE IT,1300,BT,XT,SCF,KKT,IOUT
60 I=I+1
   READ INPUT TAPE IT,1600,SCF2(I)
   IF(SCF2(I))60,100,60
100 NASP=NASP+1
   READ INPUT TAPE IT,1400,F1(NASP),FA1(NASP),FB(NASP),FB1(NASP),
   1 FC(NASP)
   IF(F1(NASP))100,110,100
110 K=0
120 K=K+1
   READ INPUT TAPE IT,1500,X(1,K),X(2,K),X(3,K),TB(K),IA(K)
   IF(ABSF(X(1,K))+ABSF(X(2,K))+ABSF(X(3,K)))130,130,120
130 K=K-1
   NC=K
140 READ INPUT TAPE IT,1600,T1,T2,T3,U1,U2,U3
   IF(ABSF(T1)+ABSF(T2)+ABSF(T3)+ABSF(U1)+ABSF(U2)+ABSF(U3))170,170,
   1 150
150 DO 160 I=1,NC
   K=K+1
   X(1,K)=T1+U1*X(1,I)
   X(2,K)=T2+U2*X(2,I)
   X(3,K)=T3+U3*X(3,I)
   IA(K)=IA(I)
160 TB(K)=TB(I)
   GO TO 140
170 SDF=0.0
   SFO=0.0
   SDF0=0.0
   DO 180 I=1,K
   J=IA(I)
   AF(I)=F1(J)
   AF1(I)=FA1(J)
   BF(I)=FB(J)
   BF1(I)=FB1(J)
180 CF(I)=FC(J)
   IF(IE)190,195,190
190 NN=1
   GO TO 200
195 NN=2
200 CALL RECIP(A1,A2,A3,AL,BE,GA,AB(1,1),AB(2,2),AB(3,3),AB(2,3),AB(1,
   1 3),AB(1,2))
   DO 230 I=1,3
   DO 230 J=1,3
   IF(I-J)220,210,220
210 AA(I,I)= AB(I,I)**2
   GO TO 230
220 AA(I,J)= 2.0*AB(I,J)*AB(I,I)*AB(J,J)
230 CONTINUE
   IF(NF)240,240,231
231 DO 232 I=1,NF
232 CALL FLSKPD(KT)
240 IF(LIB)242,243,242
242 READ INPUT TAPE KT,1000,(NME(I),I=1,12)
243 GO TO(244,2401),NN
```

```

244 IF(LINE)2403,2402,2403
2402 READ INPUT TAPE KT,2100,NR
GO TO 2401
2403 READ INPUT TAPE IT,2100,NR
2401 NO=0
N1=NR/128
245 IF(N1-NO)470,250,260
250 N2=NR-N1*128
IF(N2)470,470,270
260 N2=128
270 GO TO(280,310),NN
280 IF(LINE)281,282,281
281 READ INPUT TAPE IT,1700,(H(1,I),H(2,I),H(3,I),E(I),I=1,N2)
GO TO 283
282 READ INPUT TAPE KT,1700,(H(1,I),H(2,I),H(3,I),E(I),I=1,N2)
283 DO 300 I=1,N2
E(I)= SQRTF(E(I)+1.0)*SCF
SI(I)=0.0
DO 290 L=1,3
DO 290 M=L,3
290 SI(I)=H(L,I)*H(M,I)*AA(L,M)+SI(I)
300 SI(I)=SI(I)/4.0
GO TO 330
310 IF(LINE)311,312,311
311 DO 2311 I=1,N2
READ INPUT TAPE IT,1800, H(1,I),H(2,I),H(3,I),F(I),SCF,SI(I)
IF(SCF)2314,2314,2313
2313 JK=INTG(SCF)
2314 F(I)=SQRTF(F(I))*SCF2(JK)
2311 CONTINUE
GO TO 313
312 DO 3311 I=1,N2
READ INPUT TAPE KT,1800, H(1,I),H(2,I),H(3,I),F(I),SCF,SI(I)
IF(SCF)3314,3314,3313
3313 JK=INTG(SCF)
3314 F(I)=SQRTF(F(I))*SCF2(JK)
3311 CONTINUE
313 IF(SI(1))314,314,317
314 DO 316 I=1,N2
SI(I)=0.0
DO 315 L=1,3
DO 315 M=L,3
315 SI(I)=H(L,I)*H(M,I)*AA(L,M)+SI(I)
316 SI(I)=SI(I)/4.0
GO TO 330
317 DO 320 I=1,N2
320 SI(I)= SI(I)**2
330 DO 450 I=1,N2
A(I)=0.0
SIGK=0.0
FK(I)=0.0
DO 350 J=1,K
FA(J)=AF(J)*EXPF(-AF1(J)*SI(I))+BF(J)*EXPF(-BF1(J)*SI(I))+CF(J)
SIGK= SIGK+FA(J)**2
FK(I)=FK(I)+(FA(J)*EXPF(-TB(J)*SI(I)))**2
P(J)=0.0
DO 340 L=1,3
340 P(J)=P(J)+H(L,I)*X(L,J)
P(J)=P(J)*6.2831853
GYP(J)=FA(J)*EXPF(-TB(J)*SI(I))
350 A(I)=A(I)+GYP(J)*COSF(P(J))
FBTX=EXPF((BT*(SI(I))**XT)/2.0)/SQRTF(SIGK)
FFAX=1.0/SQRTF(FK(I))
IF(KKT-1)351,352,353

```

```
351 FK(I)=FFAX
    FJ(I)=FBTX
    GO TO 355
352 FK(I)=FFAX
    FJ(I)=FBTX
    GO TO 355
353 FK(I)=FBTX
    FJ(I)=FBTX
355 C(I)=A(I)*FK(I)
    IF(IC)370,360,370
360 SF(I)= A(I)
    EN(I)= C(I)
    GO TO 390
370 B(I)=0.0
    DO 380 J=1,K
380 B(I)=B(I)+ GYP(J)*SINF(P(J))
    D(I)=B(I)*FK(I)
    SF(I)=SQRTF(A(I)**2+B(I)**2)
    EN(I)=SF(I)*FK(I)
390 GO TO(400,410),NN
400 F(I)= E(I)/FJ(I)
    GO TO 420
410 E(I)= F(I)*FJ(I)
420 DF(I)=SIGNF(F(I),SF(I))-SF(I)
    SDF=SDF+ABSF(DF(I))
    SFO=SFO+F(I)
    DE(I)=SIGNF(E(I),EN(I))-EN(I)
    IF(F(I))422,423,422
422 SDF0=SDF0+ABSF(DF(I))
423 IF(IC)425,424,425
424 F(I)=SIGNF(F(I),SF(I))
    E(I)=SIGNF(E(I),EN(I))
    GO TO 450
425 IF(IP)440,430,440
430 TAR=E(I)/ABSF(EN(I))
    CO(I)=C(I)*TAR
    DO(I)=D(I)*TAR
    DC(I)=SIGNF(CO(I),C(I))-C(I)
    DD(I)=SIGNF(DO(I),D(I))-D(I)
    GO TO 450
440 RAT=F(I)/ABSF(SF(I))
    AO(I)=A(I)*RAT
    BO(I)=B(I)*RAT
    DA(I)=SIGNF(AO(I),A(I))-A(I)
    DB(I)=SIGNF(BO(I),B(I))-B(I)
450 CONTINUE
460 IF(IOUT)62,465,62
62 IF(IC)66,65,66
65 WRITE OUTPUT TAPE LT,1200,(H(1,I),H(2,I),H(3,I),F(I),E(I),DF(I),
    1DE(I),I=1,N2)
    GO TO 465
66 IF(IP)68,67,68
67 WRITE OUTPUT TAPE LT,1200,(H(1,I),H(2,I),H(3,I),CO(I),DO(I),DC(I),
    1DD(I),I=1,N2)
    GO TO 465
68 WRITE OUTPUT TAPE LT,1200,(H(1,I),H(2,I),H(3,I),AO(I),BO(I),DA(I),
    1DB(I),I=1,N2)
465 CALL OUTSFC(N2,IC,H,F,SF,A,B,E,EN,C,D,DF,NO,AO,BO,DA,DB,CO,DO,DC,
    1 DD,NAME,IP,DE)
    NO=NO+1
    GO TO 245
```

```

470 REWIND KT
    END FILE LT
    REWIND LT
    R=SDF/SFO
    R1=SDFO/SFO
    WRITE OUTPUT TAPE JT,1900,SFO,SDF,SDFO,R,R1
    READ INPUT TAPE IT,2100,IIJJ
    IF(IIJJ)50,480,50
480 PRINT 2000
    CALL EXIT
1000 FORMAT(12A6)
1100 FORMAT(7I10)
1200 FORMAT(7F10.4)
1300 FORMAT(3E10.4,2I10)
1400 FORMAT(5F8.4)
1500 FORMAT(4E10.4,I10)
1600 FORMAT(6E10.4)
1700 FORMAT(3F4.0,F20.5)
1800 FORMAT(4F9.2,9X,F9.2,F9.6)
1900 FORMAT(1H1,9X,12HSIGMA F(O) =,F15.3,8X,15HSIGMA /DEL F/ =,F10.3,8X
    1,15HSIGMA /DEL F/ =,F10.3,1H*///24X,2(3HR =,F9.5,9X),1H*///20X,64
    2H* THESE QUANTITIES DO NOT INCLUDE CONTRIBUTIONS FROM REFLECTIONS/
    320X,33H OBSERVED WITH ZERO INTENSITIES.)
2000 FORMAT(1H1,10X,21HTHIS JOB IS FINISHED./////////)
2100 FORMAT(I7)
    END

```

Subroutine OUTSFC

```

SUBROUTINE OUTSFC(N2,IC,H,F,SF,A,B,E,EN,C,D,DF,NO,AO,BO,DA,DB,CO,
1 DO,DC,DD,NAME,IP,DE)
    DIMENSION H(3,128),F(128),A(128),B(128),E(128),EN(128),C(128),D(12
1 8),
    DF(128),AO(128),BO(128),DA(128),DB(128),CO(128),DO(128)
2 ,DC(128),DD(128),NAME(12),DE(128),M(3,128),SF(128)
    JT=6
    N4=6*NO
    DO 50 I=1,N2
    DO 50 J=1,3
50 M(J,I)=INTG(H(J,I))
    IF(128-N2)110,110,100
100 N1=N2/22
    N3=N2-N1*22
    IF(N1)165,165,120
110 N1=5
    N3=18
120 DO 160 I=1,N1
    N4=N4+1
    J1=(I-1)*22+1
    J2=J1+21
    WRITE OUTPUT TAPE JT,1000,(NAME(K),K=1,12),N4
    IF(IC)125,150,125
125 IF(IP)140,130,140
130 WRITE OUTPUT TAPE JT,1200
    WRITE OUTPUT TAPE JT,1400,(M(1,J),M(2,J),M(3,J),F(J),SF(J),E(J),
1 EN(J),CO(J),DO(J),C(J),D(J),DC(J),DD(J),DE(J),J=J1,J2)
    GO TO 160
140 WRITE OUTPUT TAPE JT,1100
    WRITE OUTPUT TAPE JT,1300,(M(1,J),M(2,J),M(3,J),E(J),EN(J),F(J),
1 SF(J),AO(J),BO(J),A(J),B(J),DA(J),DB(J),DF(J),J=J1,J2)
    GO TO 160
150 WRITE OUTPUT TAPE JT,1500
    WRITE OUTPUT TAPE JT,1600,(M(1,J),M(2,J),M(3,J),F(J),SF(J),E(J),
1 EN(J),DF(J),DE(J),J=J1,J2)
160 CONTINUE

```

```

165 IF(N3)210,210,170
170 J1=N1*22+1
    N4=N4+1
    WRITE OUTPUT TAPE JT,1000,(NAME(K),K=1,12),N4
    IF(IC)175,200,175
175 IF(IP)190,180,190
180 WRITE OUTPUT TAPE JT,1200
    WRITE OUTPUT TAPE JT,1400,(M(1,J),M(2,J),M(3,J),F(J),SF(J),E(J),
    1 EN(J),CO(J),DO(J),C(J),D(J),DC(J),DD(J),DE(J),J=J1,N2)
    GO TO 210
190 WRITE OUTPUT TAPE JT,1100
    WRITE OUTPUT TAPE JT,1300,(M(1,J),M(2,J),M(3,J),E(J),EN(J),F(J),
    1 SF(J),AO(J),BO(J),A(J),B(J),DA(J),DB(J),DF(J),J=J1,N2)
    GO TO 210
200 WRITE OUTPUT TAPE JT,1500
    WRITE OUTPUT TAPE JT,1600,(M(1,J),M(2,J),M(3,J),F(J),SF(J),E(J),
    1 EN(J),DF(J),DE(J),J=J1,N2)
210 RETURN
1000 FORMAT(1H1,12A6//100X,4HPAGE,I4/)
1100 FORMAT(12OH0 H K L E(O) E(C) F(O) F(C)
    1A(O) B(O) A(C) B(C) DEL A DEL B DEL F /
    2)
1200 FORMAT(12OH0 H K L F(O) F(C) E(O) E(C)
    1C(O) D(O) C(C) D(C) DEL C DEL D DEL E /
    2)
1300 FORMAT(1H0,3I3,2F10.4,9F10.3)
1400 FORMAT(1H0,3I3,2F10.3,9F10.4)
1500 FORMAT(86H0 H K L F(O) F(C) E(O)
    1 E(C) DEL F DEL E /)
1600 FORMAT(2H0 ,3I4,2F12.3,2F12.4,F12.3,F12.4)
    END

```

Interatomic Distance and Angle

```

CBNDANG  BOND DISTANCE AND ANGLE PROGRAM
    DIMENSION NAME(12),N(500),X(500),Y(500),
    1Z(500),NS1(200),NS2(200),SV(200),N1(2900),N2(2900),
    2D(2900),E1(2900),E2(2900),E3(2900),ANG(2000),
    3M1(2000),M2(2000),M3(2000)
    COMMON N1,N2,D,E1,E2,E3
    EQUIVALENCE(M1,N),(M2,X),(M3,Y),(ANG,Z)
    COSDF(X)=COSF(3.14159*X/180.)
    ITP=5
    JTP=6
    KTP=9
    REWIND KTP
    READ INPUT TAPE ITP,1000,(NAME(II),II=1,12)
    READ INPUT TAPE ITP,1200,A,B,C,AL,BE,GA
    READ INPUT TAPE ITP,1500,BMX,AMX
    READ INPUT TAPE ITP,1100,XINC,YINC,ZINC
    BMX=BMX**2
    CAL=COSDF(AL)
    CBE=COSDF(BE)
    CGA=COSDF(GA)
    I=0
    IF(XINC+YINC+ZINC)107,100,107
100 I=I+1
    READ INPUT TAPE ITP,1100,X(I),Y(I),Z(I)
105 N(I)=I-1
    IF(ABSF(X(I))+ABSF(Y(I))+ABSF(Z(I)))110,110,100
107 CALL INTERP(I,N,X,Y,Z,ITP,XINC,YINC,ZINC,A,B,C,CAL,CBE,CGA)

```

110	JJ=0	02693
	KK=0	02693
	I=I-1	
	I1=I	02693
115	READ INPUT TAPE ITP,1200,T1,T2,T3,U1,U2,U3	
	IF(ABSF(T1)+ABSF(T2)+ABSF(T3)+ABSF(U1)+ABSF(U2)+ABSF(U3))140,140,	02691
1	120	02691
120	DO 130 J=1,I	02693
	I1=I1+1	02693
	N(I1)=I1-1	02693
	X(I1)=T1+U1*X(J)	02691
	Y(I1)=T2+U2*Y(J)	02691
130	Z(I1)=T3+U3*Z(J)	02691
	GO TO 115	02693
140	DO 145 J=1,I1	
	IF(XMODF(J,50)-1)142,141,142	
141	WRITE OUTPUT TAPE JTP,1800,(NAME(K),K=1,12)	
142	L1=N(J)/I	
	L2=N(J)-L1*I+1	
145	WRITE OUTPUT TAPE JTP,1700,L1,L2,X(J),Y(J),Z(J)	
150	DO190 J=1,I1	02691
	I2=J+1	02693
	DO 190 K=I2,I1	02693
	IF(N(J)/I-N(K)/I)157,155,157	
155	IF(N(J)/I)190,157,190	
157	V1=X(J)-X(K)	02693
	V2=Y(J)-Y(K)	02693
	V3=Z(J)-Z(K)	02693
	VEC=ABSF(DOTPRD(V1,V2,V3,V1,V2,V3,A,B,	02693
	1C,CAL,CBE,CGA))	02693
	IF(VEC- BMX)160,160,190	
160	IF(VEC-.49)170,180,180	02693
170	JJ=JJ+1	02693
	NS1(JJ)=N(J)	02693
	NS2(JJ)=N(K)	02693
	SV(JJ)=SQRTF(VF)	02693
	GO TO 190	02693
180	KK=KK+1	02693
	N1(KK)=N(J)	02693
	N2(KK)=N(K)	02693
	D(KK)=SQRTF(VEC)	02693
	E1(KK)=V1	02693
	E2(KK)=V2	02693
	E3(KK)=V3	02693
	IF(2900-KK)330,330,190	
190	CONTINUE	02693
	LL=0	02693
	NM=0	
	DO 322 J=1,KK	
	IF(D(J)-AMX)195,195,322	
195	KK1=J+1	02693
	DO 320 K=KK1,KK	02693
	IF(D(K)-AMX)200,200,320	
200	IF(N1(J)-N1(K))220,210,220	02693
210	M=1.	02693
	ZD=1.0	
	GO TO 276	
220	IF(N1(J)-N2(K))240,230,240	02693
230	M=2	02693
	ZD=-1.0	
	GO TO 276	
240	IF(N2(J)-N1(K))260,250,260	02693
250	M=3	02693
	ZD=-1.0	
	GO TO 276	

```

260 IF(N2(J)-N2(K))320,270,320
270 M=4
    ZD=1.0
276 LL=LL+1
    COANG=2D* DOTPRD(E1(J),E2(J),E3(J),E1(K),
    1E2(K),E3(K),A,B,C,CAL,CBE,CGA)/(D(J)*D(K))
    ANGCO=1.0-ABSF(COANG)
    IF(ANGCO)277,277,277
2277 WRITE OUTPUT TAPE JTP, 1600,N1(J),N2(J),N1(K),N2(K),ANGCO
    COANG=1.0
277 ANG(LL)=90.00-57.295780*ARCSIN(COANG)
278 GO TO(280,290,300,310),M
280 M1(LL)=N2(J)
    M2(LL)=N1(J)
    M3(LL)=N2(K)
    GO TO 315
290 M1(LL)=N2(J)
    M2(LL)=N1(J)
    M3(LL)=N1(K)
    GO TO 315
300 M1(LL)=N1(J)
    M2(LL)=N2(J)
    M3(LL)=N2(K)
    GO TO 315
310 M1(LL)=N1(J)
    M2(LL)=N2(J)
    M3(LL)=N1(K)
315 IF(2000-LL)360,360,320
360 WRITE TAPE KTP,(M1(MN),M2(MN),M3(MN),ANG(MN),MN=1,2000)
    LL=0
    NM=NM+1
320 CONTINUE
322 CONTINUE
    IF(LL)323,323,1323
1323 WRITE TAPE KTP,(M1(MN),M2(MN),M3(MN),ANG(MN),MN=1,LL)
323 CALL OUTBND(I,JJ,KK,LL,NS1,NS2,SV,
    1N1,N2,N3,D,E1,E2,E3,ANG,M1,M2,M3,
    2NAME,JTP,NM,KTP)
325 CALL EXIT
330 BMX=(SQRTF(BMX)-.25)**2
    IF(BMX-4.0)335,335,370
335 IF(I1/I-1)340,340,350
340 WRITE OUTPUT TAPE JTP,1300
    GO TO 325
350 WRITE OUTPUT TAPE JTP,1400
    I1=I
370 KK=0
    JJ=0
    GO TO 150
1000 FORMAT(12A6)
1100 FORMAT(3E10.4)
1200 FORMAT(6E10.4)
1300 FORMAT(1H1,8X,71HMORE THAN 2900 2 A. OR LESS BOND DISTANCES HAV
    1E BEEN FOUND AMONG THE///9X,71HUNTRANSLATED SET OF ATOMS. THIS PRO2691
    2BLEM IS TOO LARGE. NO RESULTS WILL///9X,10HBE ISSUED.) 02691
1400 FORMAT(1H1,8X,71HMORE THAN 2900 2 A. OR LESS BOND DISTANCES HAV
    1E BEEN FOUND. VECTORS///9X,58HWILL BE CONSIDERED FOR THE UNTRANSL02691
    2ATED SET OF ATOMS ONLY.) 02691
1500 FORMAT(2E10.4)
1600 FORMAT(10X,4I4,E12.5)
1700 FORMAT(11X,I2,1H-,I2,3F10.5)
1800 FORMAT(1H1,10X,12A6///12X,4HATOM,6X,1HX,9X,1HY,9X,1HZ/)
    END

```

Subroutine INTERP

```

SUBROUTINE INTERP(I,N,X,Y,Z,ITP,XINC,YINC,ZINC,A,B,C,CAL,CBE,CGA)
DIMENSION X(500),Y(500),Z(500),N(500),P(7),E(3,3),G(3,4),H(3)
E(1,1)=1.0
E(1,2)=B*CGA*YINC/(XINC*A)
E(1,3)=C*CBE*ZINC/(XINC*A)
E(2,1)=A*CGA*XINC/(YINC*B)
E(2,2)=1.0
E(2,3)=C*CAL*ZINC/(YINC*B)
E(3,1)=A*CBE*XINC/(ZINC*C)
E(3,2)=B*CAL*YINC/(ZINC*C)
E(3,3)=1.0
100 I=I+1
    READ INPUT TAPE ITP,1000,X(I),Y(I),Z(I)
    IF(ABS(X(I))+ABS(Y(I))+ABS(Z(I)))130,130,110
110 READ INPUT TAPE ITP,1000,{P(J),J=1,7}
    L=0
    DO 120 J=1,3
        L=L+2
        G(J,4)=LOGF(P(1)/P(L))/LOGF(P(1)/P(L+1))
        G(J,4)=(G(J,4)-1.0)/(2.0*(G(J,4)+1.0))
    DO 120 K=1,3
120 G(J,K)=E(J,K)
        CALL MATS(G,H,3,1)
        X(I)=X(I)+H(1)*XINC
        Y(I)=Y(I)+H(2)*YINC
        Z(I)=Z(I)+H(3)*ZINC
        N(I)=I-1
        GO TO 100
130 RETURN
1000 FORMAT(7E10.4)
END

```

Subroutine OUTBND

```

SUBROUTINE OUTBND(I,JJ,KK,LM,NS1,
INS2,SV,N1,N2,N3,D,E1,E2,E3,ANG,M1,
2M2,M3,NAME,JTP,NM,KTP)
C OUTPUT S.R. FOR BOND DISTANCE AND ANGLE PROGRAM
DIMENSION NS1(200),NS2(200),SV(200),
1N1(3000),N2(3000),D(3000),E1(3000),E2(3000),
2E3(3000),ANG(2000),M1(2000),M2(2000),
3M3(2000),NAME(12)
REWIND KJP
IJK=1
50 I1=KK/44
IF(KK)85,85,75
75 IF(I1)120,120,100
85 WRITE OUTPUT TAPE JTP,8000
GO TO 250
100 DO 110 J=1,I1
    WRITE OUTPUT TAPE JTP,1000,(NAME(K),K=1,12),J
    I3=J*44-22
    I2=I3-21
    DO 110 L=I2,I3
        L1=N1(L)/I
        M11=N1(L)-L1*I+1
        L2=N2(L)/I
        M22=N2(L)-L2*I+1
        LL1=N1(L+22)/I
        MM1=N1(L+22)-LL1*I +1
        LL2=N2(L+22)/I
        MM2=N2(L+22)-LL2*I +1

```

```
110 WRITE OUTPUT TAPE JTP,2000,L1,M11,L2,M22,D(L),LL1,MM1,LL2,MM2,  
    ID(L+22)  
    IF(KK-I1*44)170,170,120  
120 I4=I1*44+1  
    I5=KK-I4-21  
    IF(I5)121,122,122  
121 I5=0  
122 I6=I4+I5-1  
    I7=I6+1  
    I8=KK-I5  
    I9=I1+1  
    WRITE OUTPUT TAPE JTP,1000,(NAME(K),K=1,12),I9  
    IF(I5)150,150,130  
130 DO140 L=I4,I6  
    L1=N1(L)/I  
    M11=N1(L)-L1*I +1  
    L2=N2(L)/I  
    M22=N2(L)-L2*I+1  
    LL1=N1(L+22)/I  
    MM1=N1(L+22)-LL1*I +1  
    LL2=N2(L+22)/I  
    MM2=N2(L+22)-LL2*I +1  
140 WRITE OUTPUT TAPE JTP,2000,L1,M11,L2,M22,D(L),LL1,MM1,  
    LL2,MM2,D(L+22)  
150 DO 160 L=I7,I8  
    L1=N1(L)/I  
    M11=N1(L)-L1*I +1  
    L2=N2(L)/I  
    M22=N2(L)-L2*I +1  
160 WRITE OUTPUT TAPE JTP,3000,L1,M11,L2,M22,D(L)  
170 GO TO(175,270),IJK  
175 IF(NM)250,1176,1175  
1175 LL=2000  
    GO TO 1177  
1176 LL=LM  
    IF(LL)250,250,1177  
1177 NM=NM-1  
    READ TAPE KTP,(M1(MN),M2(MN),M3(MN),ANG(MN),MN=1,LL)  
    J1=LL/44  
176 IF(J1)200,200,180  
180 DO 190 K=1,J1  
    WRITE OUTPUT TAPE JTP,4000,(NAME(L),L=1,12),K  
    J3=K*44-22  
    J2=J3-21  
    DO 190 N=J2,J3  
    L1=M1(N)/I  
    L2=M1(N)-L1*I +1  
    L3=M2(N)/I  
    L4=M2(N)-L3*I +1  
    L5=M3(N)/I  
    L6=M3(N)-L5*I +1  
    ML1=M1(N+22)/I  
    ML2=M1(N+22)-ML1*I +1  
    ML3=M2(N+22)/I  
    ML4=M2(N+22)-ML3*I +1  
    ML5=M3(N+22)/I  
    ML6=M3(N+22)-ML5*I +1  
190 WRITE OUTPUT TAPE JTP,5000,L1,L2,L3,L4,L5,L6,ANG(N),  
    ML1,ML2,ML3,ML4,ML5,ML6,ANG(N+22)  
    IF(LL-J1*44)175,175,200  
200 J4=J1*44+1  
    J5=LL-J4-21  
    IF(J5)201,207,202  
201 J5=0
```

```

202 J6=J4+J5-1
    J7=J6+1
    J8=LL-J5
    J9=J1+1
    WRITE OUTPUT TAPE JTP,4000,(NAME(L),L=1,12),J9
    IF(J5)230,230,210
210 DO 220 N=J4,J6
    L1=M1(N)/I
    L2=M1(N)-L1*I +1
    L3=M2(N)/I
    L4=M2(N)-L3*I +1
    L5=M3(N)/I
    L6=M3(N)-L5*I +1
    ML1=M1(N+22)/I
    ML2=M1(N+22)-ML1*I +1
    ML3=M2(N+22)/I
    ML4=M2(N+22)-ML3*I +1
    ML5=M3(N+22)/I
    ML6=M3(N+22)-ML5*I +1
220 WRITE OUTPUT TAPE JTP,5000,L1,L2,L3,L4,L5,L6,ANG(N),
    1ML1,ML2,ML3,ML4,ML5,ML6,ANG(N+22)
230 DO 240 N=J7,J8
    L1=M1(N)/I
    L2=M1(N)-L1*I +1
    L3=M2(N)/I
    L4=M2(N)-L3*I +1
    L5=M3(N)/I
    L6=M3(N)-L5*I +1
240 WRITE OUTPUT TAPE JTP,6000,L1,L2,L3,L4,L5,L6,ANG(N)
    GO TO 175
250 IF(JJ)265,265,255
255 KK=JJ
    DO 260 J=1,JJ
    N1(J)=NS1(J)
    N2(J)=NS2(J)
260 D(J)=SV(J)
    WRITE OUTPUT TAPE JTP,7000
    IJK=2
    GO TO 50
265 WRITE OUTPUT TAPE JTP,9000
270 REWIND KTP
    RETURN
1000 FORMAT(1H1,6X,30HBOND DISTANCES(ANGSTROMS) FOR ,12A6,7H PAGE ,12/02693
    1//19X,24HATOM ATOM DISTANCE,27X,24HATOM ATOM DISTANCE)02693
2000 FORMAT(1H0,17X,I2,1H-,I2,3X,I2,1H-,I2,4X,F8.5,26X,I2,1H-,I2,3X,I2,
    11H-,I2,4X,F8.5)
3000 FORMAT(1H0,17X,I2,1H-,I2,3X,I2,1H-,I2,4X,F8.5)
4000 FORMAT(1H1,6X,30HBOND ANGLES (DEGREES) FOR ,12A6,7H PAGE ,12/02693
    1//2(11X,32HATOM ATOM ATOM ANGLE,11X)) 02693
5000 FORMAT(1H0,2(9X,I2,1H-,I2,3X,I2,1H-,I2,3X,I2,1H-,I2,4X,F9.4,10X)) 02693
6000 FORMAT(1H0,9X,I2,1H-,I2,3X,I2,1H-,I2,3X,I2,1H-,I2,4X,F9.4) 02693
7000 FORMAT(1H1,10X,53HTHE FOLLOWING BOND DISTANCES ARE ALL LESS THAN 002693
    1.7 A.//11X,59HBOND ANGLES INVOLVING THESE BONDS HAVE NOT BEEN CALC02693
    1ULATED.) 02693
8000 FORMAT(1H1,10X,64HNO BOND DISTANCES BETWEEN 0.7 AND 4.0 ANGSTROMS 02693
    1HAVE BEEN FOUND.) 02693
9000 FORMAT(1H1,10X,58HNO BOND DISTANCES LESS THAN 0.7 ANGSTROMS HAVE 002693
    1EEN FOUND.) 02693
    END 02693

```

Least-Squares Line and Plane Fitter

```

CLSOPLN  LEAST-SQUARES LINE AND PLANE FITTER
  DIMENSION ZZ(12),G(3,3),ZY(10),X(3,50),WXS(3), WXXS(3,3),WT(50),
  1ADJ(3,3), A(3,3),G1(3,3),XMK(3),B(3,3),YMK(3),VM(3,3),VMI(3,3),
  2VMJ(3,3), C(3,3), DI(50),DEL(50),DEL2(50),XBAR(3),VMK(3,50),
  3P(3),UMK(3,50),BV(3)
  4,Z12(3)
  ITP=5
  JTP=6
  Z34=1.0E-12
  MM=10
  READ INPUT TAPE ITP,1000,(ZZ(I),I=1,12)
  READ INPUT TAPE ITP,1003,A1,A2,A3,AL,BE,GA
  WRITE OUTPUT TAPE JTP,1020,(ZZ(I),I=1,12)
  CALL RECIP(A1,A2,A3,AL,BE,GA,B1,B2,B3,COSA,COSB,COSC)
C  SET UP MATRIX G
  G(1,1)=B1**2
  G(1,2)=B1*B2*COSC
  G(1,3)=B1*B3*COSB
  G(2,1)=G(1,2)
  G(2,2)=B2**2
  G(2,3)=B2*B3*COSA
  G(3,1)=G(1,3)
  G(3,2)=G(2,3)
  G(3,3)=B3**2
  G1(1,1)=A1**2
  G1(1,2)=A1* A2*COSF(GA*3.14159/180.0)
  G1(1,3)=A1*A3*COSF(BE*3.14159/180.0)
  G1(2,1)=G1(1,2)
  G1(2,2)=A2**2
  G1(2,3)=A2*A3*COSF(AL*3.14159/180.0)
  G1(3,1)=G1(1,3)
  G1(3,2)=G1(2,3)
  G1(3,3)=A3**2
  READ INPUT TAPE ITP,1002,N,NO,L,(ZY(I),I=1,10)
  10 DO 20 J=1,N
  20 READ INPUT TAPE ITP,1003, X(1,J),X(2,J),X(3,J),WT(J)
C  COORDINATES OF CENTROID
  DO 25 I=1,N
  IF(WT(I))25,22,25
  22 WT(I)=1.0
  25 CONTINUE
  DO 30 I=1,3
  WXS(I)=0.
  DO 30 J=1,N
  30 WXS(I)=WXS(I)+WT(J)*X(I,J)
  WS=0.
  DO 40 I=1,N
  40 WS=WS+WT(I)
  DO 50 I=1,3
  50 XBAR(I)=WXS(I)/WS
C  SET UP MATRIX A
  DO 60 I=1,3
  DO 60 J=1,3
  WXXS(I,J)=0.
  DO 55 K=1,N
  55 WXXS(I,J)=WXXS(I,J)+WT(K)*X(I,K)*X(J,K)
  60 A(I,J)=WXXS(I,J)-XBAR(I)*XBAR(J)*WS
  IF(L)65,65,61
C  EVALUATE MATRIX B FOR LINE
  61 CALL MTXMUL(3,3,3,G1,A,B)
  GO TO 75
C  SET UP ADJOINT OF A

```

```

65 ADJ(1,1)=A(2,2)*A(3,3)-A(2,3)*A(3,2)
   ADJ(2,1)=A(3,1)*A(2,3)-A(2,1)*A(3,3)
   ADJ(3,1)=A(2,1)*A(3,2)-A(3,1)*A(2,2)
   ADJ(1,2)=A(3,2)*A(1,3)-A(1,2)*A(3,3)
   ADJ(2,2)=A(1,1)*A(3,3)-A(3,1)*A(1,3)
   ADJ(3,2)=A(3,1)*A(1,2)-A(1,1)*A(3,2)
   ADJ(1,3)=A(1,2)*A(2,3)-A(1,3)*A(2,2)
   ADJ(2,3)=A(2,1)*A(1,3)-A(1,1)*A(2,3)
   ADJ(3,3)=A(1,1)*A(2,2)-A(2,1)*A(1,2)
C   EVALUATE DETERMINANT OF A
   CALL MTXMUL(3,3,3,ADJ,A,C)
   DETNA=0.
   DO 70 I=1,3
70  DETNA=DETNA+C(I,I)
   DETNA=DETNA/3.0
C   EVALUATE MATRIX B FOR PLANE
   CALL MTXMUL(3,3,3,ADJ,G,B)
75  DO 80 I=1,3
80  BV(I)=0.
   DO 90 I=1,3
   DO 90 J=1,3
90  BV(I)=BV(I)+B(J,I)**2
   BIGBV=BV(1)
   KK=1
   DO 110 I=2,3
   IF(BV(I)-BIGBV)110,110,100
100 KK=I
   BIGBV=BV(I)
110 CONTINUE
   VMN=SQRTF(BIGBV)
   DO 120 I=1,3
120  VM(I,1)=B(I,KK)/VMN
   NNN=0
C   EVALUATE VM BY ITERATION
125 CALL MTXMUL(3,3,1,B,VM,VM)
   NNN=NNN+1
   IF(MM-NNN)3,3,4
3   MM=MM+10
   Z34=Z34*100.
4   IF(40-NNN)1,1,2
1   WRITE OUTPUT TAPE JTP,1021,NO
   GO TO 235
2   IJ=1
   VMN=SQRTF(VM(1,1)**2+VM(2,1)**2+VM(3,1)**2)
   DO 126 I=1,3
126  VMI(I,1)=VM(I,1)/VMN
   DO 140 I=1,3
   Z12(I)=VM(I,1)/VMI(I,1)-1.0
   IF(ABSF((VM(I,1)/VMI(I,1))-1.0)-Z34)140,140,130
130 IJ=2
   GO TO 150
140 CONTINUE
150 DO 160 I=1,3
   VM(I,1)=VMI(I,1)
160 VM(I,I)=VMI(I,1)
   GO TO(170,125),IJ
C   NORMALIZE VECTOR VM
170 CALL MTXMUL(1,3,3,VM,G,VMI)
   CALL MTXMUL(1,3,1,VMI,VM,VMJ)
   ORM=SQRTF(VMJ(1,1))
   DO 180 I=1,3
180  VM(I,1)=VM(I,1)/ORM
   IF(L)185,185,500

```

```

500 DO 510 I=1,3
    P(I)=0.0
    DO 510 J=1,3
510 P(I)=VM(J,1)*G(J,I)+P(I)
    DL2=0.0
    DO 530 I=1,N
    DEL2(I)=0.0
    DO 520 J=1,3
    DO 520 K=1,3
520 DEL2(I)=DEL2(I)+(XBAR(J)-X(J,I))*(XBAR(K)-X(K,I))*(G1(J,K)-
    1VM(J,1)*VM(K,1))
530 DL2=DL2+DEL2(I)
    WRITE OUTPUT TAPE JTP,1025,NO,N,(ZY(I),I=1,10),XBAR(1),P(1),XBAR(2
    1),P(2),XBAR(3),P(3),(I,WT(I),DEL2(I),I=1,N)
    SATN=N-2
    IF(SATN)550,550,540
540 STND=SQRTF(DL2/SATN)
    WRITE OUTPUT TAPE JTP,1032,STND
550 DO 560 I=1,3
560 UMK(I,NO)=VM(I,1)
    GO TO 235
C
EVALUATE EIGEN VALUE FOR PLANE
185 CALL MTXMUL(3,3,1,8,VM,VMJ)
    DNEG=0.
    DO 190 I=1,3
190 DNEG=DNEG+VMJ(I,1)/VM(I,1)
    EIGEN=3.0*DETNA/DNEG
C
EVALUATE DI(PLANE TO ORIGIN DISTANCE)
    D=0.
    DO 200 I=1,3
200 D=D+VM(I,1)*XBAR(I)
C
EVALUATE D,DELTA D,(DELTA D)**2 FOR EACH POINT
    DL2=0.
    DO 220 I=1,N
    DI(I)=0.
    DO 210 J=1,3
210 DI(I)=DI(I)+VM(J,1)*X(J,I)
    DEL(I)=DI(I)-D
    DEL2(I)=DEL(I)**2
220 DL2=DL2+WT(I)*DEL2(I)
    DO 230 I=1,3
230 VMK(I,NO)=VM(I,1)
C
OUTPUT ONE
    WRITE OUTPUT TAPE JTP,1005,NO,N,(ZY(I),I=1,10),VM(1,1),VM(2,1),VM(
    13,1),D,XBAR(1),XBAR(2),XBAR(3),(I,WT(I),DI(I),DEL(I),DEL2(I),I=1,N
    2)
    WRITE OUTPUT TAPE JTP,1012,DL2,EIGEN
    AVED=0.
    DO 231 I=1,N
231 AVED=AVED+DI(I)
    ATN=N
    AVED=AVED/ATN
    SATN=N-3
    IF(SATN)232,232,233
232 WRITE OUTPUT TAPE JTP,1018,AVED
    GO TO 235
233 STND=SQRTF(DL2/SATN)
    WRITE OUTPUT TAPE JTP,1019,AVED,STND
C
INTERROGATE NEXT CARD
235 READ INPUT TAPE ITP,1002,N,NO,L,(ZY(I),I=1,10)
    IF(N)240,240,10
240 READ INPUT TAPE ITP,1013,N,L1,NO,L2
    IF(N)270,270,245
C
OUTPUT TWO

```

```

245 WRITE OUTPUT TAPE JTP,1014
    GO TO 260
250 READ INPUT TAPE ITP,1013,N,L1,NO,L2
    IF(N)270,270,260
260 IF(L1)275,275,300
275 IF(L2)280,280,340
280 DO 290 I=1,3
    XMK(I)=VMK(I,N)
290 YMK(I)=VMK(I,NO)
    LLL=1
    GO TO 350
300 IF(L2)310,310,330
310 DO 320 I=1,3
    XMK(I)=UMK(I,N)
320 YMK(I)=VMK(I,NO)
    LLL=2
    GO TO 350
330 DO 335 I=1,3
    XMK(I)=UMK(I,N)
335 YMK(I)=UMK(I,NO)
    LLL=3
    GO TO 350
340 DO 345 I=1,3
    XMK(I)=VMK(I,N)
345 YMK(I)=UMK(I,NO)
    LLL=4
350 DTPD=DOTPRD(XMK(1),XMK(2),XMK(3),YMK(1),YMK(2),YMK(3),B1,B2,B3,
    1COSA,COSB,COSC)
    WRITE OUTPUT TAPE JTP,1006
    GO TO(360,370,380,390),LLL
360 WRITE OUTPUT TAPE JTP,1016,N,NO,DTPD
    GO TO 250
370 WRITE OUTPUT TAPE JTP,1036,N,NO,DTPD
    GO TO 250
380 WRITE OUTPUT TAPE JTP,1035,N,NO,DTPD
    GO TO 250
390 WRITE OUTPUT TAPE JTP,1037,N,NO,DTPD
    GO TO 250
270 PRINT 1017
    CALL EXIT
1000 FORMAT(12A6)
1002 FORMAT(2I3,16,10A6)
1003 FORMAT(6E10.4)
1005 FORMAT(1H1,7X,12HPLANE NUMBER14,I10,6H ATOMS,7X,10A6////12X,90HEQU
    2ATION OF PLANE IS M1*X + M2*Y + M3*Z = D WHERE D IS THE ORI
    3GIN TO PLANE DISTANCE/1H0,20X,3HM1=E12.5,7H M2=E12.5,7H M3=E
    412.5,6H D=E12.5////12X,32HCOORDINATES OF CENTROID X =E12.5
    5,5X,3HY =E12.5,5X,3HZ =12.5////15X,72HATOM NUMBER WEIGHT
    6 D DELTA D (DELTA D)**2/(1H0,17X,13,6X,E10.4,
    75X,E12.5,5X,E12.5,4X,E12.5))
1006 FORMAT(1H0)
1012 FORMAT(////15X,21HSUM WT*(DELTA D)**2 =E12.5,20X,13HEIGEN VALUE =E
    112.5////)
1013 FORMAT(4I3)
1014 FORMAT(1H1,29X,40HDIHEDRAL ANGLES BETWEEN PLANES AND LINES////29X,
    134HANGLE BETWEEN PLANES P1 AND P2 = A/1H0,28X,33HANGLE BETWEEN LIN
    2ES L1 AND L2 = B/1H0,28X,36HANGLE BETWEEN LINE L AND PLANE P = C)
1016 FORMAT(1H0,28X,4HP1 =I3,10H P2 =I3,17H COSINE(A) =E12.5)
1017 FORMAT(1H1,9X,13HJOB FINISHED.////)
1018 FORMAT(1H0,14X,11HAVERAGE D =E12.5)
1019 FORMAT(1H0,14X,11HAVERAGE D =E12.5,15X,20HSTANDARD DEVIATION =E12.
    15)
1020 FORMAT(1H1,35X,12A6)
1021 FORMAT(1H1,71H A SATISFACTORY LEAST SQUARES FIT CANNOT BE FOUN
    1D FOR SET NUMBER I3,38H. EXAMINE INPUT PARAMETERS FOR ERROR.)

```

```

1025 FORMAT(1H1,8X,11HLINE NUMBERI4,I10,6H ATOMS,7X,10A6////12X,40HTHE
1PARAMETRIC EQUATIONS OF THE LINE ARE//37X,4HX = E12.5,5H + E12.5
2,4H * T//37X,4HY = E12.5,5H + E12.5,4H * T//37X,4HZ = E12.5,5H
3+ E12.5,4H * T////12X,46HTHE SQUARE DEVIATIONS FROM THE LINE, D**
42, ARE// 22X,11HATOM NUMBER,11X,6HWEIGHT,14X,4HD**2/(1H0,24X,I3,1
54X,E12.5,6X,E12.5))
1032 FORMAT(////12X,21HSTANDARD DEVIATION = E12.5)
1035 FORMAT(1H0,28X,4H L1 =I3,10H L2 =I3,17H COSINE(B) =E12.5)
1036 FORMAT(1H0,28X,4H L =I3,10H P =I3,17H SINE(C) =E12.5)
1037 FORMAT(1H0,28X,4H P =I3,10H L =I3,17H SINE(C) =E12.5)
END

```

Point to Peak Distance Calculation

```

ATDIST DETERMINES DISTANCES OF ATOMS FROM ANY POINT IN UNIT CELL
_____
DIMENSION NAME(12),N(500),X(500),Y(500),
1Z(500),NS1(200),NS2(200),SV(200),N1(5000),N2(5000),
2JNT(11),NINT(100,11),AVE(100),EDM(100),NTR(100),D(5000),TT(11),
3TV(11),DIS(11),DISA(100,11)
COSDF(X)=COSDF(3.14159*X/180.)
ITP=5
JTP=6
READ INPUT TAPE ITP,1000,(NAME(I1),I1=1,12)
READ INPUT TAPE ITP,1200,A,B,C,AL,BE,GA
READ INPUT TAPE ITP,1100,BMX
READ INPUT TAPE ITP,1100,XC,YC,ZC
BMX=BMX**2
I=0
100 I=I+1
READ INPUT TAPE ITP,1100,X(I),Y(I),Z(I)
105 N(I)=I-1
IF(ABSF(X(I))+ABSF(Y(I))+ABSF(Z(I)))110,110,100
110 JJ=0
KK=0
I=I-1
I1=1
115 READ INPUT TAPE ITP,1200,T1,T2,T3,U1,U2,U3
IF(ABSF(T1)+ABSF(T2)+ABSF(T3)+ABSF(U1)+ABSF(U2)+ABSF(U3))140,140,
1120
120 DO 130 J=1,I
I1=I1+1
N(I1)=I1-1
X(I1)=T1+U1*X(J)
Y(I1)=T2+U2*Y(J)
130 Z(I1)=T3+U3*Z(J)
GO TO 115
140 CAL=COSDF(AL)
CBE=COSDF(BE)
CGA=COSDF(GA)
150 DO190 J=1,I1
157 V1=X(J)-XC
V2=Y(J)-YC
V3=Z(J)-ZC
VEC=ABSF(DOTPRD(V1,V2,V3,V1,V2,V3,A,B,
16C,CAL,CBE,CGA))
IF(VEC-BMX)180,180,190
180 KK=KK+1
N1(KK)=N(J)
D(KK)=SQRTF(VEC)

```

```

190 CONTINUE
  WRITE OUTPUT TAPE JTP,3300,(NAME(K),K=1,12)
  DO 200 J=1,KK
    L1=N1(J)/I
    M1=N1(J)-L1*I+1
200  WRITE OUTPUT TAPE JTP,2000,L1,M1,D(J)
    READ INPUT TAPE ITP, 1100,XC,YC,ZC
    IF(ABSF(XC)+ABSF(YC)+ABSF(ZC))2190,2190,210
210  KK=0
    GO TO 190
2190 CALL EXIT
1000 FORMAT(12A6)
1100 FORMAT(3E10.4)
1200 FORMAT(6E10.4)
2000 FORMAT(1H0,17X,12,1H-,12,4X,F8.5)
3300 FORMAT(1H1,10X,12A6)
  END(1,1,0,0,0,0,1,0,0,0,0,0,0,0)

```

Form Factors for Busing Least Squares

```

CFORMBUS  FORM FACTORS FOR BUSING LEAST-SQ. REFIN. PROGRAM
  DIMENSION S(33),F(32),NC(25),BLANK(12)
  ITP=5
  JTP=6
  KT=10
  REWIND KT
  READ INPUT TAPE ITP,250, (BLANK(I),I=1,12)
  NC(1)=1
  NC(9)=2
  NC(17)=3
  NC(25)=4
  5 READ INPUT TAPE ITP,
  1 100,A,A1,B,B1,C,ATOM
  IF(A)30,30,10
10 S(1)=1.55
  DO 20 I=1,32
    F(I)=A*EXP(-A1*S(I)**2)+B*EXP(-B1*S(I)**2)+C
20 S(I+1)=S(I)-.05
  WRITE OUTPUT TAPE JTP, 200,ATOM
  WRITE OUTPUT TAPE JTP, 210
  WRITE OUTPUT TAPE JTP, 220
  WRITE OUTPUT TAPE JTP, 210
  1 230,(S(I),F(I),S(I+16),F(I+16),I=1,16)
  WRITE OUTPUT TAPEKT,250,(BLANK(I),I=1,12)
  WRITE OUTPUT TAPEKT,240,(F(I),F(I+1),F(I+2),F(I+3),F(I+4),
1F(I+5),F(I+6),F(I+7),ATOM,NC(I),I=1,25,8)
  GO TO 5
30 END FILE KT
  REWIND KT
  CALL EXIT
100 FORMAT(5F8.5,A6)
200 FORMAT(1H1,9X,A6)
210 FORMAT(1H0)
220 FORMAT(1H0,9X,7HSIN()/L,13X,11HFORM FACTOR,29X,7HSIN()/L,13X,11HFO
1RM FACTOR)
230 FORMAT(1H0,9X,F5.2,15X,F10.6,30X,F5.2,15X,F10.6)
240 FORMAT(7F9.4,F8.4,1HF,A6,I2)
250 FORMAT(12A6)
  END

```

Variance-Covariance and Atomic Parameter Input
for Busing Function and Error

CBUSVAR VARIANCE MATRIX INPUT FOR BUSING ERROR FUNCTION PROGRAM

```
DIMENSION A(18528),B(192)
ITP=5
JTP=6
I5=9
REWIND I5
READ INPUT TAPE ITP,
1 100,N,SF
READ INPUT TAPE ITP,
1 200,A1,A2,A3
DO 10 I=1,N,3
READ INPUT TAPE ITP,
1 200,B(I),B(I+1),B(I+2)
5 B(I)=(B(I)/A1)**2
B(I+1)=(B(I+1)/A2)**2
10 B(I+2)=(B(I+2)/A3)**2
SF=SF**2
M=0
DO 40 I=1,N
DO 40 J=1,N
IF(I-J)20,30,40
20 M=M+1
A(M)=0.0
GO TO 40
30 M=M+1
A(M)=B(I)
40 CONTINUE
MN=M/8
IF(M-8*MN)45,45,41
41 L=M+1
M=MN*8+8
DO 43 I=L,M
43 A(I)=0.0
45 IF(SENSESWITCH 1)46,47
46 MM=1
PUNCH 600
PUNCH 100,N,SF
GO TO 48
47 MM=2
WRITE OUTPUT TAPE I5,100,N,SF
48 K=0
DO 55 I=1,M,8
K=K+1
GO TO(51,50),MM
50 WRITE OUTPUT TAPE I5,300,A(I),A(I+1),A(I+2),A(I+3),A(I+4),A(I+5),
1A(I+6),A(I+7),K
GO TO 55
51 PUNCH 700,A(I),A(I+1),A(I+2),A(I+3),A(I+4),A(I+5),A(I+6),A(I+7)
55 CONTINUE
READ INPUT TAPE ITP,
1 200,(B(I),B(I+1),B(I+2),I=1,N,3)
IJ=N/8
IF(N-IJ*8)80,80,60
60 L=N+1
N=8*IJ+8
DO 70 I=L,N
70 B(I)=0.0
80 GO TO(81,82),MM
81 PUNCH 600
```

```

82 K=0
   DO 95 I=1,N,8
   K=K+1
   GO TO(92,90),MM
90 WRITE OUTPUT TAPE15,500,B(I),B(I+1),B(I+2),B(I+3),B(I+4),B(I+5),
   1B(I+6),B(I+7),K
   GO TO 95
92 PUNCH 800,B(I),B(I+1),B(I+2),B(I+3),B(I+4),B(I+5),B(I+6),B(I+7)
95 CONTINUE
   END FILE15
   REWIND15
   PRINT 400
   CALL EXIT
100 FORMAT(19,E10.5)
200 FORMAT(3E10.5)
300 FORMAT(8E9.4,I4,3H SD)
400 FORMAT(58H      JOB FINISHED.  SAVE TPE A5 FOR OFF-LINE CARD PUNCHIN
   1G,//////////)
500 FORMAT(8F9.6,I4,2H P)
600 FORMAT(72X)
700 FORMAT(8E9.4)
800 FORMAT(8F9.6)
   END

```

Function ARCSIN(X)

```

FUNCTION ARCSIN(X)
10 A=+1.5707963
20 B=-0.21459880
30 C=+0.08897899
40 D=-0.05017430
50 E=+0.03089188
60 F=-0.01708813
70 G=+0.00667009
80 H=-0.00126249
100 YY=X
110 IF(YY) 120,210,120
120 Y=ABSF(YY)
130 IF(Y-1.0) 140,230,280
140 Z=-SQRTF(1.0-Y)
150 Z=A+Z*(A+Y*(B+Y*(C+Y*(D+Y*(E+Y*(F+Y*(G+HY))))))
160 IF(YY) 170,210,190
170 ARCSIN= -Z
180 RETURN
190 ARCSIN=Z
200 RETURN
210 ARCSIN=0.0
220 RETURN
230 IF(YY) 260,210,240
240 ARCSIN=A
250 RETURN
260 ARCSIN=-A
270 RETURN
280 CALL ENDJOB
290 END

```

Subroutine RECIP

```

SUBROUTINE RECIP(AR,BR,CR,ALR,BER,GAR,AA,BB,CC,COSAL,COSBE,COSGA)
  ALG=ALR*3.14159/180.0
  BEG=BER*3.14159/180.0
  GAG=GAR*3.14159/180.0
  COSAR=COSF(ALG)
  COSBR=COSF(BEG)
  COSGR=COSF(GAG)
  VR=AR*BR*CR*SQRTF(1.0-COSAR*COSAR-COSBR*COSBR-COSGR*COSGR+2.*COSAR
1 *COSBR*COSGR)
  SINAR=SINF(ALG)
  SINBR=SINF(BEG)
  SINGR=SINF(GAG)
  AA=BR*CR*SINAR/VR
  BB=AR*CR*SINBR/VR
  CC=AR*BR*SINGR/VR
  COSAL=(COSBR*COSGR-COSAR)/(SINBR*SINGR)
  COSBE=(COSAR*COSGR-COSBR)/(SINAR*SINGR)
  COSGA=(COSAR*COSBR-COSGR)/(SINAR*SINBR)
  RETURN
END

```

Function DOTPROD

```

FUNCTION DOTPRD(U,V,W,X,Y,Z,B1,B2,B3,COSA,COSB,COSC)
  DOTPRD=U*X*B1**2+V*Y*B2**2+W*Z*B3**2+(V*Z+Y*W)*B2*B3*COSA+
1 (U*Z+W*X)*B1*B3*COSB+(U*Y+V*X)*B1*B2*COSC
  RETURN
END

```

Function SCAFAC

```

FUNCTION SCAFAC(A,A1,B,B1,C,S)
  SCAFAC=A*EXPF(-A1*(S**2))+B*EXPF(-B1*(S**2))+C
  RETURN
END

```

Subroutine MTXMUL

```

SUBROUTINE MTXMUL(L,M,N,A,B,C)
  DIMENSION A(3,3),B(3,3),C(3,3)
  DO 10 I=1,L
  DO 10 J=1,N
10 C(I,J)=0.
  DO 20 I=1,L
  DO 20 J=1,N
  DO 20 K=1,M
20 C(I,J)=C(I,J)+A(I,K)*B(K,J)
  RETURN
END

```

Function INTG(A)

```

FUNCTION INTG(A)
  IF(A)20,10,20
10 INTG=0
  GO TO 30
20 INTG=XSIGNF(XINTF(ABSF(A)+.1),XINTF(A))
30 RETURN
END(2,2,2,2,2)

```

UNCLASSIFIED

Naval Research Laboratory. Report 5885.
A COLLECTION OF FORTRAN PROGRAMS FOR
CRYSTAL STRUCTURE ANALYSIS, by H.G. Normant.
71 pp., April 29, 1963.

This report contains descriptions and operating instructions for a collection of general utility programs applicable to various phases of crystal structure analysis with computers. All coding is done in FORTRAN II language for 32K core IBM 704 and 7090 machines; the FORTRAN-language programs are completely listed in the Appendix.

The data input to many of the programs has been designed to be compatible with the output of the

UNCLASSIFIED (over)

1. Mathematical computer programming
2. Crystal structure - Math. anal.
3. X-ray diffraction analysis - Data - Processing

I. FORTRAN
II. Normant, H. G.

UNCLASSIFIED

Naval Research Laboratory. Report 5885.
A COLLECTION OF FORTRAN PROGRAMS FOR
CRYSTAL STRUCTURE ANALYSIS, by H.G. Normant.
71 pp., April 29, 1963.

This report contains descriptions and operating instructions for a collection of general utility programs applicable to various phases of crystal structure analysis with computers. All coding is done in FORTRAN II language for 32K core IBM 704 and 7090 machines; the FORTRAN-language programs are completely listed in the Appendix.

The data input to many of the programs has been designed to be compatible with the output of the

UNCLASSIFIED (over)

1. Mathematical computer programming
 2. Crystal structure - Math. anal.
 3. X-ray diffraction analysis - Data - Processing
- I. FORTRAN
II. Normant, H. G.

UNCLASSIFIED

Naval Research Laboratory. Report 5885.
A COLLECTION OF FORTRAN PROGRAMS FOR
CRYSTAL STRUCTURE ANALYSIS, by H.G. Normant.
71 pp., April 29, 1963.

This report contains descriptions and operating instructions for a collection of general utility programs applicable to various phases of crystal structure analysis with computers. All coding is done in FORTRAN II language for 32K core IBM 704 and 7090 machines; the FORTRAN-language programs are completely listed in the Appendix.

The data input to many of the programs has been designed to be compatible with the output of the

UNCLASSIFIED (over)

1. Mathematical computer programming
 2. Crystal structure - Math. anal.
 3. X-ray diffraction analysis - Data - Processing
- I. FORTRAN
II. Normant, H. G.

UNCLASSIFIED

Naval Research Laboratory. Report 5885.
A COLLECTION OF FORTRAN PROGRAMS FOR
CRYSTAL STRUCTURE ANALYSIS, by H.G. Normant.
71 pp., April 29, 1963.

This report contains descriptions and operating instructions for a collection of general utility programs applicable to various phases of crystal structure analysis with computers. All coding is done in FORTRAN II language for 32K core IBM 704 and 7090 machines; the FORTRAN-language programs are completely listed in the Appendix.

The data input to many of the programs has been designed to be compatible with the output of the

UNCLASSIFIED (over)

1. Mathematical computer programming
 2. Crystal structure - Math. anal.
 3. X-ray diffraction analysis - Data - Processing
- I. FORTRAN
II. Normant, H. G.

UNCLASSIFIED

automatic data reduction program XRDDR (H.G. Norman, "An X-Ray Diffraction Data Reduction Program for the IBM 704 and 7090," NRL Report 5739, Feb. 1962).

In the form presented in this report, the programs are written for use with the IBM 7090 IB Monitor system.

UNCLASSIFIED

UNCLASSIFIED

automatic data reduction program XRDDR (H.G. Norman, "An X-Ray Diffraction Data Reduction Program for the IBM 704 and 7090," NRL Report 5739, Feb. 1962).

In the form presented in this report, the programs are written for use with the IBM 7090 IB Monitor system.

UNCLASSIFIED

UNCLASSIFIED

automatic data reduction program XRDDR (H.G. Norman, "An X-Ray Diffraction Data Reduction Program for the IBM 704 and 7090," NRL Report 5739, Feb. 1962).

In the form presented in this report, the programs are written for use with the IBM 7090 IB Monitor system.

UNCLASSIFIED

UNCLASSIFIED

automatic data reduction program XRDDR (H.G. Norman, "An X-Ray Diffraction Data Reduction Program for the IBM 704 and 7090," NRL Report 5739, Feb. 1962).

In the form presented in this report, the programs are written for use with the IBM 7090 IB Monitor system.

UNCLASSIFIED