**GPTHEORY: A FORTRAN Computer Program
for Determining Molecular Symmetry Properties**

Thomas D. Bouman and Gordon L. Goodman

U of C-ALM-USAEC

ARGONNE NATIONAL LABORATORY, ARGONNE, ILLINOIS

DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency Thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

DISCLAIMER

Portions of this document may be illegible in electronic image products. Images are produced from the best available original document.

The facilities of Argonne National Laboratory are owned by the United States Government. Under the terms of a contract (W-31-109-Eng-38) between the U. S. Atomic Energy Commission, Argonne Universities Association and The University of Chicago, the University employs the staff and operates the Laboratory in accordance with policies and programs formulated, approved and reviewed by the Association.

MEMBERS OF ARGONNE UNIVERSITIES ASSOCIATION

The University of Arizona	Kansas State University	The Ohio State University
Carnegie-Mellon University	The University of Kansas	Ohio University
Case Western Reserve University	Loyola University	The Pennsylvania State University
The University of Chicago	Marquette University	Purdue University
University of Cincinnati	Michigan State University	Saint Louis University
Illinois Institute of Technology	The University of Michigan	Southern Illinois University
University of Illinois	University of Minnesota	The University of Texas at Austin
Indiana University	University of Missouri	Washington University
Iowa State University	Northwestern University	Wayne State University
The University of Iowa	University of Notre Dame	The University of Wisconsin

NOTICE

This report was prepared as an account of work sponsored by the United States Government. Neither the United States nor the United States Atomic Energy Commission, nor any of their employees, nor any of their contractors, subcontractors, or their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness or usefulness of any information, apparatus, product or process disclosed, or represents that its use would not infringe privately-owned rights.

Printed in the United States of America
Available from
National Technical Information Service
U S Department of Commerce
5285 Port Royal Road
Springfield, Virginia 22151
Price Printed Copy \$3.00, Microfiche \$0.95

ARGONNE NATIONAL LABORATORY
9700 South Cass Avenue
Argonne, Illinois 60439

GPTHEORY: A FORTRAN Computer Program
for Determining Molecular Symmetry Properties

by

Thomas D. Bouman* and Gordon L. Goodman

Chemistry Division

—NOTICE—

This report was prepared as an account of work sponsored by the United States Government. Neither the United States nor the United States Atomic Energy Commission, nor any of their employees, nor any of their contractors, subcontractors, or their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness or usefulness of any information, apparatus, product or process disclosed, or represents that its use would not infringe privately owned rights.

April 1971

*Faculty of Chemistry, Southern Illinois University, Edwardsville

TABLE OF CONTENTS

	<u>Page</u>
ABSTRACT	7
I. INTRODUCTION.	7
II. THEORY AND TECHNIQUES	11
A. Basic Group Theory.	11
B. Generation of Standard Representatives for the Symmetry Species of the Finite Point Group.	12
1. The Totally Symmetric Operator	12
2. Choice of Basis Functions	13
3. Construction of Crude Symmetry Vectors	14
4. Case of Inversion Symmetry	15
5. Imposition of a Standard Phase Choice	16
6. Example: The Octahedral Group O_h	17
C. Symmetry-adaptation of an Arbitrary Closed Set of Functions.	18
1. Allowed Function Classes.	18
2. Example: Atomic Basis Set	19
3. Symmetrization Procedure	20
D. Finite Point-group Vector-coupling Coefficients	21
1. Definition and Properties	21
2. Method of Calculation	22
3. Antisymmetric Direct Products.	24
4. Extensions and Uses	24
E. Phase Alignment of Sets of Functions Carrying the Same Symmetry Species	25
III. PROGRAM DESCRIPTIONS AND ALGORITHMS	26
A. Design Philosophy	26
B. Logic Flow.	26
C. Descriptions of the Subprograms and Their Algorithms	30
1. Driver and Initialization.	30
2. Input and Molecular-geometry Analysis	30
3. Routines Common to More Than One Major Segment	31
4. Determination of Point-group Properties	32

TABLE OF CONTENTS

	<u>Page</u>
5. Symmetry-adaptation of an Atomic Basis Set	34
6. Finite Point-group Vector-coupling Coefficients.	36
D. Definition of Variables in Common Blocks	37
1. Glossary	37
2. Blocks Containing Output from the Program	37
3. Blocks Containing Information Internal to the Program .	40
IV. INSTRUCTIONS FOR USE	44
A. Description of Input	44
B. Built-in Basis Sets	44
C. Sample Input and Program Output.	48
D. Table Limits and Error Messages	49
APPENDIX: Source-code Listings	67
ACKNOWLEDGMENTS	106
REFERENCES	107

LIST OF FIGURES

<u>No.</u>	<u>Title</u>	<u>Page</u>
1.	Disposition of the Set of Points S at Which the Operator P of Eq. 5 Is Defined, Shown for an Octahedral MX_6 Molecule.	12
2.	Structure of Matrix M from Eq. 10, Showing Off-diagonal Blocks Connecting Sets of Functions Belonging to the Same Symmetry Species	15
3.	Computer Plots of Real, Symmetry-adapted Spherical Harmonics Centered at the Origin Giving the Simplest Occurrence of Each Even-parity Representation of O_h	18
4.	Computer Plots of the Simplest Occurrence of Each Odd-parity Irreducible Representation of O_h	18
5.	Symmetry-adapted Prototype Functions ($L = 6$) Spanning All the Even-parity Representations of O_h	19
6.	Symmetry-adapted Prototype Functions ($L = 9$) Spanning All the Odd-parity Representations of O_h	19
7.	Flowchart of Overall Logic Flow	27
8.	Flowchart for Molecular-geometry Analysis Section	28
9.	Flowchart for Determination of Point-group Properties and Prototype Functions	28
10.	Flowchart for Symmetry-adaptation of Atomic Basis Set.	29
11.	Flowchart for Calculation of Vector-coupling Coefficients.	29
12.	A Possible Overlay Structure for the GPTHEORY Program.	29
13.	Example of Input Deck Including a Basis Set Supplied by the User	46
14.	Example of Input Deck with Basis Set Selected by the Program	46
15.	Sample Output from the Program for an Octahedral KrF_6 Molecule, Quantized about the Fourfold Axis.	51

LIST OF TABLES

<u>No.</u>	<u>Title</u>	<u>Page</u>
I.	Symmetry Species of the Point Group O_h	17
II.	L-values Used in the Prototype Functions for the Various Point Groups.	33
III.	Description of Input Deck	45
IV.	Prestored Minimal Valence-shell Atomic Basis Set.	46
V.	Examples of Double-zeta Slater Basis Sets	48
VI.	Summary of Input and Output for the Computer Program as Currently Implemented.	48

GPTHEORY: A FORTRAN Computer Program for Determining Molecular Symmetry Properties

by

Thomas D. Bouman and Gordon L. Goodman

ABSTRACT

Many simplifications and insights into quantum-mechanical calculations on molecules are possible if one uses the techniques and theorems of molecular point-group theory. However, because of the difficulty in translating the standard textbook methods of group theory into a form amenable to use on a computer, these simplifications have not been generally exploited in large-scale machine computations. Despite avoiding explicit consideration of the nonnumeric operations of the point group, we have succeeded in implementing a method in FORTRAN IV which allows one to generate all the relevant symmetry information about the molecule, requiring as input only the type of nuclei, their spatial coordinates, and parameters characterizing the atomic basis set. The method depends on the construction of a simple, artificial operator having the symmetry of the nuclear configuration and the generation of a set of prototype functions spanning all the irreducible representations of the group. These may be used to generate various types of symmetry information, including a symmetry-adapted atomic basis set, vibrational symmetry coordinates, and finite point-group vector-coupling coefficients. Listings of the computer program, known as GPTHEORY, and complete instructions for use are given.

I. INTRODUCTION

The theorems and techniques of group theory as applied to molecules with symmetry have long been recognized as a powerful tool for the theoretical chemist.¹⁻³ Use of symmetry-based selection rules, for example, has shed much light on the interpretation of atomic and molecular spectra^{4,5} and has led to many simplifications in quantum-mechanical calculations.⁶ Methods developed by Racah^{7,8} for atomic spectroscopy and extended to molecules by Tanabe and co-workers^{9,10} and by Griffith^{11,12} have made possible the determination of quantitative as well as qualitative relationships from the symmetry properties of a problem.

Until fairly recently, however, molecular symmetry properties have been generally ignored in organizing problems in molecular quantum mechanics for solution on large, high-speed digital computers. The resulting type of procedure, in which one allows the computer to calculate and manipulate all quantities in the same way, even those that group theory would say must vanish, may be termed a "brute-force" approach. The tendency to use this type of approach arises because of difficulties in translating the nonnumeric operations and manipulations of molecular point-group theory to a form that can be conveniently handled on a digital computer. Nevertheless, quantum chemists have recently come to appreciate again that an investigator who ignores symmetry not only gives up an important means of alleviating computational bottlenecks, but also foregoes insights that symmetry can give to the results of a large calculation. Thus, increased attention is currently being paid to techniques for implementing symmetry aspects for machine calculations.

Several different lines of attack have been followed in efforts to take account of symmetry in designing machine programs for quantum chemistry; a few examples, which are by no means exhaustive, will serve to illustrate these lines of attack. One method is to design a computer program around the specific symmetry properties of a particular point group, and thus to take full advantage of the qualitative simplifications occurring in a restricted class of molecules. Examples of this approach appear in the work of Joshi¹³ on hydrides with C_{3v} symmetry, that of Wahl¹⁴ on homonuclear diatomic molecules (symmetry $D_{\infty h}$), and that of McLean and Yoshimine¹⁵ on linear molecules in general.

A second category includes computer programs which are designed to accommodate the simplifications arising in a wider class of point groups, but which require the specification of various amounts of group-theoretical information as input. The IBMOL program system of Clementi and co-workers,¹⁶ for example, uses selection rules in eliminating the handling of zeros; however, the user must provide the program with a symmetry-adapted basis set and certain indices characterizing the point group. The POLYATOM system¹⁷ generates lists of nonvanishing integrals by extensive manipulation of input tables of symmetry operations and their effects on atoms and atomic basis functions. Symmetry- and spin-adapted Slater determinantal configuration functions for molecules with no higher than two-fold degeneracies may be generated automatically by the projection operator method of Gershgorin and Shavitt,¹⁸ if a symmetry-adapted atomic basis set is provided. A semiempirical calculation organized for symmetry has been programmed by Glarum,¹⁹ who requires the user to supply the irreducible representation matrices for the point group under consideration. Again, no degeneracies higher than two are permitted. A final example in this category is the automation by several investigators of problems in molecular vibration and rotation.²⁰ The methods have in common the requirement that the user provide the transformation from internal to symmetry coordinates.

Several attempts have been made to generate the required group-theoretical information on the computer. Possibly the first such method to be implemented was developed by Flodmark and Blokker.²¹ These authors concentrated on the detailed properties of the group itself, such as the actual irreducible representation matrices, and generated symmetry-adapted functions with an automated version of the same techniques and formalism one employs in discussing the symmetry aspects of such a problem in a group-theory textbook. The user must supply the program with the multiplication table of the group. Gabriel^{22,23} has developed and is implementing a computer-oriented method for finding symmetry-adapted functions which avoids explicit use of the representation matrices, but makes use of the generators for the group and their commutator algebra. His work may be regarded as providing a rigorous theoretical justification for the method we have developed.

An approach much closer in spirit to the one we have implemented has been proposed by Moccia.²⁴ Rather than working with the symmetry-group operations and irreducible representations themselves, he uses instead group-theoretical theorems which basis functions and operators must satisfy if they are to be symmetry-adapted. Standard techniques of calculating quantum-mechanical integrals and diagonalizing matrices are employed to generate a symmetrized basis set in a numerical way well suited to the capabilities of a digital computer. Moccia's method, however, is dependent on the assumption that the unsymmetrized atomic basis functions comprise sets of "equivalent orbitals"²⁵ which are permuted among themselves by the symmetry operations of the point group. Thus, a pre-transformation of the atomic basis functions into spatially equivalent orbitals must be performed before the method can be used; this may be a nontrivial task, especially if extended basis sets are employed. Furthermore, the construction of spatially equivalent orbitals in some cases requires hybridization of the atomic basis functions; this involves assumptions about the nature of the basis functions which go beyond those strictly determined by symmetry.

Chung and Goodman^{26,27} have extended Moccia's work to obviate the need for using "equivalent orbitals" and have succeeded in producing canonical symmetry-adapted atomic basis sets from the atomic coordinates and initial Slater-type atomic orbitals alone. Their work and the present work have influenced each other.

To obtain symmetry coordinates for their work on vibrational spectra of polymers and other molecules, Gussoni and Zerbi²⁸ have implemented a method similar in spirit to Moccia's. By setting up the $\mathbf{G}$ matrix for a set of internal nuclear-displacement coordinates and diagonalizing it, they produced a transformation that takes the original internal coordinates into symmetry coordinates.

Our present work is motivated in part by the desire to develop an efficient and systematic way of introducing symmetry into large-scale computing efforts, and also by the wish to explore new algorithms for solving standard problems--algorithms suggested by the numerical capabilities of digital computers and their associated scientific programming languages. If such a method is to be widely used, it should fulfill the requirements of speed, generality, simplicity of input, and transferability from one type of problem to another.

In the following sections, we describe a method of generating group-theoretic information on a machine in a form that is readily usable for subsequent computation. As we show, it is in fact possible to obtain virtually all relevant symmetry information, both qualitative and quantitative, for a quantum-mechanical problem without recourse to any standard textbook representation for, or explicit consideration of, the elements forming the point group. We adopt the point of view that what really interests us is not rotations, reflections, etc., nor group multiplication tables and irreducible representation matrices, nor even how functions transform under the symmetry operations. We are interested rather only in numerical relationships, such as whether a matrix element is zero or nonzero, and what ratios exist between nonzero elements. Hence we work with matrix elements from the start.

In the method to be described, we rely substantially on the following theorem in group theory, which we state without proof:

Theorem. If a matrix has a structure embodying the full symmetry of the particular nuclear geometry, its eigenvectors form bases for irreducible representations of the point group of the molecule, provided that there are no accidental degeneracies in the eigenvalues.

A matrix having this structure is the matrix of an operator that itself possesses the symmetry of the nuclear framework. In the statement of this theorem, "a structure embodying the full symmetry of the particular nuclear geometry" means a structure that is left unaltered by each operation of the point group to which the nuclear framework belongs, but one that is altered by any operation under which the nuclear framework is not invariant.

In our method, two artificial operators totally symmetric with respect to the nuclear framework are constructed, and a set of functions is chosen to span all the symmetry species occurring for the point group.* These functions then comprise a standard set against which the symmetry aspects of any other functions can be aligned, and with which other group-theoretic information can be obtained.

*It is clear at this point that linear molecules should be excluded from this method, since their possible symmetry species are not finite in number. However, this class of molecules is already well endowed with computer-oriented techniques taking advantage of symmetry.

II. THEORY AND TECHNIQUES

A. Basic Group Theory

We begin this section with a brief review of representation theory. Let φ be a set of functions $\varphi_1, \varphi_2, \dots$, associated in some way with a molecule belonging to a point group G . We assume that φ is complete in the sense that any symmetry operation $\underline{R}$ of G transforms each of the functions, φ_j , say, into a linear combination of the other functions in the set

$$\varphi_j' \equiv \underline{R}\varphi_j = \sum_k \varphi_k A_{kj}(\underline{R}), \quad (1)$$

or, in matrix-vector notation,

$$\varphi' = \varphi \underline{A}(\underline{R}). \quad (2)$$

Thus, each such operation $\underline{R}$ has a matrix $\underline{A}(\underline{R})$ associated with it. One can easily show (see, e.g., Ref. 6) that the set of matrices $\underline{A}(\underline{R})$ forms a representation of the group G . Although this representation is in general not the simplest possible, we can find a reduction matrix $\underline{U}$, corresponding to a unitary transformation of φ that generates a new set of representation matrices $\underline{B}(\underline{R})$ which factor into the simplest possible or irreducible representations of G ,

$$\underline{B}(\underline{R}) = \underline{U}^{-1} \underline{A}(\underline{R}) \underline{U}. \quad (3)$$

The same transformation applied to the functions φ converts them to a set of functions ψ which form bases for irreducible representations of G :

$$\psi = \varphi \underline{U}. \quad (4)$$

This new set of functions is said to be symmetry-adapted to the group G , and satisfies all the usual symmetry-selection rules and other relationships.

Our interest therefore centers on the determination of the reduction matrix $\underline{U}$ for a given set of functions, in a computer-oriented way. We identify three major stages in our procedure. First, the functions are transformed to a set decomposed according to definite symmetry types; second, particularly simple functions are extracted from these initial symmetrized functions; finally, a standard phase choice is imposed to yield canonical symmetrized functions.

B. Generation of Standard Representatives for the Symmetry Species of the Finite Point Group

1. The Totally Symmetric Operator

We address ourselves first to the choice of an operator P which possesses the full symmetry of any given nuclear geometry. The nuclear-attraction potential field and, of course, the complete Hamiltonian operator for a molecule are examples of such operators which normally occur in a quantum-mechanical calculation. We want to construct, however, an artificial operator for which the matrix elements are almost trivial to compute, and which has no more and no less invariance under rotations and reflections than do these naturally occurring ones.

A simple operator having these symmetry properties, one that avoids placing undue weight on symmetry-dictated nodal surfaces occurring for various functions, is the operator that annihilates a function everywhere except at a finite number of points that are members of the set S :

$$P(\underline{r}) = \sum_{\underline{r} \in S} \delta(\underline{r} - \underline{r}_v) \quad (5)$$

such that the points in S lie on circles defining a plane perpendicular to the line joining each nucleus to the center of nuclear charge of the molecule of interest, as shown in Fig. 1. Evaluation of the matrix elements of P then reduces to summing the values of the integrand at the points in S .

Fig. 1

Disposition of the Set of Points S at Which the Operator P of Eq. 5 Is Defined, Shown for an Octahedral MX_6 Molecule. ANL Neg. No. 121-4299.

This set of points defining P may be viewed either as one that goes identically into itself under all the operations of the point group G , or as an approximation to a set of symmetrically disposed circles (where the summation in Eq. 5 is replaced by a sum of line integrals). In practice,

the two interpretations appear to be equivalent, in that no practical accuracy is lost if the points do not exactly transform into one another. Transformation coefficients good to 11 significant figures are generally obtained with 24 equally spaced points on each circle, even in cases in which the circle encloses a fivefold symmetry axis. Our choice of operator has the advantage not only of making the computations simple, but also of freeing as much as possible the determination of symmetry properties from other details of any problem in which they are to be used.

2. Choice of Basis Functions

In choosing a "standard set" of functions to symmetry-adapt, we require that each symmetry species occur at least once in the representation spanned by these functions. A particularly convenient choice is the set of $2L + 1$ orbital-angular-momentum eigenfunctions, $Y_{LM}(\theta, \phi)$, of appropriately high L -value. In a spherically symmetric environment, these functions span a $(2L + 1)$ -fold-degenerate irreducible representation of the three-dimensional full rotation group $SO(3)$. As the symmetry of the environment is lowered, this erstwhile irreducible representation splits into the various irreducible representations appropriate to the finite point group. By choosing L high enough, therefore, we are assured that each possible new representation will be spanned at least once.^{2,29,30} If a center of inversion is present, two distinct L values are in fact required: an even one, L_g , for the representations that are even with respect to inversion, and an odd one, L_u , for those of odd parity. (The selection of the proper set of functions for any particular case of chemical significance, i.e., for any point group having no higher than an eightfold rotation axis, is built into the program. It can be shown that $L = 16$ is the highest one needs to span all the symmetry species that occur in these cases.)

For convenience of computation and ease of visualization, we choose the following real combinations of the Y_{LM} :¹¹

$$\left. \begin{aligned} S_0^{(L)} &= Y_{L0}; \\ S_{2M}^{(L)} &= \frac{1}{\sqrt{2}} (Y_{L,-M} - Y_{L,-M}^*); \\ S_{2M+1}^{(L)} &= \frac{1}{\sqrt{2}} (Y_{L,-M} + Y_{L,-M}^*); \end{aligned} \right\} \quad (6)$$

where

$$Y_{L,-M} = (-1)^M Y_{LM}^*{}^\dagger$$

[†]See Ref. 8, where Eqs. 2.5.6 and 2.5.28 provide the definitions used here for the Y_{LM} .

This entails no loss of generality. We index the functions sequentially by the integer K , $K = 1, 2, \dots, 2L + 1$. Let this set be denoted $S_K^{(L)}$, or simply $\underline{S}^{(L)}$; a simple radial dependence, $R(r) = r^{-1}$, is assumed a part of the functions $\underline{S}^{(L)}$ in order to avoid accidental degeneracies. The functions are centered at the center of nuclear charge, which is chosen as the origin of the master coordinate system.

3. Construction of Crude Symmetry Vectors

By the Theorem in Sec. I, a unitary transformation $\underline{W}$ which diagonalizes a matrix $\underline{P}$, totally symmetric with respect to the point-group operations,

$$\underline{W}^{-1} \underline{P} \underline{W} = \underline{P} \underline{D}, \quad (7)$$

produces eigenvectors that belong to definite symmetry types. We therefore compute the matrix of P over the set $\underline{S}^{(L)}$ and diagonalize it by means of a unitary matrix $\underline{W}$.^{*} This transformation defines a symmetry-adapted set of functions $\underline{F}^{(L)}$ through

$$\underline{F}^{(L)} = \underline{S}^{(L)} \underline{W}. \quad (8)$$

These new functions $\underline{F}^{(L)}$ do already transform irreducibly, but they are not yet in a sufficiently well-defined form to be suitable for subsequent applications.

In particular, we must first find out how many different symmetry species are spanned by the set $\underline{F}$, and identify which functions belong to which species. To do this, we sort the $\underline{F}$ first according to the degeneracies of their associated eigenvalues. We still must distinguish among species with the same degeneracy, but with different symmetry behavior. For this purpose, we choose a second totally symmetric operator, P' , of the type of Eq. 5, but defined on a different set of points S' , with matrix elements

$$P'_{\mu\nu} = \left\langle S_{\mu}^{(L)} \left| P' \right| S_{\nu}^{(L)} \right\rangle. \quad (9)$$

In general, the matrix $\underline{W}$ that diagonalized $\underline{P}$ does not completely diagonalize $\underline{P}'$, but produces a matrix,

^{*}The choice of algorithm for diagonalization in this method is not a matter of indifference. If degeneracies are present, the Givens-Householder technique introduces mixing in the eigenvectors where there is none in the matrix, corresponding to an arbitrary rotation of the coordinate axes. The Jacobi method, for one, gives eigenvectors that retain the simple structure of the original matrix, and is therefore preferable.

$$\tilde{W}^{-1} \tilde{P} \tilde{W} = \tilde{M}, \quad (10)$$

having the structure shown in Fig. 2. The nonzero off-diagonal elements of $\tilde{M}$ connect functions F_i belonging to the same symmetry species, so we can now assign a unique symmetry label to each function.

Fig. 2

Structure of Matrix $\tilde{M}$ from Eq. 10, Showing Off-diagonal Blocks Connecting Sets of Functions Belonging to the Same Symmetry Species. The blocks on the diagonal are themselves diagonal, with equal diagonal elements within a block. ANL Neg. No. 121-4451.

4. Case of Inversion Symmetry

A molecule possessing inversion symmetry will require that the above procedure be repeated for the ungerade representations. One then has two sets of functions, which together span all the irreducible representations of the point group. Since the symmetry behavior of functions in one set differs from that of corresponding members of the other set only in their inversion parity, it is useful to establish a correspondence between the two sets of symmetry labels that have been assigned. Any matrix element of P between members of the sets $\tilde{F}^{(L_g)}$ and $\tilde{F}^{(L_u)}$ must vanish because of the parity difference. However, if one first forms a new set of functions

$$Y_{L''M''} = \sum_{MM'} (L''M'' | LML'M') Y_{LM} Y_{L'M'}, \quad (11)$$

where $L'' = L_u$, $L = L'$, and the quantities $(L''M'' | LML'M')$ are the usual Clebsh-Gordan coefficients,⁸ these new functions must perforce have even parity, as the product of two sets of like parity. Since the L'' and M'' values determine the behavior of the functions toward rotations and reflections, the coefficients $\tilde{W}$ that symmetry-adapt the real analogues of the $Y_{L''M''}$ are the same as those that symmetrize $\tilde{S}^{(L_u)}$. The nonzero matrix elements

$$\langle F_{\mu}^{(L'')} | P | F_{\nu}^{(L_g)} \rangle$$

then identify symmetrized functions differing only in their behavior toward inversion.

5. Imposition of a Standard Phase Choice

If all sets of functions belonging to the same symmetry species have the same phase, the individual off-diagonal blocks $M^{(\alpha\beta)}$ of Eq. 10 are also diagonal, and we have achieved the maximum factorization of the matrix. To accomplish this, and at the same time to impose a canonical phase choice on all symmetry vectors, we proceed in two stages.

First, we find the set of functions $\tilde{S}$ that has the lowest L value capable of spanning a particular representation, without regard to possible inversion-parity distinction. Denote this set by $\tilde{S}(\min)$. The axis system and phase evinced by the eigenvectors resulting from the diagonalization of the matrix of P over this set are taken as the canonical phase for that representation.³⁰ (The way in which the set $\tilde{S}(\min)$ is identified for a particular representation is a special case of the procedure for symmetry-adapting an arbitrary set of functions, described in Sec. II.C of this report.)

Second, we align the phases of all occurrences of that representation in the set $\tilde{F}$ with the canonical phase by means of a unitary transformation $\tilde{T}$, described in Sec. II.E. If inversion symmetry is present, the procedure is best elucidated through an example. Suppose that the set of spherical harmonics with $L = 1$ is the smallest set capable of spanning a representation labeled T_{1u} in the group under consideration. Then the phases of all occurrences of T_{1u} in the set $\tilde{F}^{(L_u)}$ are aligned with that of the $L = 1$ set. This phase choice is then also imposed on all occurrences of T_{1g} in $\tilde{F}^{(L_g)}$. The problem of parity is circumvented by using an A_{1u} operator to connect the two sets of functions: One identifies the A_{1u} function in the set $\tilde{F}^{(L_u)}$ and multiplies P by it, thereby forming a new odd-parity operator,

$$P_u = F(A_{1u}) \times P. \quad (12)$$

The off-diagonal blocks of matrix elements of P_u are made diagonal by the method given in Sec. II.E.

The transformations just described may be collected into a matrix $\tilde{T}$ which, when applied to the $\tilde{F}^{(L)}$, defines a new set of functions $\tilde{\psi}^{(L)}$:

$$\tilde{\psi}^{(L)} = \tilde{F}^{(L)} \tilde{T} = \tilde{S}^{(L)} \tilde{W} \tilde{T} \triangleq \tilde{S}^{(L)} \tilde{\psi}'. \quad (13)$$

These functions now have all the well-defined symmetry behavior and phase choices that are desirable for use in calculations, and as such they could be used as our standard set of functions spanning all possible symmetry types in the point group. If the set Ψ' contains two or more representatives of a given symmetry species, however, these vectors will mix together in a way determined by the operators we have used, and hence in a way that bears no interpretive significance. To "undo" this arbitrary mixing, we may apply a final transformation $\tilde{B}$ which maximizes the number of zeroes in $\tilde{V}'$ while preserving the symmetry properties. The construction of $\tilde{B}$ is described in detail in Ref. 26. The functions $\tilde{\Psi}^{(L)}$,

$$\tilde{\Psi}^{(L)} = \tilde{\Psi}'^{(L)} \tilde{B} = \tilde{S}^{(L)} \tilde{V}' \tilde{B} = \tilde{S}^{(L)} \tilde{V}, \quad (14)$$

comprise the standard set, called prototype functions.

6. Example: The Octahedral Group O_h

As an example of the prototype functions and the standard phase choices we have implemented, we consider the octahedral group O_h . The irreducible representations of O_h and their degeneracies are indicated in Table I. The lowest L values that span all the even- and odd-parity representations are $L_g = 6$ and $L_u = 9$, respectively. If $\Gamma(L = 6)$ and $\Gamma(L = 9)$ are the corresponding representations, then under the operations making up the group O_h they reduce as follows:

$$\left. \begin{aligned} \Gamma(L = 6) &\rightarrow A_{1g} + A_{2g} + E_g + T_{1g} + 2T_{2g}; \\ \Gamma(L = 9) &\rightarrow A_{1u} + A_{2u} + E_u + 3T_{1u} + 2T_{2u}. \end{aligned} \right\} \quad (15)$$

TABLE I. Symmetry Species of the Point Group O_h

Symbol		Degeneracy
Even Parity	Odd Parity	
A_{1g}	A_{1u}	1
A_{2g}	A_{2u}	1
E_g	E_u	2
T_{1g}	T_{1u}	3
T_{2g}	T_{2u}	3

Figures 3 and 4 show the simplest representatives of each symmetry species in O_h , derived from real spherical harmonics, and indicate the standard phase choices against which the prototype functions are aligned.

The behavior of the symmetry-adapted prototype functions for $L = L_g$ and $L = L_u$ is shown in Figs. 5 and 6. These figures were generated and plotted by the computer as a program option, and illustrate a type of incidental application for which the program might be useful. The transformation coefficients making up the matrix $\tilde{V}$ in Eq. 14 are real combinations of those given in Appendix 2 of Ref. 11.

C. Symmetry-adaptation of an Arbitrary Closed Set of Functions

1. Allowed Function Classes

We have placed a good deal of emphasis thus far on the construction of the prototype functions and their phases. This effort was well spent, however, since it now becomes an easy task to symmetry-adapt virtually any set of functions associated with the molecule, that is, to find the reduction matrix $\tilde{U}$ of Eq. 4. For this to be possible, it is usually necessary to

Fig. 3

Computer Plots of Real, Symmetry-adapted Spherical Harmonics Centered at the Origin Giving the Simplest Occurrence of Each Even-parity Representation of O_h . The functions are evaluated on the surface of a unit cube, and the shaded areas represent negative function values. Subspecies labels are those used in Ref. 11. ANL Neg. No. 121-4440.

Fig. 4

Computer Plots of the Simplest Occurrence of Each Odd-parity Irreducible Representation of O_h . ANL Neg. No. 121-4438.

Fig. 5

Symmetry-adapted Prototype Functions
($L = 6$) Spanning All the Even-parity
Representations of O_h . ANL Neg.
No. 121-4439.

Fig. 6

Symmetry-adapted Prototype Functions ($L = 9$)
Spanning All the Odd-parity Representations
of O_h . ANL Neg. No. 121-4437.

require only that the set be closed, in the sense that Eq. 1 is valid. These functions may be of various types: For example, in an *ab initio* calculation of electronic properties, they may be atomic basis functions centered on the individual atoms. For problems in molecular dynamics, they may be the Cartesian displacements of the nuclei. To illustrate an important case, we will carry through the example of determining a symmetry-adapted atomic basis set.

2. Example: Atomic Basis Set

Let $\tilde{X}$ be the set of all atomic orbital basis functions to be used in a subsequent calculation; a typical member is $\chi_{nlm}^A(R_A, \theta_A, \phi_A)$, whose quantum numbers are n , ℓ , and m , and whose origin is at atom A . We may choose the functions to be real without infringing on the generality of the method. From $\tilde{X}$ we may extract a smaller set $\tilde{X}$ which contains all values of the angular indices, ℓ and m , represented in $\tilde{X}$, but with only one representative radial part for each ℓ value within a class of symmetry-related

atoms, or atom type.* Again, no loss of information is involved, since no point-group operation changes the radial part of a function (except for a change of origin), and therefore functions that differ only in their radial parts have the same transformation coefficients associated with them. The final set of symmetry-adapted functions from χ , which are to be used in a quantum-mechanical calculation, are denoted $\tilde{g}$:

$$\tilde{g} = \chi \tilde{U}. \quad (16)$$

As χ is subjected to the various symmetry operations of G , certain subsets of its constituent functions transform only among themselves and never into members of another subset. In particular, no rotation or reflection changes the quantum numbers n or ℓ of a function, nor can a rotation or reflection place the function on a new nucleus which is not related by symmetry to the original nucleus. We denote a subset with a given ℓ -value and atom type τ by $\chi^{\ell\tau}$; the subset contains $(2\ell+1) n_\tau$ basis functions, where n_τ is the number of atoms in τ . The symmetry-adapted functions we desire contain components within a given subset; we thus recognize that the transformation matrix $\tilde{U}$ also factors into submatrices $\tilde{U}^{\ell\tau}$, permitting each subset to be symmetry-adapted independently:

$$\tilde{g}^{\ell\tau} = \chi^{\ell\tau} \tilde{U}^{\ell\tau}. \quad (17)$$

Each subset in χ is treated the same way; in the remainder of this section, therefore, the superscripts will be omitted for brevity. Further, since the actual radial dependence factors out of the symmetry transformation, we ignore it and replace it by $R(r) = r^{-1}$ for ease of computation; the only requirement on $R(r)$ is that it be sufficiently distinctive to avoid accidental degeneracies.

3. Symmetrization Procedure

The initial stages of symmetry-adaptation proceed as outlined previously: the matrix of p over the subset χ is calculated and diagonalized by a transformation W . A new set of functions ρ is formed from χ ,

$$\rho = \chi W. \quad (18)$$

To determine what representations are contained in ρ and to line up phases, we reintroduce the real harmonics $\tilde{S}^{(L)}$ and calculate a connection matrix Q ,

*Operationally, we define an atom type to include all atoms of a given chemical element that are "equidistant" from the center of nuclear charge. According to the looseness of the criterion applied to determine "equidistance," a user may optionally include atoms that are not strictly related by a symmetry operation. In studies of small distortions from an equilibrium geometry, this option allows one to avoid abrupt changes of symmetry adaptation as the nuclear geometry changes continuously.

$$Q_{ij} = \langle S_i^{(L)} | P | \chi_j \rangle. \quad (19)$$

Then, letting $\tilde{V}^{(\alpha)}$ be the set of prototype transformation vectors associated with the α th prototype eigenvalue, of degeneracy n_α , and $\tilde{W}^{(\beta)}$ the set of atomic basis-function eigenvectors from eigenvalue β , of degeneracy $n_\beta = n_\alpha$, we compute the $n_\alpha \times n_\alpha$ matrix

$$\tilde{M}^{(\alpha\beta)} = \tilde{V}^{-1(\alpha)} \tilde{Q} \tilde{W}^{(\beta)}. \quad (20)$$

If $\tilde{M}^{(\alpha\beta)}$ contains any nonzero elements, then $\tilde{W}^{(\beta)}$ carries the same symmetry species as $\tilde{V}^{(\alpha)}$. The phase of $\tilde{W}^{(\beta)}$ is made to correspond to that of $\tilde{V}^{(\alpha)}$ by the method of Sec. II.E.

The new basis functions $\tilde{\sigma}'$,

$$\tilde{\sigma}' = \tilde{\rho} \tilde{T} = \tilde{\chi} \tilde{W} \tilde{T} = \tilde{\chi} \tilde{U}', \quad (21)$$

have the well-defined transformation properties necessary to induce maximum symmetry factorization in a calculation. However, if the same irreducible representation is spanned more than once in a set of functions $\tilde{\sigma}'$, the different sets still mix together in a way determined by the operator $\tilde{P}$. The method described in Ref. 26 determines a final transformation $\tilde{B}'$, which produces a simple set of coefficients in $\tilde{U}$:

$$\tilde{\sigma} = \tilde{\sigma}' \tilde{B}' = \tilde{\chi} \tilde{U}' \tilde{B}' = \tilde{\chi} \tilde{U}. \quad (22)$$

The procedure is repeated for each set $\tilde{\chi}^{\ell\tau}$.

D. Finite Point-group Vector-coupling Coefficients

1. Definition and Properties

Let $\tilde{F}_\Gamma$ and $\tilde{G}_{\Gamma'}$ be two sets of functions forming bases for the irreducible representations Γ and Γ' , respectively, of the point group. Consider a new set $\tilde{H}$ with $n_\Gamma \times n_{\Gamma'}$ members formed by taking all possible products of pairs $\tilde{F}_{\Gamma\gamma} \times \tilde{G}_{\Gamma'\gamma'}$, where γ and γ' label the different subspecies in Γ and Γ' . The elements in $\tilde{H}$ provide a new basis set for representations of the point group. A representation Γ'' is contained in the direct product $\Gamma \times \Gamma'$, if symmetry-adapted functions $\tilde{H}_{\Gamma''\gamma''}$ can be constructed from $\tilde{H}$ by means of a suitable unitary transformation:¹²

$$\tilde{H}_{\Gamma''\gamma''} = \sum_{\gamma} \sum_{\gamma'} (\Gamma''\gamma'' | \Gamma\gamma \Gamma'\gamma') \tilde{F}_{\Gamma\gamma} \tilde{G}_{\Gamma'\gamma'}. \quad (23)$$

The coefficients $(\Gamma''\gamma''|\Gamma\gamma\Gamma'\gamma')$ in Eq. 23 are called vector-coupling coefficients. In the full rotation group, these coefficients occur in the coupling of two angular momenta to form a third:

$$Y_{\ell''m''} = \sum_m \sum_{m'} (\ell''m''|\ell m \ell' m') Y_{\ell m} Y_{\ell' m'}. \quad (24)$$

In this case, the coefficients are also called Clebsh-Gordan coefficients.⁸ The vector-coupling coefficients are defined to be orthonormal in the sense:

$$\left. \begin{aligned} \sum_{\gamma} \sum_{\gamma'} (\Gamma_1 \gamma_1 | \Gamma \gamma \Gamma' \gamma') (\Gamma \gamma \Gamma' \gamma' | \Gamma_2 \gamma_2) &= \delta_{\Gamma_1 \Gamma_2} \delta_{\gamma_1 \gamma_2} \delta(\Gamma, \Gamma', \Gamma_1); \\ \sum_{\Gamma''} \sum_{\gamma''} (\Gamma \gamma_1 \Gamma' \gamma'_1 | \Gamma'' \gamma'') (\Gamma'' \gamma'' | \Gamma \gamma_2 \Gamma' \gamma'_2) &= \delta_{\gamma_1 \gamma_2} \delta_{\gamma'_1 \gamma'_2}; \end{aligned} \right\} \quad (25)$$

where $\delta(\Gamma, \Gamma', \Gamma_1)$ is unity if $\Gamma \times \Gamma'$ contains Γ_1 , and is zero otherwise. Because vector-coupling coefficients completely describe the geometric relationships between various basis functions, any quantum-mechanical matrix element can be factored into a part which is determined by only geometric properties and a second part which depends on rotationally invariant properties of the particular wave functions and operator:

$$\langle F_{\alpha''} \Gamma'' \gamma'' | A_{\Gamma \gamma} | G_{\alpha'} \Gamma' \gamma' \rangle = (\Gamma'' \gamma'' | \Gamma \gamma \Gamma' \gamma') \langle F_{\alpha''} \Gamma'' || A_{\Gamma} || G_{\alpha'} \Gamma' \rangle. \quad (26)$$

The second factor on the right is called the reduced matrix element, and the theorem that assures this factorization is called the Wigner-Eckart theorem.¹² If Γ , Γ' , and Γ'' are degenerate irreducible representations, considerable labor may be saved in calculating matrix elements among the members of the degenerate sets by calculating the reduced matrix element only once.

2. Method of Calculation

To compute the vector-coupling coefficients for finite point groups, we use the fact that the real spherical harmonics in terms of which we expand prototype functions for the finite group are themselves symmetry-adapted to the full rotation group. In showing the derivation, we find it convenient to revert to the complex functions Y_{LM} , which are then related to the prototype functions $\psi^{(L)}$ of Eq. 14 by a complex transformation $\zeta^{(L)}$:

$$\psi^{(L)} = Y^{(L)} \zeta^{(L)}, \quad (27)$$

or

$$\psi_{\alpha\Gamma\gamma}^{(L)} = \sum_{M=-L}^L Y_{LM} C_{LM}^{\alpha\Gamma\gamma}. \quad (28)$$

The coefficients $C_{LM}^{\alpha\Gamma\gamma}$ are related to the elements of the matrix $\underline{Y}$ of Eq. 14 in the following way: First, let the matrix $\underline{C}$ be written

$$\underline{C} = \underline{A} + i\underline{B},$$

where $\underline{A}$ and $\underline{B}$ are real matrices. Then,

$$\left. \begin{aligned} A_{L0}^{\alpha\Gamma\gamma} &= V_{L1}^{\alpha\Gamma\gamma}; & B_{L0}^{\alpha\Gamma\gamma} &= 0; \\ A_{LM}^{\alpha\Gamma\gamma} &= \frac{(-1)^M}{\sqrt{2}} V_{L,2M+1}^{\alpha\Gamma\gamma}; & B_{LM}^{\alpha\Gamma\gamma} &= \frac{(-1)^M}{\sqrt{2}} V_{L,2M}^{\alpha\Gamma\gamma}; \\ A_{L,-M}^{\alpha\Gamma\gamma} &= (-1)^M A_{LM}^{\alpha\Gamma\gamma}; & B_{L,-M}^{\alpha\Gamma\gamma} &= (-1)^{M+1} B_{LM}^{\alpha\Gamma\gamma}. \end{aligned} \right\} \quad (29)$$

Now consider the matrix element

$$\left\langle \psi_{\alpha''\Gamma''\gamma''}^{(L'')} \middle| \psi_{\alpha'\Gamma'\gamma'}^{(L')} \middle| \psi_{\alpha\Gamma\gamma}^{(L)} \right\rangle.$$

In view of Eq. 26, this can be written

$$\left\langle \psi_{\alpha''\Gamma''\gamma''}^{(L'')} \middle| \psi_{\alpha'\Gamma'\gamma'}^{(L')} \middle| \psi_{\alpha\Gamma\gamma}^{(L)} \right\rangle = (\Gamma''\gamma'' | \Gamma\gamma\Gamma'\gamma') \left\langle \psi_{\alpha''\Gamma''}^{(L'')} \middle| \psi_{\alpha'\Gamma'}^{(L')} \middle| \psi_{\alpha\Gamma}^{(L)} \right\rangle. \quad (30)$$

By Eq. 28, however, we can also write

$$\left\langle \psi_{\alpha''\Gamma''\gamma''}^{(L'')} \middle| \psi_{\alpha'\Gamma'\gamma'}^{(L')} \middle| \psi_{\alpha\Gamma\gamma}^{(L)} \right\rangle = \sum_{M''} \sum_{M'} \sum_M C_{L''M''}^{*\alpha''\Gamma''\gamma''} C_{L'M'}^{\alpha'\Gamma'\gamma'} C_{LM}^{\alpha\Gamma\gamma} \left\langle Y_{L''M''} \middle| Y_{L'M'} \middle| Y_{LM} \right\rangle. \quad (31)$$

Equating these two expressions and applying the Wigner-Eckart theorem to the matrix element over spherical harmonics, we obtain the final expression:

$$(\Gamma''\gamma''|\Gamma\gamma\Gamma'\gamma') = \frac{(-1)^{L''} \langle Y_{L''} \| Y_{L'} \| Y_L \rangle}{\langle \psi_{\alpha''\Gamma''} \| \psi_{\alpha'\Gamma'} \| \psi_{\alpha\Gamma} \rangle} \sum_{M'} \sum_M C_{L'',M+M'}^{\alpha''\Gamma''\gamma''} C_{L'M'}^{\alpha'\Gamma'\gamma'} C_{LM}^{\alpha\Gamma\gamma} (-1)^{M+M'} \begin{pmatrix} L & L' & L'' \\ M & M' & -M-M' \end{pmatrix}. \quad (32)$$

The quantity in front of the summation signs in Eq. 32 is not calculable directly in our method. Its magnitude is determined, however, by the condition of Eq. 25 on the vector-coupling coefficients. We are left with an arbitrary choice of sign for each trio of representations $(\Gamma, \Gamma', \Gamma'')$. In the absence of a well-defined means of imposing other sign choices, we choose the quantity

$$\frac{(-1)^{L''} \langle Y_{L''} \| Y_{L'} \| Y_L \rangle}{\langle \psi_{\alpha''\Gamma''} \| \psi_{\alpha'\Gamma'} \| \psi_{\alpha\Gamma} \rangle}$$

to be positive for all representations and prototype L values. Since the vector-coupling coefficients are not dependent upon the inversion parity of the representations involved, we may take $L = L' = L'' = L_g$, the value that spans all the even representations of the point group.

3. Antisymmetric Direct Products

Equation 32 fails for some combinations $(\Gamma, \Gamma', \Gamma'')$ if all three sets of spherical harmonics have even L values, because of the behavior of the 3- j symbols under permutation of L values. This condition arises when two of the Γ 's are the same and the third belongs to the set of antisymmetric squares (see, e.g., Ref. 11). The summation may be made not to vanish if one of the sets of Y_{LM} is antisymmetric. Let the functions $Y_{L''M''}$, Y_{LM} , $Y_{L'M'}$ of Eq. 24 be chosen so that $L = L' = L_g$, and $L'' = L_u$. The functions $Y_{L''M''}$ are then not usual spherical harmonics, because they are antisymmetric with respect to interchange of L and L' , have even inversion parity (see Eq. 11), and in fact are multiplied throughout by $i \equiv \sqrt{-1}$. The coefficients $C_{L''M''}^{\alpha''\Gamma''\gamma''}$ that symmetry-adapt the $Y_{L''M''}$ can be inserted into Eq. 32 to obtain the desired antisymmetric vector-coupling coefficients.

4. Extensions and Uses

Vector-coupling coefficients are produced by the method described in this report for any finite point group, even for one such as the icosahedral group which has two linearly independent sets of coefficients for certain combinations of irreducible representations.¹¹ With proper

attention to phase choices, it should be possible also to generate the V , W , and X coefficients of Griffith,¹² which are the respective analogues of the Wigner 3-j, 6-j, and 9-j symbols.⁸

E. Phase Alignment of Sets of Functions Carrying the Same Symmetry Species

Let $\tilde{V}^{(\alpha)}$ and $\tilde{W}^{(\beta)}$ be the sets of transformation vectors associated with eigenvalues α , β , respectively, of degeneracy $n_\alpha = n_\beta$. Here $\tilde{V}^{(\alpha)}$ and $\tilde{W}^{(\beta)}$ need not arise from the same set of functions, but $\tilde{V}^{(\alpha)}$ is assumed to have the standard phase choice. We seek a unitary transformation $\tilde{T}^{(\alpha\beta)}$,

$$\tilde{W}^{(\beta)} = \tilde{W}^{(\beta)} \tilde{T}^{(\alpha\beta)}, \quad (33)$$

such that

$$\tilde{M}^{(\alpha\beta)} \tilde{T}^{(\alpha\beta)} = \tilde{M}_D^{(\alpha\beta)}, \quad (34)$$

where $\tilde{M}^{(\alpha\beta)}$ is the matrix connecting $\tilde{V}^{(\alpha)}$ and $\tilde{W}^{(\beta)}$, and $\tilde{M}_D^{(\alpha\beta)}$ is diagonal, with equal diagonal elements λ . $\tilde{M}^{(\alpha\beta)}$ is defined by

$$\tilde{M}^{(\alpha\beta)} = \tilde{V}^{-1(\alpha)} \tilde{P} \tilde{W}^{(\beta)}, \quad (35)$$

where $\tilde{P}$ is the matrix of a totally symmetric operator over the basis functions represented in $\tilde{V}^{(\alpha)}$ and $\tilde{W}^{(\beta)}$.

Suppressing the superscripts and rewriting, we obtain

$$\tilde{M} = \tilde{M}_D \tilde{T}^{-1}, \quad (36)$$

or

$$m_{ij} = \sum_k \lambda_{ik} \delta_{ik} t_{kj}^{-1} = \lambda t_{ij}^{-1}. \quad (37)$$

Since $\tilde{T}$ is unitary, we may write

$$\sum_j |m_{ij}|^2 = \lambda^2 \sum_j |t_{ij}^{-1}|^2 = \lambda^2 \cdot 1. \quad (38)$$

Then

$$\lambda = \sqrt{\sum_j |m_{ij}|^2}.$$

and $t_{ij}^{-1} = t_{ji} = m_{ij}^*/\lambda$. These small transformations $\mathbb{T}^{(\alpha\beta)}$ are all that need be calculated, although formally they may be collected in one unitary transformation $\mathbb{T}$ applied to the entire matrix $\mathbb{W}$.

III. PROGRAM DESCRIPTIONS AND ALGORITHMS

A. Design Philosophy

The program described in this report (which we call GPTHEORY) is designed with a view to interfacing with any of several types of applications programs, as well as being a "stand-alone" program in and of itself. To facilitate interfacing, the output from the program is organized into block-common storage areas whose contents may optionally be dumped on- to an external storage device. Input to the program is purposely kept to a bare minimum, namely, the number and types of atoms, their Cartesian coordinates in some convenient axis system, and "something" that characterizes the basis set. The latter may be as little as a control integer, since the program contains several prestored Slater-type atomic basis sets, from which it will extract the necessary information. Section IV of this report contains complete descriptions of the input and the output.

A potential user of this program may well envision applications that have not been specifically provided for, and may therefore need to modify the structure of the program. Hence, we have tried to make the package as modular as possible, well-defined tasks being relegated to subroutines. Furthermore, "sockets" have been provided for the user to "plug in" his own applications module with minimum trauma.

Extensive matrix manipulations and inner products, such as are used in this method, tend to involve accumulation of roundoff errors that can seriously degrade the discrimination of "zero" and "nonzero" quantities vital to our approach. This problem is handled quite well by the introduction of threshold parameters: Matrix elements and certain other quantities are compared with the zero threshold at frequent stages in the computation, and set equal to zero if they fall below. In this way, cumulative errors are minimized, and up to 15-digit accuracy is maintained in the transformation coefficients.

A good deal of intermediate printout, which is suppressed under normal operation, may be invoked as a diagnostic tool in case of trouble. Indeed, if termination occurs under conditions allowing the program to remain in control, a restart is made with intermediate output to aid in troubleshooting.

B. Logic Flow

The program is divided into several sections that are logically almost independent of one another. Overall control is therefore vested in a

short driver program, MAIN, which calls the appropriate segments, controls restarting in case of program-detected errors, and provides an interface to applications programs.

Figures 7-11 show flowcharts of the several major parts of the program. This version of the program is not organized into an overlay structure; if space is a limitation, however, the control sections (subroutines and COMMON blocks) can be put into overlays as shown in Fig. 12. (The tree structure will, of course, differ somewhat for different machines: It is shown here for the IBM 360 Linkage Editor.)

Fig. 7. Flowchart of Overall Logic Flow

Fig. 8. Flowchart for Molecular-geometry Analysis Section

Fig. 9. Flowchart for Determination of Point-group Properties and Prototype Functions

Fig. 10. Flowchart for Symmetry-adaptation of Atomic Basis Set

Fig. 11. Flowchart for Calculation of Vector-coupling Coefficients (V.C.C.'s)

Fig. 12. A Possible Overlay Structure for the GPTHEORY Program

C. Descriptions of the Subprograms and Their Algorithms

1. Driver and Initialization

- a. MAIN. Acts as the driver and overall logic controller. Initialization functions are performed (see MAIN0060 through MAIN0140 in the appendix), the appropriate computation segments are called, and error restarting with intermediate printout is made when possible.
- b. NOFLOW. Suppresses exponent underflow messages issued by OS/360. Supplied as a machine language deck.
- c. SYMSUM. Prints a summary of the symmetry properties of the point group, and stores several cross-reference tables.
- d. PTGRUP. Determines the Schönflies symbol for the point group of the molecule by a table look-up based on the order of the group, the number of irreducible representations, the presence of a center of inversion, and complex conjugate representations.
- e. FILOUT. Dumps unformatted contents of output COMMON blocks onto data set 20. (May be a dummy data set.)
- f. APPLY. Provides dummy subroutine, which user replaces with his own application.

2. Input and Molecular-geometry Analysis

- a. MOLCUL. Inputs problem identification, atomic Cartesian coordinates, and program-control parameters and thresholds. The origin is translated to the center of nuclear charge, and spherical polar coordinates of the atoms are computed. The subroutine determines the distinct atom-types, assigns the atom to them, tests for linearity of the molecule, determines if a center of inversion is present, and constructs several cross-reference tables. Some checks are made on the suitability of the input.

- b. ALPTAB. Generates a table of factorials and a set of coefficients C_{lmr} to be used in computing normalized associated Legendre polynomials. The formula used is

$$C_{lmr} = \frac{(-1)^m}{2^l} \sqrt{\frac{(2l+1)(l-m)!}{2(l+m)!}} (-1)^r \frac{(2l-2r)!}{r!(l-r)!(l-2r-m)!}$$

- c. GRID. Computes Cartesian and spherical polar coordinates of "totally symmetric" grid points for the molecule. If inversion symmetry is present, the grid is made inversion-symmetric, point for point. The grid

consists of NDIV equally spaced points on each circle centered on the lines joining each atom to the origin. The circle is located 9/10 of the distance from the origin to the atom, and subtends an angle ω at the origin

$$\omega = 2 \tan^{-1}[(\tan 4^\circ)(1 - Z/150)],$$

where Z is the nuclear charge on the atom.

d. DATE1. Consists of the OS/360 assembly language routine that prints out the current calendar date.

3. Routines Common to More Than One Major Segment

a. ADAPT. Supervises symmetry-adaptation of arbitrary closed set of functions.

b. PSYM. Calculates matrix of P over either one or two sets of functions by the formula

$$\langle F_i | P | G_j \rangle = \sum_{v \in S} F_i(\mathbf{r}_v) G_j(\mathbf{r}_v).$$

A control parameter, KALC, determines the kind of computation: If $KALC < 0$, F_i and G_j are assumed to come from two different sets of functions, otherwise from the same set; if $|KALC| = 9$, a center of inversion is present and the matrix element is evaluated over only one atom of each inversion-symmetric pair.

c. THLMK. Computes value of set of $2L + 1$ real spherical harmonics with radial dependence at a grid point. The functions are centered at the origin of the master coordinate system and are defined as follows for $m \geq 1$:

$$FUNCT(1) \equiv S_{\ell,1} = (r\sqrt{2})^{-1} \theta_{\ell 0}(\theta);$$

$$FUNCT(2m) \equiv S_{\ell,2m} = r^{-1} \theta_{\ell m}(\theta) \sin(m\varphi)(-1)^m;$$

$$FUNCT(2m+1) \equiv S_{\ell,2m+1} = r^{-1} \theta_{\ell m}(\theta) \cos(m\varphi)(-1)^m.$$

For each value of m , a call is made to

d. THETAX. Evaluates the normalized associated Legendre polynomial $\theta_{\ell m}(\theta)$:

$$\theta_{\ell m}(\theta) = \sin^m \theta \sum_{r=0}^v C_{\ell m r} (\cos \theta)^{\ell-m-2r}, \quad v = \text{int}\left(\frac{\ell-m}{2}\right),$$

where the C_{lmr} were defined in subroutine ALPTAB. Special formulas are used for the special cases of $\sin \theta = 0$ and $\cos \theta = 0$.

e. MATPRT. Is a matrix output routine for square matrices of arbitrary size. The matrices are printed out six columns at a time, with an appropriate alphanumeric label for each row. If eigenvalues are associated with particular columns of the matrix, they are also output.

f. EIGEN. Diagonalizes a real, symmetric matrix using the Jacobi method. This method is used because no mixing is introduced among vectors belonging to the same degenerate eigenvalue if they are not connected in the original matrix. The eigenvectors are arranged in order of descending eigenvalue.

g. DEGEN. Determines the number of distinct eigenvalues and their degeneracies. Two eigenvalues are considered to be degenerate if the absolute value of their ratio differs from unity by less than CRTDEG (q.v.).

h. SORT1. Sorts eigenvalues and eigenvectors in ascending order of elements in an integer array associated with the matrix. Thus, eigenvectors may be sorted according to representation index, degeneracy, etc.

i. LINEUP. Determines whether two different sets of functions belong to the same irreducible representation, and, if so, rotates one set to align its phase with that of the other set. The phase transformation is carried out by the method of Sec. II.E.

j. CLEAN. Sorts out arbitrary mixing of multiply-occurring representatives of a given symmetry species, to give coefficients that are more easily interpretable. For each symmetry species, the number of sets of vectors belonging to it is determined, and a call is made to:

k. SCAN. Determines the member of a set with the smallest number of nonzero coefficients, and identifies the coefficients within a vector that differ at most by a sign.

l. WIG3J. Computes set of Wigner 3-j symbols for given ℓ values. Current version assumes $\ell_1 = \ell_2 = \text{even}$. The formula used is given on pp. 45-46 of Ref. 8.

m. SYMB3J. When ℓ_3 is odd, assigns the correct phase to the 3-j symbol.

4. Determination of Point-group Properties

a. GROUP. Is the supervisor for this section. On the basis of results of a preliminary call to PROSYM (q.v.) with $L = 4$, determines

lowest L-values of spherical harmonics spanning all irreducible representations of even (and odd) parity for the particular nuclear configuration. Calls PROSYM to obtain prototype functions, then PHASER to impose standard phase choice. Table II shows the L-values selected for each point group.

TABLE II. L-values Used in the Prototype Functions
for the Various Point Groups

L	Groups
0	(C ₁)
1	C _{2h} , C _i
2	D ₂ , C _{2v} , D _{2h} , C ₃ , C _s , C ₂
3	C _{4h} , D _{3d} , D _{2h} , C _{6h} , S ₆
4	D _{2d} , C ₃ , D ₃ , C _{3v} , D _{3d} , C _{3h} , C ₄ , D ₄ , S ₄ , C _{4v} , C _{4h} , D _{4h} , C ₆ , S ₆ , C _{6h} , C ₇ , C ₈ , S ₈
5	D _{4h} , D _{5d}
6	D _{3h} , C ₅ , D ₅ , C _{5v} , C _{5h} , D ₆ , D _{6h} , D _{6v} , T, T _h , T _d , O, O _h
7	-
8	D _{4d} , D _{8h}
9	D _{6h} , D _{8h} , O _h , T _h
10	D _{5h}
11	-
12	D _{6d} , K, K _h
13	-
14	-
15	K _h
16	D _{7h} , D _{8d}

b. PROSYM. Supervises determination of a symmetry-adapted set of spherical harmonics.

c. ANTSYM. Generates a set of even-parity, odd- ℓ spherical harmonics to be used in establishing the correspondence of the odd-parity (odd- ℓ) symmetry vectors with the even-parity (even- ℓ) ones.

The formulas used are as follows: Let us write $Y_{LM} = C_{LM} + iS_{LM}$. The functions C_{LM} and S_{LM} are then related to the $S_{L,K}$ by a transformation like that defined by Eqs. 29. The antisymmetric functions $\hat{S}_{L,K}$ are then given by

$$\hat{S}_{L',1} = \sum_{M_1=1}^L (-1)^{M_1+1} \left[S_{LM_1}^1 C_{LM_1}^2 - C_{LM_1}^1 S_{LM_1}^2 \right] \begin{pmatrix} L & L & L' \\ M_1 & -M_1 & 0 \end{pmatrix};$$

$$\hat{S}_{L',2M_3} = -\frac{1}{\sqrt{2}} \sum_{M_1=-L}^L \left[S_{LM_1}^1 S_{LM_2}^2 - C_{LM_1}^1 C_{LM_2}^2 \right] \begin{pmatrix} L & L & L' \\ M_1 & M_2 & -M_3 \end{pmatrix};$$

$$\hat{S}_{L',2M_3+1} = -\frac{1}{\sqrt{2}} \sum_{M_1=-L}^L \left[C_{LM_1}^1 S_{LM_2}^2 + S_{LM_1}^1 C_{LM_2}^2 \right] \begin{pmatrix} L & L & L' \\ M_1 & M_2 & -M_3 \end{pmatrix};$$

where $M_3 \geq 0$, $M_2 = M_3 - M_1$, and the superscripts indicate the grid on which the function is evaluated.

d. LINCOM. Calculates matrix of P' (i.e., over the second totally symmetric grid) over the crude symmetry vectors. Nonzero off-diagonal blocks identify sets of vectors belonging to the same symmetry species.

e. PHASER. Imposes a standard phase on all symmetry vectors. The standard phase for a given representation is taken as that evinced by the lowest L-value set of harmonics spanning that representation. If a parity distinction exists, the functions of opposite parity are given the same phase by means of an A_{1u} operator which couples them. Any arbitrary mixing of the final phased vectors is sorted out by subroutine CLEAN.

f. AUFUN. Multiplies an odd-parity spherical harmonic by the A_{1u} symmetry vector to change its parity.

5. Symmetry-adaptation of an Atomic Basis Set

a. BASIS. Constructs the necessary internal bookkeeping tables for the atomic basis set, which has been made available by:

b. INBAS. Generates an atomic basis set for the whole molecule, based either on input values or on table look-ups. Optimized minimal and double-zeta Slater-type basis functions are stored for H-Xe (valence shell only), and H-Kr, respectively. The value of LFLOW(4) controls the choice of stored basis set or the input of different values. (See Sec. IV.A below for description of input.)

c. ORBTLS. Supervises symmetry-adaptation of atomic basis set. Generates symmetrized functions for each distinct ℓ -value in each atom type, and sets up the necessary tables of information about the symmetry functions.

d. BASFUN. Evaluates a set of atomic basis functions at a particular grid point. For each atom-centered function, the coordinates of the grid point are transformed to the atom-centered axis system, and a call is made to:

e. VALUE. Evaluates the $2\ell + 1$ atomic basis functions at a grid point. The angular parts of the functions are given in real Cartesian representation for speed of computation, and an overall radial dependence of r^{-1} or r^{-2} is affixed. Functions up through $\ell = 4$ are implemented.

The formulas used are:

$$S_1 = R^{-2},$$

$$P_1 = ZR^{-2},$$

$$P_2 = YR^{-2},$$

$$P_3 = XR^{-2},$$

$$D_1 = \frac{1}{2}(3Z^2 - R^2) R^{-4},$$

$$D_2 = \sqrt{3}YZR^{-4},$$

$$D_3 = \sqrt{3}XZR^{-4},$$

$$D_4 = \sqrt{3}XYR^{-4},$$

$$D_5 = \frac{\sqrt{3}}{2}(X^2 - Y^2) R^{-4},$$

$$F_1 = Z(5Z^2 - 3R^2) R^{-4},$$

$$F_2 = \frac{\sqrt{6}}{2}Y(5Z^2 - R^2) R^{-4},$$

$$F_3 = \frac{\sqrt{6}}{2}X(5Z^2 - R^2) R^{-4},$$

$$F_4 = \sqrt{60}XYZR^{-4},$$

$$F_5 = \sqrt{15}Z(X^2 - Y^2) R^{-4},$$

$$F_6 = \frac{\sqrt{10}}{2}Y[3(R^2 - Z^2) - 4Y^2] R^{-4},$$

$$F_7 = \frac{\sqrt{10}}{2}X[4X^2 - 3(R^2 - Z^2)] R^{-4},$$

$$G_1 = \left(\frac{35}{2} Z^4 - 15Z^2 R^2 + \frac{3}{2} R^4 \right) R^{-6},$$

$$G_2 = \frac{\sqrt{10}}{2} YZ(14Z^2 - 6R^2) R^{-6},$$

$$G_3 = \frac{\sqrt{10}}{2} XZ(14Z^2 - 6R^2) R^{-6},$$

$$G_4 = \sqrt{20} XY(7Z^2 - R^2) R^{-6},$$

$$G_5 = \frac{\sqrt{20}}{2} (X^2 - Y^2)(7Z^2 - R^2) R^{-6},$$

$$G_6 = \sqrt{70} YZ[3(R^2 - Z^2) - 4Y^2] R^{-6},$$

$$G_7 = \sqrt{70} XZ[4X^2 - 3(R^2 - Z^2)] R^{-6},$$

$$G_8 = 4\sqrt{35} XY(2X^2 - R^2 + Z^2) R^{-6},$$

and

$$G_9 = \sqrt{35}[4X^2(X^2 - R^2 + Z^2) + (R^2 - Z^2)^2] R^{-6}.$$

6. Finite Point-group Vector-coupling Coefficients

a. VECUPL. Calculates vector-coupling coefficients for the finite point group, and supervises related operations.

For the "symmetric" vector-coupling coefficients, the formula is

$$\begin{aligned} (\Gamma'' \gamma'' | \Gamma \gamma \Gamma' \gamma') &= \eta \sum_m \sum_{m'} (-1)^{m+m'} \begin{pmatrix} \ell & \ell & \ell'' \\ m & m' & -m-m' \end{pmatrix} \\ &\times [(AA' - BB') A'' + (BA' + AB') B'']. \end{aligned}$$

For the "antisymmetric" coefficients, the last factor is replaced by

$$[(BA' + AB') A'' - (AA' - BB') B''],$$

where $A' \equiv A'_{\ell m'}$, etc., and the A's and B's are defined in Eqs. 29.

b. ODDGA. Constructs set of transformation coefficients for odd-L-value prototype functions.

c. ORTHOG. Performs Schmidt orthogonalization of doubly-occurring sets of coupling coefficients for the special cases of point groups T , T_h , K , and K_h .

d. VPRINT. Prints out tables of selection rules and coupling coefficients.

D. Definition of Variables in Common Blocks

1. Glossary

The following definitions are used in the program:

Angular quantum number: A combined index denoting the irreducible representation and the subspecies. Assigned sequentially from $A_{1g} = 1$.

Basis index: An index denoting a particular set of n , ℓ , and ζ values in the atomic basis set. There are $2\ell + 1$ times the number of atoms of that element functions belonging to a given basis index.

K-value: An integer, $1 \leq K \leq 2\ell + 1$, indexing the real basis functions belonging to an ℓ -value; $K = 2m$ indexes the function corresponding to $\sin(m\phi)$; $K = 2m + 1$ indexes the $\cos(m\phi)$ function.

Prototype function: A linear combination of real spherical harmonic functions which is symmetry-adapted and has a canonical phase.

Radial quantum number: An index denoting the particular occurrence of a symmetry species in the symmetry-adapted atomic basis set. That is, the third A_{1g} function would have a radial quantum number = 3, an angular quantum number = 1.

Transformation block: A set of symmetry transformation coefficients for atomic basis functions, characterized by a particular atom-type and ℓ -value.

Transformation index: An index given to the symmetry-adaptation of a particular set of atomic basis functions, i.e., a particular basis index on a particular atom-type. Since the transformation coefficients are determined by the ℓ -value of the functions and their atom-type, several transformation indices may refer to the same block of coefficients.

2. Blocks Containing Output from the Program

a. Block/OUT1/. Information about the nuclear geometry and chemical makeup of the molecule.

POS(I,IA): The Ith Cartesian coordinate for atom No, IA.

RADIUS(IA), THETA(IA), PHI(IA): The spherical polar coordinates for atom IA.

NATOMS: The number of atoms in the molecule.

NATYPE: The number of atom-types.

NUMT(IT): The number of atoms in atom-type IT.

NUCT(IT): The nuclear charge (or atomic number) of atoms in atom type IT.

ITSYMB(IT): The chemical symbol of atoms in type IT.

ITYPA(IN,IT): The loading index of the INth atom of atom-type IT.

ITYPT(IA): The atom-type to which atom IA belongs.

ITYPI(IA): The running index of atom IA within its atom-type.

INVAT(IA): The index of the atom inversion-symmetric with respect to atom IA (if a center of inversion is present).

b. Block/OUTB/. Information about the unsymmetrized atomic basis functions.

BCOEF(I): The self-consistent field coefficient of the Ith Slater-type orbital in the appropriate SCF atomic orbital basis vector.

ZETA(IB,IT): The orbital exponent of the function corresponding to basis index IB in atom-type IT.

NVALUE(IB,IT): The principal quantum number for basis index IB in atom-type IT.

LVALUE(IB,IT): The corresponding value of ℓ .

NBAST(IT): The number of basis indices (sets of r , ℓ , ζ values) on atoms of type IT.

ICRK(IT): The number of core electrons on atoms of type IT.

NASB(IT): The number of atomic symmetry blocks (distinct ℓ -values) for atom-type IT.

IBLKA(L+1,IT): The index of the block of transformation coefficients for the particular L-value in atom-type IT.

NBLK: The number of transformation blocks making up the array ASCOEF.

NBVB(IBLK): The number of basis vectors from functions in IBLK.

NBIB(IBLK): The number of basis indices represented in IBLK.

ITB(IBLK): The atom-type for the transformation block IBLK.

LVB(IBLK): The L-value characterizing the block IBLK.

IBIT(IB,IBLK): The basis index corresponding to running index IB within the block IBLK.

IBCFST(IBLK): The index in BCOEF of the first coefficient belonging to functions in transformation block IBLK.

NOR: The total number of orbitals in the basis set.

LAT(LAO): The index of the atom on which orbital LAO is centered.

IBT(LAO): The basis index of orbital LAO.

IKT(LAO): The k-value ($k = 1, 2, \dots, 2\ell + 1$: a sequential index numbering the real atomic orbitals of given ℓ) for orbital LAO.

LAOFST(IB,IA): The running index of the first member of basis index IB on atom IA.

c. Block/OUT2/. Information about the symmetry transformation of the atomic orbitals and about the new, symmetry-adapted functions.

ASCOEF(IARRAY): The symmetry-transformation coefficient indexed by IARRAY.

ITRANT(IB,IT): The transformation index corresponding to basis index IB of atom-type IT.

LAOT(IARUN,ITRANS): The index of the IARUNth orbital in the transformation block indexed by ITRANS [= ITRANT(IB,IT)].

ITRAN(ISO): The transformation index corresponding to symmetry-adapted basis function ISO.

ISRUNT(ISO): The running index of symmetry orbital ISO within its transformation index.

LANGT(ISO): The "angular quantum number" of symmetry orbital ISO.

IRADT(ISO): The corresponding "radial quantum number."

IFIRST(ITRANS): The index of the first coefficient in ASCOEF indexed by ITRANS.

ISOT(ISRUN,ITRANS): The index of the ISRUNth symmetry orbital in the transformation block ITRANS.

IFB(IBLK): The index in ASCOEF of the first coefficient in the transformation block IBLK.

ISOQ(ANG,IRAD): The symmetry orbital whose angular and radial quantum numbers are ANG and IRAD, respectively.

NRADA(IREP): The number of times the representation IREP occurs in the atomic basis set.

d. Block/OUT3/. Information about the point group to which the molecule belongs.

NREP: The number of irreducible representations in the point group.

NDEGA(IREP): The degeneracy of the irreducible representation IREP.

LANGA(I,IREP): The angular quantum number assigned to the Ith row of representation IREP.

IREPA(IANG): The representation to which angular quantum number IANG belongs.

IORDER: The order of the group.

LVAL1: The ℓ -value of the prototype functions that span all the even representations of the group.

LVAL2: The ℓ -value spanning all the odd representations. (Set = 0 if no center of inversion.)

IRTYPE(IREP): The Mulliken symbol (approximately) for the representation IREP.

IRPAR(IREP): The parity symbol (g or u) for the representation IREP.

IODD: Equals 1 if a center of inversion is present, 0 otherwise.

IRLOWL(IREP): Low ℓ -value spanning the representation IREP. Set = -1, if no low ℓ -value spans the representation.

e. Block/OUT4/. Information about the finite point-group vector-coupling coefficients.

VCCOEF(IVSUB): The normalized, unsymmetrized vector-coupling coefficient indexed by IVSUB.

IVCCB(IDPR,IR3): The index of the coefficient block referring to IDPR = IR2 + IR1 * (IR1 - 1)/2 and IR3, where IR2 $\leq$ IR1. If IVCCB(IDPR,IR3) = 0, IR3 is not contained in the direct product of IR1 x IR2.

IFVCC(IVCB): The address of the first coefficient in VCCOEF of the coefficient in VCCOEF of the coefficient block IVCB.

3. Blocks Containing Information Internal to the Program

a. Block/CONTRL/. Control parameters and thresholds.

CRTZRO: The zero threshold. A quantity A is set to zero if $|A| \leq$ CRTZRO. Default is 10^{-8} .

CRTDEG: The degeneracy criterion. A and B are considered degenerate if $\text{CRTDEG} < A/B < (\text{CRTDEG})^{-1}$. Default is 1.00000001.

ISTOP: Abnormal-termination flag. If program detects an error and retains control, ISTOP is set to 1.

INTER: Intermediate-printout flag. If INTER = 1, many intermediate results are printed out for diagnostic purposes.

LFLOW(I): If LFLOW(I) = 1, the Ith major program segment is executed. See input description (Sec IV.A below) for details.

b. Block/PROTO/. Information about the prototype functions spanning the irreducible representations of the group.

COPRO(ISUB): The transformation coefficient indexed by ISUB; coefficients are packed columnwise into COPRO.

EVP(IEV): The eigenvalue associated with the IEVth transformation vector; used only in output to show members of a degenerate set.

IREPL(I): The irreducible representation index assigned to the Ith eigenvector (or set of degenerate eigenvectors).

NEVALP(IPAR): The number of distinct eigenvalues in the set of prototype functions of parity IPAR (= 1 for g, 2 for u).

c. Block/GIRD/. Information about the points making up the domain of the totally symmetric operator P.

RGRID(LA,IGRID): The radial distance of points about atom LA and belonging to grid IGRID, from the origin.

CTHGR(IND): The cosine of the θ -coordinate of grid point IND, as measured from the master axis system.

PHIGR(IND): The ϕ -coordinate of grid point IND.

NGRID: The total number of points in each grid.

NDIV: The number of equally spaced points on each circle comprising the grid.

d. Block/XYZ/. Further information about the grid points.

XG(IND), YG(IND), ZG(IND): The Cartesian coordinates of grid point IND in the master coordinate system.

X(LA), Y(LA), Z(LA): The Cartesian coordinates of atoms LA in the master coordinate system. (These are identical with POS(I,LA), I = 1, 2, 3.)

e. Block/WORK/. A work area for use within major segments of the program. Content varies, but variables most frequently occurring are:

RES(I,J): A matrix containing the matrix elements of P over one or two sets of functions.

EV(I) or VALU(I): The Ith eigenvalue resulting from the diagonalization of RES.

C(I,J): The eigenvector coefficients that diagonalize RES.

B(I,J): A scratch array, same size as RES or C.

NDEG(K): The degeneracy of the Kth distinct eigenvalue of RES.

NEVAL: The number of distinct eigenvalues of RES.

IREPLB(K): The irreducible representation index assigned to the Kth set of eigenvectors.

LABEL, INDEX: Hollerith and integer arrays used to label rows of matrices printed out.

f. Block/PHASE/. Control variables for phasing of prototype functions.

IFLGG(I) (or IFLAG(I)): Flag set to 1 if representation I has been phased properly.

IPHASE: Control flag determining which functions are phased against which in subroutine LINEUP.

g. Block/ALP/. Parameter communication among routines PSYM, THLMK, AND THETAX.

RCR: Reciprocal of distance of current grid point from origin. SQ2OR: $RCR/\sqrt{2}$.

W (or X): (Cos θ) of current grid point.

Y: (Sin θ) of current grid point.

LC(or L): ℓ -value of current set of spherical harmonics.

h. Block/UNGRAD/. Information about A_{1u} transformation vector (if any).

CAU(I): Ith coefficient in A_{1u} transformation vector.

LODD: ℓ -value spanning the A_{1u} representation.

LODEG: $2 \cdot LODD + 1$.

i. Block/WIGNER/. 3-j symbols for set of ℓ -values.

S3J(MSUB1, M3P1): 3-j symbol $\begin{pmatrix} \ell & \ell & \ell' \\ m_1 & m_2 & m_3 \end{pmatrix}$, where ℓ' is even, $MSUB1 = |m_2 - m_1|/2 + 1$, $M3P1 = m_3 + 1$, and $m_3 \geq 0$, $m_1 \geq \frac{1}{2}m_3$.

Z3J(MSUB1, M3P1): 3-j symbol as above, except ℓ' is odd.

j. Block/WORK2/. Communication among routines ORBTLS, BASFUN, and VALUE.

XP or X } Cartesian coordinates of current grid point in an atom-centered
YP or Y } system.
ZP or Z }

RP or R: Distance of current grid point from current atom.

JSUB or ISUB: Index of last function before set on current atom.

LDEG: $2 \cdot L + 1$, where L is the ℓ -value of the current set of basis functions.

NUMB: The number of atoms in the current atom-type.

IT: The index of the current atom-type.

k. Block/ALP1/. Information that is invariant from molecule to molecule.

THETAC(ISUB): The coefficient $C_{\ell m r}$ for which ISUB equals $8m + 4\ell(\ell+1) + r + 1$.

FACT(N): The value of $(N-1)!$.

l. Block/FILES/. Data-set reference numbers.

INPUT: The logical unit number for input; usually = 5.

IOUT: The logical unit number for output; usually = 6.

m. Block/ASYMI/. Parameter communication between subroutines PROSYM and ANTSYM.

LA or L: The even ℓ -value used to form the odd- ℓ set of spherical harmonics. $LA = \max(2, \frac{1}{2}\ell_{\text{even}} + 1)$.

MGRID or NGRID: The total number of points in a grid.

L3P1: $L_{\text{odd}} + 1$.

LAP1 or LP1: $LA + 1$.

LADEG or LL: $2 \cdot LA + 1$.

n. Block/SCAN1/. Communication between subroutines CLEAN and SCAN.

IFLAG(K): A flag indicating that the Kth component of the column vector has already been examined $IFLAG(K) = 1$.

NSC(IC): The number of coefficients in the vector having the ICth distinct numerical absolute value.

SAMEC(IC): The value of the ICth distinct numerical magnitude in the vector.

JM: The index of the particular eigenvector being examined.

IC: The number of distinct numerical values counted in the vector.

IV. INSTRUCTIONS FOR USE

A. Description of Input

The following discussion applies to the IBM 360/75 version of the program, using the NAMELIST capability of IBM FORTRAN IV. Since the input is so brief and is handled entirely in subroutines MOLCUL and INBAS, the user who does not have the NAMELIST capability available will find it simple to write his own input format. The input is in three logical components:

Component A1. Problem identification and control parameters. The parameter card (or cards) is in NAMELIST format (see IBM OS/360 Manual C28-6515, "FORTRAN IV Language," for details), and the name of the list is PARAMS. Only those parameters for which the default option is not used need be specified.

Component A2. Cartesian coordinates of atoms and their chemical symbols. In NAMELIST format, with the list ATOM. Default options are provided. Each atom requires at least one card.

Component A3. Atomic basis function parameters. (Optional; used only if LFLOW(4) = 9.) In standard A, I, or F formats.

Table III gives a detailed description of the variable lists and their formats; Figs. 13 and 14 show typical input decks with and without user-supplied basis sets, respectively.

B. Built-in Basis Sets

A feature that was felt to be particularly useful in facilitating the use of the program is the prestoring of several standard, optimized atomic basis sets for each of a wide range of elements. This was done because in a large, multicenter calculation, complete optimization of the basis set will probably not be feasible and the "best atom" sets from the literature are likely to be taken as the starting point. Two sets have been implemented: a minimal, valence-shell basis set for elements hydrogen through xenon,

and a double-zeta set for elements hydrogen through krypton. Tables IV and V show the two types of basis sets we have stored, and give literature references.

Since the user may have his own input scheme for basis function parameters, the input or assignment routine is made completely separate from the generation of cross-reference tables needed for the symmetry-adaptation bookkeeping. Thus the input routine, INBAS, may be replaced by the user's own module, as long as the arrays (NBAST, CRCHRG, NVALUE, LVALUE, ZETA, and BCOEF) are properly filled and transmitted through COMMON/OUTB/.

TABLE III Description of Input Deck

Card No	Variable	Description and Format	Default Option
<u>Component A1 Job I.D. and Control</u>			
11	IDENTO1	Problem identification (MAG, A3)	None
	JOB	Run-case flag. An asterisk (*) in column 80. Job terminates if a slash (/) is encountered in column 80.	None
12		Parameters in NAMELIST format (nameist name = PARAMS)	
	NATOMS	Number of atoms in the molecule	None
	CRIZAO	Zero threshold	1.0×10^{-8}
	CRIDEG	Degeneracy criterion	1.0000001
	CRTTYP	Atom-type criterion. Two atoms of the same element are put into the same atom-type when $ R_i - R_j \leq \text{CRTTYP}$	1.0×10^{-5}
	NTCHK	The anticipated number of atom-types, used as a check on the input	1
	INTER	Intermediate printout flag. = 1 if desired 0 otherwise	0
	SCALE	Factor by which all Cartesian coordinates are multiplied	1.0
	NDIV	Number of equally spaced points on each grid circle	24
	LFLOWN10	Logical flow control array. A "*" in the appropriate element means that branch is taken	
	LFLOW11	Symmetry-adapted atomic basis set	1
	LFLOW12	Vibron-rotation symmetry analysis (not yet implemented)	1
	LFLOW13	Finite point-group vector-coupling coefficients	1
	LFLOW14	Choice of atomic basis set	1
		= 1 Stored minimal valence-shell Slater basis	
		= 2 Stored double-zeta Slater basis	
		= 9 Basis set read in by user	
	LFLOW15-LFLOW20	Available for further logic control	0
<u>Component A2 Atomic Coordinates</u>			
21 to 2 NATOMS		Each coordinate is expressed internally in the form $X = XOP(IX)$	
		Parameters in NAMELIST format (nameist name = ATOM)	
	ELMNT	The chemical symbol for the element (e.g., 'FE')	None
	X, Y, Z	The Cartesian coordinates, which may or may not be modified by operators	0.0
	XOP, YOP, ZOP	Operations, including 'SIN', 'COS', 'IP', 'SQRT', ' ' (no operations). If 'SIN' or 'COS' is used, the operand must be in degrees	' '
	FX, FY, FZ	The operands of XOP, YOP, and ZOP	1.0
<u>Component A3 Basis-set Input</u>			
			None
31	NASB(1)	The number of atomic symmetry blocks for atom-type 1	
	ICRKT(1)	The number of "core" electrons (21)	
32	LVB	The l -value of functions in the symmetry block	
	NB10	The number of sets of m, C_l values in the block	
	NBVB	The number of basis vectors formed from the primitive functions $\leq \text{NB10} (31)$	
33 to 3 NB10+2	NVALUE	The principal quantum number of the functions	
	ZETA	The orbital exponent. 1000 set to a card, (35 F10.5)	
3 NB10+3 to 3 NB10+2+NBVB	(BCN11, 1-1, NB10)	The basis-vector coefficients (Each vector begins a new card, BFO.00)	
		Repeat from card 32 for each atomic symmetry block in atom-type 1	
		Repeat from card 31 for each atom-type 1	


```

STAGGERED ETHANE - EXTENDED BASIS SET
CPARAMS NATOMS=8,NTCHEK=2,LFLOW(3)=0,LFLOW(4)=9 &END
&ATOM ELMNT='C',Z=1, &END
&ATOM ELMNT='H',Y=2.,Z=2. &END
&ATOM ELMNT='H',X=1.,XDP='SQRT',FX=3.,Y=-1.,Z=2. &END
&ATOM ELMNT='H',X=-1.,XDP='SQRT',FX=3.,Y=-1.,Z=2. &END
&ATOM ELMNT='C',Z=-1, &END
&ATOM ELMNT='H',Y=-2.,Z=-2. &END
&ATOM ELMNT='H',X=1.,XDP='SQRT',FX=3.,Y=1.,Z=-2. &END
&ATOM ELMNT='H',X=-1.,XDP='SQRT',FX=3.,Y=1.,Z=-2. &END
-- 2 0
-- 0 4 2
1 9.153
1 5.382
2 1.428
3 3.076
0.08307 0.47100 -0.00049 0.00373
-0.00604 -0.22225 0.77062 0.28311
1 3 1
2 5.157
2 2.177
2 1.150
0.02472 0.39516 0.64475
1 0
0 1 1
1 1.000
1.000

```

Fig. 13. Example of Input Deck Including a Basis Set Supplied by the User

```

ICOSAEDRAL H(12) SKELETON
CPARAMS NATOMS=12, LFLOW(4)=2, INTER=1 &END
&ATOM ELMNT='O',Z=1.0,ZDP='SQRT',FZ=1.25 &END
&ATOM ELMNT='B',Y=1.0,Z=0.5 &END
&ATOM ELMNT='B',X=1.0,XDP='SIN',FX=72.0,Y=1.0,YDP='COS',FY=72.0,Z=0.5 &END
&ATOM ELMNT='B',X=1.0,XDP='SIN',FX=144.0,Y=1.0,YDP='COS',FY=144.0,Z=0.5 &END
&ATOM ELMNT='B',X=-1.0,XDP='SIN',FX=144.0,Y=1.0,YDP='COS',FY=144.0,Z=0.5 &END
&ATOM ELMNT='B',X=-1.0,XDP='SIN',FX=72.0,Y=1.0,YDP='COS',FY=72.0,Z=0.5 &END
&ATOM ELMNT='B',Y=-1.0,Z=-0.5 &END
&ATOM ELMNT='B',X=-1.0,XDP='SIN',FX=108.0,Y=1.0,YDP='COS',FY=108.0,Z=-.5 &END
&ATOM ELMNT='B',X=-1.0,XDP='SIN',FX=36.0,Y=1.0,YDP='COS',FY=36.0,Z=-.5 &END
&ATOM ELMNT='B',X=1.0,XDP='SIN',FX=36.0,Y=1.0,YDP='COS',FY=36.0,Z=-.5 &END
&ATOM ELMNT='B',X=1.0,XDP='SIN',FX=108.0,Y=1.0,YDP='COS',FY=108.0,Z=-.5 &END
&ATOM ELMNT='B',Z=-1.0,ZDP='SQRT',FZ=1.25 &END

```

Fig. 14. Example of Input Deck with Basis Set Selected by the Program

TABLE IV. Prestored Minimal Valence-shell Atomic Basis Set

Element	n	l	ζ	Element	n	l	ζ
H	1	S	1.0000	Ge	4	S	2.0109
He	1	S	1.6875		3	D	5.4171
Li	2	S	0.640		4	P	1.6951
Be	2	S	0.956	As	4	S	2.2360
B	2	S	1.288		3	D	5.7928
	2	P	1.211		4	P	1.8623
C	2	S	1.623	Se	4	S	2.4394
	2	P	1.487		3	D	6.1590
N	2	S	1.924		4	P	2.0718
	2	P	1.917	Br	4	S	2.6382
O	2	S	2.246		3	D	6.5197
	2	P	2.227		4	P	2.2570
F	2	S	2.564	Kr	4	S	2.8289
	2	P	2.550		3	D	6.8753
Ne	2	S	2.879		4	P	2.4423
	2	P	2.879	Rb	5	S	0.9969
Na	3	S	0.8358	Sr	5	S	1.2141
Mg	3	S	1.1025	Y	5	S	1.2512
Al	3	S	1.3724		4	D	3.9896
	3	P	1.3552	Zr	5	S	1.2891
Si	3	S	1.6344		4	D	3.2679
	3	P	1.4284	Nb	5	S	1.3392
P	3	S	1.8806		4	D	3.0796
	3	P	1.6288	Mo	5	S	1.3952
S	3	S	2.1223		4	D	3.1110
	3	P	1.8273	Tc	5	S	1.4453
Cl	3	S	2.3561		4	D	3.2205
	3	P	2.0387	Ru	5	S	1.4905
Ar	3	S	2.5856		4	D	3.3470
	3	P	2.2547	Rh	5	S	1.5286
K	4	S	0.8738		4	D	3.4937
Ca	4	S	1.0995	Pd	5	S	1.5675
Sc	4	S	1.1581		4	D	3.6476
	3	D	2.3733	Ag	5	S	1.6057
Ti	4	S	1.2042		4	D	3.8064
	3	D	2.7138	Cd	5	S	1.6384
V	4	S	1.2453		4	D	3.9692
	3	D	2.9943	In	5	S	1.9023
Cr	4	S	1.2833		4	D	4.2354
	3	D	3.2522		5	P	1.6940
Mn	4	S	1.3208	Sn	5	S	2.1257
	3	D	3.5094		4	D	4.4925
Fe	4	S	1.3585		5	P	1.8204
	3	D	3.7266	Sb	5	S	2.3222
Co	4	S	1.3941		4	D	4.7436
	3	D	3.9518		5	P	1.9989
Ni	4	S	1.4277	Te	5	S	2.5076
	3	D	4.1765		4	D	4.9900
Cu	4	S	1.4606		5	P	2.1617
	3	D	4.4002	I	5	S	2.6807
Zn	4	S	1.4913		4	D	5.2335
	3	D	4.6261		5	P	2.3223
Ga	4	S	1.7667	Xe	5	S	2.8436
	3	D	5.0311		4	D	5.4733
	4	P	1.5554		5	P	2.4849

References:

H-Ne: P. Bagus and T. L. Gilbert, unpublished work.

Na-Xe: E. Clementi et al., J. Chem. Phys. **38**, 2686 (1963); J. Chem. Phys. **47**, 1300 (1967).

TABLE V. Examples of Double-zeta Slater Basis Sets

n	l	ζ	SCF Coefficients			
			C_1	C_2	C_3	C_4
<u>Kr</u>						
1	0	28.9410	0.30191	-0.47092	0.18287	-0.06059
1	0	37.8590	0.71223	0.02751	-0.00684	0.00387
2	0	15.5500	0.02495	1.38570	-0.96946	0.31961
2	0	18.8680	-0.03566	-0.27267	0.36941	-0.12187
3	0	6.5065	0.00393	-0.03026	0.73304	-0.37484
3	0	8.3700	-0.00544	0.06030	0.41654	-0.10581
4	0	2.4358	0.00025	-0.00133	-0.00322	0.70702
4	0	4.0461	-0.00075	0.00422	0.02389	0.45131
2	1	15.4600	0.93533	-0.42145	0.11673	
2	1	29.2240	0.05118	-0.01096	0.00227	
3	1	8.1315	0.07231	0.44406	-0.14806	
3	1	6.0579	-0.03686	0.64246	-0.18834	
4	1	1.8095	-0.00221	-0.00943	0.57945	
4	1	3.2184	0.00572	0.00309	0.54963	
3	2	5.1816	0.70696			
3	2	10.3260	0.37915			
<u>F</u>						
1	0	8.3660	1.06848	-0.24322		
2	0	10.8850	-0.07301	-0.02281		
2	0	2.6270	0.00379	1.03628		
2	1	4.1800	0.35507			
2	1	1.8480	0.72760			

References

- He, Na-Ar: Clementi *et al.*, J. Chem. Phys. **40**, 1944 (1964).
 Li-Ne: Bagus and Gilbert, unpublished work.
 K-Kr: Clementi *et al.*, J. Chem. Phys. **47**, 1865 (1967).

C. Sample Input and Program Output

Table VI summarizes the input and output for the program. Figures 13-15 show input decks and sample output.

TABLE VI Summary of Input and Output for the Computer Program as Currently Implemented

<u>Input</u>	
Chemical symbols and Cartesian coordinates of atoms	
Parameters describing basis set (either read in or selected from the tables stored in program)	
Control parameters (or use of default options)	
<u>Output</u>	
Characterization of all symmetry species in point group	
Identification of point group	
Set of symmetry-adapted prototype functions	
Transformation coefficients for converting atomic basis set to symmetry-adapted set	
Labels and degeneracies for symmetry species spanned by basis set	
Internal bookkeeping information	
Selection rules for direct products of representations	
Finite point-group vector-coupling coefficients	

D. Table Limits and Error Messages

The following are the limits of various variables according to the current implementation of the program:

No. of atoms ≤ 30
Total No. of basis functions ≤ 120
No. of atom-types ≤ 10
No. of transformation blocks ≤ 50
Maximum matrix size $\leq 50 \times 50$
No. of irreducible representations ≤ 14
No. of ℓ -values per atom-type ≤ 5
No. of basis vector coefficients ≤ 500
No. of symmetry transformation coefficients ≤ 3000
 ℓ -value of basis function ≤ 4
No. of points per grid ≤ 720
 ℓ -value of prototype functions ≤ 16

Due principally to the large matrices provided for, the program requires roughly 388K bytes of core storage without overlays. If space is a consideration, of course, the matrix sizes can be reduced.

Several error messages, whose content should be self-explanatory, are included in the program for debugging purposes and should not arise under normal operation, unless a mistake has been made in the input. See the program listings for the specific messages.

**** GPTHEORY - AUTOMATED MOLECULAR POINT GROUP THEORY ****
 THOMAS D. BOUMAN AND GORDON L. GOODMAN
 CHEMISTRY DIVISION, ARGONNE NATIONAL LABORATORY
 ARGONNE, ILLINOIS 60439

PROBLEM TITLE -- KRYPTON HEXAFLUORIDE, DOUBLE-ZETA BASIS SET

9/14/71

INPUT DESCRIPTION OF MOLECULE

NUMBER OF ATOMS = 7
 ASSUMED NO. OF EQUIVALENT SETS (ATOM TYPES) = 2
 CARTESIAN COORDINATES (IN FORM $X = A*FUNC(B)$)

ATOM NUMBER	CHEMICAL SYMBOL	A	X FUNC	B	A	Y FUNC	B	A	Z FUNC	B
1	KR	0.0		1.000	0.0		1.000	0.0		1.000
2	F	1.0000		1.000	0.0		1.000	0.0		1.000
3	F	0.0		1.000	1.0000		1.000	0.0		1.000
4	F	0.0		1.000	0.0		1.000	1.0000		1.000
5	F	-1.0000		1.000	0.0		1.000	0.0		1.000
6	F	0.0		1.000	-1.0000		1.000	0.0		1.000
7	F	0.0		1.000	0.0		1.000	-1.0000		1.000

THRESHOLDS AND CONTROL PARAMETERS

ZERO THRESHOLD = 1.000-08
 DEGENERACY TEST (RATIO OF EIGENVALUES) = 1.00000001
 ATOM EQUIVALENCE THRESHOLD = 1.000-05
 NO. OF GRID POINTS PER ATOM = 24
 SCALE FACTOR MULTIPLYING COORDINATES = 10.00000
 PROGRAM OPTIONS SELECTED -
 POINT GROUP PROPERTIES
 SYMMETRY-ADAPTED ATOMIC BASIS SET
 DOUBLE-ZETA SLATER BASIS
 VIBRATION-ROTATION SYMMETRY
 VECTOR-COUPLING COEFFICIENTS

Fig. 15. Sample Output from the Program for an Octahedral KrF_6 Molecule, quantized about the Fourfold Axis

MOLECULAR GEOMETRY ANALYSIS

ATOM NUMBER	CHEMICAL SYMBOL	ATOMIC NUMBER (Z)	CARTESIAN COORDINATES IN CENTER OF CHARGE SYSTEM		
			X	Y	Z
1	KR	36	0.0	0.0	0.0
2	F	9	1.00000000	0.0	0.0
3	F	9	0.0	1.00000000	0.0
4	F	9	0.0	0.0	1.00000000
5	F	9	-1.00000000	0.0	0.0
6	F	9	0.0	-1.00000000	0.0
7	F	9	0.0	0.0	-1.00000000

ATOM NUMBER	CHEMICAL SYMBOL	ATOMIC NUMBER (Z)	SPHERICAL COORDINATES IN CENTER OF CHARGE SYSTEM		
			R	THETA	PHI
1	KR	36	0.0	0.0	0.0
2	F	9	1.00000000	1.57079633	0.0
3	F	9	1.00000000	1.57079633	1.57079633
4	F	9	1.00000000	0.0	0.0
5	F	9	1.00000000	1.57079633	3.14159265
6	F	9	1.00000000	1.57079633	4.71238898
7	F	9	1.00000000	3.14159265	0.0

THE PROGRAM DISTINGUISHES 2 ATOM TYPES, AS FOLLOWS...

ATOM-TYPE INDEX	CHEMICAL SYMBOL	NO. OF ATOMS	NUCLEAR CHARGE	UPPER : LOADING INDICES OF THE ATOMS IN THIS ATOM-TYPE LOWER : RUNNING INDICES WITHIN THE ATOM-TYPE						
1	KR	1	36	1						
2	F	6	9	2	3	4	5	6	7	
				1	2	3	4	5	6	

CENTER OF INVERSION - YES

SUMMARY OF POINT GROUP PROPERTIES

NO. OF IRREDUCIBLE REPRESENTATIONS = 10

COMPLEX-CONJUGATE PAIRS OF I.R.'S. = NO

ORDER OF THE GROUP = 48

THE POINT GROUP FOR THIS NUCLEAR CONFIGURATION IS O_h

L VALUES OF REAL SPHERICAL HARMONICS USED TO SPAN ALL THE I.R.'S IN THIS GROUP = 6 9

REPRESENTATION INDEX	DEGENERACY	SPECIES LABEL	PHASE STANDARD	ANGULAR QUANTUM NUMBERS		
1	1	A ₁ (G)	L = 0	1		
2	1	A ₂ (G)	(L = 3)	2		
3	2	E(G)	L = 2	3	4	
4	3	T ₁ (G)	L = 2	5	6	7
5	3	T ₂ (G)	(L = 1)	8	9	10
6	1	A ₁ (U)	(L = 0)	11		
7	1	A ₂ (U)	L = 3	12		
8	2	E(U)	(L = 2)	13	14	
9	3	T ₁ (U)	(L = 2)	15	16	17
10	3	T ₂ (U)	L = 1	18	19	20

ATOMIC BASIS SET AND SYMMETRY ADAPTATION

TRANSFORMATION BLOCK 1

ATOM-TYPE 1, S FUNCTIONS FOR ELEMENT KR

TOTAL OF 1 BASIS FUNCTIONS FOR EACH BASIS INDEX

BLOCK OF SYMMETRY TRANSFORMATION COEFFICIENTS STARTS AT LOCATION 1 OF ASCOEF

BASIS INDEX	TRANSF. INDEX	QUANTUM NUMBERS N ZETA	BASIS VECTORS			
			C(1)	C(2)	C(3)	C(4)
1	1	1 28.9410	0.30191	-0.47092	0.18287	-0.06059
2	2	1 37.8590	0.71223	0.02751	-0.00684	0.00387
3	3	2 15.5500	0.02495	1.38570	-0.96946	0.31961
4	4	2 18.8680	-0.03566	-0.27267	0.36941	-0.12187
5	5	3 6.5065	0.00393	-0.03026	0.73304	-0.37484
6	6	3 8.3700	-0.00544	0.06030	0.41654	-0.10581
7	7	4 2.4358	0.00025	-0.00133	-0.00322	0.70702
8	8	4 4.0461	-0.00075	0.00422	0.02389	0.45131

Fig. 15 (Contd.)

TRANSFORMATION BLOCK 2

ATOM-TYPE 1, P FUNCTIONS FOR ELEMENT KR

TOTAL OF 3 BASIS FUNCTIONS FOR EACH BASIS INDEX

BLOCK OF SYMMETRY TRANSFORMATION COEFFICIENTS STARTS AT LOCATION 2 OF ASCOEF

BASIS INDEX	TRANSF. INDEX	QUANTUM NUMBERS N ZETA	BASIS VECTORS C(1) C(2) C(3)
9	9	2 15.4600	0.93533 -0.42145 0.11673
10	10	2 29.2240	0.05118 -0.01096 0.00727
11	11	3 8.1315	0.07231 0.44406 -0.14806
12	12	3 6.0579	-0.03686 0.64246 -0.18834
13	13	4 1.8095	-0.00221 -0.00943 0.57946
14	14	4 3.2184	0.00572 0.00309 0.54963

TRANSFORMATION BLOCK 3

ATOM-TYPE 1, D FUNCTIONS FOR ELEMENT KR

TOTAL OF 5 BASIS FUNCTIONS FOR EACH BASIS INDEX

BLOCK OF SYMMETRY TRANSFORMATION COEFFICIENTS STARTS AT LOCATION 11 OF ASCOEF

BASIS INDEX	TRANSF. INDEX	QUANTUM NUMBERS N ZETA	BASIS VECTORS C(1)
15	15	3 5.1816	0.70696
16	16	3 10.3260	0.37915

TRANSFORMATION BLOCK 4

ATOM-TYPE 2, S FUNCTIONS FOR ELEMENT F

TOTAL OF 6 BASIS FUNCTIONS FOR EACH BASIS INDEX

BLOCK OF SYMMETRY TRANSFORMATION COEFFICIENTS STARTS AT LOCATION 36 OF ASCOEF

BASIS INDEX	TRANSF. INDEX	QUANTUM NUMBERS N ZETA	BASIS VECTORS C(1) C(2)
1	17	1 8.3660	1.06848 -0.24322
2	18	2 10.8850	-0.07301 -0.02281
3	19	2 2.6270	0.00379 1.03628

Fig. 15 (Contd.)

TRANSFORMATION BLOCK 5

ATOM-TYPE 2, P FUNCTIONS FOR ELEMENT F

TOTAL OF 18 BASIS FUNCTIONS FOR EACH BASIS INDEX

BLOCK OF SYMMETRY TRANSFORMATION COEFFICIENTS STARTS AT LOCATION 72 OF ASCOEF

BASIS INDEX	TRANSF. INDEX	QUANTUM NUMBERS M ZETA	BASIS VECTORS C(1)
4	20	2 4.1800	0.35507
5	21	2 1.8420	0.72760

FINAL SYMMETRY VECTORS FOR ATOM-TYPE 1, L = 0

COLUMNS 1 TO 1

EIGENVALUES

2.17594403180 02

EIGENVECTORS

S (1) 1.00000000000 00

ANGULAR QUANTUM NUMBER ASSIGNED TO EACH SYMMETRY VECTOR

1

FINAL SYMMETRY VECTORS FOR ATOM-TYPE 1, L = 1

COLUMNS 1 TO 3

EIGENVALUES

5.90043256960 01 5.90043256960 01 5.90043256960 01

EIGENVECTORS

Z (1)	1.00000000000 00	0.0	0.0
Y (1)	0.0	1.00000000000 00	0.0
X (1)	0.0	0.0	1.00000000000 00

ANGULAR QUANTUM NUMBER ASSIGNED TO EACH SYMMETRY VECTOR

18 19 20

Fig. 15 (Contd.)

FINAL SYMMETRY VECTORS FOR ATOM-TYPE 1, L = 2

COLUMNS 1 TO 5

EIGENVALUES

	1.07398349170 02	1.07398349170 02	9.32568278960-01	9.32568278960-01	9.32568278960-01
--	------------------	------------------	------------------	------------------	------------------

EIGENVECTORS

3Z2-1 (1)	1.00000000000 00	0.0	0.0	0.0	0.0
Y2 (1)	0.0	0.0	1.00000000000 00	0.0	0.0
X2 (1)	0.0	0.0	0.0	1.00000000000 00	0.0
XY (1)	0.0	0.0	0.0	0.0	1.00000000000 00
X2-Y2 (1)	0.0	1.00000000000 00	0.0	0.0	0.0

ANGULAR QUANTUM NUMBER ASSIGNED TO EACH SYMMETRY VECTOR

3 4 5 6 7

FINAL SYMMETRY VECTORS FOR ATOM-TYPE 2, L = 0

COLUMNS 1 TO 6

EIGENVALUES

	1.40685306330 05	1.28764295680 05	1.28764295680 05	1.30711485040 05	1.30711485040 05	1.30711485040 05
--	------------------	------------------	------------------	------------------	------------------	------------------

EIGENVECTORS

S (1)	4.08248290460-01	-2.88675134590-01	5.00000000000-01	0.0	0.0	7.07106781190-01
S (2)	4.08248290460-01	-2.88675134590-01	-5.00000000000-01	0.0	7.07106781190-01	0.0
S (3)	4.08248290460-01	5.77350269190-01	0.0	7.07106781190-01	0.0	0.0
S (4)	4.08248290460-01	-2.88675134590-01	5.00000000000-01	0.0	0.0	-7.07106781190-01
S (5)	4.08248290460-01	-2.88675134590-01	-5.00000000000-01	0.0	-7.07106781190-01	0.0
S (6)	4.08248290460-01	5.77350269190-01	0.0	-7.07106781190-01	0.0	0.0

ANGULAR QUANTUM NUMBER ASSIGNED TO EACH SYMMETRY VECTOR

1 3 4 18 19 20

Fig. 15 (Contd.)

FINAL SYMMETRY VECTORS FOR ATOM-TYPE 2, 1 = 1

COLUMNS 1 TO 6

EIGENVALUES

	2.46743136650 03	1.11948183790 03	1.11968183790 03	2.35581503810 02	2.35581503810 02	2.35581503810 02
EIGENVECTORS						
Z (1)	0.0	0.0	0.0	0.0	5.00000000000-01	0.0
Y (1)	0.0	0.0	0.0	0.0	0.0	5.00000000000-01
X (1)	-4.08248290460-01	2.88675134590-01	-5.00000000000-01	0.0	0.0	0.0
Z (2)	0.0	0.0	0.0	5.00000000000-01	0.0	0.0
Y (2)	-4.08248290460-01	2.88675134590-01	5.00000000000-01	0.0	0.0	0.0
X (2)	0.0	0.0	0.0	0.0	0.0	5.00000000000-01
Z (3)	-4.08248290460-01	-5.77350269190-01	0.0	0.0	0.0	0.0
Y (3)	0.0	0.0	0.0	5.00000000000-01	0.0	0.0
X (3)	0.0	0.0	0.0	0.0	5.00000000000-01	0.0
Z (4)	0.0	0.0	0.0	0.0	-5.00000000000-01	0.0
Y (4)	0.0	0.0	0.0	0.0	0.0	-5.00000000000-01
X (4)	4.08248290460-01	-2.88675134590-01	5.00000000000-01	0.0	0.0	0.0
Z (5)	0.0	0.0	0.0	-5.00000000000-01	0.0	0.0
Y (5)	4.08248290460-01	-2.88675134590-01	-5.00000000000-01	0.0	0.0	0.0
X (5)	0.0	0.0	0.0	0.0	0.0	-5.00000000000-01
Z (6)	4.08248290460-01	5.77350269190-01	0.0	0.0	0.0	0.0
Y (6)	0.0	0.0	0.0	-5.00000000000-01	0.0	0.0
X (6)	0.0	0.0	0.0	0.0	-5.00000000000-01	0.0

Fig. 15 (Contd.)

FINAL SYMMETRY VECTORS FOR ATOM-TYPE 2, L = 1

COLUMNS 7 TO 12

EIGENVALUES

2.21967664230 02 2.21967664230 02 2.21967664230 02 2.25324194780 02 2.25324194780 02 2.25324194780 02

EIGENVECTORS

Z (1)	0.0	-5.00000000000-01	0.0	0.0	0.0	-5.00000000000-01
Y (1)	5.00000000000-01	0.0	0.0	0.0	5.00000000000-01	0.0
X (1)	0.0	0.0	0.0	0.0	0.0	0.0
Z (2)	0.0	0.0	5.00000000000-01	0.0	0.0	5.00000000000-01
Y (2)	0.0	0.0	0.0	0.0	0.0	0.0
X (2)	-5.00000000000-01	0.0	0.0	-5.00000000000-01	0.0	0.0
Z (3)	0.0	0.0	0.0	0.0	0.0	0.0
Y (3)	0.0	0.0	-5.00000000000-01	0.0	-5.00000000000-01	0.0
X (3)	0.0	5.00000000000-01	0.0	5.00000000000-01	0.0	0.0
Z (4)	0.0	5.00000000000-01	0.0	0.0	0.0	-5.00000000000-01
Y (4)	-5.00000000000-01	0.0	0.0	0.0	5.00000000000-01	0.0
X (4)	0.0	0.0	0.0	0.0	0.0	0.0
Z (5)	0.0	0.0	-5.00000000000-01	0.0	0.0	5.00000000000-01
Y (5)	0.0	0.0	0.0	0.0	0.0	0.0
X (5)	5.00000000000-01	0.0	0.0	-5.00000000000-01	0.0	0.0
Z (6)	0.0	0.0	0.0	0.0	0.0	0.0
Y (6)	0.0	0.0	5.00000000000-01	0.0	-5.00000000000-01	0.0
X (6)	0.0	-5.00000000000-01	0.0	5.00000000000-01	0.0	0.0

Fig. 15 (Contd.)

FINAL SYMMETRY VECTORS FOR ATOM-TYPE 2, L = 1

COLUMNS 13 TO 18

EIGENVALUES

	1.19621623810 03	1.19621623810 03	1.19621623810 03	2.27001086590 02	2.27001086590 02	2.27001086590 02
--	------------------	------------------	------------------	------------------	------------------	------------------

EIGENVECTORS

Z (1)	5.000000000000-01	0.0	0.0	0.0	0.0	0.0
Y (1)	0.0	5.000000000000-01	0.0	0.0	0.0	0.0
X (1)	0.0	0.0	0.0	0.0	0.0	7.07106781190-01
Z (2)	5.000000000000-01	0.0	0.0	0.0	0.0	0.0
Y (2)	0.0	0.0	0.0	0.0	7.07106781190-01	0.0
X (2)	0.0	0.0	5.000000000000-01	0.0	0.0	0.0
Z (3)	0.0	0.0	0.0	7.07106781190-01	0.0	0.0
Y (3)	0.0	5.000000000000-01	0.0	0.0	0.0	0.0
X (3)	0.0	0.0	5.000000000000-01	0.0	0.0	0.0
Z (4)	5.000000000000-01	0.0	0.0	0.0	0.0	0.0
Y (4)	0.0	5.000000000000-01	0.0	0.0	0.0	0.0
X (4)	0.0	0.0	0.0	0.0	0.0	7.07106781190-01
Z (5)	5.000000000000-01	0.0	0.0	0.0	0.0	0.0
Y (5)	0.0	0.0	0.0	0.0	7.07106781190-01	0.0
X (5)	0.0	0.0	5.000000000000-01	0.0	0.0	0.0
Z (6)	0.0	0.0	0.0	7.07106781190-01	0.0	0.0
Y (6)	0.0	5.000000000000-01	0.0	0.0	0.0	0.0
X (6)	0.0	0.0	5.000000000000-01	0.0	0.0	0.0

ANGULAR QUANTUM NUMBER ASSIGNED TO EACH SYMMETRY VECTOR

1	3	4	5	6	7	8	9	10	15	16	17	18	19	20	18	19	20
---	---	---	---	---	---	---	---	----	----	----	----	----	----	----	----	----	----

NO. OF TIMES EACH REPRESENTATION OCCURS

13	0	7	4	2	0	0	0	2	13
----	---	---	---	---	---	---	---	---	----

Fig. 15 (Contd.)

SELECTION RULES FOR DIRECT PRODUCTS OF IRREDUCIBLE REPRESENTATIONS

REP.1 X REP.2		REP.3 -->				
		A ₁ B ₁ (G) 1	A ₁ B ₂ (G) 2	E(G) 3	T(G) 4	T ₂ (G) 5
1(A ₁ B ₁ (G)) X 1(A ₁ B ₁ (G))		1	0	0	0	0
2(A ₁ B ₁ (G)) X 1(A ₁ B ₁ (G))		0	1	0	0	0
2(A ₁ B ₁ (G)) X 2(A ₁ B ₁ (G))		1	0	0	0	0
3(E(G)) X 1(A ₁ B ₁ (G))		0	0	1	0	0
3(E(G)) X 2(A ₁ B ₁ (G))		0	0	-1	0	0
3(E(G)) X 3(E(G))		1	-1	1	0	0
4(T ₁ (G)) X 1(A ₁ B ₁ (G))		0	0	0	1	0
4(T ₁ (G)) X 2(A ₁ B ₁ (G))		0	0	0	0	1
4(T ₁ (G)) X 3(E(G))		0	0	0	1	1
4(T ₁ (G)) X 4(T ₁ (G))		1	0	1	1	-1
5(T ₂ (G)) X 1(A ₁ B ₁ (G))		0	0	0	0	1
5(T ₂ (G)) X 2(A ₁ B ₁ (G))		0	0	0	1	0
5(T ₂ (G)) X 3(E(G))		0	0	0	1	1
5(T ₂ (G)) X 4(T ₁ (G))		0	1	1	-1	1
5(T ₂ (G)) X 5(T ₂ (G))		1	0	1	1	-1

VECTOR COUPLING COEFFICIENTS

REPRESENTATIONS 1 X 1 --> 1

1ANG3 -->

1ANG1	1ANG2	1
1	1	1.000000000

Fig. 15 (Contd.)

```

REPRESENTATIONS 2 X 2 --> 1
                                IANG3 -->
    IANG1  IANG2                1
    2      2                    1.0000000000

```

```

REPRESENTATIONS 3 X 3 --> 1
                                IANG3 -->
    IANG1  IANG2                1
    3      3                    -0.707106781
    3      4                     0.0
    4      3                     0.0
    4      4                    -0.707106781

```

```

REPRESENTATIONS 4 X 4 --> 1
                                IANG3 -->
    IANG1  IANG2                1
    5      5                    -0.577350269
    5      6                     0.0
    5      7                     0.0
    6      5                     0.0
    6      6                    -0.577350269
    6      7                     0.0
    7      5                     0.0
    7      6                     0.0
    7      7                    -0.577350269

```

```

REPRESENTATIONS 5 X 5 --> 1
                                IANG3 -->
    IANG1  IANG2                1
    8      8                    0.577350269
    8      9                     0.0
    8     10                     0.0
    9      8                     0.0
    9      9                    0.577350269
    9     10                     0.0
   10      8                     0.0
   10      9                     0.0
   10     10                    0.577350269

```

Fig. 15 (Contd.)

REPRESENTATIONS 2 X 1 --> 2

IANG3 -->

IANG1	IANG2	2
2	1	1.000000000

REPRESENTATIONS 3 X 3 --> 2

IANG3 -->

IANG1	IANG2	2
3	3	0.0
3	4	0.707106781
4	3	-0.707106781
4	4	0.0

REPRESENTATIONS 5 X 4 --> 2

IANG3 -->

IANG1	IANG2	2
8	5	0.0
8	6	0.0
8	7	0.577350269
9	5	0.0
9	6	0.577350269
9	7	0.0
10	5	0.577350269
10	6	0.0
10	7	0.0

REPRESENTATIONS 3 X 1 --> 3

IANG3 -->

IANG1	IANG2	3	4
3	1	-1.000000000	0.0
4	1	0.0	-1.000000000

Fig. 15 (Contd.)

REPRESENTATIONS 3 X 2 --> 3

IANG3 -->

IANG1	IANG2	3	4
3	2	0.0	1.000000000
4	2	-1.000000000	0.0

REPRESENTATIONS 3 X 3 --> 3

IANG3 -->

IANG1	IANG2	3	4
3	3	-0.707106781	0.0
3	4	0.0	0.707106781
4	3	0.0	0.707106781
4	4	0.707106781	0.0

REPRESENTATIONS 4 X 4 --> 3

IANG3 -->

IANG1	IANG2	3	4
5	5	-0.408248290	0.707106781
5	6	0.0	0.0
5	7	0.0	0.0
6	5	0.0	0.0
6	6	-0.408248290	-0.707106781
6	7	0.0	0.0
7	5	0.0	0.0
7	6	0.0	0.0
7	7	0.816496581	0.0

REPRESENTATIONS 5 X 4 --> 3

IANG3 -->

IANG1	IANG2	3	4
8	5	0.0	0.0
8	6	0.0	0.0
8	7	0.0	0.816496581
9	5	0.0	0.0
9	6	0.707106781	-0.408248290
9	7	0.0	0.0
10	5	-0.707106781	-0.408248290
10	6	0.0	0.0
10	7	0.0	0.0

Fig. 15 (Contd.)

REPRESENTATIONS 5 X 5 --> 3

IANG3 -->

IANG1	IANG2	3	4
8	8	0.816496581	0.0
8	9	0.0	0.0
8	10	0.0	0.0
9	8	0.0	0.0
9	9	-0.408248290	-0.707106781
9	10	0.0	0.0
10	8	0.0	0.0
10	9	0.0	0.0
10	10	-0.408248290	0.707106781

REPRESENTATIONS 4 X 3 --> 4

IANG3 -->

IANG1	IANG2	5	6	7
5	1	-1.000000000	0.0	0.0
6	1	0.0	-1.000000000	0.0
7	1	0.0	0.0	-1.000000000

REPRESENTATIONS 4 X 3 --> 4

IANG3 -->

IANG1	IANG2	5	6	7
5	3	-0.500000000	0.0	0.0
5	4	0.866025404	0.0	0.0
6	3	0.0	-0.500000000	0.0
6	4	0.0	-0.866025404	0.0
7	3	0.0	0.0	1.000000000
7	4	0.0	0.0	0.0

REPRESENTATIONS 4 X 4 --> 4

IANG3 -->

IANG1	IANG2	5	6	7
5	5	0.0	0.0	0.0
5	6	0.0	0.0	-0.707106781
5	7	0.0	-0.707106781	0.0
6	5	0.0	0.0	-0.707106781
6	6	0.0	0.0	0.0
6	7	-0.707106781	0.0	0.0
7	5	0.0	-0.707106781	0.0
7	6	-0.707106781	0.0	0.0
7	7	0.0	0.0	0.0

Fig. 15 (Contd.)

REPRESENTATIONS 5 X 2 --> 4

[ANG3 -->

[ANG1	[ANG2	5	6	7
8	2	0.0	0.0	1.000000009
9	2	0.0	1.000000000	0.0
10	2	1.000000000	0.0	0.0

REPRESENTATIONS 5 X 3 --> 4

[ANG3 -->

[ANG1	[ANG2	5	6	7
8	3	0.0	0.0	0.0
8	4	0.0	0.0	1.000000000
9	3	0.0	0.866025404	0.0
9	4	0.0	-0.500000000	0.0
10	3	-0.866025404	0.0	0.0
10	4	-0.500000000	0.0	0.0

REPRESENTATIONS 5 X 4 --> 4

[ANG3 -->

[ANG1	[ANG2	5	6	7
8	5	0.0	0.707106781	0.0
8	6	-0.707106781	0.0	0.0
8	7	0.0	0.0	0.0
9	5	0.0	0.0	-0.707106781
9	6	0.0	0.0	0.0
9	7	0.707106781	0.0	0.0
10	5	0.0	0.0	0.0
10	6	0.0	0.0	0.707106781
10	7	0.0	-0.707106781	0.0

REPRESENTATIONS 5 X 5 --> 4

[ANG3 -->

[ANG1	[ANG2	5	6	7
8	8	0.0	0.0	0.0
8	9	-0.707106781	0.0	0.0
8	10	0.0	-0.707106781	0.0
9	8	-0.707106781	0.0	0.0
9	9	0.0	0.0	0.0
9	10	0.0	0.0	-0.707106781
10	8	0.0	-0.707106781	0.0
10	9	0.0	0.0	-0.707106781
10	10	0.0	0.0	0.0

Fig. 15 (Contd.)

REPRESENTATIONS 4 X 2 --> 5

[ANG3 -->

[ANG1	[ANG2	8	9	10
5	2	0.0	0.0	1.000000000
6	2	0.0	1.000000000	0.0
7	2	1.000000000	0.0	0.0

REPRESENTATIONS 4 X 3 --> 5

[ANG3 -->

[ANG1	[ANG2	8	9	10
5	3	0.0	0.0	-0.866025404
5	4	0.0	0.0	-0.500000000
6	3	0.0	0.866025404	0.0
6	4	0.0	-0.500000000	0.0
7	3	0.0	0.0	0.0
7	4	1.000000000	0.0	0.0

REPRESENTATIONS 4 X 4 --> 5

[ANG3 -->

[ANG1	[ANG2	8	9	10
5	5	0.0	0.0	0.0
5	6	-0.707106781	0.0	0.0
5	7	0.0	0.707106781	0.0
6	5	0.707106781	0.0	0.0
6	6	0.0	0.0	0.0
6	7	0.0	0.0	-0.707106781
7	5	0.0	-0.707106781	0.0
7	6	0.0	0.0	0.707106781
7	7	0.0	0.0	0.0

REPRESENTATIONS 5 X 1 --> 5

[ANG3 -->

[ANG1	[ANG2	8	9	10
8	1	1.000000000	0.0	0.0
9	1	0.0	1.000000000	0.0
10	1	0.0	0.0	1.000000000

Fig. 15 (Contd.)

REPRESENTATIONS 5 X 3 --> 5

LANG3 -->

LANG1	LANG2	8	9	10
8	3	1.000000000	0.0	0.0
8	4	0.0	0.0	0.0
9	3	0.0	-0.500000000	0.0
9	4	0.0	-0.866025404	0.0
10	3	0.0	0.0	-0.500000000
10	4	0.0	0.0	0.866025404

REPRESENTATIONS 5 X 4 --> 5

LANG3 -->

LANG1	LANG2	8	9	10
8	5	0.0	-0.707106781	0.0
8	6	0.0	0.0	-0.707106781
8	7	0.0	0.0	0.0
9	5	-0.707106781	0.0	0.0
9	6	0.0	0.0	0.0
9	7	0.0	0.0	-0.707106781
10	5	0.0	0.0	0.0
10	6	-0.707106781	0.0	0.0
10	7	0.0	-0.707106781	0.0

REPRESENTATIONS 5 X 5 --> 5

LANG3 -->

LANG1	LANG2	8	9	10
8	8	0.0	0.0	0.0
8	9	0.0	0.0	-0.707106781
8	10	0.0	0.707106781	0.0
9	8	0.0	0.0	0.707106781
9	9	0.0	0.0	0.0
9	10	-0.707106781	0.0	0.0
10	8	0.0	-0.707106781	0.0
10	9	0.707106781	0.0	0.0
10	10	0.0	0.0	0.0

Fig. 15 (Contd.)

APPENDIX

Source-code Listings

INDEX FOR PROGRAM: GPTHEORY				PAGE				COMMON CROSS-REFERENCE TABLE FOR: GPTHEORY				PAGE	
DATE = 4/07/71													
ROUTINE	PAGE	ROUTINE	PAGE	ROUTINE	PAGE	ROUTINE	PAGE						
ADAPT	16					PSYN	18	WIG3J	29	VPRINT	73		
ALPTAB	8					THETAX	21						
ANTSYN	32					THLMK	20	/GIRD/		/PHASE/			
APPLY	76							GRID	9	LINEUP	33		
AUFUN	40					/ALP1/		MOLCUL	2	PHASER	36		
DASFUN	62					ALPTAB	8	PROSYN	13	PROSYN	13		
BASIS	48					THETAX	21	PSYN	18				
CLEAN	41					WIG3J	29	THLMK	20	/PROTO/			
DEGEN	25									LINEUP	33		
EIGEN	23					/ASYM/		BASIS	48	ODDGA	70		
FILEUT	77					ANTSYN	32	FILEUT	77	PHASER	36		
GRID	9					PROSYN	13	INBAS	51	PROSYN	13		
GROUP	11							ORBTLS	58	VECUPL	65		
INBAS	51					/CONTR/							
LINCOM	27					ADAPT	16	/OUT1/		/SCAN/			
LINEUP	33					BASIS	48	BASFUN	62	CLEAN	41		
MAIN	1					GRID	9	BASIS	48	SCAN	43		
NATPRT	22					GROUP	11	FILEUT	77				
MOLCUL	2					INBAS	51	GRID	9	/UNGRAD/			
ODDGA	70					LINCOM	27	INBAS	51	AUFUN	40		
ORBTLS	58					LINEUP	33	MOLCUL	2	PHASER	36		
ORTHOG	72					MAIN	1	ORBTLS	58				
PHASER	36					MOLCUL	2	PSYN	18	/WIGNE/			
PROSYN	13					ODDGA	70			SYMB3J	31		
PSYN	18					ORBTLS	58	/OUT2/		VECUPL	65		
PTGROUP	46					ORTHOG	72	BASIS	48	WIG3J	29		
SCAN	43					PHASER	36	FILEUT	77				
SORT1	26					PROSYN	13	ORBTLS	58	/WORK1/			
SYMB3J	31					PSYN	18			ADAPT	16		
SYNSUM	44					PTGROUP	46	/OUT3/		LINEUP	33		
THETAX	21					SYNSUM	44	ADAPT	16	ODDGA	70		
THLMK	20					VECUPL	65	CLEAN	41	ORBTLS	58		
VALUE	63							FILEUT	77	PHASER	36		
VECUPL	65					/FTLES/		GROUP	11	PROSYN	13		
VIBROT	75					ADAPT	16	LINEUP	33	VECUPL	65		
VPRINT	73					BASIS	48	MOLCUL	2				
WIG3J	29					INBAS	51	ODDGA	70	/WORK2/			
						LINEUP	33	ORBTLS	58	BASFUN	62		
						MAIN	1	PHASER	36	ORBTLS	58		
						NATPRT	22	PROSYN	13	VALUE	63		
						MOLCUL	2	PTGROUP	46				
						ODDGA	70	SYNSUM	44	/XYZ/			
						ORBTLS	58	VECUPL	65	BASFUN	62		
						PHASER	36	VPRINT	73	GRID	9		
						PROSYN	13						
						PTGROUP	46	/OUT4/					
						SYNSUM	44	FILEUT	77				
						VECUPL	65	ORTHOG	72				
						VPRINT	73	VECUPL	65				

PROGRAM: GPTHEORY	ROUTINE: MAIN	PAGE: 1	PROGRAM: GPTHEORY	ROUTINE: MOLCUL	PAGE: 2
C*** PROGRAM GPTHEORY	MAIN0010		SUBROUTINE MOLCUL (JORDER)	MOLC0010	
C*** AUTOMATION OF MOLECULAR POINT GROUP THEORY	MAIN0020		C*** READS AND ANALYZES DATA PERTAINING TO NUCLEAR GEOMETRY	MOLC0020	
C*** T.D. BOUMAN AND G.L. GOODMAN	MAIN0030		IMPLICIT REAL*8 (A-H,O-Z)	MOLC0030	
C*** CHEMISTRY DIVISION, ARGONNE NATIONAL LABORATORY	MAIN0040		INTEGER*4 SYMNDL	MOLC0040	
C*** 4720 S. CASS AVE., ARGONNE, ILLINOIS 60439	MAIN0050		INTEGER*4 ELNMT,XOP,YOP,ZOP	MOLC0050	
IMPLICIT REAL*8 (A-H,O-Z)	MAIN0060		DIMENSION X(130),Y(130),Z(130)	MOLC0060	
COMMON/CNTAL/CRTZRO,CRTDEG,ISTOP,INTER,LFLOW(10)	MAIN0070		DIMENSION IDENT (20),PI(2),FC(3),JOP(3),RCCT(10),NUC(32),IIND(16),	MOLC0070	
COMMON/FILES/INPUT,ISUT	MAIN0080		ISYMBOL(103),IASYMB(30)	MOLC0080	
INPUT*5	MAIN0090		COMMON/OUT/POB(3,30),RADIUS(30),TMEFA(30),PMIC(30),MADMS,MATYPE;	MOLC0090	
ISUT*6	MAIN0100		1 NUC(1),NUC(10),ISYMB(10),ITYPA(10,10),ITYPT(30),ITYPL(30),	MOLC0100	
REWIND 20	MAIN0110		2 INVTAT(30)	MOLC0110	
CALL MFLUM	MAIN0120		COMMON/OUT/YNREP,NDEJAT(4),TANGA(5,14),IREPAT(2),IORDER,LVAL1,	MOLC0120	
ISTOP=0	MAIN0130		1 LVAL2,ITYPT(14),IRPAR(14),IDDO,IRLON(14)	MOLC0130	
C*** INPUT AND MOLECULAR GEOMETRY ANALYSIS	MAIN0140		COMMON/CNTAL/CRTZRO,CRTDEG,ISTOP,INTER,LFLOW(10)	MOLC0140	
10 CALL MOLCUL (JORDER)	MAIN0150		COMMON/GIR3/GRH(10,30,3),CHGR(1447),PMIGR(1447),NGR(10),NOIV	MOLC0150	
IF (ISTOP.EQ.1) GO TO 10	MAIN0170		COMMON/FILES/INPUT,ISUT	MOLC0160	
IF (JORDER.EQ.1) GO TO 5	MAIN0180		DATA 10/' /,HYES/3HYES/,HMO/3HMO /	MOLC0170	
C*** DETERMINATION OF POINT GROUP PROPERTIES	MAIN0190		DATA 10/' /,OS/,SIN/,17/,SQRT/,10/,17/	MOLC0190	
1 CALL GROUP(1SUM,JSUM)	MAIN0200		DATA SYMBOL/'H',HE',LE',BE',B',C',N',O',F',NE',NA',	MOLC0200	
CALL SYMSUM(1SUM,JSUM)	MAIN0210		1,'AG',AL',ST',P',S',CL',AR',X',CA',SE',Y',V',CN',	MOLC0210	
IF (ISTOP.EQ.0) GO TO 4	MAIN0220		2,'HW',FE',CO',NI',CU',ZU',GA',GE',AS',SE',BR',KA',RB',	MOLC0220	
IF (INTER.EQ.1) GO TO 10	MAIN0230		3,'SR',Y',ZK',NB',MH',TC',RU',RH',PO',AG',CD',IN',SM',	MOLC0230	
INTER=1	MAIN0240		4,'SA',TE',I',XE',CS',BA',LA',CE',PR',ND',PM',SH',EU',	MOLC0240	
ISTOP=0	MAIN0250		5,'GB',TB',DY',MO',EA',TN',YB',LU',NR',TA',W',RE',OS',	MOLC0250	
WRITE (ISUT,20)	MAIN0260		6,'IR',PT',AU',HG',TL',PD',BI',PO',AT',RN',FR',RA',AC',	MOLC0260	
20 FORMAT (77) RESTART WITH INTERMEDIATE OUTPUT //	MAIN0270		7,'TH',PA',U',NP',PU',AN',CN',KK',CF',FS',FN',MO',NE',	MOLC0270	
GO TO 1	MAIN0280		8,'LW' /	MOLC0280	
C*** SYMMETRY-ADAPTATION OF ATOMIC BASIS SET	MAIN0290		NAMLIST/PARAMS/NTOMS,CRTZRO,CRTDEG,CRTYPT,NTCHK,INTER,SCALE,	MOLC0290	
4 IF (LFLOW(1).EQ.0) GO TO 7	MAIN0300		1 NOIV,LFLOW	MOLC0300	
5 CALL BASIS	MAIN0310		NAMLIST/ATOM/ELNMT,X,Y,Z,XOP,YOP,ZOP,PX,PY,PZ	MOLC0310	
IF (ISTOP.EQ.1) GO TO 10	MAIN0320		CALEND=DATE(1,DUMMY)	MOLC0320	
IF (JORDER.EQ.1) GO TO 3	MAIN0330		PI=3.141592653589793	MOLC0330	
CALL ORBLS	MAIN0340		DR=PI/180.00	MOLC0340	
IF (ISTOP.EQ.0) GO TO 7	MAIN0350		IF (ISTOP.EQ.1) GO TO 203	MOLC0350	
IF (INTER.EQ.1) GO TO 10	MAIN0360		C*** INPUT OF MOLECULAR GEOMETRY AND JOB PARAMETERS	MOLC0360	
INTER=1	MAIN0370		READ (INPUT,300) IDENT,JOB	MOLC0370	
ISTOP=0	MAIN0380		300 FORMAT (19A4,A3,A1)	MOLC0380	
WRITE (ISUT,20)	MAIN0390		IF (JOB.EQ.1) STOP	MOLC0390	
GO TO 1	MAIN0400		WRITE (ISUT,302)	MOLC0400	
7 IF (LFLOW(2).EQ.0) GO TO 8	MAIN0410		302 FORMAT (1H,35X,1000) GPTHEORY - AUTOMATED MOLECULAR POINT GROUP	MOLC0410	
C*** VIBRATION-ROTATION SYMMETRY ANALYSIS	MAIN0420		THEORY *****/TAX,TENOMAS D. BOUMAN AND GORDON L. GOODMAN//	MOLC0420	
CALL VIBROT	MAIN0430		2 42X,'CHEMISTRY DIVISION, ARGONNE NATIONAL LABORATORY//54X,	MOLC0430	
IF (ISTOP.EQ.0) GO TO 9	MAIN0440		3 'ARGONNE, ILLINOIS 60439'//)	MOLC0440	
IF (INTER.EQ.1) GO TO 10	MAIN0450		1000 WRITE (ISUT,301) IDENT,CALEND	MOLC0450	
INTER=1	MAIN0460		301 FORMAT(' PROBLEM TITLE -- ',20A4,A3//)	MOLC0460	
ISTOP=0	MAIN0470		LVAL1=0	MOLC0470	
WRITE (ISUT,20)	MAIN0480		LVAL2=0	MOLC0480	
GO TO 1	MAIN0490		IORDER=0	MOLC0490	
C*** FINITE POINT GROUP VECTOR-COUPLING COEFFICIENTS	MAIN0500		ISTOP=0	MOLC0500	
8 IF (LFLOW(3).EQ.0) GO TO 3	MAIN0510		INTER=0	MOLC0510	
CALL VECUPL	MAIN0520		NTCHK=1	MOLC0520	
3 CONTINUE	MAIN0530		NOIV=24	MOLC0530	
C*** INTERFACE TO APPLICATIONS PROGRAMS	MAIN0540		DO 99 I=5,10	MOLC0540	
CALL APPLY	MAIN0550		99 LFLOW(1)=0	MOLC0550	
CALL FICUT	MAIN0560		DO 98 I=1,4	MOLC0560	
GO TO 10	MAIN0570		98 LFLOW(1)=1	MOLC0570	
END			SCALE=1.00	MOLC0580	
			CRTZRO=1.0-3	MOLC0590	

PROGRAM: GPTHEORY	ROUTINE: MOLCUL	PAGE: 3	PROGRAM: GPTHEORY	ROUTINE: MOLCUL	PAGE: 4
CRTDEG=.00000001	MOLC0590		220 POSIJ,11*POSIJ,11*PCIJ1)*SCALE	MOLC1170	
CRTTYP=1.0D-5	MOLC0590		IF (NATOMS.LE.2) GO TO 203	MOLC1180	
READ (INPUT,PARAMS)	MOLC0610		WRITE (IOUT,72) CRTZRD,CRTDEG,CRTTYP,NDIV,SCALE	MOLC1190	
IF (NATOMS.GT.30) GO TO 150	MOLC0620		72 FORMAT(// ' THRESHOLDS AND CONTROL PARAMETERS' /1X,34(1H-1)/5X,'ZERO	MOLC1200	
IF (NATOMS.GT.2) GO TO 23	MOLC0630		1THRESHOLD = ',1PEB.2/5X,'DEGENERACY TEST (RATIO OF EIGENVALUES) =	MOLC1210	
IF (NATOMS.EQ.0) GO TO 203	MOLC0640		2',OPF11.8/5X,'ATOM EQUIVALENCE THRESHOLD = ',1PEB.2/5X,'NO. OF CR	MOLC1220	
WRITE (IOUT,20)	MOLC0650		3D POINTS PER ATOM = ',13/5X,'SCALE FACTOR MULTIPLYING COORDINATES	MOLC1230	
20 FORMAT (/' MOLECULE IS DIATOMIC.')	MOLC0660		4* ',OPF10.5)	MOLC1235	
WRITE (IOUT,101)	MOLC0670		WRITE (IOUT,73)	MOLC1240	
101 FORMAT (/' ONLY THE REPRESENTATIONS SPANNED BY THE INPUT L-VALUES	MOLC0680		74 FORMAT (5X,'PROGRAM OPTIONS SELECTED'//10X,'POINT GROUP PROPERTIES	MOLC1250	
WILL BE CHARACTERIZED.// ' NO VECTOR-COUPLING COEFFICIENTS WILL BE	MOLC0690		15')	MOLC1260	
203 COMPUTED.')	MOLC0700		IF (INTER.EQ.1) WRITE (IOUT,74)	MOLC1270	
23 WRITE (IOUT,59) NATOMS,NTCHEK	MOLC0710		74 FORMAT (10X,'INTERMEDIATE PRINTOUT')	MOLC1280	
40 FORMAT (/' INPUT DESCRIPTION OF MOLECULE' /1X,28(1H-1)/5X,'NUMBER OF	MOLC0720		IF (LFLOW1).EQ.1) WRITE (IOUT,75)	MOLC1290	
LATIONS = ',12/5X,'ASSUMED NO. OF EQUIVALENT SETS (ATOM TYPES) = ',	MOLC0730		75 FORMAT (10X,'SYMMETRY-ADAPTED ATOMIC BASIS SET')	MOLC1300	
2 12/)	MOLC0740		IF (LFLOW1).EQ.1) WRITE (IOUT,76)	MOLC1310	
WRITE (IOUT,70)	MOLC0750		76 FORMAT (12X,'MINIMAL SLATER BASIS')	MOLC1320	
70 FORMAT (5X,'CARTESIAN COORDINATES (IN FORM X = A*FUNC1) 1//5X,'	MOLC0760		IF (LFLOW1).EQ.2) WRITE (IOUT,77)	MOLC1330	
1 ' ATOM CHEMICAL',11X,'X',23X,'Y',23X,'Z'/5X,'NUMBER SYMBOL',	MOLC0770		77 FORMAT (12X,'DOUBLE-ZETA SLATER BASIS')	MOLC1340	
2 2X,314X,'A',4X,'FUNC',4X,'B',5X1/)	MOLC0780		IF (LFLOW1).EQ.9) WRITE (IOUT,78)	MOLC1350	
DO 220 I=1,NATOMS	MOLC0790		78 FORMAT (12X,'BASIS SET READ IN')	MOLC1360	
X=0.	MOLC0800		IF (LFLOW12).EQ.1) WRITE (IOUT,79)	MOLC1370	
Y=0.	MOLC0810		79 FORMAT (10X,'VIBRATION-ROTATION SYMMETRY')	MOLC1380	
Z=0.	MOLC0820		IF (LFLOW13).EQ.1) WRITE (IOUT,80)	MOLC1390	
XOP=1B	MOLC0830		80 FORMAT (10X,'VECTOR-COUPLING COEFFICIENTS')	MOLC1400	
YOP=1B	MOLC0840		C*** TRANSLATION TO CENTER OF NUCLEAR CHARGE COORDINATE SYSTEM	MOLC1410	
ZOP=1B	MOLC0850		1005 FN=0.	MOLC1420	
FX=1.00	MOLC0860		DO 104 J=1,3	MOLC1430	
FY=1.00	MOLC0870		104 P1J)=0.	MOLC1440	
FZ=1.00	MOLC0880		DO 3 I=1,NATOMS	MOLC1450	
READ (INPUT,ATOM)	MOLC0890		FN1=NUC1)	MOLC1460	
WRITE (IOUT,71) 1,ELMNT,X,XOP,FX,Y,YOP,FY,Z,ZOP,FZ	MOLC0900		DO 110 J=1,3	MOLC1470	
71 FORMAT (7X,12,7X,A2,3X,31PB,4,1X,A4,FB,3,3X1)	MOLC0910		110 P1J)=P1J)+POS1J,1)*FN1	MOLC1480	
IASYMB(1)=ELMNT	MOLC0920		3 FN=FN+FN1	MOLC1490	
POS1,1)=X	MOLC0930		DO 111 J=1,3	MOLC1500	
POS2,1)=Y	MOLC0940		111 P1J)=P1J)/FN	MOLC1510	
POS3,1)=Z	MOLC0950		DO 400 I=1,NATOMS	MOLC1520	
1OP1)=XOP	MOLC0960		R2=0.	MOLC1530	
1OP2)=YOP	MOLC0970		DO 401 J=1,3	MOLC1540	
1OP3)=ZOP	MOLC0980		POS1J,1)=POS1J,1)-P1J)	MOLC1550	
FC1)=FX	MOLC0990		401 R2=R2+POS1J,1)*2	MOLC1560	
FC2)=FY	MOLC1000		RADIUS(1)=DSQRT(R2)	MOLC1570	
FC3)=FZ	MOLC1010		XX1)=POS1,1)	MOLC1580	
DO 222 N=1,103	MOLC1020		YY1)=POS2,1)	MOLC1590	
NCH=N	MOLC1030		ZZ1)=POS3,1)	MOLC1600	
IF (IASYMB(1).EQ.SYMBOL(N)) GO TO 223	MOLC1040		400 CONTINUE	MOLC1610	
222 CONTINUE	MOLC1050		C*** SPHERICAL POLAR COORDINATES IN C.C. SYSTEM	MOLC1620	
WRITE (IOUT,224) IASYMB(1),1	MOLC1060		DO 4 I=1,NATOMS	MOLC1630	
224 FORMAT (/' THE CHEMICAL SYMBOL',42,' INPUT ON ATOM-PARAMETER CARD	MOLC1070		THETA(1)=0.	MOLC1640	
IND. ',12,' IS INCORRECTLY PUNCHED.')	MOLC1080		PHI(1)=0.	MOLC1650	
GO TO 203	MOLC1090		W=POS3,1)	MOLC1660	
223 NUC1)=NCH	MOLC1100		IF (RADIUS(1).LE.CRTZRD) GO TO 4	MOLC1670	
DO 220 J=1,3	MOLC1110		IF (DABS(DABS(W)-RADIUS(1)).GT.1.D-7) GO TO 305	MOLC1680	
IF (FC1J).EQ.0) FC1J)=1.00	MOLC1120		IF (POS3,1).GE.0.) GO TO 4	MOLC1690	
IF (1DPIJ).EQ.1) FC1J)=DCOS(DR*FC1J))	MOLC1130		THETA(1)=PI	MOLC1700	
IF (1DPIJ).EQ.1) FC1J)=DSIN(DR*FC1J))	MOLC1140		GO TO 4	MOLC1710	
IF (1DPIJ).EQ.1) FC1J)=1.00/FC1J)	MOLC1150		305 THETA(1)=DARCOS(POS1J,1)/RADIUS(1))	MOLC1720	
IF (1DPIJ).EQ.1) FC1J)=DSQRT(FC1J))	MOLC1160		IF (DABS(POS2,1).LE.CRTZRD.AND.DABS(POS1,1).LE.CRTZRD) GO TO 4	MOLC1730	

PROGRAM: GETHEDRY	ROUTINE: MOLECU	PAGE: 5	PROGRAM: GETHEDRY	ROUTINE: MOLECU	PAGE: 6
PHI(1)=DATANZ(POS(2,1),POS(1,1))	MOLC1740		ITYP(1)=NUNR	MOLC2320	
IF (PHI(1).LT.0.) PHI(1)=PHI(1)+2.00*PI	MOLC1750		ITYP(1)=NUNR	MOLC2330	
CONTINUE	MOLC1760		GO TO 24	MOLC2340	
C*** MOLECULAR GEOMETRY ANALYSIS	MOLC1770		25 CONTINUE	MOLC2350	
WRITE(10UT,81)	MOLC1780		NATYPE=NATYPE+1	MOLC2360	
81 FORMAT(1H,'MOLECULAR GEOMETRY ANALYSIS',19,2Y(1H-17))	MOLC1790		NUC(1)=NUC(1)+1	MOLC2370	
WRITE(10UT,205)	MOLC1800		ITSYMB(1)=JASYMB(1)	MOLC2380	
205 FORMAT(1Z5X,'ATOM',3X,'CHEMICAL',4X,'ATOMIC',6X,'CRYSTAL COORDINATES IN CENTER OF CHARGE SYSTEM')	MOLC1810		RCCT(1)=RADIUS(1)	MOLC2390	
WRITE(10UT,60)	MOLC1820		MUNT(1)=NATYPE	MOLC2400	
60 FORMAT(24X,'NUMBER',3X,'SYMBOL',3X,'NUMBER (Z)',10X,'X',16X,'Y',	MOLC1830		ITYP(1)=NATYPE	MOLC2410	
16X,'Z')	MOLC1840		ITYP(1)=1	MOLC2420	
204 FORMAT(126X,12,7X,A2,8X,12,4X,3F17,6/)	MOLC1850		ITYP(1),NATYPE)=14	MOLC2430	
WRITE(10UT,204) I,IASYMB(1),NUC(1),IPUS(J,I),J,I,3I,1,NATONS	MOLC1860		24 CONTINUE	MOLC2440	
WRITE(10UT,50)	MOLC1870		IF (NATYPE.EQ.NATONS) JORDER=1	MOLC2450	
50 FORMAT(1Z5X,'ATOM',3X,'CHEMICAL',4X,'ATOMIC',6X,'SPHERICAL COORDINATES IN CENTER OF CHARGE SYSTEM',24X,'NUMBER',3X,'SYMBOL',3X,'NUMBER')	MOLC1880		JORDER=10ORDER	MOLC2460	
28X(12),10X,'H',14X,'METHA',13X,'PHI')	MOLC1890		IF (NATYPE.GT.10) GO TO 140	MOLC2470	
WRITE(10UT,204) I,IASYMB(1),NUC(1),RADIUS(1),THETA(1),PHI(1),	MOLC1900		IF (NATYPE.NE.NTCHEM) GO TO 200	MOLC2480	
I=1,NATONS)	MOLC1910		C*** TEST FOR CENTER OF INVERSION	MOLC2490	
C*** TEST FOR LINEAR MOLECULE	MOLC1920		NAT=NATONS	MOLC2500	
IF (NATONS.EQ.2) GO TO 310	MOLC1930		MINV=0	MOLC2510	
IS=2	MOLC1940		DO 61 IT=1,NATYPE	MOLC2520	
IF (RADIUS(1).LE.CRTZRO) IS=3	MOLC1950		IF (RCCT(1).GT.CRTZRO) GO TO 65	MOLC2530	
LIN=1	MOLC1960		NAT=NATONS-1	MOLC2540	
IS=IS-1	MOLC1970		GO TO 61	MOLC2550	
DO 104 I=IS,NATONS	MOLC1980		65 NUNB=NUNT(1)	MOLC2560	
PHI(1)=DND(1,PHI(1)+PI,2.00*PI)	MOLC1990		DO 62 IN=1,NUNB	MOLC2570	
IF (DSIN(THETA(1)).LE.1.0-7) PHI(1)=D.	MOLC2000		IA=ITYP(1),IN=1	MOLC2580	
IF ((DABS(THETA(1))-THETA(1)).LE.CRTZRO.AND.DABS(PHI(1))-PHI(1)).LE.CRTZRO	MOLC2010		THPI=PI-THETA(1)	MOLC2590	
1 .LE.CRTZRO) .OR. (DABS(THETA(1))-PI+THETA(1)).LE.CRTZRO	MOLC2020		PHI(1)=DND(1,PHI(1)+PI,2.00*PI)	MOLC2600	
2 .AND.DABS(DABS(PHI(1))-PHI(1)).LE.CRTZRO) GO TO 104	MOLC2030		IF (DSIN(THETA(1)).LE.1.0-7) PHI(1)=D.	MOLC2610	
LIN=0	MOLC2040		DO 63 IN=1,NUNB	MOLC2620	
104 CONTINUE	MOLC2050		JA=ITYP(1),IT	MOLC2630	
IF (LIN.EQ.0) GO TO 21	MOLC2060		IF (DABS(THETA(1))-THPI).GT.CRTZRO.OR.DABS(PHI(1))-PHI(1)).GT.	MOLC2640	
WRITE(10UT,105)	MOLC2070		1 CRTZRO) GO TO 63	MOLC2650	
105 FORMAT(1H,'MOLECULE IS LINEAR.')	MOLC2080		MINV=MINV+1	MOLC2660	
WRITE(10UT,101)	MOLC2090		INVT(1)=JA	MOLC2670	
310 READ(10UT,315) LVAL1,LVAL2	MOLC2100		GO TO 62	MOLC2680	
315 FORMAT(215)	MOLC2110		63 CONTINUE	MOLC2690	
LFLQW(3)=D	MOLC2120		1000=0	MOLC2700	
IF (MAX(LVAL1,LVAL2).GT.16) GO TO 150	MOLC2130		GO TO 160	MOLC2710	
C*** GENERATE AND LABEL ATOM-TYPES, OTHER TABLES	MOLC2140		62 CONTINUE	MOLC2720	
21 NUNT(1)=1	MOLC2150		61 CONTINUE	MOLC2730	
NATYPE=1	MOLC2160		IF (MINV.EQ.NAT) 1000=1	MOLC2740	
RCCT(1)=RADIUS(1)	MOLC2170		160 IF (NATANDIV.GT.72) GO TO 150	MOLC2750	
MUNT(1)=NUC(1)	MOLC2180		GO TO 201	MOLC2760	
ITSYMB(1)=IASYMB(1)	MOLC2190		200 WRITE(10UT,202)	MOLC2770	
ITYP(1)=1	MOLC2200		202 FORMAT(1H,'THE CALCULATED NUMBER OF ATOM-TYPES DOES NOT AGREE WITH YOUR ASSUMPTION. PLEASE RECHECK YOUR INPUT.')	MOLC2780	
ITYP(1)=1	MOLC2210		150 STOP	MOLC2790	
ITYP(1)=1	MOLC2220		201 WRITE(10UT,51) NATYPE	MOLC2800	
DO 24 IA=2,NATONS	MOLC2230		51 FORMAT(1H,'THE PROGRAM DISTINGUISHES ',12,' ATOM TYPES, AS FOLLOWS')	MOLC2810	
DO 25 I=1,NATYPE	MOLC2240		15...'	MOLC2820	
IF (DABS(RADIUS(1))-RCCT(1)).GT.CRTZRO.OR.NUC(1).NE.NUC(1))	MOLC2250		WRITE(10UT,53)	MOLC2830	
1 GO TO 25	MOLC2260		53 FORMAT(1Z18X,'ATOM-TYPE CHEMICAL NO. OF NUCLEAR UPPER : 13,MOLC2840		
1011 NUNB=NUNT(1)+1	MOLC2270		1400 INDICES OF THE ATOMS IN THIS ATOM-TYPE',20X,'INDEX SYMB,MOLC2850		
NUNT(1)=NUNB	MOLC2280		20L ATOMS CHARGE LOWER : RUNNING INDICES WITHIN THE ATOM-TYPE,MOLC2860		
ITYP(1)=1	MOLC2290		3YX(18X,91'-'),3X,8(1'-'),3X,6(1'-'),3X,7(1'-'),3X,5(1'-')/)	MOLC2870	
	MOLC2300		DO 54 I=1,NATYPE	MOLC2880	
	MOLC2310			MOLC2890	

PROGRAM: GPTHEORY	ROUTINE: MOLCUL	PAGE: 7	PROGRAM: GPTHEORY	ROUTINE: ALPTAB	PAGE: 8
NUMB=NUMT(1)	MOLC2900		SUBROUTINE ALPTAB	ALPT0010	
WRITE (10UT,55) IF,ITSYMBITF,NUMB,NUCT(1),TIFYPAT,ITS,1+1,NUMB	MOLC2910		C*** SETS UP TABLE OF COEFFICIENTS FOR CALCULATION OF ASSOCIATED	ALPT0020	
55 FORMAT (1/21X,12,10X,42,8X,12,7X,12,6X,1215,11X/60X,1215)	MOLC2920		C*** LEGENDRE POLYNOMIALS, L-LE, 16, AND TABLE OF FACTORIALS	ALPT0030	
54 WRITE (10UT,56) (I,I=1,NUMB)	MOLC2930		IMPLICIT REAL*8 (A-H,O-Z)	ALPT0040	
56 FORMAT (60X,1215)	MOLC2940		COMMON/ALP1/THETAC(1224),FACT(150)	ALPT0050	
IF (1ORDER.EQ.1) WRITE (10UT,100)	MOLC2950		FACT(1)=1.00	ALPT0060	
100 FORMAT (32X,5(SINCE THIS MOLECULE HAS NO NON-TRIVIAL SYMMETRY &LENE	MOLC2960		DO 3 J=2,50	ALPT0070	
INTS,740X,'FURTHER CALCULATIONS OF SYMMETRY-RELATED'740X,'PROPERTY	MOLC2970		F(1)=1	ALPT0080	
RES WILL BE BYPASSED.')	MOLC2980		3 FACT(1)=FACT(1-1)*F(1)	ALPT0090	
HINV=MND	MOLC2990		LMAX=16	ALPT0100	
IF(1000.EQ.1) HINV=MYES	MOLC3000		THETAC(1)=DSORT(0.500)	ALPT0110	
WRITE(10UT,82) HINV	MOLC3010		DO 4 L1=1,LMAX	ALPT0120	
82 FORMAT (/// CENTER OF INVERSION - ',43/1.	MOLC3020		TL(1)=2*TL1+1	ALPT0130	
IF (1STOP.EQ.01 GO TO 999	MOLC3030		TL=2*TL1	ALPT0140	
203 READ (INPUT,3001 IDENT, JOB	MOLC3040		A=THETAC(1)*DSORT(TL(1)/TL	ALPT0150	
IF (1JOB.EQ.1MOL16F) STOP	MOLC3050		LP1=TL+1	ALPT0160	
IF (1JOB.NE.1MOL151) GO TO 203	MOLC3060		JSUB=4*LI*LP1	ALPT0170	
WRITE (10UT,3021	MOLC3070		DO 4 MPI=1,LP1	ALPT0180	
GO TO 1000	MOLC3080		MJ=MP1-1	ALPT0190	
150 WRITE (10UT,1511	MOLC3090		JSUB=(JSUB+8*MJ	ALPT0200	
151 FORMAT (/// ONE OR MORE OF THE INPUT PARAMETERS WILL CAUSE ARAY	MOLC3100		ANORM=A*DSORT(FACT(LP1-MJ1)/FACT(LP1+MJ1))	ALPT0210	
LIMITS TO BE EXCEEDED. CALCULATION ABANDONED.')	MOLC3110		LM=(11-MJ)/2+1	ALPT0220	
1STOP=1	MOLC3120		PHASE=-1.00	ALPT0230	
GO TO 201	MOLC3130		IF (2*(MJ/2).NE.MJ) PHASE=-PHASE	ALPT0235	
999 CALL ALPTAB	MOLC3132		DO 2 IP=1,LM	ALPT0240	
CALL GRID(1,1000)	MOLC3134		MU=IP-1	ALPT0250	
CALL GRID(2,1000)	MOLC3136		PHASE=-PHASE	ALPT0260	
RETURN	MOLC3138		2 THETAC(JSUB+IP1)=ANORM*PHASE*FACT(2*(LI-MJ)+1)/(FACT(MU+1)*FACT(LP1	ALPT0270	
END	MOLC3140		1 -MU)*FACT(LP1-MJ-2*MJ))	ALPT0280	
			4 CONTINUE	ALPT0290	
			37 THETA=D.	ALPT0300	
			RETURN	ALPT0310	
			END	ALPT0320	

PROGRAM: GRIMEDRY	ROUTINE: GRID	PAGE: 2	PROGRAM: GRIMEDRY	ROUTINE: GRID	PAGE: 10
SUBROUTINE GRID (IGRID, IODD)	GRID0010		34 ALPH=0.		GRID0590
C**** CALCULATES GRID OF POINTS REFLECTING SYMMETRY OF MOLECULE, WHERE	GRID0020		DO 3 NPT=1,NDIV		GRID0600
C**** FUNCTION VALUES WILL BE SAMPLED	GRID0030		IND=IND+1		GRID0610
IMPLICIT REAL*8 IA-H,O-Z	GRID0040		COSAL=DCOS(ALPH)		GRID0620
COMMON/OUT1/POS(3,30),RAD(US(30),THETA(30),PHI(30),NATOMS,NATYPE,	GRID0050		CTH=COSIN*COSDM+SINTN*SINUM*CUSAL		GRID0630
1 NURT(10),NUCT(10),TSYMB(10),ITYPAT(30,10),ITYPT(30),ITYPI(30),	GRID0060		C=IGRIDNDI*CTH		GRID0640
2 INVAT(30)	GRID0070		NOIX=1.00-CTH*CTH		GRID0650
COMMON/CONTR/CRIZRO,CRIZEG,ISTOP,INTER,LFLOW(10)	GRID0080		IF (RADIX,LF,CRIZRO) PADYX=0.		GRID0660
COMMON/GRID/RCR(10(30,3),CTHGR(144),PHIGR(144),NGRID,NDIV	GRID0090		V=DSORT(RADIX)		GRID0670
COMMON/XYZ/XG(1440),YG(1440),ZG(1440),XI(30),YI(30),ZI(30)	GRID0100		RST=V*RCR		GRID0680
DIMENSION IAFLAG(30)	GRID0110		IF (V*SINTY,LE,CRIZRO) GO TO 4		GRID0690
C>>>> CHANGE NEXT CARD IF DIMENSION OF, E.G., 'RADIUS' IS CHANGED	GRID0120		21 W=(COSDM-CUSTN*CTH)/TSINTN*V		GRID0700
MAXNAT=30	GRID0130		IF (DABS(DABSIN)-1.00).GT.(CRIZRO) GO TO 30		GRID0710
PI=3.141592653589793	GRID0140		IF (W)32,30,33		GRID0720
PIPL=2.00*PI	GRID0150		32 W=-1.00		GRID0730
A1=DTAN (PI/45.00)	GRID0160		GO TO 30		GRID0740
A2=150.00	GRID0170		33 W=1.00		GRID0750
IRASE=0	GRID0180		30 PMNPH=DARCOS(W)		GRID0760
NGRID=NDIV*NATOMS	GRID0190		IF (ALPH-PI)5,6,7		GRID0770
DO 44 IA=1,NATOMS	GRID0200		5 PHIG=PHI(IA)*PMNPH		GRID0780
IF (RADIUS(IA).LE.(CRIZRO)) NGRID=NGRID-MDIV	GRID0210		GO TO 8		GRID0790
44 CONTINUE	GRID0220		6 PHIG=PHI(IA)		GRID0800
IF ((IGRID.EQ.2) IRASE=NGRID	GRID0230		GO TO 8		GRID0810
41 IND=IBASE	GRID0240		7 PHIG=PHI(IA)*PMNPH		GRID0820
FUIV=NDIV	GRID0250		GO TO 8		GRID0830
C**** GRID CONSISTS OF 4 CIRCLE OF 'NDIV' POINTS ABOUT THE RADIUS VECTOR	GRID0260		4 PHIG=ALPH		GRID0840
C**** JOINING EACH ATOM TO THE ORIGIN	GRID0270		8 PMTGR(IND)*PHIG		GRID0850
DO 50 IA=1,NATOMS	GRID0280		IF (IODD.EQ.0) GO TO 53		GRID0860
IAFLAG(IA)=0	GRID0290		C**** COORDINATES OF INVERSION-SYMMETRIC POINTS		GRID0870
IACC=0	GRID0300		JNO=JNDIV*NPT		GRID0880
DO 1 IA=1,NATOMS	GRID0310		CTHGR(JNO)=CTH		GRID0890
RIT=RADIUS(IA)	GRID0320		PHIGR(JNO)=PHIG+PI		GRID0900
IF (RIT,GT,CRIZRO) GO TO 55	GRID0330		IF (PMIGR(JNO).GT.PI) PHIGR(JNO)=PHIG-PI		GRID0910
IACC=1	GRID0340		53 CONTINUE		GRID0920
GO TO 1	GRID0350		XG(IND)=RST*DCOS(PHIG)		GRID0930
55 IF (IODD.EQ.0) GO TO 51	GRID0360		YG(IND)=RST*DSIN(PHIG)		GRID0940
IF (IAFLAG(IA).EQ.0) GO TO 45	GRID0370		ZG(IND)=RCR*CTH		GRID0950
IND=IND+MDIV	GRID0380		IF (IODD.EQ.0) GO TO 3		GRID0960
GO TO 1	GRID0390		ZG(IND)=-ZG(IND)		GRID0970
45 IP1=IA+1	GRID0400		XG(JNO)=-XG(IND)		GRID0980
IF (IP1,GT,NATOMS) GO TO 1	GRID0410		YG(JNO)=-YG(IND)		GRID0990
J8=INVAT(IA)	GRID0420		3 ALPH=ALPH+2.00*PI/FUIV		GRID1000
50 JNDIV=(JA-1-IACC)*MDIV+IBASE	GRID0430		1 CONTINUE		GRID1010
IAFLAG(JA)=1	GRID0440		IF (IGRID.NE.1) RETURN		GRID1020
C**** ALGORITHM FOR ANGULAR OPENING SUBTENDED BY GRID CIRCLES	GRID0450		DO 10 J=1,NATOMS		GRID1030
51 IT=ITYPT(IA)	GRID0460		X(1)=POS(1,1)		GRID1040
20 ZIT=NUCT(1)1	GRID0470		Y(1)=POS(2,1)		GRID1050
OPSIDE=A*(1.00-ZIT/A2)	GRID0480		10 ZIT=POS(3,1)		GRID1060
IF (IGRID.EQ.2) OPSIDE=1.500*OPSIDE	GRID0490		RETURN		GRID1070
HYPOT=DSORT(1.00+OPSIDE*OPSIDE)	GRID0500		END		GRID1080
COSDM=1.00/HYPOT	GRID0510				
SINDM=OPSIDE/HYPOT	GRID0520				
RCR=RGRI(1A,IGRID)	GRID0530				
IF (IODD.EQ.1) RGR(1JA,IGRID)=RCR	GRID0540				
C**** ASSIGNMENT OF GRID COORDINATES	GRID0550				
COSTN=DCOS(THETA(IA))	GRID0560				
SINTN=DSIN(THETA(IA))	GRID0570				

PROGRAM: GPINEORY	ROUTINE: P47SYN	PAGE: 13	PROGRAM: GPINEORY	ROUTINE: PROSYN	PAGE: 14
SUBROUTINE PROSYN (L,IDE,MR,ISUM,LOEGL,IFLAG)	PROS0010		CALL PPSY (L,LL,THLML,LL,THLML,2,KALC,RES,B,NMAX)	PROS0590	
C*** SUPERVISES DETERMINATION OF IRREDUCIBLE REPRESENTATIONS SPANNED	PROS0020		LL=L	PROS0600	
C*** BY A PARTICULAR SET OF SPHERICAL HARMONICS	PROS0030		LL=LL	PROS0610	
IMPLICIT REAL*8 (A-H,O-Z)	PROS0040		C*** TRANSFORM MATRIX WITH VECTORS - RESULTING MATRIX WILL BE BLOCK	PROS0620	
REAL*8 LABEL,LABEL2	PROS0050		C*** DIAGONAL EXCEPT FOR BLOCKS CONNECTING VECTORS BELONGING TO THE SAME	PROS0630	
COMMON/GRU/GR10(30,3),CTMGR1(447),PMIGR1(447),MGR10,MIV	PROS0060		C*** REPRESENTATION	PROS0640	
COMMON/CONTR/CKT(40),CRDEG,ISTOP,INTER,LFLOW(10)	PROS0070		CALL CINCOR (RES,EV,NDEG,NMAX,LL,C,B,NEVAL,IREPLT,MR,LOEGL)	PROS0650	
COMMON/OUT3/NREP,NDEG(14),IANG(15),IREP(32),IORDER,LVAL1,	PROS0080		IDENT2(13)=IDENT1	PROS0660	
1 LVAL2,IRTP(14),IRPAR(14),LODB,ILOW(14)	PROS0090		IDENT2(14)=IDE(2)	PROS0670	
COMMON/PROT/COPT(1802),EVP(151),IREPLT(21),NEVALP(21)	PROS0100		DO 30 I=1,LL	PROS0680	
COMMON/PHASE/PHGG(20),PHASE	PROS0110		LABEL2(11)=MOL5	PROS0690	
COMMON/ASYM/LA,MGR10,L3P1,LAP1,LADEG	PROS0120		30 IDENT2(11)=	PROS0700	
COMMON/FILES/INPUT,LOUT	PROS0130		IF IINTER.EQ.1 CALL MATPRT (LL,0,IDENT2,RES,EV,NMAX,LABEL2,INDEX2)	PROS0710	
COMMON/NORX1/DUMMY(8000)	PROS0140		DO 6 I=1,LL	PROS0720	
DIMENSION LABEL(50),INDEX(50),IDENT1(20),LOEGL(14),IDE(2),	PROS0150		6 EV(1)=RES(1,1)	PROS0730	
1 NDEGL(33),IREPLT(33),RES(50,50),EVAL(50,50),B(50,50),NDEG(50)	PROS0160		C*** CHECK TO MAKE SURE DEGENERACIES HAVEN'T BEEN DEGRADED	PROS0740	
DIMENSION IDENT2(20),LABEL2(50),INDEX2(50)	PROS0170		CALL DEGEN (LL,EV,NDEGL,NEVAL,CRDEG)	PROS0750	
EQUIVALENCE (RES,DUMMY), (EV,DUMMY(250)), (C,DUMMY(255)),	PROS0180		IF (NEVAL.NE.NEVAL1) GO TO 7	PROS0760	
1 (B,DUMMY(505)), (NDEG,DUMMY(7975)), (NEVAL,DUMMY(8000)),	PROS0190		CALL SORT1 (C,B,EV,NMAX,LL,NDEG,-NEVAL,IREPLT)	PROS0770	
2 (NDEGL,DUMMY(755)), (IREPLT,DUMMY(7950)), (LABEL,DUMMY(7875)),	PROS0200		NAT=NR	PROS0780	
3 (INDEX,DUMMY(7925)), (LABEL2,DUMMY(756)), (INDEX2,DUMMY(7626))	PROS0210		NREP=NR	PROS0790	
DATA IDENT1/4H MAT,4HRIK,4HOF P,4H OVE,4H SP,4HHEAT,4HCAL,	PROS0220		IND=0	PROS0800	
14HARM,4HONIC,4HS,10*4H /	PROS0230		IBASE=0	PROS0810	
DATA IDENT2/4H MAT,4HRIK,4HOF P,4H OVE,4H SP,4HHEAT,4HVECT,4HORS	PROS0240		ISI=0	PROS0820	
1,4HOF F,4HJST,4H MAT,4HRIK,8*4H /	PROS0250		ISUB=0	PROS0830	
DATA MOL3/7H COS /,MOL4/7H SIN /,MOL5/7H E.V. /	PROS0260		NEVALP(1)=NEVAL	PROS0840	
EXTERNAL THLML,ANTSYM	PROS0270		GO TO 13	PROS0850	
NMAX=50	PROS0280		C*** IF L IS ODD, LINE UP PHASES OF ODD REPRESENTATIONS WITH EVEN ONES	PROS0860	
PHASE=3	PROS0290		20 LA=MAX(12,L/2+1)	PROS0870	
IF (IFLAG.EQ.2) GO TO 1	PROS0300		CALL WIG3J (LA,LA,1,ISTOP)	PROS0880	
IDENT1(11)=IDE(1)	PROS0310		MGR10=MGR1D	PROS0890	
IDENT1(12)=IDE(2)	PROS0320		LAP1=LA+1	PROS0900	
INDEX(1)=0	PROS0330		LADEG=2*LA+1	PROS0910	
LABEL(1)=MOL3	PROS0340		L3P1=L+1	PROS0920	
DO 60 M=1,L	PROS0350		CALL ADAPT (L,LL,ANTSYM,9,1,IDENT1,2)	PROS0930	
M2=2*M	PROS0360		DO 41 I=1,NEVAL	PROS0940	
INDEX(M2)=M	PROS0370		41 IREPLT(11)=IREPLT(11)+NRT	PROS0950	
INDEX(M2+1)=M	PROS0380		IND=NRT	PROS0960	
LABEL(M2)=MOL4	PROS0390		NR=NRT	PROS0970	
60 LABEL(M2+1)=MOL5	PROS0400		NREP=2*NR	PROS0980	
LL=2*LL+1	PROS0410		NEVALP(2)=NEVAL	PROS0990	
KALC=1	PROS0420		IBASE=NEVALP(1)	PROS1000	
IF (LODB.NE.0) KALC=9	PROS0430		ISI=LL	PROS1010	
C*** GENERATE CRUDE SYMMETRY VECTORS	PROS0440		ISUB=ISI+ISI	PROS1020	
CALL ADAPT (L,LL,THLML,KALC,1,IDENT1,1)	PROS0450		C*** STORE VECTORS FOR LATER USE	PROS1030	
IF (L.NE.4) GO TO 1	PROS0460		13 DO 15 I=1,NEVAL	PROS1040	
ISUM=0	PROS0470		15 IREPLT(11)=IBASE+IREPLT(11)	PROS1050	
DO 2 I=1,NEVAL	PROS0480		DO 40 J=1,NR	PROS1060	
2 ISUM=ISUM+NDEG(13)+2	PROS0490		40 NDEG(11)=IND+LOEGL(1)	PROS1070	
IF (ISUM.NE.13) RETURN	PROS0500		DO 11 J=1,LL	PROS1080	
1 IF (INTER.EQ.0) GO TO 3	PROS0510		EVP(1)=ISI+EVC(J)	PROS1090	
WRITE (LOUT,200)	PROS0520		DO 31 I=1,LL	PROS1100	
200 FORMAT (// 'DEGENERACIES OF EIGENVALUES FOR PROTOTYPE FUNCTION(S)')	PROS0530		ISUB=ISUB+1	PROS1110	
WRITE (LOUT,100) NDEGL(1),I=1,NEVAL	PROS0540		11 CPROT(SUB)=C(1,1)	PROS1120	
100 FORMAT (16F5)	PROS0550		RETURN	PROS1130	
3 IF (ISTOP.EQ.1) RETURN	PROS0560		7 ISTOP=1	PROS1140	
IF (2*(L/2).NE.L) GO TO 20	PROS0570		WRITE (LOUT,501)	PROS1150	
C*** CALCULATE SECOND MATRIX OF SYMMETRY OPERATOR	PROS0580		50 FORMAT (1H 'DEGENERACIES OF MATRIX 2 DO NOT AGREE WITH THOSE IN 4')	PROS1160	

PROGRAM: CPINEOSY	SUBROUTINE: PROSYN	PAGE: 15	PROGRAM: CPINEOSY	SUBROUTINE: ADAPT	PAGE: 15
1 MATRIX 1. CALCULATION ABANDONED. *)	PROS1170		SUBROUTINE ADAPT (L,N,F2,KALC,MPAR,IDENT,IFPRO)	ADAP0010	
RETURN	PROS1180	C***	SYMMETRY-ADAPTS SET OF FUNCTIONS DEFINED BY FUNCTION F2	ADAP0020	
END	PROS1190		IMPLICIT REAL*8 (A-H,O-Z)	ADAP0030	
			COMMON/CONTAL/CATRO,CRTDEC,ISTOP,INTEN,LFLOW(10)	ADAP0040	
			COMMON/OUT3/HREP,NDEGAIL(4),IANGA(5,14),IREP(132),ICORDER,LVALL,	ADAP0050	
			1 LVAC2,IRTYPE(4),IRPART(4),I000,IRLOW(14)	ADAP0060	
			COMMON/FILES/INPUT, IOUT	ADAP0070	
			COMMON/HORR1/DUMMY(8000)	ADAP0080	
			DIMENSION LABEL(50),INDEX(50),IREPL(50),IDENT(10),RES(50,53),	ADAP0090	
			1 EV(50),C(50,50),R(50,50),NDEG(50)	ADAP0100	
			EQUIVALENCE (RES,DUMMY),(EV,DUMMY(2501)),(C,DUMMY(2551)),	ADAP0110	
			1 IR,DUMMY(5051)),(LABEL,DUMMY(78751)),(INDEX,DUMMY(79251)),	ADAP0120	
			2 (IREPL,DUMMY(79501)),(NDEG,DUMMY(79751)),(NEVAL,DUMMY(80001))	ADAP0130	
			EXTERNAL F2,THLMX	ADAP0140	
			NNAX=50	ADAP0150	
			IF (IFPRO.EQ.2) GO TO 10	ADAP0160	
		C***	CALCULATE CONNECTION MATRIX OVER BASIS FUNCTIONS	ADAP0170	
			CALL PSYM (L,N,F2,L,N,F2,1,KALC,RES,B,NNAX)	ADAP0180	
			IF (INTER.EQ.1) CALL MATPR (N,0,IDENT,RES,EV,NNAX,LABEL,INDEX)	ADAP0190	
		C***	DIAGONALIZE TO GET CRUDE SYMMETRY VECTORS	ADAP0200	
			CALL EIGEN (RES,B,N,EV,N,SRNORM,NNAX)	ADAP0210	
			DO 1 I=1,N	ADAP0220	
			DO 1 J=1,N	ADAP0230	
			IF (DABS(RES(I,J))-C.CRTZRO) RES(I,J)=0.	ADAP0240	
			1 CONTINUE	ADAP0250	
		C***	FIND DEGENERACIES	ADAP0260	
			CALL DEGEN (N,EV,NDEG,NEVAL,CRTDEC)	ADAP0270	
		C***	SORT VECTORS BY DEGENERACY	ADAP0280	
			CALL SORT1 (RES,C,EV,NNAX,N,NDEG,NEVAL,NDEG)	ADAP0290	
			IF (INTER.EQ.1) CALL MATPR (N,1,IDENT,C,EV,NNAX,LABEL,INDEX)	ADAP0300	
			IF (IFPRO.EQ.1) RETURN	ADAP0310	
			10 DO 2 I=1,NEVAL	ADAP0320	
			2 (IREPL(I))=0	ADAP0330	
		C***	LINE UP PHASES WITH PROTOTYPE FUNCTIONS	ADAP0340	
			DO 3 IPAR=1,2	ADAP0350	
			IF (MPAR.EQ.0) GO TO 4	ADAP0360	
			IF (MPAR.EQ.1.AND.IPAR.EQ.2) GO TO 3	ADAP0370	
			IF (MPAR.EQ.-1.AND.IPAR.EQ.1) GO TO 3	ADAP0380	
			4 LV=LVALL	ADAP0390	
			IF (IPAR.EQ.2) LV=LVAC2	ADAP0400	
			IF (LV.EQ.0) GO TO 3	ADAP0410	
			LVD=2*LV+1	ADAP0420	
			CALL PSYM (LV,LVD,THLMX,L,N,F2,2,-KALC,RES,B,NNAX)	ADAP0430	
			CALL LINEUP (N,IPAR)	ADAP0440	
			3 CONTINUE	ADAP0450	
			DO 5 I=1,NEVAL	ADAP0460	
			IF (IREPL(I).EQ.0) GO TO 6	ADAP0470	
			5 CONTINUE	ADAP0480	
		C***	SORT VECTORS BY REPRESENTATION	ADAP0490	
			CALL SORT1 (C,B,EV,NNAX,N,NDEG,NEVAL,IREPL)	ADAP0500	
			IF (IFPRO.EQ.2) RETURN	ADAP0510	
		C***	UNDO ARBITRARY MIXING OF SETS BELONGING TO SAME REP.	ADAP0520	
			CALL CLEAN (C,M,NDEG,NEVAL,IREPL,NNAX,CRTZRO,0)	ADAP0530	
			RETURN	ADAP0540	
			6 WRITE (IOUT,7) (IREPL(I),I=1,NEVAL)	ADAP0550	
			7 FORMAT (// ' ONE OR MORE VECTORS NOT SYMMETRY-ADAPTED. CALCULATION	ADAP0560	
			1 ABANDONED. '//2015)	ADAP0570	
			1 STOP=1	ADAP0580	

PROGRAM: GRTHEORY	ROUTINE: ADAPT	PAGE: 12	PROGRAM: GRTHEORY	ROUTINE: PSYX	PAGE: 18
RETURN	ADAP0590		SUBROUTINE PSYM IL1,N1,F1,L2,M2,F2,IGRID,KALC,RES,R,NMAX)	PSYM0010	
END	ADAP0590		C**** CALCULATES MATRIX OF SYMMETRY OPERATOR OVER FUNCTIONS SPECIFIED	PSYM0020	
			C**** BY F1, F2	PSYM0030	
			IMPLICIT REAL*8 (A-H,O-Z)	PSYM0040	
			COMMON/OUT1/POS(3,30),RAD(US(30),THETA(30),PHI(30),NATDMS,NATYPE,	PSYM0050	
			1,NUT(10),MULT(10),TTSYN(10),ITYP(10,10),ITYPE(30),ITYPE(10),	PSYM0060	
			2,INVT(30)	PSYM0070	
			COMMON/CONTROL/CRTRD,CRTRD,ISTOP,INTER,LFLDNC(10)	PSYM0080	
			COMMON/GRID/GRID(30,3),CINGR(1447),PHIGR(1447),NGRID,NDIV	PSYM0090	
			COMMON/ALPZCR,SQZOR,M,Y,LC	PSYM0100	
			DIMENSION FUNCT1(100),FUNCT2(100)	PSYM0110	
			DIMENSION IAPLAG(30)	PSYM0120	
			DIMENSION RESCNMAX,M1,RINMAX)	PSYM0130	
			EXTERNAL F1,F2	PSYM0140	
			ROOTHF=DSQRT(0.500)	PSYM0150	
			IBASE=0	PSYM0160	
			IF (SGRID.EQ.2) IBASE=NGRID	PSYM0170	
			4) ISUB=M1*N2	PSYM0180	
			DO 3) I=1,ISUB	PSYM0190	
			3) R(1)=0.	PSYM0200	
			IND=IBASE	PSYM0210	
			DO 4) I=1,NATDMS	PSYM0220	
			4) IAPLAG(I)=0	PSYM0230	
			C**** LOOP OVER GRID POINTS	PSYM0240	
			DO 1) IA=1,NATDMS	PSYM0250	
			IF (RADIUS(IA).LE.CRVZRD) GO TO 1	PSYM0260	
			IF (1/ABS(KALC).NE.0) GO TO 3	PSYM0270	
			C**** IF /KALC/=0, TAKE ONLY ONE ATOM FROM EACH INVERSION-SYMMETRIC	PSYM0280	
			C**** PAIR. CONTRIBUTION TO MATRIX ELEMENT FROM OTHER ATOM IS IDENTICAL	PSYM0290	
			IF (IAPLAG(IA).EQ.0) GO TO 5	PSYM0300	
			IND=IND+NDIV	PSYM0310	
			GO TO 1	PSYM0320	
			5) JA=INVT(IA)	PSYM0330	
			IAPLAG(JA)=1	PSYM0340	
			3) RCR=L.DD/RGRD(IA,IGRID)	PSYM0350	
			SQZOR=ROOTHF*RCR	PSYM0360	
			DO 2) NPT=1,NDIV	PSYM0370	
			IND=IND+1	PSYM0380	
			CALL F1 (LI,N1,FUNCT1,IND)	PSYM0390	
			IF (KALC.LT.0) GO TO 39	PSYM0400	
			DO 25) I=1,N1	PSYM0410	
			25) FUNCT2(I)=FUNCT1(I)	PSYM0420	
			GO TO 24	PSYM0430	
			39) CALL F2 (L2,M2,FUNCT2,IND)	PSYM0440	
			24) ISUB=0	PSYM0450	
			DO 10) I=1,N1	PSYM0460	
			LIM=I	PSYM0470	
			IF (KALC.LT.0) LIM=M2	PSYM0480	
			DO 10) J=1,LIM	PSYM0490	
			ISUB=ISUB+1	PSYM0500	
			10) R(ISUB)=R(ISUB)+FUNCT1(I)*FUNCT2(J)	PSYM0510	
			2) CONTINUE	PSYM0520	
			1) CONTINUE	PSYM0530	
			C**** CLEAN UP MATRIX	PSYM0540	
			ISUB=0	PSYM0550	
			DO 20) I=1,M1	PSYM0560	
			LIM=I	PSYM0570	
			IF (KALC.LT.0) LIM=M2	PSYM0580	

PROGRAM: GPTHEORY	ROUTINE: PSYN	PAGE: 12	PROGRAM: GPTHEORY	ROUTINE: INLX	PAGE: 20
DO 20 J=1,LIM	PSYN0590		SUBROUTINE INLX (L,LL,FUNCT,IND)	THLM0010	
ISUB=ISUB+1	PSYN0600	C***	CALCULATES FUNCTION VALUES OF REAL SPHERICAL HARMONICS CENTERED	THLM0020	
IF (DABS(RISUB),LE.CRFZRO) R(ISUB)=0.	PSYN0610	C***	AT ORIGIN, EVALUATED AT GRID POINT 'IND'	THLM0030	
RESCT,JF*R(ISUB)	PSYN0620		IMPLICIT REAL*8 (A-H,O-Z)	THLM0040	
IF (KALC.GT.0) RESIJ,II*R(ISUB)	PSYN0630		COMMON/GIRU/RGRID130,31,CTHGR114471,PHIGR114471,NGRTO,NDIV	THLM0050	
20 CONTINUE	PSYN0640		COMMON/ALP/RCR,SOZOR,W,Y,EC	THLM0060	
RETURN	PSYN0650		DIMENSION FUNCT (40)	THLM0070	
END	PSYN0660		DIMENSION COSHPI301,SINHPI301	THLM0080	
			LC=L	THLM0090	
			W=CTHGR(IND)	THLM0100	
			RADIX=1.00-W*W	THLM0110	
			IF (RADIX.LE.1.0-8) RADIX=0.	THLM0120	
			Y=OSQRT(RADIX)	THLM0130	
			PHIX=PHIGR(IND)	THLM0140	
			FUNCT(1)=THETA(XI)*SOZOR	THLM0150	
			IF (L.EQ.0) RETURN	THLM0160	
			SINHPI(1)=OSIN(PHIX)	THLM0170	
			COSHPI(1)=DCOS(PHIX)	THLM0180	
			DO 1 M=1,L	THLM0190	
			IF (M.EQ.1) GO TO 10	THLM0200	
			SINHPI(M)=SINHPI(M-1)*COSHPI(1)+COSHPI(M-1)*SINHPI(1)	THLM0210	
			COSHPI(M)=COSHPI(M-1)*COSHPI(1)-SINHPI(M-1)*SINHPI(1)	THLM0220	
		10	M2=M*M	THLM0230	
			FACTOR=THETA(XI)*RCR	THLM0240	
			IF (2*(M/2),NE.M) FACTOR=-FACTOR	THLM0250	
			FUNCT(M2+1)=FACTOR*COSHPI(M)	THLM0260	
		1	FUNCT(M2)=FACTOR*SINHPI(M)	THLM0270	
			RETURN	THLM0280	
			END		

PROGRAM: GRTHEDBY	ROUTINE: THETAX	PAGE: 21	PROGRAM: GRTHEDBY	ROUTINE: MATPAT	PAGE: 22
FUNCTION THETAX (M)	THET0010	SUBROUTINE MATPAT (N, IEV, IDENT, A, B, NMAX, LABEL, INDEX)	MATP0010		
C*** CALCULATION OF VALUE OF THETAX(L,M) (X)	THET0020	MATRIX OUTPUT ROUTINE	MATP0020		
IMPLICIT REAL*8 (A-H,O-Z)	THET0030	IMPLICIT REAL*8 (A-H,O-Z)	MATP0030		
COMMON/ALP1/THETAC(1224),FACT(50)	THET0040	REAL*8 LABEL	MATP0040		
COMMON/ALP2/ACR,SQZOR,X,Y,L	THET0050	COMMON/FILES/INPUT,IDUT	MATP0050		
ISUB=8*M+4*L*(L+1)	THET0060	DIMENSION IDENT(20),A(NMAX,N),B(N)	MATP0060		
MAX(P=(L-M)/2+1	THET0070	DIMENSION LABEL(NMAX),INDEX(NMAX)	MATP0070		
LM2=L-M+2	THET0080	NFIN=0	MATP0080		
THETAX=0.	THET0090	NINIT=NFIN+1	MATP0090		
IF (DABS(X).LE.1.0-7) GO TO 60	THET0100	NFIN=NFIN+6	MATP0100		
IF (DABS(Y).LE.1.0-7) GO TO 50	THET0110	IF (NFIN.GT.N) NFIN=N	MATP0110		
GO 34 IF=1,MATP	THET0120	WRITE (IDUT,10) IDENT,NINIT,NFIN	MATP0120		
34 THETAX=THETAX+THETAC(ISUB+1)*X**ILM2-2*(P)	THET0130	FORMAT (1ML,20A4//) COLUMNS ',E2,' TO ',L2I	MATP0130		
THETAX=THETAX+Y**H	THET0140	IF (IEV.EQ.0) GO TO 8	MATP0140		
RETURN	THET0150	WRITE (IDUT,5)	MATP0150		
60 IF (2*(L-M)/2).EQ.L-M) THETAX=THETAC(ISUB+MAX(P)	THET0160	FORMAT (/ ' EIGENVALUES')	MATP0160		
RETURN	THET0170	WRITE (IDUT,6) (BII),I=NINIT,NFIN)	MATP0170		
50 IF (M.EQ.0) THETAX=THETAC(ISUB+1)*X**L	THET0180	FORMAT (/J2X,1P6E20.10)	MATP0180		
RETURN	THET0190	WRITE (IDUT,7)	MATP0190		
END	THET0200	FORMAT (/ ' EIGENVECTORS')	MATP0200		
		DO 11 I=1,N	MATP0210		
		WRITE (IDUT,2) LABEL(I),INDEX(I),IAC(I,J),J=NINIT,NFIN)	MATP0220		
		FORMAT (/1X,A7,1MC,12,1HE,1P6E20.10)	MATP0230		
		IF (NFIN.LT.N) GO TO 1	MATP0240		
		RETURN	MATP0250		
		END	MATP0260		

PROGRAM: GPHERRY	RUNTIME: EIGEN	PAGE: 23	PROGRAM: GPHERRY	ROUTINE: EIGEN	PAGE: 24
SUBROUTINE EIGEN (A,S,N,EV,NM,ONORMZ,NMAX)	EIGE0010	192 M=K+1	EIGE0590		
C**** JACOBI METHOD - DIAGONALIZES REAL, SYMMETRIC MATRIX 'A', STORES	EIGE0020	IF (M.GT.N) GO TO 199	EIGE0600		
C**** EIGENVALUES IN 'EV', EIGENVECTORS IN 'S', 'EV' AND 'S' ARE SORTED	EIGE0030	DO 198 I=N,N	EIGE0610		
C**** IN ORDER OF DECREASING EIGENVALUE. EIGENVECTORS ARE COLUMNS OF	EIGE0040	TINTL=A(I,I)	EIGE0620		
IMPLICIT REAL*8 (A-H,O-Z)	EIGE0050	A(I,I)=A(I,I)+SINT*TINTL*COST	EIGE0630		
DIMENSION A(NMAX,N),S(NMAX,N),EV(NMAX)	EIGE0060	198 A(J,I)=A(J,I)+COST*TINTL*SINT	EIGE0640		
RDDTMP=DSQRT(10.500)	EIGE0070	199 DO 215 I=1,M	EIGE0650		
RHO=.0001	EIGE0080	TINTL=S(I,I)	EIGE0660		
DO 30 I=1,N	EIGE0090	S(I,K)=S(I,K)+SINT*S(I,K)*COST	EIGE0670		
S(I,I)=1.00	EIGE0100	215 S(I,J)=S(I,J)+COST*TINTL*SINT	EIGE0680		
IF (I.EQ.N) GO TO 31	EIGE0110	COST2=COST+COST	EIGE0690		
K=I+1	EIGE0120	SINT2=SINT*SINT	EIGE0700		
DO 30 J=K,N	EIGE0130	V4=.0001	EIGE0710		
S(I,J)=0.	EIGE0140	A(J,J)=V1*COST2+V3*SINT2-V4	EIGE0720		
30 S(J,I)=0.	EIGE0150	A(N,K)=V1*SINT2+V3*COST2+V4	EIGE0730		
31 TINTL=0.	EIGE0160	A(I,K)=0.	EIGE0740		
DO 60 I=2,N	EIGE0170	240 CONTINUE	EIGE0750		
K=I-1	EIGE0180	C**** THRESHOLD TEST	EIGE0760		
DO 60 J=1,K	EIGE0190	IF (IND.NE.1) GO TO 260	EIGE0770		
60 TINTL=TINTL+A(I,J)+2	EIGE0200	IND=0	EIGE0780		
THR=DSQRT(2.00*TINTL)	EIGE0210	GO TO 95	EIGE0790		
B=N	EIGE0220	260 IF (THR.GT.ONORMZ) GO TO 90	EIGE0800		
ONORMZ=THR*B	EIGE0230	C**** SORTING OF EIGENVALUES AND EIGENVECTORS	EIGE0810		
IND=0	EIGE0240	DO 285 K=1,N	EIGE0820		
90 THR=THR/B	EIGE0250	DO 285 J=K,N	EIGE0830		
IF (N.EQ.1) GO TO 310	EIGE0260	IF (A(K,K).GE.A(I,J)) GO TO 285	EIGE0840		
C**** LOOP THROUGH OFF-DIAGONAL ELEMENTS	EIGE0270	TEMP=A(K,K)	EIGE0850		
95 DO 240 K=2,N	EIGE0280	A(K,K)=A(I,J)	EIGE0860		
KK=K-1	EIGE0290	A(I,J)=TEMP	EIGE0870		
DO 240 J=1,KK	EIGE0300	DO 280 L=1,N	EIGE0880		
IF (DABS(A(I,K)).LE.THR) GO TO 240	EIGE0310	TEMP=S(L,K)	EIGE0890		
IND=1	EIGE0320	S(L,K)=S(L,J)	EIGE0900		
V1=A(I,J)	EIGE0330	280 S(L,J)=TEMP	EIGE0910		
V2=A(I,K)	EIGE0340	285 CONTINUE	EIGE0920		
V3=A(K,K)	EIGE0350	310 DO 300 I=1,N	EIGE0930		
UM=0.500*(V1-V3)	EIGE0360	EV(I)=A(I,I)	EIGE0940		
IF (UM.NE.0.1) GO TO 155	EIGE0370	DO 300 J=1,N	EIGE0950		
OMEGA=-1.00	EIGE0380	300 A(I,J)=S(I,J)	EIGE0960		
GO TO 160	EIGE0390	RETURN	EIGE0970		
155 OMEGA=-V2/DSQRT(V2*V2+UM*UM)*DSIGN(1.0,UM)	EIGE0400	END	EIGE0980		
160 RADIX=1.00-OMEGA*OMEGA	EIGE0410				
IF (RADIX.LT.1.0-10) RADIX=0.	EIGE0420				
SINT=ROOTH*OMEGA/DSQRT(1.00+DSQRT(RADIX))	EIGE0430				
RADIX=1.00-SINT*SINT	EIGE0440				
IF (RADIX.LT.1.0-10) RADIX=0.	EIGE0450				
COST=DSQRT(RADIX)	EIGE0460				
L=J-1	EIGE0470				
DO 185 I=1,L	EIGE0480				
TINTL=A(I,I)	EIGE0490				
A(I,K)=A(I,J)*SINT+TINTL*COST	EIGE0500				
185 A(I,J)=A(I,J)*COST-TINTL*SINT	EIGE0510				
L=J+1	EIGE0520				
M=K-1	EIGE0530				
IF (L.GT.M) GO TO 192	EIGE0540				
DO 191 I=L,M	EIGE0550				
TINTL=A(I,I)	EIGE0560				
A(I,K)=A(I,J)*SINT+TINTL*COST	EIGE0570				
191 A(J,I)=A(J,I)*COST-TINTL*SINT	EIGE0580				

PROGRAM: GPTRCXY	ROUTINE: DEGEN	PAGE: 25	PROGRAM: GPTRCXY	ROUTINE: SORT1	PAGE: 26
SUBROUTINE DEGEN (NMAT, EV, NDEG, NEVAL, CRTDEG)	DEGE0010		SUBROUTINE SORT1 (RES, C, EV, NMAX, NBASIS, NDEG, NEVLL, IARRAY)	SORT0010	
C*** DETERMINE NUMBER OF DISTINCT EIGENVALUES AND THEIR DEGENERACIES	DEGE0020		C*** SORTS VECTORS IN ASCENDING ORDER OF ELEMENTS OF 'IARRAY'.	SORT0020	
IMPLICIT REAL*8 (A-H, O-Z)	DEGE0030		C*** IF NEVLL.GT.0, SORTED VECTORS ARE STORED IN 'C', OTHERWISE IN	SORT0030	
DIMENSION EV(NMAT), NDEG(NMAT)	DEGE0040		C*** 'RES', 'IARRAYS' 'EV', 'NDEG', AND 'IARRAY' ARE REORDERED.	SORT0040	
IEV=1	DEGE0050		IMPLICIT REAL*8 (A-H, O-Z)	SORT0050	
NEVAL=0	DEGE0060		DIMENSION RES(NMAX, NBASIS), C(NMAX, NBASIS), EV(NBASIS), NDEG(NBASIS)	SORT0060	
10 LDEG=1	DEGE0070		DIMENSION IARRAY(NBASIS), NOG(100)	SORT0070	
1PI=IEV+1	DEGE0080		DIMENSION NOT(100), EV(100)	SORT0080	
NEVAL=NEVAL+1	DEGE0090		NEVAL=IABS(NEVLL)	SORT0090	
IF (1PI.GT.NMAT) GO TO 30	DEGE0100		JBASE=0	SORT0100	
DO 3 J=1PI, NMAT	DEGE0110		IEV=0	SORT0110	
IF (EV(IPI)/EV(J)).GT.CRTDEG.OR.(EV(IPI)/EV(J)).LT.1.D0/CRYDEG) GO TO 4	DEGE0120		IAMAX=0	SORT0120	
3 LDEG=LDEG+1	DEGE0130		DO 2 I=1, NEVAL	SORT0130	
4 NDEG(NEVAL)=LDEG	DEGE0140		IF (IARRAY(I).GT.IAMAX) IAMAX=IARRAY(I)	SORT0140	
IEV=IEV+LDEG	DEGE0150		2 CONTINUE	SORT0150	
GO TO 10	DEGE0160		DO 50 I=1, IAMAX	SORT0160	
30 NDEG(NEVAL)=LDEG	DEGE0170		JBASE=0	SORT0170	
NSUM=0	DEGE0180		DO 51 JEV=1, NEVAL	SORT0180	
DO 5 I=1, NEVAL	DEGE0190		ND=NDEG(JEV)	SORT0190	
6 NSUM=NSUM+NDEG(I)	DEGE0200		NDB=IARRAY(JEV)	SORT0200	
IF (NSUM.GT.NMAT) NEVAL=NEVAL-1	DEGE0210		IF (NDB.NE.I) GO TO 51	SORT0210	
RETURN	DEGE0220		IEV=IEV+1	SORT0220	
END	DEGE0230		NDI=IEV+NDB	SORT0230	
			NDC(IEV)=ND	SORT0240	
			DO 52 M=1, ND	SORT0250	
			IBM=JBASE+M	SORT0260	
			JBM=JBASE+M	SORT0270	
			EV(IBM)=EV(JBM)	SORT0280	
			DO 53 J=1, NBASIS	SORT0290	
			52 C(J, IBM)=RES(J, JBM)	SORT0300	
			IBASE=IBASE+ND	SORT0310	
			51 JBASE=JBASE+ND	SORT0320	
			50 CONTINUE	SORT0330	
			55 J=0	SORT0340	
			DO 53 I=1, NEVAL	SORT0350	
			IARRAY(I)=NDI(I)	SORT0360	
			NDEG(I)=NOG(I)	SORT0370	
			NDI=NDEG(I)	SORT0380	
			DO 54 M=1, NDI	SORT0390	
			J=J+1	SORT0400	
			54 EV(J)=EV(I)	SORT0410	
			53 CONTINUE	SORT0420	
			IF (NEVLL.GT.0) RETURN	SORT0430	
			DO 1 I=1, NBASIS	SORT0440	
			DO 1 J=1, NBASIS	SORT0450	
			1 RES(I, J)=C(I, J)	SORT0460	
			RETURN	SORT0470	
			END	SORT0480	

PROGRAM: GRTHEORY	ROUTINE: LINCOM	PAGE: 27	PROGRAM: GRTHEORY	ROUTINE: LINCOM	PAGE: 28
SUBROUTINE LINCOM (RES,EV,NDEG,NDIM,LL,C,R,NEVAL,IREFL,NREP,LDEG) LINCO010			DO 41 K=1,LL		LINC0590
C*** TRANSFORMS MATRIX OF PROTOTYPE FUNCTIONS WITH CRUDE SYMMETRY VECTOR	LINC0010		ISUB=ISUB+1		LINC0600
IMPLICIT REAL*8 (A-H,O-Z)	LINC0020		IF (R(ISUB).EQ.0.) GO TO 41		LINC0610
DIMENSION IREFL(33),LDEG(14),ISKIP(33)	LINC0030		A=A+C(R*(IND)*R(IISUB)*C(LPNIND))		LINC0620
DIMENSION RES(NDIM,LL),EV(11),C(NDIM),R(NDIM),NDEG(NDIM)	LINC0040		41 CONTINUE		LINC0630
COMMON/CONTRL/CRYZRO,CRYDEG,ISTOP,INTER,LFLON(10)	LINC0050		IF (DABS(A).LE.CRYZRO) A=0.		LINC0640
ISUB=0	LINC0060		4 RESIM,N)=A		LINC0650
DO 7 J=1,LL	LINC0070		IF (ISKIP(J).EQ.1) GO TO 3		LINC0660
ISKIP(J)=0	LINC0080		DO 40 M=J,NDJ		LINC0670
IREFL(J)=0	LINC0090		IF (DABS(RESIM,11).GT.CRYZRO) GO TO 42		LINC0680
DO 7 J=1,LL	LINC0100		40 CONTINUE		LINC0690
ISUB=ISUB+1	LINC0110		GO TO 3		LINC0700
7 K(IISUB)*RES(I,J)	LINC0120		42 ISKIP(J)=1		LINC0710
NREP=0	LINC0130		IREFL(J)=NREP		LINC0720
50 DO 5 I=1,LL	LINC0140		3 J=J+NDJ		LINC0730
DO 5 J=1,I	LINC0150		I=I+NDI		LINC0740
5 RES(I,J)=0.	LINC0160		1 CONTINUE		LINC0750
J=1	LINC0170		DO 6 I=1,LL		LINC0760
DO 1 I=1,NEVAL	LINC0180		DO 6 J=1,I		LINC0770
NDI=NDEG(I)	LINC0190		6 RES(J,1)=RES(I,J)		LINC0780
INDIC=0	LINC0200		RETURN		LINC0790
NDI=I+NDI-1	LINC0210		END		LINC0800
IF (ISKIP(I).NE.0) GO TO 51	LINC0220				
NREP=NREP+1	LINC0230				
IREFL(I)=NREP	LINC0240				
LDEG(NREP)=NDI	LINC0250				
C*** CALCULATE BLOCKS ALONG DIAGONAL	LINC0260				
51 DO 2 M=1,NDI	LINC0270				
NIND=(M-1)*NDIM	LINC0280				
DO 2 M=1,M	LINC0290				
A=0.	LINC0300				
ISUB=0	LINC0310				
NIND=(M-1)*NDIM	LINC0320				
DO 21 L=1,LL	LINC0330				
LPNIND=L+NIND	LINC0340				
DO 21 K=1,LL	LINC0350				
ISUB=ISUB+1	LINC0360				
IF (R(ISUB).EQ.0.) GO TO 21	LINC0370				
A=A+C(R*(MIND)*R(IISUB)*C(LPNIND))	LINC0380				
21 CONTINUE	LINC0390				
IF (DABS(A).LE.CRYZRO) A=0.	LINC0400				
2 RESIM,N)=A	LINC0410				
J=NDI+1	LINC0420				
C*** CALCULATE OFF-DIAGONAL BLOCKS WITH SAME DEGENERACY	LINC0430				
IF (I.EQ.NEVAL) GO TO 1	LINC0440				
IPI=I+1	LINC0450				
DO 3 J=IPI,NEVAL	LINC0460				
NDJ=NDEG(J)	LINC0470				
IF (NDI.NE.NDJ) GO TO 3	LINC0480				
NDJI=J+NDJ-1	LINC0490				
DO 4 M=J,NDJI	LINC0500				
NIND=(M-1)*NDIM	LINC0510				
DO 4 M=1,NDI	LINC0520				
A=0.	LINC0530				
ISUB=0	LINC0540				
NIND=(M-1)*NDIM	LINC0550				
DO 41 L=1,LL	LINC0560				
LPNIND=L+NIND	LINC0570				

PROGRAM: GRIPEORY	ROUTINE: WIG3J	PAGE: 29	PROGRAM: GRIPEORY	ROUTINE: WIG3J	PAGE: 30
SUBROUTINE WIG3J (L1,L2,L3,(STOP)	WIG30010		SUM=SUM+ PHASE / (FACT(KP1)*FACT(N1-K)*FACT(N3-K)*FACT(N4-K)	WIG30590	
C**** CALCULATE ALL NON-REDUNDANT 3-J SYMBOLS FOR GIVEN L-VALUES	WIG30020		1 *FACT(N6-K)*FACT(N7-K))	WIG30600	
IMPLICIT REAL*8 (A-H,O-Z)	WIG30030		5 PHASE=-PHASE	WIG30610	
COMMON/WIGMER/SSJ(17,17),ZSJ(17,17)	WIG30040		6 PROD=FACTOR*SUM	WIG30620	
COMMON/F(LS/INPUT,EDUT	WIG30050		IF (IPABS(PROD).LE.1.0D-8) PROD=0.	WIG30630	
COMMON/ALP1/INETACT(224),FACT(50)	WIG30060		IF (MARK.EQ.0) GO TO 3	WIG30640	
IF (L3.GT.L1+L2.OR.L3.LT.(ABS(L1-L2)+1) GO TO 100	WIG30070		23J(MSUB1,M3P1)=PROD	WIG30650	
C**** CURRENT VERSION ASSUMES L1=L2=EVEN	WIG30080		GO TO 2	WIG30660	
IF (L1.NE.L2) GO TO 100	WIG30090		3 S3J(MSUB1,M3P1)=PROD	WIG30670	
MARK=0	WIG30100		2 CONTINUE	WIG30680	
L3MF=L3/2	WIG30110		RETURN	WIG30690	
IF (2*L3MF.NE.L3) MARK=1	WIG30120		C**** ERROR MESSAGE	WIG30700	
LP1=L3+1	WIG30130		100 WRITE (IOUT,101)	WIG30710	
LIP1=L1+1	WIG30140		101 FORMAT (77* 'ILLEGAL ARGUMENT IN WIG3J')	WIG30720	
C**** LOOP OVER RESTRICTED RANGE OF M-VALUES	WIG30150		1STOP=1	WIG30730	
DO 2 M3P1=1,LP1	WIG30160		RETURN	WIG30740	
M3=M3P1-1	WIG30170		END	WIG30750	
M02=M3/2	WIG30180				
M1MIN=M02+1	WIG30190				
IF (2*M02.NE.M3) M1MIN=M02+2	WIG30200				
DO 2 M1P1=M1MIN,LIP1	WIG30210				
M1=M1P1-1	WIG30220				
M2=M3-M1	WIG30230				
MSUB1=ABS(M2-M1)/2+1	WIG30240				
C**** COMPUTATION OF 3-J SYMBOL FOR GENERAL ARGUMENTS	WIG30250				
N1=L1+L2-L3	WIG30260				
N11=L1-L2+L3	WIG30270				
N12=L2-L1+L3	WIG30280				
N2=L1+M1	WIG30290				
N3=L1-M1	WIG30300				
N4=L2+M2	WIG30310				
N5=L2-M2	WIG30320				
N6=L3+M3	WIG30330				
N7=L3-M3	WIG30340				
N8=L1+L2+L3+1	WIG30350				
FACTOR=(FACT(N1+1)*FACT(N11+1)*FACT(N12+1)/FACT(MB+L3)*FACT(N2+1)	WIG30360				
1 *FACT(N3+1)*FACT(N4+1)*FACT(N5+1)*FACT(N6+1)*FACT(N7+1)	WIG30370				
FACTOR=DSQRT(FACTOR)	WIG30380				
LLM=L1-L2-M3	WIG30390				
IF (2*ILLM/2).NE.LLM) FACTOR=-FACTOR	WIG30400				
N6=L3-L2+M1	WIG30410				
N7=L3-L1-M2	WIG30420				
KMIN=MAX(0,-N6,-N7)	WIG30430				
KMAX=MIND(1,N3,N4)	WIG30440				
N1=N1+1	WIG30450				
N3=N3+1	WIG30460				
N4=N4+1	WIG30470				
N6=N6+1	WIG30480				
N7=N7+1	WIG30490				
SUM=0.	WIG30500				
IF (KMIN.GT.0) GO TO 4	WIG30510				
KMIN=1	WIG30520				
SUM=L.D0/(FACT(N1)*FACT(N3)*FACT(N4)*FACT(N6)*FACT(N7))	WIG30530				
IF (KMAX.EQ.0) GO TO 6	WIG30540				
4 PHASE=L.D0	WIG30550				
IF (2*(KMIN/2).NE.KMIN) PHASE=-1.D0	WIG30560				
DO 5 K=KMIN,KMAX	WIG30570				
KP1=K+1	WIG30580				

```

FUNCTION SYMB3J (M1,M2,M3)
C*** ASSIGNS PROPER PHASE TO 3-J SYMBOL WHERE 13 IS DDD.
IMPLICIT REAL*8 (A-H,O-Z)
COMMON/HIGNER/SSJ(17,17),Z9J(17,17)
M3P1=IABS(M3)+1
MSUB=IABS(M2-M1)/2+1
SYMB3J=Z9J(MSUB,M3P1)
IF (M2.GT.M1) SYMB3J=-SYMB3J
RETURN
END

```

```

SYMB0010
SYMB0020
SYMB0030
SYMB0040
SYMB0050
SYMB0060
SYMB0070
SYMB0080
SYMB0090
SYMB0100

```

```

SUBROUTINE ANTSYM (L3,LL3,FUNCT,IND)
C*** CALCULATES VALUE OF ANTISYMMETRIC PRODUCT OF PROTOTYPE FUNCTIONS
IMPLICIT REAL*8 (A-H,O-Z)
COMMON/ASYM7/L,NGRID,L3P1,LPI,LL
DIMENSION FUNCT(50),SSINI(33),SSIN2(33),SCOS1(33),SCOS2(33),
1 FUN1(50),FUN2(50)
DATA ROOTHF/0.7071067811865475/,RTHF0/7.0710678118654752/
CALL FHLNR (L,LL,FUN1,IND)
IND2=IND-NGRID
CALL FHLNR (L,LL,FUN2,IND2)
SSINI(LPI)=0.
SSIN2(LPI)=0.
SCOS1(LPI)=FUN1(1)
SCOS2(LPI)=FUN2(1)
PHASE=1.00
DO 1 M=1,L
PHASE=-PHASE
M2=2*M
M2P1=M2+1
NM=LPI-M
NP=LPI+M
SSINI(NM)=-FUN1(M2)*ROOTHF
SSIN2(NM)=-FUN2(M2)*ROOTHF
SSINI(MP)=-PHASE*SSINI(NM)
SSIN2(MP)=-PHASE*SSIN2(NM)
SCOS1(MN)=FUN1(M2P1)*ROOTHF
SCOS2(MN)=FUN2(M2P1)*ROOTHF
SCOS1(MP)=PHASE*SCOS1(MN)
1 SCOS2(MP)=PHASE*SCOS2(MN)
DO 2 M3P1=1,L3P1
M3=M3P1-1
IF (M3.GT.0) GO TO 4
SUM=0.
PHASE=-1.00
DO 5 M1=1,L
K=M1-LPI
PHASE=-PHASE
SYMB=SYMB3J(M1,-M1,0)*PHASE
5 SUM=SUM+SSINI(K)*SCOS2(K)-SCOS1(K)*SSIN2(K)*SYMB
FUNCT(1)=SUM*10.00
GO TO 2
4 M2=2*M3
SUM=0.
SUMC=0.
DO 3 K=1,LL
M1=K-LPI
M2=M3-M1
K2=M2+LPI
IF (K2.GT.LL) GO TO 3
SYMB=SYMB3J(M1,M2,M3)
SUM=SUM+SCOS1(K)*SSIN2(K2)+SSINI(K)*SCOS2(K2)*SYMB
SUMC=SUMC+SSINI(K)*SSIN2(K2)-SCOS1(K)*SCOS2(K2)*SYMB
3 CONTINUE
FUNCT(M32)=-SUMC*RTHF0
FUNCT(M32+1)=-SUMA*RTHF0
2 CONTINUE
RETURN
END

```

PROGRAM: GPHICOR	ROUTINE: LINEUP	PAGE: 33	PROGRAM: GPHICOR	ROUTINE: LINEUP	PAGE: 34
SUBROUTINE LINEUP (NDASIS,IPAR)	LINE0010		ISUBK=ISUB1+K	LINE0590	
C*** LINE UP SYMMETRY VECTORS WITH PROTOTYPE FUNCTIONS	LINE0020		IF (COPROD(SUBK),EQ.0.) GO TO 40	LINE0600	
IMPLICIT REAL*8 (A-H,O-Z)	LINE0030		KSUB=K-NDIM	LINE0610	
COMMON/CONTROL/CRTRD,CRTOG,ISTOP,INTER,LFLOW(10)	LINE0040		DO 4 L=1,NDASIS	LINE0620	
COMMON/FUNTS/ANREP,NOEGAI(4),IANGA(5,14),IREPA(32),IORDER,LVAL1,	LINE0050		ASUB=KSUB+NDIM	LINE0630	
1 LVAL2,IRTYPE(14),IRPAR(14),IORD,IRLOW(14),	LINE0060		5 A=A+COPROD(SUBK)*P(KSUB)*C(JSUB1+L)	LINE0640	
COMMON/PROTGD/COPROD(1802),EVP(36),IREPL(25),NEVALP(2)	LINE0070		60 CONTINUE	LINE0650	
COMMON/PHASE/IFLAG(20),IPHASE	LINE0080		IF (OABS(A).LE.CRTZRO) A=0.	LINE0660	
COMMON/FILES/INPUT,IOUT	LINE0090		7 RECON(M,N)=A	LINE0670	
COMMON/WORK/DUMMY(8000)	LINE0100		IF (M.GT.1) GO TO 9	LINE0680	
DIMENSION P(2500),EV(50),C(2500),RECON(5,5),CC(5,5),TEMP(5),	LINE0110		C*** TEST FOR NON-ZERO ELEMENT	LINE0690	
1 NDEGB(50),IREPLB(50)	LINE0120		DO 8 N=1,NDI	LINE0700	
0 DIMENSION IPHOL(8),IRTYPE(2),JFLAG(20)	LINE0130		IF (OABS(IRECON(1,N)).GT.1.0D-7) GO TO 19	LINE0710	
EQUIVALENCE IP,DUMMY(1,EV,DUMMY(2501)),CC,DUMMY(2551),	LINE0140		8 CONTINUE	LINE0720	
1 IRECON,DUMMY(5051),CC,DUMMY(50761),TEMP,DUMMY(51011),	LINE0150		LABEL=LABEL	LINE0730	
2 (IREPLB,DUMMY(79501)),(NDEGB,DUMMY(79751)),(NEVALB,DUMMY(80001))	LINE0160		GO TO 3	LINE0740	
DATA IPHOL/4HBASI,4HS FN,4HPRDT,4H FN,4HPRDT,4H FN,4H NOT,4HHNL	LINE0170		C*** IF INTERACTION IS NON-ZERO, LINE UP PHASES	LINE0750	
1G/	LINE0180		19 IF (IPHASE.NE.1.FR .IFLAG(LABEL,EQ.0) GO TO 40	LINE0760	
NDIM=50	LINE0190		IRTYPE(1)=IPHOL(1)	LINE0770	
IP=IPHASE+1	LINE0200		IRTYPE(2)=IPHOL(8)	LINE0780	
IRTYPE(1)=IPHOL(2+IPHASE+1)	LINE0210		40 IF (INTER.EQ.1) WRITE (IOUT,30) 1,IRTYPE	LINE0790	
IRTYPE(2)=IPHOL(2+IPHASE+2)	LINE0220		30 FORMAT (//A PHASE CONNECTION MATRIX FOR EIGENVALUE NO. ',I2,3X,244	LINE0800	
NEVAL=NEVALP(IPAR)	LINE0230		1, ' REPHASED.')	LINE0810	
LL=2+LVAL1+3	LINE0240		IREPLB(1)=LABEL	LINE0820	
IPASE=0	LINE0250		9 ALAM=0.	LINE0830	
ISUB=0	LINE0260		DO 10 N=1,NDI	LINE0840	
(FIPAR,EQ.1) GO TO 4	LINE0270		10 ALAM=ALAM+RECON(M,N)**2	LINE0850	
IRATE=NEVALP(1)	LINE0280		TEMP(M)=DSQRT(ALAM)	LINE0860	
ISUB=LL**2	LINE0290		IF (INTER.EQ.0) GO TO 50	LINE0870	
LL=2+LVAL2+1	LINE0300		IF (IPHASE.NE.1) SDTO 43	LINE0880	
4 J1=1	LINE0310		IF (IFLAG(LABEL,NF,0) LABEL=LABEL	LINE0890	
DO 61 IEV=1,NEVAL	LINE0320		IRTYPE(1)=IPHOL(3)	LINE0900	
61 JFLAG(IEV)=0	LINE0330		IRTYPE(2)=IPHOL(4)	LINE0910	
DO 1 I=1,NEVALB	LINE0340		43 IMH=1+M+1	LINE0920	
NDI=NDEGB(1)	LINE0350		WRITE (IOUT,35) IMH, (RECON(M,N),N=1,NDI)	LINE0930	
I1=1	LINE0360		35 FORMAT (1 SYD. FM. ',I2,1P5F2D.10)	LINE0940	
LABEL=0	LINE0370		50 IF (IPHASE.EQ.3) GO TO 1	LINE0950	
DO 3 IEV=1,NEVAL	LINE0380		5 CONTINUE	LINE0960	
LABEL=IREPL(IEV+IBASE)	LINE0390		DO 11 M=1,NDI	LINE0970	
NOEG=NOEGAI(LABEL)	LINE0400		DO 11 N=1,NDI	LINE0980	
IF (NOEG.NE.NDI.DX,LABEL,EQ.LABEL) GO TO 3	LINE0410		11 CC(M,N)=RECON(M,N)/TEMP(M)	LINE0990	
GO TO (51,200,300,51), IPI	LINE0420		IF (IPHASE=1) 20,26,28	LINE1000	
200 IF (IREPLB(1)).NE.0.AND.(IFLAG(LABEL,NF,0) GO TO 1	LINE0430		C*** ROTATE PROTOTYPE FUNCTIONS PARALLEL TO FIRST OCCURRENCE OF THAT	LINE1010	
GO TO 51	LINE0440		C*** REP. (LOWEST L VALUE)	LINE1020	
300 IF (JFLAG(1,1).EQ.1) GO TO 3	LINE0450		28 IF (IFLAG(LABEL,NF,0) GO TO 20	LINE1030	
IRTYPE(1)=IPHOL(3)	LINE0460		GO TO 27	LINE1040	
IRTYPE(2)=IPHOL(4)	LINE0470		28 IF (IFLAG(LABEL,NF,0) GO TO 3	LINE1050	
IF (IFLAG(LABEL,NF,0) GO TO 51	LINE0480		27 ISUB2=ISUB+1+LL	LINE1060	
IRTYPE(1)=IPHOL(1)	LINE0490		DO 21 M=1,LL	LINE1070	
IRTYPE(2)=IPHOL(2)	LINE0500		ISUB2=ISUB2+1	LINE1080	
C*** CALCULATE BLOCK CONNECTING SYMMETRY-ADAPTED SET OF ATOMIC FUNCTION	LINE0510		DO 22 N=1,NDI	LINE1090	
C*** WITH SYMMETRIZED PROTOTYPES OF SAME DEGENERACY	LINE0520		A=0.	LINE1100	
51 DO 5 M=1,NDI	LINE0530		ISUB1=ISUB2	LINE1110	
ISUB1=((1+M)*LL+ISUB	LINE0540		DO 23 N=1,NDI	LINE1120	
(N) 7 N=1,NDI	LINE0550		ISUB1=ISUB1+LL	LINE1130	
ISUB1=1+1+M+NDIM	LINE0560		23 A=A+COPROD(SUB1)*CC(M,N)	LINE1140	
A=0.	LINE0570		IF (OABS(A).LE.CRTZRO) A=0.	LINE1150	
GO 60 K=L,LL	LINE0580		22 TEMP(M)=A	LINE1160	

PROGRAM: GRITHEORY	ROUTINE: LIVEP	PAGE: 35	PROGRAM: GRITHEORY	ROUTINE: PHASE8	PAGE: 36
ISUB1=ISUB2	LINE1170		SUBROUTINE PHASE8	PHAS0010	
DO 24 N=1,NDI	LINE1180		C*** ROTATES PROTOTYPE FUNCTIONS INTO STANDARD PHASE (AXIS) CHOICE.	PHAS0020	
ISUB1=ISUB1+LL	LINE1190		IMPLICIT REAL*8 (A-H,O-Z)	PHAS0030	
24 COPROD(SUB1)=TEMPINI	LINE1200		REAL*8 LABEL	PHAS0040	
21 CONTINUE	LINE1210		COMMON/OUT3/NREP,NDEGA(14),LANGA(5,14),TREP(132),IORDER,LVAL1,	PHAS0050	
JFLAG(IEV)=1	LINE1220		1 LVAL2,IRTYPE(14),IRPART(14),IORD,IRLOW(14)	PHAS0060	
GO TO 3	LINE1230		COMMON/PROTCD/COPROD(1802),EVP(55),IREPL(25),NEVALP(2)	PHAS0070	
C*** ROTATE BASIS VECTORS PARALLEL TO PROTOTYPE VECTORS	LINE1240		COMMON/CONTROL/CRIZRO,CRIDEG,ISTOP,INTER,CFLOW(10)	PHAS0080	
20 JN2=J1+NDIM	LINE1250		COMMON/PHASE/IFLAG(120),IPHASE	PHAS0090	
DO 12 K=1,NBASIS	LINE1260		COMMON/UNGRAB/CRUI(33),CDD,CDDC	PHAS0100	
JN2=JN2+1	LINE1270		COMMON/FILES/INPUT,IOUT	PHAS0110	
DO 13 N=1,NDI	LINE1280		COMMON/MORF/DUMMY(8000)	PHAS0120	
A=0.	LINE1290		DIMENSION LABEL(50),INDEX(50),IDENT(20),IREPL(50),NDEGB(50)	PHAS0130	
JN1=JN2	LINE1300		DIMENSION C(50,50),REST(50,50),H(2500),EVI(50),ITITLE(20)	PHAS0140	
DO 14 M=1,NDI	LINE1310		EQUIVALENCE (C,DUMMY(2551)),LABEL,DUMMY(7875)),INDEX,	PHAS0150	
JN1=JN1+NDIM	LINE1320		1 DUMMY(7925)),IREPLB,DUMMY(7950)),NDEGB,DUMMY(7975)),	PHAS0160	
14 A=AVE(JN1)+CCIM,N1	LINE1330		2 IREVALB,DUMMY(8000)),IRES,DUMMY(1),EVI,DUMMY(2501)),B,DUMMY(5051))	PHAS0170	
IF IDABSI(1,4E,CRIZRO) A=0.	LINE1340		DIMENSION ITPAR(2)	PHAS0180	
13 TEMPINI=A	LINE1350		DATA IDENT/4M STA,4MNDAR,4MO PH,4HASE,4MCHOI,4MCE V,4MECTO,4HRS	PHAS0190	
JN1=JN2	LINE1360		1,12*4H /, ITITLE/4H FIM,4HAL P,4HROFO,4HTYPE,4H FUN,4HCTIO,	PHAS0200	
DO 15 N=1,NDI	LINE1370		2 4HMS, 4MEVEN,4H PAR,4HITY,10*4H /, ITPAR/4MEVEN,4H DDD/	PHAS0210	
JN1=JN1+NDIM	LINE1380		EXTERNAL THLMK,AUFUN	PHAS0220	
15 C(JN1)=TEMPINI	LINE1390		NNAK=50	PHAS0230	
12 CONTINUE	LINE1400		IPHASE=1	PHAS0240	
GO TO 1	LINE1410		NR=NREP	PHAS0250	
3 (I=I)+NDEG	LINE1420		LL1=2*LVAL1+1	PHAS0260	
1 J1=J1+NDI	LINE1430		DO 6 I=1,NREP	PHAS0270	
RETURN	LINE1440		IRLOW(I)=1	PHAS0280	
END	LINE1450		6 IFLAG(I)=0	PHAS0290	
			C*** IDENTIFY A(IG) PROTOTYPE FUNCTION BY LINING UP WITH L=0	PHAS0300	
			KALC=1	PHAS0310	
			IF (ICDD.NE.0) KALC=9	PHAS0320	
			CALL ADAPT (0.),THLMK,KALC,1,IDENT,01	PHAS0330	
			IRAG=IREPLB(1)	PHAS0340	
			IF (INTER.EQ.0) GO TO 40	PHAS0350	
			WRITE (IOUT,30)	PHAS0360	
			30 FORMAT (// 'REPRESENTATION INDICES'//)	PHAS0370	
			WRITE (IOUT,35) IRAG	PHAS0380	
			35 FORMAT (16F51)	PHAS0390	
			40 IFLAG(IRAG)=1	PHAS0400	
			IRLOW(IRAG)=0	PHAS0410	
			IF (ICDD.EQ.0) GO TO 10	PHAS0420	
			NR=NREP/2	PHAS0430	
			IFLAG(IRAG+NR)=1	PHAS0440	
			NEV1=NEVALP(1)	PHAS0450	
			NEVAL=NEVALP(2)	PHAS0460	
			LL2=2*LVAL2+1	PHAS0470	
			LODB=LVAL2	PHAS0480	
			LODEG=LL2	PHAS0490	
			C*** FIND THE A(10) FUNCTION	PHAS0500	
			NNON=0	PHAS0510	
			DO 1 I=1,NEVAL	PHAS0520	
			IRI=IREPL(I+NEV1)	PHAS0530	
			IF INDEG(IRI).EQ.11 NNON=NNON+1	PHAS0540	
			1 CONTINUE	PHAS0550	
			IVEC=1	PHAS0560	
			IF (NNON.EQ.1) GO TO 2	PHAS0570	
			IRAU=IRAG+NR	PHAS0580	

PROGRAM: GRINBODY	ROUTINE: PHASEB	PAGE: 31	PROGRAM: GRINBODY	ROUTINE: PHASEB	PAGE: 30
DD 3 (=L,NFVAL	PHAS0590		ISUB=ISUB+1	PHAS117C	
IF (IREPL(I+NEV11,NE,IRAU) GO TO 3	PHAS0600	22 C(I,J)=COPRO(I,ISUB)		PHAS118C	
IYEC=1	PHAS0610	IEV=NEVALP(I)		PHAS119C	
GO TO 2	PHAS0620	NEVALB=NEVALP(2)		PHAS1200	
3 CONTINUE	PHAS0630	DD 31 I=L,NEVALB		PHAS121C	
GO TO 100	PHAS0640	IREP=IREPL(I+IEV)		PHAS1220	
2 LS=LL1+LL2	PHAS0650	51 NOEGB(I)=NOEGB(IREP)		PHAS1230	
IBASE=LS+(IYEC-1)*LL2	PHAS0660	CALL LINEUP (LL2,I)		PHAS124C	
DD 4 I=1,LL2	PHAS0670	IPHASE=0		PHAS125C	
4 CAU(I)=COPRO(IBASE+I)*1.003	PHAS0680	ISUB=LS		PHAS126C	
C*** F(10 STANDARD PHASES FOR REFS. WHOSE PAIVITY MATCHES LOW L VALUE	PHAS0690	DD 23 J=L,LL2		PHAS127C	
10 LMAX=MAX0(LVAL1,LVAL2)	PHAS0700	DD 23 I=1,LL2		PHAS1280	
DD 5 L=L1,LMAX	PHAS0710	ISUB=ISUB+1		PHAS1290	
LDEN=2*L+1	PHAS0720	23 COPRO(ISUB)=C(I,J)		PHAS1300	
MPAR=-1	PHAS0730	C*** CLEAN UP FINAL VECTORS AND PRINT OUT		PHAS1310	
IF (1000.EQ.0.UR.2*(L/2).EQ.0) MPAR=1	PHAS0740	50 DD 55 (PAR=1,2		PHAS132C	
CALL ADAPT (L,LDEN,THLX,KALE,MPAR,IDENT,0)	PHAS0750	IF (IPAR.EQ.2) GO TO 56		PHAS1330	
IF (INTER.FO.0) GO TO 45	PHAS0760	IBASE=0		PHAS1340	
WRITE (IOUT,30)	PHAS0770	NEVAL=NEVALP(I)		PHAS1350	
WRITE (IOUT,35) (IREPLB(I),J=1,NEVALP)	PHAS0780	IEV=0		PHAS1360	
45 NEVAL=NEVALP(I)	PHAS0790	IS1=0		PHAS137C	
IEV=0	PHAS0800	LL=LL1		PHAS1380	
IF (2*(L/2).EQ.0.UR.1000.EQ.0) GO TO 7	PHAS0810	GO TO 57		PHAS1390	
IEV=NEVAL	PHAS0820	56 IF (1000.EQ.0) RETURN		PHAS1400	
NEVAL=NEVALP(2)	PHAS0830	IBASE=LS		PHAS1410	
7 DD 8 J=1,NEVAL	PHAS0840	NEVAL=NEVALP(2)		PHAS1420	
IREP=IREPL(J+IEV)	PHAS0850	IS1=LL1		PHAS1430	
IF (1IFLAG(IREP).EQ.1) GO TO 8	PHAS0860	LL=LL2		PHAS1440	
DD 9 K=L,NEVALB	PHAS0870	IEV=NEVALP(I)		PHAS1450	
IF (IREPLB(K),NE,IREP) GO TO 9	PHAS0880	57 (SUB=IBASE		PHAS1460	
IRLOW(IREP)=L	PHAS0890	DD 59 I=L,NEVAL		PHAS1470	
IFLAG(IREP)=1	PHAS0900	IREP=IREPL(I+IEV)		PHAS1480	
IF (1000.EQ.0) GO TO 20	PHAS0910	NDEGB(I)=NDEGB(IREP)		PHAS1490	
NRT=NR	PHAS0920	59 IREPLB(I)=IREP		PHAS1500	
IF (2*(L/2).NE.0) NRT=NR	PHAS0930	DD 50 J=L,LL		PHAS1510	
IFLAG(IREP+NRT)=1	PHAS0940	EV(I)=EVP(J+IS1)		PHAS1520	
20 DD 14 IN=1,NK	PHAS0950	DD 50 I=L,LL		PHAS1530	
IF (1IFLAG(IN).EQ.0) GO TO 4	PHAS0960	ISUB=ISUB+1		PHAS154C	
14 CONTINUE	PHAS0970	58 C(I,J)=COPRO(ISUB)		PHAS1550	
GO TO 15	PHAS0980	CALL CLEAN (C,LL,NDEGB,NEVAL,IREPLB,NMAX,CRTZRO,1)		PHAS1560	
9 CONTINUE	PHAS0990	IF (INTER.EQ.0) GO TO 65		PHAS1570	
8 CONTINUE	PHAS1000	ITITLE(I)=THPART(PART		PHAS1580	
5 CONTINUE	PHAS1010	CALL MATPRT (LL,3,ITITLE,C,EV,NMAX,LABEL,(NULX)		PHAS1590	
C*** REPHASE PROTOTYPES AGAINST EACH OTHER	PHAS1020	WRITE (IOUT,105)		PHAS1600	
DD 16 I=1,NR	PHAS1030	105 FORMAT (/' REPRESENTATION INDEX FOR EACH SET OF VECTORS'/)		PHAS1610	
IF (1IFLAG(I).EQ.0) GO TO 110	PHAS1040	WRITE (IOUT,35) (IREPL(I),I=1,NEVAL)		PHAS1620	
16 CONTINUE	PHAS1050	65 ISUB=IBASE		PHAS1630	
15 DD 11 I=1,NREP	PHAS1060	DD 60 J=1,LL		PHAS1640	
11 IFLAG(I)=IRLOW(I)+1	PHAS1070	DD 60 I=1,LL		PHAS1650	
IPHASE=2	PHAS1080	ISUB=ISUB+1		PHAS1660	
IF (1000.EQ.0) GO TO 50	PHAS1090	60 COPRO(ISUB)=C(I,J)		PHAS1670	
IF (INTER.NF.0) WRITE (IOUT,140)	PHAS1100	55 CONTINUE		PHAS1680	
150 FORMAT (/' FINAL PHASING OF EVEN-ODD PROTOTYPES. ODD FUNCTIONS	PHAS1110	RETURN		PHAS1690	
IRE "BASIS FMS,")	PHAS1120	C*** ERROR MESSAGES		PHAS1700	
CALL PSYM (LVAL1,LL1,THLX,LVAL2,LL2,AUFUN,1,-9,RES,B,NMAX)	PHAS1130	100 ISTOP=1		PHAS1710	
ISUB=LS	PHAS1140	WRITE (IOUT,345)		PHAS1720	
DD 22 J=1,LL2	PHAS1150	345 FORMAT (/' & (I) FUNCTION NOT FOUND.'/)		PHAS1730	
DD 22 I=1,LL2	PHAS1160	RETURN		PHAS1740	

PROGRAM: GPINERY	ROUTINE: PHASE	PAGE: 39	PROGRAM: GPINERY	ROUTINE: AVEW	PAGE: 40
110 WRITE (1001,120)	PHAS1750		SUBROUTINE AUFUM (L,LL,FUNCT,IND)	AUFU0010	
120 FORMAT (77,' ERROR IN PHASING ROUTINE.')	PHAS1760	C***	EVALUATES ODD-L SPHERICAL HARMONIC YINCS A(LU) FUNCTION	AUFU0020	
1STDP*)	PHAS1770		IMPLICIT REAL*8 (A-H,O-Z)	AUFU0030	
RETURN	PHAS1780		DIMENSION FUNCT (100),FUNCT(133)	AUFU0040	
END	PHAS1790		COMMON/UNGRAD/CAU(33),LODD,LODEG	AUFU0050	
			CALL THLMK (L,LL,FUNCT,IND)	AUFU0060	
			IF (L.EQ.LODD) GO TO 3	AUFU0070	
			CALL THLMK (LODD,LODEG,FUNCT,IND)	AUFU0080	
			GO TO 4	AUFU0090	
			3 DO 5 I=1,LL	AUFU0100	
			5 FUNCT(I)=FUNCT(I)	AUFU0110	
			4 SUM=0.	AUFU0120	
			DO 2 J=1,LODEG	AUFU0130	
			IF (CAU(J).EQ.0.) GO TO 2	AUFU0140	
			SUM=SUM+FUNCT(I)*CAU(J)	AUFU0150	
			2 CONTINUE	AUFU0160	
			DO 3 I=1,LL	AUFU0170	
			1 FUNCT(I)=FUNCT(I)*SUM	AUFU0180	
			RETURN	AUFU0190	
			END	AUFU0200	

PROGRAM: GPWEDBY	ROUTINE: CLEAN	PAGE: 31	PROGRAM: GPWEDBY	ROUTINE: CLEAY	PAGE: 32
SUBROUTINE CLEAN IC,NBASIS,NDEG,NEVAL,IREPLB,NMAX,CRTZRO,IOPT)	CLEA0010	12	SUM=SUM+EM(K,1)*ED(K)*EM(K,1)	CLEA0590	
C*** UNDO ARBITRARY MIXING OF FUNCTIONS OF SAME SYMMETRY TYPE	CLEA0020		R(I,J)=SUM	CLEA0600	
IMPLICIT REAL*8 (A-H,O-Z)	CLEA0030	11	R(J,I)=SUM	CLEA0610	
COMMON/OUT3/NREP,NDEGAIL4,IJANGA15,I41,IREPAT321,IOPTOR,LVALL,	CLEA0040		CALL EIGEN (R,EN,NSETS,TEMP,NSETS,SMOKE,201	CLEA0620	
1 LVALL2,IRTP114,IJPAK114,IOOD,IRLOML14)	CLEA0050		GO TO 13	CLEA0630	
DIMENSION C(NMAX,NBASIS),NDEG(NEVAL),IREPLB(NEVAL)	CLEA0060	C*** IF IC=NSETS, TRANSFORMATION MATRIX R = EMT*TRANSPOSEI		CLEA0640	
OTMENSION ISV(20),EM(20,20),R(20,20),TEMP(20),ED(20),DIAG(20),	CLEA0070	9	DO 14 I=1,NSETS	CLEA0650	
1 IC(1)	CLEA0080		DO 14 J=1,NSETS	CLEA0660	
COMMON/SCAN1/IFLAG(50),NSC(50),SAMEC(50),JM,IC	CLEA0090	14	R(I,J)=EM(J,I)	CLEA0670	
EQUIVALENCE (TEMP,SAMEC)	CLEA0100	C*** TRANSFORM ALL PARTNER VECTORS IN DEGENERATE SETS		CLEA0680	
DATA DIAG/2.00,3.00,5.00,7.00,11.00,13.00,17.00,19.00,23.00,	CLEA0110	13	DO 15 M=1,NDIR	CLEA0690	
1 29.00,31.00,37.00,41.00,43.00,47.00,53.00,59.00,61.00,67.00,	CLEA0120		DO 16 K=1,NBASIS	CLEA0700	
2 71.00/	CLEA0130		DO 17 I=1,NSETS	CLEA0710	
DO 1 J=1,NREP	CLEA0140		II=(SV(I)*M-1	CLEA0720	
NDIR=NDEGAILR)	CLEA0150		SUM=0.	CLEA0730	
C*** FIND NUMBER OF SETS BELONGING TO THIS REP.	CLEA0160		DO 18 J=1,NSETS	CLEA0740	
NSETS=0	CLEA0170		J1=ISV(J)+M-1	CLEA0750	
II=1	CLEA0180	18	SUM=SUM+R(J1)*R(J,I)	CLEA0760	
DO 2 I=1,NEVAL	CLEA0190		IF (DABS(SUM).LE.CRTZRO) SUM=0.	CLEA0770	
NDI=NDEGILEV)	CLEA0200	17	TEMP(I)=SUM	CLEA0780	
LABEL=IREPLB(IEV)	CLEA0210		DO 19 I=1,NSETS	CLEA0790	
IF (LABEL.NE.IRI GO TO 2	CLEA0220		II=ISV(I)+M-1	CLEA0800	
NSETS=NSETS+1	CLEA0230	19	C(K,I)=TEMP(I)	CLEA0810	
ISVINSETS=II	CLEA0240	16	CONTINUE	CLEA0820	
2 II=II+NDI	CLEA0250	15	CONTINUE	CLEA0830	
IF (INSETS.LE.1) GO TO 1	CLEA0260	1	CONTINUE	CLEA0840	
C*** FIND VECTOR WITH SMALLEST NUMBER OF DISTINCT NONZERO COEFFICIENTS	CLEA0270		RETURN	CLEA0850	
DO 20 M=1,NDIR	CLEA0280		END	CLEA0860	
JM=ISV(M)+M-1	CLEA0290				
CALL SCAN IC,NBASIS,NMAX,CRTZRO)	CLEA0300				
20 ICMIN=IC	CLEA0310				
ICMIN=NBASIS	CLEA0320				
MINM=0	CLEA0330				
DO 24 M=1,NDIR	CLEA0340				
IF (ICMIN).GE.(CMIN) GO TO 24	CLEA0350				
ICMIN=IC(M)	CLEA0360				
MINM=M-1	CLEA0370				
24 CONTINUE	CLEA0380				
C*** CALCULATE ICMIN*NSETS MATRIX	CLEA0390				
DO 3 J=1,NSETS	CLEA0400				
JM=ISV(J)+MINM	CLEA0410				
CALL SCAN IC,NBASIS,NMAX,CRTZRO)	CLEA0420				
DO 7 KC=1,IC	CLEA0430				
PKC=MSC(KC)	CLEA0440				
7 EM(KC,J)=DSORT(PKC)*SAMEC(KC)	CLEA0450				
3 CONTINUE	CLEA0460				
IF ((IC.EQ.NSETS) GO TO 9	CLEA0470				
IF (IOPT.EQ.1) RETURN	CLEA0480				
C*** IF IC.NE.NSETS, FORM SQUARE NSETS*NSETS MATRIX AND DIAGONALIZE	CLEA0490				
DO 8 KC=1,NSETS	CLEA0500				
8 ED(KC)=DIAG(KC)	CLEA0510				
NPL=NSETS+1	CLEA0520				
DO 10 KC=NPL,IC	CLEA0530				
10 ED(KC)=DIAG(NSETS)	CLEA0540				
DO 11 I=1,NSETS	CLEA0550				
DO 11 J=1,I	CLEA0560				
SUM=0.	CLEA0570				
DO 12 K=1,IC	CLEA0580				

PROGRAM: GPINEDRY	ROUTINE: SCAN	PAGE: 93	PROGRAM: GPINEDRY	ROUTINE: SYNSUM	PAGE: 94
SUBROUTINE SCAN (C,NBASIS,NMAX,CRTZRO)	SCAN0010		SUBROUTINE SYNSUM (ISUM,JSUM)	SYNS0010	
IMPLICIT REAL*8 (A-H,O-Z)	SCAN0020	C***	SUMMARIZES SYMMETRY PROPERTIES OF MOLECULE AND GENERATES CASE-	SYNS0020	
COMMON/SCAN1/ (FLAG(50),NSC(50),SANG(50),JM,IC	SCAN0030	C***	REFERENCE TABLES	SYNS0030	
DIMENSION (NMAX,NBASIS)	SCAN0040		IMPLICIT REAL*8 (A-H,O-Z)	SYNS0040	
IC=0	SCAN0050		DIMENSION LDEG(14),LDEG0(14),NOLJ(14),NOL1(14),NOL2(14)	SYNS0050	
DO 23 K=1,NBASIS	SCAN0060		COMMON/CONTROL/CRTZRO,CRTZOG,ISTOP,INTER,LFLOW(10)	SYNS0060	
NSC(K)=0	SCAN0070		COMMON/FILES/INPUT, IOUT	SYNS0070	
SAHFCKT=0.	SCAN0080		COMMON/DUTY/NREP,NDEGAT(4), (ANGATS(4), (REPAC(32), (ORDER,LVAL1,	SYNS0080	
23 IF (FLAG(K).EQ.1) GO TO 21	SCAN0090		L LVAL2,IRTYP(14),IRPAR(14),IDDD,IRLOW(14)	SYNS0090	
AC=DABSC(K,JM)	SCAN0100		DATA (HOL1/4H A,B,6H E,4H T,4H U,4H V/, (HOL2/3H G),	SYNS0100	
IF (AC.LE.CRTZRO) GO TO 21	SCAN0110		L 3H I),3H /	SYNS0110	
JC=JC+1	SCAN0120		DATA HYES/3HYES/, HNO/3HNO /,LFP /1H1/,IRP/1H1/	SYNS0120	
NSC(K)=1	SCAN0130		ICMPLX=0	SYNS0130	
SANG(K)=C(K,JM)	SCAN0140		IF (ISUM.EQ.13.AND.JSUM.EQ.6) ICMPLX=1	SYNS0140	
KPI=K+1	SCAN0150		IF (ISUM.EQ.15.AND.JSUM.EQ.5) ICMPLX=1	SYNS0150	
IF (KPI.GT.NBASIS) GO TO 21	SCAN0160		IF (ISUM.EQ.15.AND.JSUM.EQ.10) ICMPLX=1	SYNS0160	
DO 22 L=KPI,NBASIS	SCAN0170		IF (ISUM.EQ.17.AND.JSUM.EQ.9) ICMPLX=1	SYNS0170	
ACL=DABSC(L,JM)	SCAN0180		IF (ISUM.EQ.17.AND.JSUM.EQ.18) ICMPLX=1	SYNS0180	
IF (ABS(ACL-AC).GT.CRTZRO) GO TO 22	SCAN0190		IF (ISUM.EQ.23.AND.JSUM.EQ.14) ICMPLX=1	SYNS0190	
IF (FLAG(L)=1	SCAN0200	50	ORDER=0	SYNS0200	
NSC(L)=NSC(K)+1	SCAN0210		DO 35 I=1,NREP	SYNS0210	
22 CONTINUE	SCAN0220		IRORCR=IORDER+NDEGAT(I)**2	SYNS0220	
21 CONTINUE	SCAN0230		IF (NDEGAT(I).EQ.2.AND.ICMPLX.EQ.1) IORDER=IORDER-2	SYNS0230	
RETURN	SCAN0240	35	CONTINUE	SYNS0240	
END	SCAN0250		IF (IDDD.EQ.1) NR=NREP/2	SYNS0250	
	SCAN0260		ION=0	SYNS0260	
	SCAN0270		DO 20 I=1,NREP	SYNS0270	
			IDEG=NDEGAT(I)	SYNS0280	
			IRTYP(I)=HOL1(IDEG)	SYNS0290	
			IRPAR(I)=HOL2(I)	SYNS0300	
			IF (IDDD.EQ.1.AND.I.GT.NREP/2) IRPAR(I)=HOL2(2)	SYNS0310	
			IF (IDDD.EQ.0) IRPAR(I)=HOL2(3)	SYNS0320	
			DO 20 IO=1,IDEG	SYNS0330	
			ION=ION+1	SYNS0340	
			(ANGAT(I),1)=ION	SYNS0350	
		20	REPAC(ION)=1	SYNS0360	
			HCOMP=MNO	SYNS0370	
			IF (ICMPLX.EQ.1) HCOMP=HYES	SYNS0380	
			WRITE (IOUT,10)	SYNS0390	
		10	FORMAT (3H,'SUMMARY OF POINT GROUP PROPERTIES'/1H,33(1H-1/)	SYNS0400	
			WRITE (IOUT,4) NREP,HCOMP,IORDER	SYNS0410	
		4	FORMAT (5X,'NO. OF IRREDUCIBLE REPRESENTATIONS = ',12/5X,'COMPLEX-C	SYNS0420	
			IONJUGATE PAIRS OF I.R.S. = ',12/5X,'ORDER OF THE GROUP = ',13)	SYNS0430	
		C***	DETERMINE LABEL OF MOLECULAR POINT GROUP	SYNS0440	
			CALL PTGRP (ICMPLX)	SYNS0450	
			WRITE (IOUT,32) LVAL1,LVAL2	SYNS0460	
		32	FORMAT (5X,'L VALUES OF REAL SPHERICAL HARMONICS USED TO SPAN ALL	SYNS0470	
			THE I.R.S. IN THIS GROUP - ',215)	SYNS0480	
		100	WRITE (IOUT,21)	SYNS0490	
		21	FORMAT (7/33X,'REPRESENTATION DEGENERACY SPECIES PHASE',	SYNS0500	
			1 9X,'ANGULAR'/33X,'INDEX',22X,'LABEL STANDARD QUANTUM NUMBERS	SYNS0510	
			2'/33X,141'-',1.3X,101'-',1.3X,111'-',1.3X,81'-',1.3X,151'-',1/)	SYNS0520	
			DO 23 I=1,NREP	SYNS0530	
			IOEC=NDEGAT(I)	SYNS0540	
			ILF=HOL2(3)	SYNS0550	
			IR=ILF	SYNS0560	
			IR=1	SYNS0570	
			IF (IRLOW(I).GE.0) GO TO 24	SYNS0580	
				SYNS0590	

PROGRAM: GPINFORM	ROUTINE: SYNSUM	PAGE 45	PROGRAM: GPINFORM	ROUTINE: PIGRUP	PAGE 46
IF (LODD.EQ.0) GO TO 24	SYMS0600		SYMS0600	SYMS0600	PTGR0010
IR=1-NR	SYMS0610		SYMS0610	SYMS0610	PTGR0020
IF (1.LE.NR) IR=1+NR	SYMS0620		SYMS0620	SYMS0620	PTGR0030
ICF=ILP	SYMS0630		SYMS0630	SYMS0630	PTGR0040
IR=IRP	SYMS0640		SYMS0640	SYMS0640	PTGR0050
24 WRITE (IOUT,22) 1,1DEG,IRTYPE(1),IRPAR(1),1LF,1RLOWL(1),1RY,1TANSA	SYMS0650		SYMS0650	SYMS0650	PTGR0060
1 1D,11,1D=1,1DEG	SYMS0660		SYMS0660	SYMS0660	PTGR0070
22 FORMAT (3X,12,15X,11,6X,2A4,3X,A1,3HL *,12,A1,'X',515)	SYMS0670		SYMS0670	SYMS0670	PTGR0080
23 CONTINUE	SYMS0680		SYMS0680	SYMS0680	PTGR0090
IF (1STOP.EQ.0) RETURN	SYMS0690		SYMS0690	SYMS0690	PTGR0100
WRITE (IOUT,36)	SYMS0700		SYMS0700	SYMS0700	PTGR0110
36 FORMAT (1H,'ERROR IN DETERMINATION OF SYMMETRY PROPERTIES. CALCUL	SYMS0710		SYMS0710	SYMS0710	PTGR0120
ATION ABANDONED.')	SYMS0720		SYMS0720	SYMS0720	PTGR0130
RETURN	SYMS0730		SYMS0730	SYMS0730	PTGR0140
END	SYMS0740		SYMS0740	SYMS0740	PTGR0150
			SYMS0750	SYMS0750	PTGR0160
			SYMS0760	SYMS0760	PTGR0170
			SYMS0770	SYMS0770	PTGR0180
			SYMS0780	SYMS0780	PTGR0190
			SYMS0790	SYMS0790	PTGR0200
			SYMS0800	SYMS0800	PTGR0210
			SYMS0810	SYMS0810	PTGR0220
			SYMS0820	SYMS0820	PTGR0230
			SYMS0830	SYMS0830	PTGR0240
			SYMS0840	SYMS0840	PTGR0250
			SYMS0850	SYMS0850	PTGR0260
			SYMS0860	SYMS0860	PTGR0270
			SYMS0870	SYMS0870	PTGR0280
			SYMS0880	SYMS0880	PTGR0290
			SYMS0890	SYMS0890	PTGR0300
			SYMS0900	SYMS0900	PTGR0310
			SYMS0910	SYMS0910	PTGR0320
			SYMS0920	SYMS0920	PTGR0330
			SYMS0930	SYMS0930	PTGR0340
			SYMS0940	SYMS0940	PTGR0350
			SYMS0950	SYMS0950	PTGR0360
			SYMS0960	SYMS0960	PTGR0370
			SYMS0970	SYMS0970	PTGR0380
			SYMS0980	SYMS0980	PTGR0390
			SYMS0990	SYMS0990	PTGR0400
			SYMS1000	SYMS1000	PTGR0410
			SYMS1010	SYMS1010	PTGR0420
			SYMS1020	SYMS1020	PTGR0430
			SYMS1030	SYMS1030	PTGR0440
			SYMS1040	SYMS1040	PTGR0450
			SYMS1050	SYMS1050	PTGR0460
			SYMS1060	SYMS1060	PTGR0470
			SYMS1070	SYMS1070	PTGR0480
			SYMS1080	SYMS1080	PTGR0490
			SYMS1090	SYMS1090	PTGR0500
			SYMS1100	SYMS1100	PTGR0510
			SYMS1110	SYMS1110	PTGR0520
			SYMS1120	SYMS1120	PTGR0530
			SYMS1130	SYMS1130	PTGR0540
			SYMS1140	SYMS1140	PTGR0550
			SYMS1150	SYMS1150	PTGR0560
			SYMS1160	SYMS1160	PTGR0570
			SYMS1170	SYMS1170	PTGR0580
			SYMS1180	SYMS1180	PTGR0590
			SYMS1190	SYMS1190	PTGR0600

PROGRAM: GRMCOBY	ROUTINE: PIGRUP	PAGE: 97	PROGRAM: GRMCOBY	ROUTINE: BASIS	PAGE: 98
6 IPTGRPI2)*IMOL(21	PTGR0590		SUBROUTINE BASIS		BAS10010
50 WRITE (1007,100) IPTGRP	PTGR0600	C***	READS IN OR LOOKS UP ATOMIC ORBITAL PARAMETERS AND GENERATES CROSS		BAS10020
100 FORMAT(IX,'THE POINT GROUP FOR THIS NUCLEAR CONFIGURATION IS',IX,6	PTGR0610	C***	REFERENCE TABLES		BAS10030
1A4)	PTGR0620		IMPLICIT REAL*8 (A-H,O-Z)		BAS10040
RETURN	PTGR0630		COMMON/FILES/INPUT,1007		BAS10050
END	PTGR0640		COMMON/DUT1/POS(3,30),RADJUS(30),THETA(30),PHI(30),NATOMS,NATYPE,		BAS10060
			1 NUNT(10),MULT(10),ITSYMB(10),ITYPA(10,10),ITYP(130),ITYP(130),		BAS10070
			2 INVAT(30)		BAS10080
			COMMON/DUTB/BCOEF(500),ZETA(20,10),NVALUE(20,10),LVALUE(20,10),		BAS10090
			1 MBAST(10),ICRKT(10),NASH(10),IBLKAT(5,10),MBLK,NBV8(50),NB1B150,		BAS10100
			2 1TB150,LVB150,IB1F(20,30),1BCFS(150),NOR.EAT(120),1BT(120),		BAS10110
			3 1RT(120),1ADFS(120,30)		BAS10120
			COMMON/DUT2/ASCDEF(3000),ITRANT(20,10),1ADT(40,80),ITRANI(120),		BAS10130
			1 1ESNUNT(120),1ANGT(120),1RADT(120),1FIRST(100),1SOY(40,80),1F8(50),		BAS10140
			2 1SOQ(32,30),NRADA(14)		BAS10150
			COMMON/CONTROL/ERTZRB,ERTDEG,ISTOP,INTER,LFCOMT(10)		BAS10160
			DIMENSION (IMOL(5),IRUM(100))		BAS10170
			DATA IMOL/INS,IMP,THD,IMF,ING/, 1HOLC1/2HC1/, 1HOLC2/1H/		BAS10180
			NMAX=50		BAS10190
			IAO=0		BAS10200
			ITJ=0		BAS10210
			IFT=1		BAS10220
			IFIT=1		BAS10230
			IBLK=0		BAS10240
			CALL INBAS		BAS10250
			IF (1STOP.EQ.1) RETURN		BAS10260
			C*** ASSIGN TRANSFORMATION BLOCK PARAMETERS		BAS10270
			16 DO 150 IT=1,NATYPE		BAS10280
			LNF=0		BAS10290
			NH=NUNT(IT)		BAS10300
			NB=NBAST(IT)		BAS10310
			NBR=0		BAS10320
			DO 156 LPO=1,10		BAS10330
			L=LPO-1		BAS10340
			DO 156 1B=1,NB		BAS10350
			IF (L.NE.LVALUE(1B,1)) GO TO 159		BAS10360
			3 NBR=NBR+1		BAS10370
			ITJ=ITJ+1		BAS10380
			(TRANTINBR,IT)=ITJ		BAS10390
			IF (LPO.LE.LRF) GO TO 165		BAS10400
			LRF=LPO		BAS10410
			IBLK=IBLK+1		BAS10420
			IFJ=IFJ+1		BAS10430
			IFB(IBLK)=IFJ		BAS10440
			IFJ=(IFJ+INH*(20LRF-1)))*2		BAS10450
			165 IFIRST(ITJ)=IFJ		BAS10460
			159 IF (NBR.EQ.NB) GO TO 161		BAS10470
			156 CONTINUE		BAS10480
			C*** GENERATE BASIS FUNCTION REFERENCE TABLES		BAS10490
			161 DO 150 IN=1,NH		BAS10500
			1A=ITYPA(IN,1)		BAS10510
			DO 150 1B=1,NB		BAS10520
			LDEG=2*LVALUE(1B,1)+1		BAS10530
			1ADFS(1B,1A)=1A0+1		BAS10540
			DO 150 K=1,LDEG		BAS10550
			1A0=1A0+1		BAS10560
			1AT(1A0)=1A		BAS10570
			1BT(1A0)=1B		BAS10580

PROGRAM: GTHEORY	ROUTINE: BASIS	PAGE: 49	PROGRAM: GTHEORY	ROUTINE: BASIS	PAGE: 50
150 IKT(IAO)=K	BAS10590		ZT=ZETA(1B1,IT)	BAS11160	
NOR=IAO	BAS10600		IFBC=IBCFST(1BLK)+1B-1	BAS11170	
IF INOR.GT.120 GO TO 200	BAS10610		ITRANS=ITRANT(1B1,IT)	BAS11180	
IAI=1	BAS10620		NBV=INB-ITRNB*IFBC	BAS11190	
ITC=0	BAS10630		WRITE (1OUT,11) 1B1,ITRANS,N,ZT,IBCF(11),1*IFBC,NBV,NB1	BAS11200	
DO 151 IT=1,NATYPE	BAS10640		11 FORMAT (14,6X,12,7X,12,F9.4,5X,9F10.5)	BAS11210	
NB=NBAS(1IT)	BAS10650		10 CONTINUE	BAS11220	
DO 152 IB=1,NB	BAS10660		7 WRITE (1OUT,204)	BAS11230	
ITR=ITR+1	BAS10670		IF (INTER.EQ.0) RETURN	BAS11240	
IARUN=0	BAS10680		DO 21 I=1,80	BAS11250	
DO 153 IAO=IAI,NOR	BAS10690		21 IROUT=0	BAS11260	
IA=IAT(IAO)	BAS10700		WRITE (1OUT,201)	BAS11270	
ITO=ITYPT(IA)	BAS10710		20 FORMAT (11H,17X,' BASIS CHEMICAL ATOM QUANTUM NUMBERS ORBITBAS11280		
IBD=IBT(IAO)	BAS10720		1AL TRANS. TRANS. NUMBERING INDEX ATOM BASIS*/18X,'FUNCTION	BAS11290	
ITRANS=ITRANT(IAO,ITO)	BAS10730		2YMBOL INDEX N L K EXPONENT BLOCK INDEX WITHIN TBAS11300		
IF (ITRANS.NE.ITR) GO TO 153	BAS10740		3RANS. TYPE INDEX*/18X,96(1H-1/)	BAS11310	
IARUN=IARUN+1	BAS10750		IAT=1	BAS11320	
IADT(IARUN,ITRANS)=IAO	BAS10760		DO 25 IAO=1,NOR	BAS11330	
153 CONTINUE	BAS10770		IA=IAT(IAO)	BAS11340	
152 CONTINUE	BAS10780		IF (IA.NE.IAT) WRITE (1OUT,204)	BAS11350	
151 IA=IAI+IARUN	BAS10790		IAT=IA	BAS11360	
NBLK=1BLK	BAS10800		IT=ITYPT(IA)	BAS11370	
IF INBLK.GT.50 GO TO 200	BAS10810		IB=IBT(IAO)	BAS11380	
NCF=1F1-1	BAS10820		CP(1,VALUE(1B,IT)+1	BAS11390	
IF (NCF.LE.3000) GO TO 400	BAS10830		ITRANS=ITRANT(1B,IT)	BAS11400	
WRITE (1OUT,201)	BAS10840		IRUNITRANS=IRUNITRANS+1	BAS11410	
201 FORMAT (11H,' NUMBER OF TRANSFORMATION COEFFICIENTS EXCEEDS 3000'	BAS10850		25 WRITE(1OUT,26) IAO,ITSYMB(1T),IA,NVALUE(1B,IT),1HOL(1P1),IKT(IAO),IBAS11420		
ISTOP=1	BAS10860		1ETAT(1B,IT),1BLKAT(1P1,IT),ITRANS,IRUNITRANS,IT,1B	BAS11430	
GO TO 400	BAS10870		26 FORMAT (18X,15,8X,A2,6X,82,6X,12,3X,A1,14,4X,F9.4,4X,12,5X,13,8X,	BAS11440	
200 WRITE (1OUT,202)	BAS10880		1 13,9X,12,3X,13/1	BAS11450	
202 FORMAT (11H,'PARAMETER SIZE ERROR IN SUBROUTINE BASIS')	BAS10890		RETURN	BAS11460	
ISTOP=1	BAS10900		END	BAS11470	
C*** PRINTOUT OF CROSS-REFERENCE TABLES	BAS10910				
400 WRITE (1OUT,5)	BAS10920				
5 FORMAT (11H,'ATOMIC BASIS SET AND SYMMETRY ADAPTATION*/1X,40(1H-1/)	BAS10930				
11	BAS10935				
DO 7 IALK=1,NBLK	BAS10940				
IT=ITB(1BLK)	BAS10950				
1=LYB(1BLK)	BAS10960				
NB=NBIB(1BLK)	BAS10970				
WRITE (1OUT,6) 1BLK,IT,1HOL(1L+1),ITSYMB(1T)	BAS10980				
6 FORMAT (7' TRANSFORMATION BLOCK '12//' ATOM-TYPE '12,' ' ,	BAS10990				
1 AL' FUNCTIONS FOR ELEMENT 'A2/1	BAS11000				
NSIZE=12*1+1)*NUNT(1T)	BAS11010				
WRITE (1OUT,19) NSIZE,1F8(1BLK)	BAS11020				
19 FORMAT (7' TOTAL OF '13,' BASIS FUNCTIONS FOR EACH BASIS INDEX//	BAS11030				
1 ' BLOCK OF SYMMETRY TRANSFORMATION COEFFICIENTS STARTS AT LOCATIO	BAS11040				
2N '14,' OF ASCOEF//)	BAS11050				
WRITE (1OUT,8)	BAS11060				
8 FORMAT (7' BASIS TRANSF. QUANTUM NUMBERS BASIS VECTORS')	BAS11070				
NV=NBVB(1BLK)	BAS11080				
WRITE (1OUT,9) (1HOLC1,1V,1HOLC2,1V=1,NV)	BAS11090				
9 FORMAT (7' INDEX INDEX*, 6X, 'M ZETA*,5X,10(4X,A2,12,A1,1X)	BAS11100				
WRITE (1OUT,204)	BAS11110				
204 FORMAT (7'	BAS11120				
DO 10 IB=1,NB	BAS11130				
1B1=IBT(1B,1BLK)	BAS11140				
N=NVALUE(1B,1T)	BAS11150				

PROGRAM: GPIMCURY	ROUTINE: INBAS	PAGE: 53	PROGRAM: GPIMCURY	ROUTINE: INBAS	PAGE: 54
D 5.1905,7.3342,1.4598,2.5623,12.930,21.248,6.8436,4.4787,1.3857, INBA1170	A -.02413,0.04730,-.00153,.00398,-.13437,-.00449,-.60087,-.11632, INBA1750				
E 2.7424,3.8359,8.3989,24.928,34.078,13.720,18.130,5.4824,7.4731, INBA1180	B .54834,.56956,-.00422,.01445,-.02953,.00031,.14467,-.03022, INBA1760				
F 1.6823,2.8215,13.562,25.645,4.1667,5.5727,1.6232,3.1345,4.2259, INBA1190	C -.13567,-.15909,.67605,.43415,.46987,.13709,-.02895,-.00309, INBA1770				
G 9.0546,25.203,34.967,13.913,18.000,9.8328,4.0342,1.6451,3.0598, INBA1200	D -.35118,-.03204,.28760,.82391,.75478,.36564,.66518,.34106, INBA1780				
H 14.610,27.313,9.7272,9.0506,1.1714,3.4127,4.3554,9.3545,28.303, INBA1210	E .00510,-.00618,.00094,-.00150,.00008,-.00017,-.45273,-.07401, INBA1790				
I 37.064,15.316,18.803,6.6824,9.5169,2.3318,4.0670,15.061,28.333, INBA1220	F 1.23731,-.17787,-.02037,.04578,-.00125,.00282,.15120,-.01914, INBA1800				
J 9.0037,6.2067,1.8954,3.5888,4.8816,9.9206,28.941,37.859,15.550, INBA1230	G -.66710,-.17267,.58189,.54381,-.00328,.01433,-.03024,.00178, INBA1810				
DATA A0114 / 19.868,6.5065,8.3700,2.4358,4.0461,15.560,29.224, INBA1240	H .16484,-.04900,-.09679,-.17711,.52578,.37466,.89362,.11307, INBA1820				
L 8.1315,6.0579,1.8095,3.2184,5.1816,10.3267, INBA1250	I .03560,-.01121,-.34713,-.03038,.47688,.57076,.76316,.35383, INBA1830				
DATA BC1 / 1.0,0.18159,0.84289,0.86288,0.15286,0.00021,-.0.18792, INBA1260	J .67678,.52641,.00164,-.00026,.00034,-.00051,-.00002,-.00008, INBA1840				
L -.00093,1.02666,0.87859,0.13422,-.000051,-.0.21603,0.03249,1.02631, INBA1270	DATA BC31 / -.46634,.09153,1.25591,-.19999,-.02676,.05279,-.00095, INBA1850				
2.0.38708,0.12215,0.00062,-.0.24194,0.00684,1.03149,0.21506,0.84089, INBA1280	L .006157, INBA1860				
3 0.90077,0.10695,0.00187,-.0.27677,0.01014,1.03413,0.25925,0.80295, INBA1290	DATA BC4 / 1.5438,-.02252,-.67793,.18395,.50633,-.61280,-.00240, INBA1870				
4 1.08517,-.0.0035,0.00235,-.0.23530,-.0.01893,1.03375,0.28840,0.7781, INBA1300	L .02360,-.03390,.00667,.13145,-.03351,-.19444,-.10244,-.87696, INBA1880				
5 1.07582,-.0.08065,0.00313,-.0.24017,-.0.02125,1.03535,0.32655,7.4660, INBA1310	2 .26578,.06370,.16559,.01755,.00062,-.34925,-.03851,.32197, INBA1890				
6 1.06846,-.0.07301,0.00379,-.24322,-.02281,1.03628,.35507,.72760, INBA1320	3 .80538,.73165,.40849,.60565,.40204,.00787,-.01079,.00124, INBA1900				
7 1.06232,-.06461,-.00439,-.24534,-.02408,1.03688,.36896,.71694, INBA1330	4 -.00197,-.00010,-.00025,-.46254,.07740,1.28802,-.22335,-.01370, INBA1910				
9 75919,.26678,.00088,.00143,.00016,-.00030,-.34000,.03768,.39901, INBA1340	5 .94773,-.00115,.00309,.15605,-.01825,-.73462,.22283,.85512, INBA1920				
A .68259,-.00049,.00052,.05260,-.00652,-.07599,-.10472,-.79808, INBA1350	6 .49965,-.00398,.01505,-.03217,-.00322,.16597,-.05322,-.14590, INBA1930				
B .25733,.73678,.33147,-.60598,.46103,-.00283,-.00183,.00040,-.00084, INBA1360	7 -.13103,-.70162,.41307,.90636,.09190,.04112,-.00943,-.37082, INBA1940				
C -.34742,-.02775,-.33845,.75162,.00171,-.00572,-.07001,-.00758, INBA1370	8 -.02072,-.34236,.72134,.72813,.39159,.59688,-.41187,-.00733, INBA1950				
D -.10297,-.01383,-.64423,-.43983,-.74494,-.31623,.39604,.61459,.00939, INBA1380	9 -.01137,-.00127,-.00203,.00010,-.00025,-.46769,.07583,1.24500, INBA1960				
E -.01221,-.00103,-.00223,-.02916,-.01558,-.68867,.61647,.00149, INBA1390	A -.17773,-.01770,.04412,-.00100,.00272,.13367,-.01443,-.70405, INBA1970				
F -.00347,.07899,.00174,-.18830,-.12383,.64318,.45352,.77360, INBA1400	B .18834,.61291,.52512,-.00571,.01398,-.03067,-.00221,-.15595, INBA1980				
G .26973,-.00662,.02313,-.14620,-.04726,.76411,.30883,.22649, INBA1410	C -.04458,-.13450,-.13631,.69836,.41776,.91337,-.07491,-.05562, INBA1990				
H .79217,.01514,-.02761,.30143,-.00327,-.31339,-.07524,-.66752, INBA1420	D -.01391,-.37778,-.01492,.36084,.76903,.71876,.40304,.44554, INBA2000				
I .47866,.00246,-.00440,.00076,.01924,-.27440,-.00891,-.67830, INBA1430	E .56468,.00939,-.01174,.00084,-.00143,.00006,-.00015,-.45546, INBA2010				
J .42977,.77760,.26218,-.00473,.01824,-.17507,-.05446,.69577,.38127, INBA1440	F .05245,1.18610,-.09988,-.01256,.03593,-.00065,.00177,.14376, INBA2020				
DATA BC2 / 7.4977,25361,.00143,.00244,.00012,-.00041,-.49236, INBA1450	G -.00094,-.67009,.14521,-.64109,.49954,-.00480,.41702,-.02713, INBA2030				
1 1.2790,.98322,.19647,.04050,-.11416,-.11559,-.02116,-.41650, INBA1460	H -.00137,.14666,.03524,-.13397,-.13100,.69261,.42361,.89679, INBA2040				
2 -.00731,.77817,.39127,.78393,.25344,-.00352,.01426,-.19689, INBA1470	I .10142,.03915,-.00864,-.37991,-.02214,-.41525,.70533,.71157, INBA2050				
3 -.05826,-.60660,.47850,.53980,.46677,.00369,-.00317,-.00036, INBA1480	J .41464,.49325,.51462,.00551,-.00931,.00071,-.00128,.00005, INBA2060				
4 -.00129,-.45693,.08381,.68067,.49502,-.02803,-.09538,.12583, INBA1490	DATA BC3 / -.00013,-.47476,.07494,.16369,-.01321,.04039, INBA2070				
5 -.01924,-.41437,-.07015,.72311,.48648,.75824,.28161,.00287, INBA1500	L -.00068,.00192,.15523,-.01146,-.72787,.19702,.69361,-.45118, INBA2080				
6 .00204,-.20738,-.06876,.55335,.55074,-.22143,.79375,-.01508, INBA1510	2 -.00443,-.01764,-.02993,.00141,.15584,-.04499,-.15270,-.11220, INBA2090				
7 -.02554,.00143,-.00476,-.34337,-.06389,.43577,.72240,.00642, INBA1520	3 .72260,.39740,.90375,.08931,.04431,-.00881,-.38885,-.01787, INBA2100				
8 -.01993,.10635,.01811,-.34780,-.14259,.69985,.48475,.76189, INBA1530	4 .36604,.75500,.69757,.43271,.56740,.44064,.01005,-.01387, INBA2110				
9 .27656,.00498,-.00117,-.23282,-.06743,.67838,.63171,.35071, INBA1540	5 .00087,-.00179,.00006,-.00014,-.48478,.00788,1.37427,-.30751, INBA2120				
A .65354,.00694,-.00912,.00065,-.00263,-.44690,.04291,.63649, INBA1550	6 -.01221,-.04366,-.00065,.00184,.16148,-.01830,-.86612,.31280, INBA2130				
B .58355,.03133,-.12009,1.3582,-.01047,-.45658,-.10403,.68854, INBA1560	7 .73748,.41625,-.00511,.01735,.02138,-.00002,-.15160,.04305, INBA2140				
C .56846,.62615,.45922,.01945,-.06999,-.23934,-.11117,.72656, INBA1570	8 .09849,.09759,-.72048,-.39635,.89818,.10137,.03356,-.00537, INBA2150				
D .43099,.77926,.22770,.00354,-.00462,.00101,-.00142,-.00015, INBA1580	9 .39846,-.02035,.43411,.69645,.69521,.44925,.51296,.49324, INBA2160				
E -.00031,-.39810,.05070,1.11024,-.06090,-.01632,-.03337,-.00179, INBA1590	A .00341,-.00582,.00035,-.00070,.00003,-.00006,-.49614,.09498, INBA2170				
F .00340,.10928,-.00170,-.48580,.06286,.36935,.71592,-.00385, INBA1600	B 1.24127,-.16234,-.01107,.03504,-.00062,.00147,.16362,-.01821, INBA2180				
G .00922,-.01899,-.00100,.09788,-.01455,-.04442,-.17608,.74202, INBA1610	C -.72478,-.18966,.55056,.60534,-.00249,.00729,-.02346,-.00228, INBA2190				
H .32095,.28733,.04573,-.42376,-.67764,.84278,.17591,.02374, INBA1620	D .15949,-.05126,-.05278,.71895,.51599,.59477,.88755,.11397, INBA2200				
I -.00576,.46278,.54811,.02112,-.02590,.00210,-.00361,.00020, INBA1630	E .03143,-.00605,-.38770,-.02536,.48983,.64044,.69487,.43560, INBA2210				
J -.00044,-.41612,.03117,1.21977,-.13724,-.01838,.03750,-.00097, INBA1640	F .26726,.74626,.02116,-.03121,.00146,-.00270,.00012,-.00026, INBA2220				
DATA BC3 / 0.0222,-.14019,-.00826,-.72706,-.23825,.75652,-.36790, INBA1650	G -.44786,.00512,1.28168,-.15704,-.01318,.03804,-.00087,.00199, INBA2230				
L -.00544,-.01702,.03251,-.00137,-.18955,.06811,.19369,-.11258, INBA1660	H .17029,-.00169,-.86319,.28996,.58647,.57498,-.00411,.01166, INBA2240				
I -.61595,-.68659,.87022,-.14168,.03427,-.01145,-.31473,-.03823, INBA1670	M -.03289,-.00322,.23248,-.09601,-.19348,.49694,.60674, INBA2250				
3 .48646,.61279,.43866,.37563,.02017,-.02835,.00190,-.00645, INBA1680	J .92478,.05707,.07310,-.03168,-.00088,.00675,-.39181,-.01146, INBA2260				
4 .00017,-.00047,-.60680,.01604,1.18618,-.11670,-.00793,.05023, INBA1690	DATA BC6 / 1.26343,.76006,-.00256,.10185,.05674,.00306,-.00522, INBA2270				
5 -.00059,-.00177,-.13267,.00208,-.88835,.17866,.99639,.39647, INBA1700	L -.20708,.95615,.15095,.74031,.37327,.27375,.74022,-.01831, INBA2280				
6 -.00991,.05843,-.03230,.00043,.15140,-.04054,-.29233,-.01502, INBA1710	2 -.02695,.00138,-.00254,.00011,-.00025,-.43471,-.01087,1.22777, INBA2290				
7 .84798,.25199,.85312,-.16093,.02487,-.00560,-.32486,-.04357, INBA1720	3 -.10323,-.01141,.03644,-.00069,.00164,.16452,.00360,-.81330, INBA2300				
B .38439,.71247,.81198,.30493,.56091,.44768,.00458,-.00820, INBA1730	4 .23365,-.65816,.50750,-.00289,.01029,-.03668,-.00403,.23616, INBA2310				
9 .00107,-.00156,.00009,-.00022,-.43033,.05288,1.16499,-.09189, INBA1740	5 -.08087,-.15364,-.18275,.45342,.66194,.87283,.19162,.03095, INBA2320				

PROGRAM: GPINEDRY	ROUTINE: INBAS	PAGE: 45	PROGRAM: GPINEDRY	ROUTINE: INBAS	PAGE: 56
6 -.03119,-.00081,.00276,-.38302,-.03619,.62023,.49383,-.00213,	INBA2330		13 FORMAT (BFI0.8)	INBA2910	
7 .01453,-.07385,.00740,-.12584,-.12484,.00427,.31649,.70035,.60948,	INBA2343		DD 9 IBI=1,NBI	INBA2920	
8 .19636,.81497,.01832,-.02648,.00103,-.00191,.00007,-.00017,	INBA2350		BCOEF1 (BFST+ISUB)*OUT1261)	INBA2930	
9 -.45154,-.00998,.121808,-.07289,-.01345,.03675,-.00080,.00189,	INBA2360		9 ISUB=ISUB+1	INBA2940	
4 .18269,-.32043,-.90776,.30443,.63255,.53181,-.00538,.01670,	INBA2370		7 CONTINUE	INBA2950	
8 .06606,.00233,-.29411,-.11899,-.16548,-.20568,-.47504,-.63742,	INBA2380		IBCFST (IBLK)=IBFST	INBA2960	
C .92134,.06221,-.06160,-.02036,-.00126,.00386,-.40594,-.01242,	INBA2390		IBFST=IBFST+NBI*NBV	INBA2970	
0 .24398,-.82473,-.01145,-.05798,.08935,.00298,-.05037,-.22726,	INBA2400		NBT=NBT+NBI	INBA2980	
E .42895,-.27999,-.71617,-.38692,-.17486,-.83676,-.00785,-.01682,.00040,	INBA2410		3 IBLK=IBLK+1	INBA2990	
F -.00093,-.00002,-.00003,-.46228,-.01232,-.11118,.05264,-.00530,	INBA2420		NBAST (ITI)=NBT	INBA3000	
G .02422,-.00019,.00049,.17590,.00865,-.76204,.16523,.71794,	INBA2430		10 CONTINUE	INBA3010	
H .45554,-.00385,.01469,.04711,.00497,-.25806,.06881,.22989,	INBA2440		RETURN	INBA3020	
I .17806,-.47164,-.64988,.92089,.05183,.06775,-.01455,-.00097,	INBA2450		1 DD 157 IT=1,NATYPE	INBA3030	
J .00296,-.41231,-.00833,.13274,.42648,-.01164,.06609,.09860,	INBA2460		MZ=NUCT (IT)	INBA3040	
DATA EC7 / .00243,-.02336,-.28098,.82121,.29550,.70183,.39352,	INBA2470		16 IF (MZ.GT.MZ) GO TO 20	INBA3050	
1 .33846,.67846,.03171,-.04440,.00313,-.00592,.00023,-.00069,	INBA2480		NBAST (ITI)=NBASL (MZ,LV)	INBA3060	
2 -.45917,.02244,.140547,-.31512,-.01672,-.06469,-.00090,.00283,	INBA2490		IF INBAS1111).LT.01 GO TO 20	INBA3070	
3 .17032,-.00080,-.95336,.37395,.10174,.10306,-.00690,.04629,	INBA2503		NASK (IT)=NASL (MZ,LV)	INBA3080	
4 -.05884,.00491,.27763,-.09485,-.46036,.01625,.78349,.36510,	INBA2510		ICRK (IT)=ICRL (MZ,LV)	INBA3090	
5 .92724,-.05085,-.06919,-.02298,-.00338,.00427,-.41861,-.00867,	INBA2520		NSUB=NSUB+1	INBA3100	
6 .19213,.87604,-.00829,.05405,.10742,.00261,-.03805,-.29596,	INBA2530		NSUB=NSUB+1	INBA3110	
7 .78969,.34026,.71133,.37898,.30191,.71223,.02495,-.03966,	INBA2540		DD 100 IZ=1,MZ	INBA3120	
8 .00393,-.00344,.00025,-.00075,-.47092,.02751,.138570,-.27267,	INBA2550		IF (IZ.NE.MZ) GO TO 101	INBA3130	
A -.03026,.06030,-.00133,.00422,.16287,-.00664,-.66946,.36941,	INBA2560		GO TO 110	INBA3140	
B .73304,-.41654,-.00322,.02389,-.06059,.00387,.31761,-.12187,	INBA2570		101 NSUB=NSUB+NASL (IZ,LV)	INBA3150	
C -.37484,-.10581,.70702,.45131,.93333,.05118,.07231,-.03686,	INBA2580		100 NSUB=NSUB+NBASL (IZ,LV)	INBA3160	
D -.00221,.00572,-.42145,-.01096,.44406,.64246,-.00943,.00309,	INBA2590		110 IB=0	INBA3170	
E .11671,.00227,-.14806,-.18834,-.57945,-.54963,-.70696,-.37915/	INBA2600		NAS=NASR (IT)	INBA3180	
MYSDPH=2	INBA2610		DD 105 IAS=L,NAS	INBA3190	
MZ=54	INBA2620		IFBI (IBLK)=IF	INBA3200	
LV=FLOW (4)	INBA2630		NS=NSUB+IAS-1	INBA3210	
IBFST=1	INBA2640		LVBI (IBLK)=LVALU (INS)	INBA3220	
IBLK=1	INBA2650		L=LVBI (IBLK)	INBA3230	
C*** READ IN OR ASSIGH BASIS SET PARAMETERS	INBA2660		IBLK (L+1,IT)=IBLK	INBA3240	
IF (LV.GT.MYSDPH.AND.LV.NE.9) GO TO 200	INBA2670		NBIB (IBLK)=NBIL (NS)	INBA3250	
IF (LV.NE.9) GO TO 1	INBA2680		NBVB (IBLK)=NBVL (NS)	INBA3260	
NBT=0	INBA2690		IBFSL=IBFL (NS)	INBA3270	
DD 10 IT=1,NATYPE	INBA2700		NBT=NBT+IBLK	INBA3280	
READ (INPUT,11) NASR (IT),ICRK (IT)	INBA2710		NBV=NBVB (IBLK)	INBA3290	
11 FORMAT (16F1)	INBA2720		DD 120 IBI=1,NBI	INBA3300	
NAS=NASR (IT)	INBA2730		IB=IB+1	INBA3310	
IB=0	INBA2740		IBIT (IB,IBLK)=IB	INBA3320	
DD 3 IAS=1,NAS	INBA2750		NS=NSUB+IB-1	INBA3330	
ITR (IBLK)=IT	INBA2760		NVALU (IB,IT)=NVALU (INS)	INBA3340	
READ (INPUT,11) LVBI (IBLK),NBIB (IBLK),NBVB (IBLK)	INBA2770		ZETA (IB,IT)=RADIL (INS)	INBA3350	
L=LVBI (IBLK)	INBA2780		120 LVALU (IB,IT)=LVBI (IBLK)	INBA3360	
IBLK (L+1,IT)=IBLK	INBA2790		ISUB=0	INBA3370	
NBT=NBT+IBLK	INBA2800		DD 130 IBV=1,NBV	INBA3380	
DD 5 IBI=1,NBI	INBA2810		DD 131 IBI=1,NBI	INBA3390	
IB=IB+1	INBA2820		BCOEF (IBFST+ISUB)=BCOFL (IBFSL+ISUB)	INBA3400	
IBIT (IB,IBLK)=IB	INBA2830		131 ISUB=ISUB+1	INBA3410	
LVALU (IB,IT)=LVBI (IBLK)	INBA2840		130 CONTINUE	INBA3420	
5 READ (INPUT,6) NVALU (IB,IT),ZETA (IB,IT)	INBA2850		IBCFST (IBLK)=IBFST	INBA3430	
6 FORMAT (15,F10.5)	INBA2860		IBFST=IBFST+NBT*NBV	INBA3440	
NBV=NBVB (IBLK)	INBA2870		105 IBLK=IBLK+1	INBA3450	
ISUB=0	INBA2880		137 CONTINUE	INBA3460	
DD 7 IBV=1,NBV	INBA2890		RETURN	INBA3470	
READ (INPUT,13) BCY (IB),IBI=1,NBI	INBA2900		200 WRITE (IOUT,202)	INBA3480	

PROGRAM: GPTEGNY	ROUTINE: INHAS	PAGE: 57	PROGRAM: GPTEGNY	ROUTINE: ORBILS	PAGE: 58
202 FORMAT (1H1, 'PARAMETER SIZE ERROR IN SUBROUTINE BASIN')	INBA3490		SUBROUTINE ORBILS	ORBT0010	
1STOP=1	INBA3500		C**** CALCULATES SYMMETRY-ADAPTED BASIS ORBITALS	ORBT0020	
RETURN	INBA3510		(IMPLICIT REAL*8 (A-H,O-Z))	ORBT0030	
20 WRITE (10OUT,22) NZ,LV	INBA3520		REAL*8 LABEL	ORBT0040	
22 FORMAT (1H1, ' STORED BASIS SET OPTION IS NOT IMPLEMENTED FOR NUC	INBA3530		COMMON/CONTROL/CRTZRO,CRTDEG,1STOP,INTER,LFLDNL10)	ORBT0050	
2CLEAR CHARGE ',12,' AND SOPHISTICATION LEVEL ',12,'. ' RESUBMIT Q	INBA3540		COMMON/OUT1/POS13,301,RAD1US1301,THETA1301,PHI1301,NATONS,NATYPR,	ORBT0060	
2SING BASIS SET INPUT OPTION.')	INBA3550		1 NUNTC101,NUCT1101,11SYMB1101,11TYP130,103,11TPT1301,11TPI1301,	ORBT0070	
1STOP=1	INBA3560		2 INVT1301	ORBT0080	
RETURN	INBA3570		COMMON/OUT0/BCOEF15001,ZETA120,101,NVALUE120,101,LVALUE120,101,	ORBT0090	
END	INBA3580		1 NBAST1101,1CRK1101,NASB1101,1BLK15,101,NBLK,NBVS1501,NBT1501,	ORBT0100	
			2 1TB1501,LVB1501,1BST120,501,1BCFST1501,MOR,1AT11201,1BT11201,	ORBT0110	
			3 1KTT1201,1ADFST120,301	ORBT0120	
			COMMON/OUT2/ASCOEF130001,1TRAN1120,101,1AOT140,801,1TRAN11201,	ORBT0130	
			1 1SRUNY11201,1ANGF11201,1RADT11201,1FINST1801,15OT140,801,1FB1501,	ORBT0140	
			2 1SQ1132,301,NRAD1141	ORBT0150	
			COMMON/OUT3/NREP,NOEGA1141,TANGA15,141,1NEPA1321,1ORDER,LVAL1,	ORBT0160	
			1 LVAL2,1RTYPE141,1RPAR1141,1000,1RLDNL1141	ORBT0170	
			COMMON/WORK2/DUMMY141,JSUB,LCDEG,NUMB,11	ORBT0180	
			COMMON/FILES/INPUT,1OUT	ORBT0190	
			COMMON/WORK1/DUMMY180001	ORBT0200	
			DIMENSION LABEL1501,INDEX1501,HOLANG111,61,1REPLB1501,10ENT1201,	ORBT0210	
			1 1HOL1101,1HOL2171,1ANGB1501,1RES1150,501,VALU1501,C150,501,	ORBT0220	
			2 B150,501,NDEGB1501,1HOL121,1HOL121,1REPT1201	ORBT0230	
			EQUIVALENCE (RES1,DUMMY1,1VALU,DUMMY1250111,1C,DUMMY1255111,	ORBT0240	
			1 1B,DUMMY1505111,11ANGB,DUMMY1756111,	ORBT0250	
			2 1LABEL,DUMMY1787511,1INDEX,DUMMY1792511,11REPLB,DUMMY1795011,	ORBT0260	
			3 1NDEGB,DUMMY1797511,1NEVALB,DUMMY180001)	ORBT0270	
			DATA 1HOL14H CRU,4HDE S/, 1HOL14H FIN,4HAL S/	ORBT0280	
			DATA 10ENT14H CRU,4HDE S,4HYMME,4HTRY,4HYECT,4HORS,4HFOR,4HATDOR	ORBT0290	
			1N,4H-TYP,114H /, 1HOL14HE 1,4HE 2,4HE 3,4HE 4,4HE 5,	ORBT0300	
			2 4HE 6,4HE 7,4HE 8,4HE 9,4HE 10/, 1HOL14H. 1 /, 1HOL24H= 0	ORBT0310	
			3,4H= 1,4H= 2,4H= 3,4H= 4,4H= 5 /	ORBT0320	
			DATA 1HOLANG17H S,10*2H,7H Z,7H Y,7H X,8*2H	ORBT0330	
			1,7H 3/2-1,7H YZ,7H XZ,7H XY,7H XZ-YZ,6*2H,	ORBT0340	
			2 7H 0 COS,7H 1 SIN,7H 1 COS,7H 2 SIN,7H 2 COS,7H 3 SIN,	ORBT0350	
			3 7H 3 COS,7H 4 SIN,7H 4 COS,2*2H,11*2H /	ORBT0360	
			EXTERNAL BASFUM	ORBT0370	
			NNAX=50	ORBT0380	
			00 5 E=1,NREP	ORBT0390	
			5 NRADA11=0	ORBT0400	
			ISO=0	ORBT0410	
			DO 1 1BLK=1,NBLK	ORBT0420	
			C**** SYMMETRY-ADAPT SET OF BASIS FUNCTIONS FOR GIVEN L-VALUE, ATOM-TYPE	ORBT0430	
			1T=1TB1BLK1	ORBT0440	
			L=LVB1BLK1	ORBT0450	
			1PL=L+1	ORBT0460	
			10ENT11=1HOL111	ORBT0470	
			10ENT121=1HOL121	ORBT0480	
			10ENT1101=1HOL11111	ORBT0490	
			10ENT1111=1HOL13	ORBT0500	
			10ENT1121=1HOL21LP11	ORBT0510	
			LCDEG=2*PL+1	ORBT0520	
			NUMB=NUMT111	ORBT0530	
			1=0	ORBT0540	
			KFLAG=0	ORBT0550	
			00 30 1A=1,NUMB	ORBT0560	
			IF (RAD1US1A).GT.CRTZRO) KFLAG=1	ORBT0570	
			00 30 N=1,LCDEG	ORBT0580	

PROGRAM: GPINEDRY	ROUTINE: ORBIL5	PAGE: 39	PROGRAM: GPINEDRY	ROUTINE: ORBIL5	PAGE: 60
(*)+1	ORBT0590		DD 150 J=1,NOR	ORBT1160	
INDEX(I)=IA	ORBT0600		IF (IANGT(I).LT.IANGT(J)) GO TO 150	ORBT1170	
30 LABEL(1)=HMLANG(N,LP1)	ORBT0610		IF (IANGT(1).EQ.IANGT(J).AND.(IRADT(1).LT.IRADT(J)) GO TO 150	ORBT1180	
NBASIS=LOEG*NUMB	ORBT0620		ITEMP=IANGT(I)	ORBT1190	
NPAR=0	ORBT0630		IANGT(I)=IANGT(J)	ORBT1200	
IF (1000.EQ.0.OR.KFLAG.EQ.1) GO TO 40	ORBT0640		IANGT(J)=ITEMP	ORBT1210	
NPAR=1	ORBT0650		ITEMP=IRADT(I)	ORBT1220	
IF (2*(L/2).NE.L) NPAR=-1	ORBT0660		IRADT(I)=IRADT(J)	ORBT1230	
40 CALL ADAPT (L,NBASIS,BASFUN,1,NPAR,IDENT,0)	ORBT0670		IRADT(J)=ITEMP	ORBT1240	
IF (11STOP.EQ.1) RETURN	ORBT0680		ITEMP=ISRUNT(I)	ORBT1250	
IDENT(1)=IHML(1)	ORBT0690		ISRUNT(I)=ISRUNT(J)	ORBT1260	
IDENT(2)=IHML(2)	ORBT0700		ISRUNT(J)=ITEMP	ORBT1270	
CALL MATRT (NBASIS,L,IDENT,C,VALU,NMAX,LABEL,INDEX)	ORBT0710		ITEMP=ITRANI(I)	ORBT1280	
C*** GENERATE CROSS-REFERENCE TABLES	ORBT0720		ITRANI(I)=ITRANI(J)	ORBT1290	
IND=0	ORBT0730		ITRANI(J)=ITEMP	ORBT1300	
DO 11 I=1,NEVALB	ORBT0740	150 CONTINUE		ORBT1310	
NDI=NDGB(I)	ORBT0750	151 CONTINUE		ORBT1320	
IN=IREPLB(I)	ORBT0760		ON 152 I=1,NOR	ORBT1330	
DO 11 J=1,NDI	ORBT0770		IANG=IANGT(I)	ORBT1340	
IND=IND+1	ORBT0780		IRAD=IRADT(I)	ORBT1350	
11 IANGB(IND)=IANG(J,IR)	ORBT0790		IREP(I)=IREP(IANG)	ORBT1360	
WRITE (10UT,10) IANGB(I),I=1,NBASIS	ORBT0800		ISRUNT=ISRUNT(I)	ORBT1370	
10 FORMAT (///' ANGULAR QUANTUM NUMBER ASSIGNED TO EACH SYMMETRY VECTOR	ORBT0810		ITRANS=ITRANI(I)	ORBT1380	
10M'//20151	ORBT0820		ISOT(I)=ISUM,ITRANSI=1	ORBT1390	
NG=NBASIT(I)	ORBT0830	152 ISQCIANG,IRAD)=1		ORBT1400	
DO 6 I=1,NH	ORBT0840	C*** PRINTOUT OF TABLES		ORBT1410	
1VI=LVALUF(I,IT)	ORBT0850	WRITE (10UT,50) INRAD(I),I=1,NREPI		ORBT1420	
IF (LVI.NE.LI GO TO 6	ORBT0860	50 FORMAT (///' NO. OF TIMES EACH REPRESENTATION OCCURS'//L415)		ORBT1430	
ITRANS=ITRANI(I,IT)	ORBT0870	IF (INTER.EQ.0) RETURN		ORBT1440	
IF (IFIRSTITRANS)	ORBT0880	WRITE (10UT,155)		ORBT1450	
ISRUN=0	ORBT0890	155 FORMAT (1M,139X,'*** REFERENCE TABLES FOR SYMMETRY-ADAPTED BASIS		ORBT1460	
ISR=IF1	ORBT0900	I ***//)		ORBT1470	
DO 8 J=1,NEVALB	ORBT0910	WRITE (10UT,100)		ORBT1480	
NDJ=NDGB(IJ)	ORBT0920	100 FORMAT (19X,'SYMMETRY REP. SPECTES QUANTUM NUMBERS ATORBT1495			
IRJ=IREPLB(IJ)	ORBT0930	1M ATOM-TYPE TRANS. RUNNING INDEX//19X,*FUNCTION INDEX		ORBT1500	
C>>>> CHANGE NEXT CARD IF CORRESPONDING DIMENSION IS CHANGED	ORBT0940	2LABEL RADIAL ANGULAR SYMBOL INDEX INDEX WITHIN		ORBT1510	
IF (INRAD(IJ).GT.20) GO TO 20	ORBT0950	3TRANS.//19X,91CM-1//)		ORBT1520	
INRAD(IJ)=INRAD(IJ)+1	ORBT0960	IAONT=1		ORBT1530	
DO 13 M=L,NDJ	ORBT0970	DO 101 ISQ=1,NOR		ORBT1540	
IF (ISRUN.GT.NMAX) GO TO 20	ORBT0980	ITRANS=ITRANI(SQ)		ORBT1550	
ISRUN=ISRUN+1	ORBT0990	IANG=IANGT(ISQ)		ORBT1560	
LABF2=0	ORBT1000	IREP=IREP(IANG)		ORBT1570	
DO 12 K=L,NBASIS	ORBT1010	IRAD=IRADT(ISQ)		ORBT1580	
ASCDEF(ISR)=C(K,ISRUN)	ORBT1020	IF (IANG.NE.IAONT) WRITE (10UT,104)		ORBT1590	
12 ISR=ISR+1	ORBT1030	104 POKYAT I(K)		ORBT1600	
ISO=ISO+1	ORBT1040	IAONT=IANG		ORBT1610	
IANG=IANGB(ISRUN)	ORBT1050	DO 105 IT=1,NATYPE		ORBT1620	
IRAD=INRAD(IJ)	ORBT1060	NG=NBASIT(I)		ORBT1630	
IANGT(ISO)=IANG	ORBT1070	DO 102 I=1,NH		ORBT1640	
ISRUNT(ISO)=ISRUN	ORBT1080	IF (ITRANI(I,IT).NE.ITRANS) GO TO 102		ORBT1650	
ITRANI(ISO)=ITRANS	ORBT1090	GO TO 101		ORBT1660	
13 CONTINUE	ORBT1100	102 CONTINUE		ORBT1670	
8 CONTINUE	ORBT1110	105 CONTINUE		ORBT1680	
6 CONTINUE	ORBT1120	101 WRITE (10UT,103) ISO,IREP,ITYPE,IREPS,ITRANS,IREPS,IRAD,IANG,		ORBT1690	
1 CONTINUE	ORBT1130	1 I (SYMB(I),I,ITRANS,ISRUNT(ISO)		ORBT1700	
DO 151 J=1,NOR	ORBT1140	103 FORMAT (19K,15,7X,12,4X,2A4,5X,12,8X,12,8X,13,9X,13/)		ORBT1710	
IF (IANGT(I).EQ.1) GO TO 151	ORBT1150	RETURN		ORBT1720	
		20 WRITE (10UT,21)		ORBT1730	

PROGRAM: GPTHEORY	ROUTINE: DRBTL5	PAGE: 01	PROGRAM: GPTHEORY	ROUTINE: BASFMM	PAGE: 02
21 FORMAT (2// ' ARRAY LIMIT EXCEEDED IN SUBROUTINE DRBTL5')	DRBT1740		SUBROUTINE BASFMM (IL,NBASIS,FUNCT,IND)	BASF0010	
1 STOP=1	DRBT1750	C*** CALCULATES FUNCTION VALUES OF REAL BASIS FUNCTIONS CENTERED		BASF0020	
RETURN	DRBT1760	C*** ON ATOMS		BASF0030	
END	DRBT1770	IMPLICIT REAL*8 (A-H,O-Z)		BASF0040	
		COMMON/OUT1/PDS13,301,RADIUS(301,THETA(301),PHI(301),NATOMS,NATYPE,		BASF0050	
		1 NUMT(101),MUCT(101),ISYMB(101),ITYPAT(10,101),ITVPT(301),ITVPT(301),		BASF0060	
		2 INVAT(1301)		BASF0070	
		COMMON/WORK/2/XP,YP,ZP,RP,JSUB,LOEG,NUMB,LY		BASF0080	
		COMMON/XYZ/XG(14401,YG(14401,ZG(14401,XI(301),Y(301),Z(301)		BASF0090	
		DIMENSION FUNCT(100)		BASF0100	
		JSUB=0		BASF0110	
		DO 1 IN=1,NUMB		BASF0120	
		IA=ITYPAT(IN,101)		BASF0130	
		XP=XG(INDI-XI(IA)		BASF0140	
		YP=YG(INDI-YI(IA)		BASF0150	
		ZP=ZG(INDI-ZI(IA)		BASF0160	
		RP=XP*XP+YP*YP+ZP*ZP		BASF0170	
		CALL VALUE (IL,FUNCT)		BASF0180	
		1 JSUB=JSUB+LOEG		BASF0190	
		RETURN		BASF0200	
		END		BASF0210	

PROGRAM: GRINEXDY	ROUTINE: VALUE	PAGE: 63	PROGRAM: GRINEXDY	ROUTINE: VALUE	PAGE: 64
SUBROUTINE VALUE (L,FUNCT)	VALU0010		FACTOR=3.00*(R2-Z2)	VALU0590	
C**** RETURNS VALUES OF 2*L+1 BASIS FUNCTIONS AT POINT X,Y,Z	VALU0020		FUNCT(IISUB+1)=CONST*Y*(FACTOR-4.00*Y2)	VALU0600	
IMPLICIT REAL*8 (A-H,O-Z)	VALU0030		FUNCT(IISUB+7)=CONST*X*(4.00*X2-FACTOR)	VALU0610	
COMMON/NORR2/X,Y,Z,R2,ISUB,LDEC,NUMB,IT	VALU0040		FACTOR=FACTOR/3.00	VALU0620	
DIMENSION FUNCT(1100)	VALU0050		FUNCT(IISUB+8)=RT15 *X*Y*4.00*(2.00*X2-FACTOR)/R5	VALU0630	
DATA R00Y271.41421356237309507, R00Y371.73205080756887737,	VALU0060		FUNCT(IISUB+9)=RT15 *14.00*X2*(X2-FACTOR)*FACTOR*FACTOR/R5	VALU0640	
1 HRT10/L.5813883008418967, HROUT6/L.22474487139158907,	VALU0070		RETURN	VALU0650	
2 R00T15/3.87298334620741687, RT15/5.41607978309961607,	VALU0080		END	VALU0660	
3 RT70/8.36660026534075547	VALU0090				
LP1=LP1	VALU0100				
GO TO (1,2,3,4,5), LP1	VALU0110				
C**** S FUNCTIONS	VALU0120				
1 FUNCT(IISUB+1)=1.00/R2	VALU0130				
RETURN	VALU0140				
C**** P FUNCTIONS	VALU0150				
2 R2=1.00/R2	VALU0160				
FUNCT(IISUB+11)=Z*R2	VALU0170				
FUNCT(IISUB+2)=Y*R2	VALU0180				
FUNCT(IISUB+3)=X*R2	VALU0190				
RETURN	VALU0200				
C**** D FUNCTIONS	VALU0210				
3 R3=1.00/(R2*R2)	VALU0220				
FUNCT(IISUB+11)=0.500*(3.00*Z2-R2)*R3	VALU0230				
R2=RODT3*Z*R3	VALU0240				
FUNCT(IISUB+2)=Y*R2	VALU0250				
FUNCT(IISUB+3)=X*R2	VALU0260				
FUNCT(IISUB+4)=RODT3*X*Y*R3	VALU0270				
FUNCT(IISUB+5)=RODT3*(X*X-Y*Y)*(1.500*R3)	VALU0280				
RETURN	VALU0290				
C**** F FUNCTIONS	VALU0300				
4 X2=X*X	VALU0310				
Y2=Y*Y	VALU0320				
Z2=Z*Z	VALU0330				
R4=R2*R2	VALU0340				
FACT=5.00*Z2-R2	VALU0350				
FUNCT(IISUB+11)=Z*(FACT-2.00*R2)/R4	VALU0360				
FACTOR=HROUT6*FACT/R4	VALU0370				
FUNCT(IISUB+2)=Y*FACTOR	VALU0380				
FUNCT(IISUB+3)=X*FACTOR	VALU0390				
FUNCT(IISUB+4)=RODT15*2.00*X*Y*Z/R4	VALU0400				
FUNCT(IISUB+5)=RODT15*Z*(X2-Y2)/R4	VALU0410				
FACTOR=3.00*(R2-Z2)	VALU0420				
FUNCT(IISUB+6)=HRT10*Y*(FACTOR-4.00*Y2)/R4	VALU0430				
FUNCT(IISUB+7)=HRT10*X*(4.00*X2-FACTOR)/R4	VALU0440				
RETURN	VALU0450				
C**** G FUNCTIONS	VALU0460				
5 X2=X*X	VALU0470				
Y2=Y*Y	VALU0480				
Z2=Z*Z	VALU0490				
R5=R2*R2*R2	VALU0500				
FUNCT(IISUB+11)=(17.500*Z2*Z2-15.00*(Z2*R2+1.500*R2*R2))/R5	VALU0510				
FACTOR=HRT10*Z*(14.00*Z2-6.00*R2)/R5	VALU0520				
FUNCT(IISUB+2)=FACTOR*Y	VALU0530				
FUNCT(IISUB+3)=FACTOR*X	VALU0540				
FACTOR=RODT2 *HRT10*(17.00*Z2-R2)/R5	VALU0550				
FUNCT(IISUB+4)=X*Y*FACTOR*2.00	VALU0560				
FUNCT(IISUB+5)=(X2-Y2)*FACTOR	VALU0570				
CONST=RT10*Z/R5	VALU0580				

PROGRAM: GRIMERY	ROUTINE: VECUPL	PAGE: 65	PROGRAM: GRIMERY	ROUTINE: VECUPL	PAGE: 66
SUBROUTINE VECUPL	VEC00010		DO 85 I=1,L02	VEC00590	
C**** CALCULATES FINITE POINT-GROUP VECTOR COUPLING COEFFICIENTS	VEC00020		IF (DABS(AA(1:1)).LE.CRT(PO)) AA(1)=0.	VEC00600	
IMPLICIT REAL*8 (A-H,O-Z)	VEC00030		IF (DABS(DB(1:1)).LE.CRT(RO)) DB(1)=0.	VEC00610	
COMMON/CONTROL/CATZRO,CRTDEG,ISTOP,INTER,LFLOW(10)	VEC00040		85 CONTINUE	VEC00620	
COMMON/PROTCD/CPROD(LN02),CVP(56),IREPL(75),NEVALP(2)	VEC00050		C**** GENERATE REQUIRED 3-J SYMBOLS	VEC00630	
COMMON/OUT3/NAEP,NOEGAT(4),TANGAT(5,14),IREPAT(32),IDROR,LVAL1,	VEC00060		22 CALL WIG3J(LVAL1,LVAL1,LVAL1,ISTOP)	VEC00640	
1 LVAL2,IRTYPI(14),IRPAR(14),LODD,IRLOWL(14)	VEC00070		IF (ISTOP.EQ.1) RETURN	VEC00650	
COMMON/OUT4/VCCDEF(2350),IVCCB(85,14),IFVCC(50)	VEC00080		C**** TRIPLE LOOP OVER SYMMETRY SPECIES AND SUBSPECIES	VEC00660	
COMMON/NICMER(533127,17),23J(13,17)	VEC00090		IF (INTER.NE.0) WRITE (IDUT,1100)	VEC00670	
COMMON/FILES/IMPUL,LOUT	VEC00100		1100 FORMAT (' UNNORMALIZED V.C.C.S?')	VEC00680	
COMMON/NDK1/DUMMY(8000)	VEC00110		IFV=1	VEC00690	
DIMENSION VCTEMP(75),AA(1089),AB(1089),AD(961),BD(961),IREPLB(50),	VEC00120		IVC=0	VEC00700	
1 NDEGS(150)	VEC00130		ISUB=1	VEC00710	
EQUIVALENCE (AA,DUMMY), (AB,DUMMY(1090)), (AD,DUMMY(5051)),	VEC00140		LAB3=0	VEC00720	
1 (BD,DUMMY(6012)), (VCTEMP,DUMMY(6973)), (IREPLB,DUMMY(7950)),	VEC00150		NARK=0	VEC00730	
2 (NDEGS,DUMMY(7975)), (INEVALS,DUMMY(8000))	VEC00160		DO 9 I3=1,NEVAL	VEC00740	
DATA IHS/IHS7,IHA/IHA7	VEC00170		LABE13=IREPL(13)	VEC00750	
ROOT2=DSQRT(2.00)	VEC00180		IR3=LABE13	VEC00760	
NEVAL=NEVALP(1)	VEC00190		NDEG3=NDEGA(LABE13)	VEC00770	
LOEG=2*LVAL1+1	VEC00200		IF (LAB3.EQ.LABE13) GO TO 9	VEC00780	
LP1=LVAL1+1	VEC00210		IF (NDEG3.LE.NOMAX) LAB3=LABE13	VEC00790	
NOMAX=3	VEC00220		IF (NARK.EQ.1) LAB3=LABE13	VEC00800	
IF (IDROR.EQ.24.AND.NREP.EQ.6) NOMAX=2	VEC00230		IF (NARK.EQ.1) IR3=IR3+2	VEC00810	
IF (IDROR.EQ.12.AND.NREP.EQ.3) NOMAX=2	VEC00240		IF (NARK.EQ.1.AND.NOMAX.EQ.2) IR3=IR3-1	VEC00820	
IANTI=0	VEC00250		ISUB=-1	VEC00830	
NAQN=0	VEC00260		LAB1=0	VEC00840	
NR=NREP	VEC00270		DO 5 I1=1,NEVAL	VEC00850	
IF (LODD.NE.0) NR=NREP/2	VEC00280		LABE11=IREPL(1)	VEC00860	
NR=NR	VEC00290		NDEG1=NDEGA(LABE11)	VEC00870	
C**** DETERMINE IF SOME V.C.C.S ARE ANTISYMMETRIC	VEC00300		IF (LAB1.EQ.LABE11) GO TO 5	VEC00880	
DO 50 I=1,NP	VEC00310		LAB1=LABE11	VEC00890	
IF (NDEGA(1),GT.NOMAX) NR2=NR+2	VEC00320		ISUB2=-1	VEC00900	
50 NAQN=NAQN+NDEGA(1)	VEC00330		DO 7 I2=1,1	VEC00910	
IF (NAQN.GT.NR) IANTI=1	VEC00340		LABE12=IREPL(I2)	VEC00920	
IF (INOMAX.EQ.2) NR2=NR2-1	VEC00350		NDEG2=NDEGA(LABE12)	VEC00930	
ISUB=0	VEC00360		IF (LAB2.EQ.LABE12) GO TO 7	VEC00940	
LODD=0	VEC00370		LAB2=LABE12	VEC00950	
IF (IANTI.EQ.1) CALL ODDG (LODD,LDEG)	VEC00380		IDP2=LABE12+LABE11-1/2	VEC00960	
LOP1=LODD+1	VEC00390		IF (LODD.EQ.0) GO TO 21	VEC00970	
C**** STORE PROTOTYPE VECTORS IN CONVENIENT FORM	VEC00400		LAB1NR=LABE11+NR	VEC00980	
DO 1 J=1,LDEG	VEC00410		LAB2NR=LABE12+NR	VEC00990	
JJ=LP1+LDEG*(J-1)	VEC00420		IRNR2=IR3+NR2	VEC01000	
AA(JJ)=CPROD(ISUB+1)	VEC00430		LTEM=LAB1NR+LAB2NR-1/2	VEC01010	
BD(JJ)=0.	VEC00440		IDPR1=LAB2NR+LTEM	VEC01020	
PHASE=1.00	VEC00450		IDPR2=LABE12+LTEM	VEC01030	
DO 2 M=1,LVAL1	VEC00460		IDPR3=LABE11+LAB2NR+LAB2NR-1/2	VEC01040	
JP=JJ+M	VEC00470		21 IVSUB=0	VEC01050	
JN=JJ+M	VEC00480		DO 10 IG3=1,NDEG3	VEC01060	
ISUB2M=ISUB+2*M	VEC00490		ISG3=(ISUB3+IG3)*LDEG	VEC01070	
PHASE=-PHASE	VEC00500		IF (NARK.EQ.0) GO TO 27	VEC01080	
COFF=CPROD(ISUB2M+1)/ROOT2	VEC00510		IF (NOMAX.EQ.2) GO TO 60	VEC01090	
AA(JP)=PHASE*COFF	VEC00520		IF (NDEG1.LT.5.OR.NDEG2.LT.4) GO TO 13	VEC01100	
AA(JN)=COFF	VEC00530		IF (NDEG3.EQ.4.AND.NDEG2.EQ.4) GO TO 13	VEC01110	
COFF=CPROD(ISUB2M+1)/ROOT2	VEC00540		IF (NDEG3.EQ.4) GO TO 24	VEC01120	
BB(JP)=PHASE*COFF	VEC00550		GO TO 27	VEC01130	
2 981JM=-COFF	VEC00560		60 IF (NDEG1.LT.3.OR.NDEG2.LT.3) GO TO 13	VEC01140	
1 ISUB=ISUB+LDEG	VEC00570		GO TO 24	VEC01150	
LO2=LOP1+LDEG	VEC00580		C**** SUBSPECIES LOOP FOR SYMMETRIC VECTOR COUPLING COEFFICIENTS	VEC01160	

PROGRAM: GPINORDY	ROUTINE: VECU1	PAGE: 57	PROGRAM: GPINORDY	ROUTINE: VECU1	PAGE: 58
27 SNORM=0.	VECUI180		1XN 34 K=1,LOEG	VECUI180	
IVS=1	VECUI190		M1=K-LP1	VECUI190	
DO 6 IG1=1,NDEG1	VECUI200		K1=ISG1+K	VECUI190	
ISG1=(ISUB1+IG1)*LOEG	VECUI210		PHASE=2.00	VECUI190	
DO 8 IG2=1,NDEG2	VECUI220		DO 34 N=1,LOEG	VECUI190	
ISG2=(ISUB2+IG2)*LOEG	VECUI230		PHASE=-PHASE	VECUI190	
SUM=AA1ISG1+LP1)*AA1ISG2+LP1)*AA1ISG3+LP1)*S3J11,1)	VECUI240		M3=N-LOP1	VECUI190	
DO 11 K=LP1,LOEG	VECUI250		M3=ISG3+N	VECUI190	
M1=K-LP1	VECUI260		M2=M3-M1	VECUI190	
K1=ISG1+K	VECUI270		IF (IABSIN2).GT.LVAL1) GO TO 34	VECUI190	
LIN=LOEG-M1	VECUI280		KN2=ISG2+LPL+M2	VECUI190	
DO 11 N=1,LIN	VECUI290		T1=0.	VECUI190	
IF (M1.EQ.0.AND.N.LE.LP1) GO TO 11	VECUI300		T2=0.	VECUI190	
M2=ISG2+N	VECUI310		IF (IABSIN1.EQ.0.) GO TO 87	VECUI190	
M2=N-LP1	VECUI320		T1=IABS(K1)*ABS(KN2)-AA(K1)*AA(KN2)+BO(N3)	VECUI190	
M3=M1+M2	VECUI330		87 IF (IABSIN1.EQ.0.) GO TO 88	VECUI190	
PHASE=2.00	VECUI340		T2=IABS(K1)*AA(KN2)+AA(K1)*BB(KN2)+AO(N3)	VECUI190	
IF (I2*(M3/2).NE.M3) PHASE=-PHASE	VECUI350		NA Y3=T1+T2	VECUI190	
MSUB1=IABS(M2-M1)/2+1	VECUI360		IF (T3.EQ.0.) GO TO 34	VECUI190	
MSUB2=IABS(M3+1)	VECUI370		SUM=SUM+T3*SYNBSJ(M1,M2,M3)*PHASE	VECUI190	
IF (MSUB2.GT.LP1) GO TO 11	VECUI380		34 CONTINUE	VECUI190	
KN3=M3+LPL+ISG3	VECUI390		IF (IABS(SUM).LE.1.00-6) SUM=0.	VECUI190	
T1=0.	VECUI400		VCTEMP(IVS)=SUM	VECUI190	
T2=0.	VECUI410		SNORM=SNORM+SUM*SUM	VECUI190	
IF (IABS(KN3).EQ.0.) GO TO 80	VECUI420		IVS=IVS+1	VECUI190	
T1=(AA(K1)*AA(IN2)-BB(K1)*BB(IN2))*AA(KN3)	VECUI430		33 CONTINUE	VECUI190	
80 IF (IABS(KN3).EQ.0.) GO TO 82	VECUI440		IFLAG=IMA	VECUI190	
T2=(AA(K1)*BB(IN2)+BB(K1)*AA(IN2))*BB(KN3)	VECUI450		IF (SNORM.LE.1.00-6) GO TO 13	VECUI190	
82 T3=T1+T2	VECUI460		23 RTNORM=DSORT(SNORM)	VECUI190	
IF (T3.EQ.0.) GO TO 11	VECUI470		IF (IG3.GT.1) GO TO 35	VECUI190	
SUM=SUM+T3*S3J(MSUB1,MSUB2)*PHASE	VECUI480		IVCB=IVCB+1	VECUI190	
11 CONTINUE	VECUI490		IAYB=IVCB	VECUI190	
IF (IABS(SUM).LE.CRT2RDI SUM=0.	VECUI500		IF (IFLAG.EQ.1MA) IAYB=-IVCB	VECUI190	
VCTEMP(IVS)=SUM	VECUI510		IVCB(IOPR,IR3)=IAYB	VECUI190	
SNORM=SNORM+SUM*SUM	VECUI520		IF (VCB(IVCB)=IFV	VECUI190	
IVS=IVS+1	VECUI530		35 IVS=1	VECUI190	
6 CONTINUE	VECUI540		IF (INTER.EQ.0) GO TO 100	VECUI190	
6 CONTINUE	VECUI550		ND12=NDEG1-NDEG2	VECUI190	
IFLAG=IHS	VECUI560		WRITE (IDUT,1000) LABEL1,LABEL2,LABEL3,IFLAG,VCTEMP(IVS),JVS=1.	VECUI190	
IF (SNORM.GT.CRT2RDI) GO TO 23	VECUI570		1000	VECUI190	
IF (LABEL1.EQ.LABEL2.AND.NDEG1.GT.1) GO TO 24	VECUI580		1000	VECUI190	
IF (LABEL1.EQ.LABEL3.AND.NDEG1.GT.1) GO TO 24	VECUI590		C*** STORE V.C.C.S AND THEIR ADDRESS TABLES	VECUI190	
IF (LABEL2.EQ.LABEL3.AND.NDEG2.GT.1) GO TO 24	VECUI600		DO 12 IG1=1,NDEG1	VECUI190	
GO TO 13	VECUI610		DO 12 IG2=1,NDEG2	VECUI190	
C*** SUBSPECIES LOOP FOR ANTISYMMETRIC V.C.C.S	VECUI620		VCCOE(IIV+IVSUB)=VCTEMP(IVS)/RTNORM	VECUI190	
24 TSURD=-1	VECUI630		IVSUB=IVSUB+1	VECUI190	
DO 30 I=1,NEVALS	VECUI640		IVS=IVS+1	VECUI190	
IF (I*REPLB11.EQ.LABEL3) GO TO 31	VECUI650		12 CONTINUE	VECUI190	
30 ISUBO=ISUBO+NDEGS11)	VECUI660		IF (IG3.GT.1) GO TO 10	VECUI190	
GO TO 13	VECUI670		IF (IG3.EQ.0) GO TO 10	VECUI190	
31 ISG3=ISUBO+IG3)*LOEG	VECUI680		IVCB(IOPR,IRNR2)=0	VECUI190	
SNORM=0.	VECUI690		IVCB(IOPR1,IR3)=IAYB	VECUI190	
IVS=1	VECUI700		IVCB(IOPR1,IRNR2)=0	VECUI190	
DO 33 IG1=1,NDEG1	VECUI710		IVCB(IOPR2,IRNR2)=IAYB	VECUI190	
ISG1=(ISUB1+IG1)*LOEG	VECUI720		IVCB(IOPR2,IR3)=0	VECUI190	
DO 33 IG2=1,NDEG2	VECUI730		IVCB(IOPR3,IR3)=0	VECUI190	
ISG2=(ISUB2+IG2)*LOEG	VECUI740		IVCB(IOPR3,IRNR2)=IAYB	VECUI190	
SUM=0.	VECUI750		GO TO 10	VECUI190	

PROGRAM: GRIMERY	ROUTINE: VECU1	PAGE 69	PROGRAM: GRIMERY	ROUTINE: DUGA	PAGE 70
13 IVCCB(IOPR,IR3)=0	VECU2340		SUBROUTINE DUGA ILODD,LL	00060010	
IF (ILODD.EQ.0) GO TO 7	VECU2350	C***	GENERATES SYMMETRY-ADAPTED SET OF ANTISYMMETRIC PROTOTYPE FUNCTION	00060020	
IVCCB(IOPR,IRNR2)=0	VECU2360		IMPLICIT REAL*8 IA-H,O-Z	00060030	
IVCCB(IOPR,IR3)=0	VECU2370		REAL*8 LABEL	00060040	
IVCCB(IOPR,IRNR2)=0	VECU2380		COMMON/OUT3/MREP,NDEGA(14),IANGA(5,14),IREPA(32),IORDER,LVAL	00060060	
IVCCB(IOPR2,IRNR2)=0	VECU2390		1 LVAL2,IRFP(14),IRPAR(14),ILODD,ILODD(14)	00060070	
IVCCB(IOPR2,IR3)=0	VECU2400		COMMON/PROTO/COPRO(1802),EVP(56),REPL(25),NEVALP(2)	00060080	
IVCCB(IOPR3,IR3)=0	VECU2410		COMMON/CNTAL/CRIZRD,CRIDEG,ISTOP,INTER,LFLON(10)	00060090	
IVCCB(IOPR3,IRNR2)=0	VECU2420		COMMON/FILES/INPUT,LOUT	00060100	
GO TO 7	VECU2430		COMMON/MARK/DUMMY(8000)	00060110	
10 CONTINUE	VECU2440		DIMENS ION LABEL(50),INDEX(50),IDENT(20),AD(96),BD(96),VALU(50)	00060120	
IFV=JFV+NDEG1*NDEG2*NDEG3	VECU2450		1,C(50,50),IREPL(50),NDEG(50)	00060130	
7 ISUB2=ISUB2+NDEG2	VECU2460		EQUIVALENCE (VALU,DUMMY(250)),IC,DUMMY(255)),IAD,DUMMY(505)),	00060140	
5 ISUB1=ISUB1+NDEG1	VECU2470		1 TWO,DUMMY(6012)),LABEL,DUMMY(7875)),INDEX,DUMMY(7925)),	00060150	
IF (MARK.EQ.0) GO TO 91	VECU2480		2 IIREPL8,DUMMY(7950)),INDEG,DUMMY(7975)),INEVAL,DUMMY(8000))	00060160	
MARK=0	VECU2490		DIMENSION LXA(5)	00060165	
GO TO 9	VECU2500		EXTERNAL TMLNK	00060170	
91 IF (NDEG3.GT.NDMAK MARK=1	VECU2510		DATA HOL1/TH SIN 7, HOL2/TH COS 7	00060185	
9 ISUB3=ISUB3+NDEG3	VECU2520		DATA IDENT/4H EVE,4H PA,4H RITY,4H DD,4H D-L,4H FMC,4H TION,	00060190	
IF (IORDER.EQ.60.DR.IORDER.EQ.120) CALL ORTHOG	VECU2530		1 4HS AN,4H TISY,4H HMET,4H RIC,4H IN P,4H ARTI,4H CLE,4H EXCH,4H ANGE,	00060200	
CALL VPRINT (NDMAK)	VECU2540		2 444H /	00060210	
RETURN	VECU2550		DATA LAB/2,1,3,5,4/	00060215	
END	VECU2560		NMAK=50	00060220	
			ILODD=LVAL2	00060230	
			IF (ILODD.EQ.0) GO TO 2	00060240	
			NEVAL=NEVALP(2)	00060250	
			IBASE=NEVALP(3)	00060260	
			DO I 1=1,NEVAL	00060270	
			IREPL(I)=IREPL(I+IBASE)-NREP/2	00060280	
			LABL6=IREPL(I)	00060290	
			1) NDEG11=NDEGA(LABL6)	00060300	
			LL=2*ILODD+1	00060310	
			LL1=2*LVAL1+1	00060320	
			ISUB=LL1+LL1	00060330	
			DO J 1=1,LL	00060340	
			VALU(J)=EVP(J+LL1)	00060350	
			DO I 1=1,LL	00060360	
			ISUB=ISUB+1	00060370	
			1 C(I,J)=COPRO(I,ISUB)	00060380	
			GO TO 8	00060390	
			2 IF (LVAL1.EQ.2) ILODD=3	00060400	
			IF (LVAL1.EQ.4) ILODD=5	00060410	
			IF (LVAL1.EQ.6) ILODD=9	00060420	
			IF (LVAL1.GT.6) ILODD=15	00060430	
			3 LL=2*ILODD+1	00060440	
			8 CALL RTGJJ (LVAL1,LVAL1,ILODD,ISTOP)	00060450	
			IF (ISTOP.EQ.1) RETURN	00060460	
			INDEX11=0	00060470	
			LABEL11=HOL2	00060480	
			DO 20 M=1,ILODD	00060490	
			M2=2*M	00060500	
			M2P1=M2+1	00060510	
			INDEX(M2)=M	00060520	
			INDEX(M2P1)=M	00060530	
			LABEL(M2)=HOL1	00060540	
			20 LABEL(M2P1)=HOL2	00060550	
			IF (ILODD.EQ.1) GO TO 7	00060560	
			4 CALL ADAPT (ILODD,LL,TMLNK,1,1,IDENT,0)	00060570	

PROGRAM: LPTHEORY	ROUTINE: D00G6	PAGE: 71	PROGRAM: OPTHEORY	ROUTINE: ORTHOG	PAGE: 72
IF 11STOP.EQ.1) RETURN	000G0580		SUBROUTINE ORTHOG	ORTH0010	
IF (NREP.EQ.5.AND.IORDER.EQ.24) GO TO 30	000G0581	C*** SCATTY ORTHOGONALIZES DOUBLY-OCCURRING SETS OF V.C.C.'S FOR POINT	ORTH0520		
GO TO 7	000G0582	C*** GROUPS K AND K (H)	ORTH0030		
30 DO 35 I=1,NEVAL	000G0583	IMPLICIT REAL*8 (A-H,O-Z)	ORTH0040		
ILAB=IREPLB(I)	000G0584	COMMON/CONTROL/CATZRO,CRTDEG,1STOP,INTER,LFLDW101	ORTH0050		
0) 36 J=1,5	000G0585	COMMON/OUT4/VCCDEF(2350),IVCCB(85,14),IFVCC(50)	ORTH0060		
IF (ILAB.EQ.J) IREPLB(I)=LAB(I)	000G0586	DIMENSION A(25),B(25)	ORTH0070		
36 CONTINUE	000G0587	DIMENSION C(25,10),NDEG(2),IREPLB(2)	ORTH0080		
35 CONTINUE	000G0588	NCOEF=20	ORTH0090		
7 IF (INTER.EQ.0) GO TO 4	000G0590	IND=14	ORTH0100		
CALL MATPAT (LL,I,IDENT,C,VALU,MHAX,LABEL,INDEX)	000G0590	4 IVCB1=IABS(IVCCB(IND,5))	ORTH0110		
WRITE (IOUT,12)	000G0610	IVCB2=IABS(IVCB(IND,7))	ORTH0120		
(2 FORMAT (/// REPRESENTATION INDEX ASSIGNED TO EACH SET OF VECTORS*)	000G0620	IFV1=IVCCB(IVCB1)-1	ORTH0130		
WRITE (IOUT,10) (IREPLB(I),I=1,NEVAL)	000G0630	IFV2=IVCCB(IVCB2)-1	ORTH0140		
10 FORMAT (//16J5)	000G0640	KK=0	ORTH0150		
4 ROOT2=DSORT(2.00)	000G0650	DO 2 K=1,5	ORTH0160		
LPI=LDD+1	000G0660	SUM=0.	ORTH0170		
ISUB=0	000G0670	DO 1 I=1,NCOEF	ORTH0180		
DO 5 I=1,LL	000G0680	J=I+KK	ORTH0190		
JJ=ISUB+LPI	000G0690	AT1=VCCDEF(IFV1+J)	ORTH0200		
AD1JJ=C(I,I)	000G0700	B11=VCCDEF(IFV2+J)	ORTH0210		
801JJ=0.	000G0710	3 SUM=SUM+AT1*B11	ORTH0220		
PHASE=1.00	000G0720	IF (K.GT.1) GO TO 5	ORTH0230		
DO 6 M=1,LDD	000G0730	SNORM=DSORT(1.00-SUM*SUM)	ORTH0240		
JP=JJ+M	000G0740	5 DO 3 I=1,NCOEF	ORTH0250		
JM=JJ+M	000G0750	J=I+KK	ORTH0260		
M2=2*M	000G0760	TERM=(B11-SUM*AT11)/SNORM	ORTH0270		
PHASE=-PHASE	000G0770	IF (IABS(TERM).LE.CATZRO) TERM=0.	ORTH0280		
COEF=C(INZ+1,I)/ROOT2	000G0780	3 VCCDEF(IFV2+J)=TERM	ORTH0290		
AD1JP=PHASE*COEF	000G0790	2 KK=KK+NCOEF	ORTH0300		
AD1JM=COEF	000G0800	IF (IND.EQ.14) GO TO 6	ORTH0310		
COEF=C(INZ,I)/ROOT2	000G0810	C*** SIMPLIFY SETS OF COEFFICIENTS FOR $V \times V = V$	ORTH0320		
801JP=PHASE*COEF	000G0820	DO 7 I=1,2	ORTH0330		
8 801JM=-COEF	000G0830	NDEG(I)=5	ORTH0340		
5 ISUB=ISUB+LL	000G0840	7 IREPLB(I)=5	ORTH0350		
RETURN	000G0850	KK=0	ORTH0360		
END	000G0860	DO 8 K=1,5	ORTH0370		
		DO 9 I=1,25	ORTH0380		
		J=I+KK	ORTH0390		
		C(I,K)=VCCDEF(IFV1+J)	ORTH0400		
		9 C(I,K+5)=VCCDEF(IFV2+J)	ORTH0410		
		8 KK=KK+25	ORTH0420		
		CALL CLEAN IC,25,NDEG,2,IREPLB,25,CRTZRO,0)	ORTH0430		
		KK=0	ORTH0440		
		DO 10 K=1,5	ORTH0450		
		DO 11 I=1,25	ORTH0460		
		J=I+KK	ORTH0470		
		VCCDEF(IFV1+J)=C(I,K)	ORTH0480		
		11 VCCDEF(IFV2+J)=C(I,K+5)	ORTH0490		
		10 KK=KK+25	ORTH0500		
		IF 4(IND.EQ.15) RETURN	ORTH0510		
		6 IND=15	ORTH0520		
		NCOEF=25	ORTH0530		
		GO TO 4	ORTH0540		
		END	ORTH0550		

PROGRAM: GRTEORY	ROUTINE: VPRINT	PAGE: 13	PROGRAM: GRTEORY	ROUTINE: VPRINT	PAGE: 14
SUBROUTINE VPRINT (NDMAX)	VPR10010		IR3=13	VPR10590	
C**** PRINTOUT OF SELECTION RULE AND V.C.C. TABLES	VPR10020		ND3=NDEGAI(3)	VPR10600	
IMPLICIT REAL*8 (A-H,O-Z)	VPR10030		IF (MARK.EQ.1) IR3=IR3+7	VPR10610	
COMMON/OUT5/NREP,NDFGA(14),IANGA(5,14),IREPAT(32),IDRDER,LVAL1,	VPR10040		IF (MARK.EQ.1.AND.NDMAX.EQ.2) IR3=(R3-1	VPR10620	
1 LVAL2,IRTP(14),IRPAR(14),IDDO,IRLOH(14)	VPR10050		DO 2 I=1,NR	VPR10630	
COMMON/OUT4/VCCDEF(2350),VCCB(85,14),IFVCC(SDI	VPR10060		ND1=NDEGAI(1)	VPR10640	
COMMON/F(15)/INPUT,IOUT	VPR10070		DO 2 I2=1,I1	VPR10650	
01NEWS(DN IDP(14),LINE(2)	VPR10080		ND2=NDEGAI(2)	VPR10660	
DATA LINE/4H----,3H---	VPR10090		ND12=ND1*ND2	VPR10670	
NR=NREP	VPR10100		IND=I2+I1*(I1-1)/2	VPR10680	
IF (IDDO.NE.0) NR=NR/2	VPR10110		IVCA=ABS(VCCB(IND,IR3))	VPR10690	
NLARK=2	VPR10120		IF (IVCB.EQ.0) GO TO 2	VPR10700	
LARK=0	VPR10130		WRITE (IOUT,3) I1,I2,I3	VPR10710	
DO 20 I=1,NR	VPR10140		3 FORMAT (///' REPRESENTATIONS ',I2,' X ',I2,' --> ',I2//	VPR10720	
IF (NDEGAI(1).GT.NDMAX) LARK=1	VPR10150		WRITE (IOUT,5) (IANGA(1,I3),I=1,ND3)	VPR10730	
20 CONTINUE	VPR10160		5 FORMAT (28X,' IANG3 --> ',I2/4X,' IANG1 IANG2',14X,I2,4(19X,I2))	VPR10740	
IF (NDMAX.FO.2) NLARK=1	VPR10170		WRITE (IOUT,25)	VPR10750	
WRITE (IOUT,10)	VPR10180		25 FORMAT (1X)	VPR10760	
10 FORMAT (1H,' SELECTION RULES FOR DIRECT PRODUCTS OF IRREDUCIBLE REPR	VPR10190		IVSUB=0	VPR10770	
PRESENTATIONS',1X,66(1H-1//)	VPR10200		(FV=IFVCC(IVCB)	VPR10780	
WRITE (IOUT,11)	VPR10210		DO 4 I=1,ND1	VPR10790	
11 FORMAT (28X,' REP.3 --> ')	VPR10220		IANG1=IANGA(1,I1)	VPR10800	
WRITE (IOUT,12) (IRTP(I),IRPAR(I),I=1,NR)	VPR10230		DO 4 J=1,ND2	VPR10810	
12 FORMAT (28X,14(A4,A3))	VPR10240		IANG2=IANGA(1,I2)	VPR10820	
WRITE (IOUT,13) (I,I=1,NR)	VPR10250		INIT=IFV+IVSUB	VPR10830	
13 FORMAT (1X,' REP.1 X REP.2',5X,14I7)	VPR10260		IND=INIT*(ND3-1)*ND12	VPR10840	
WRITE (IOUT,14) (LINE(I),I=1,Z),J=1,NR)	VPR10270		WRITE (IOUT,6) (IANG1,IANG2,(VCCDEF(I),K=INIT,IND,ND12)	VPR10850	
14 FORMAT (1X,25(1H-1,2X,14(A4,A3))	VPR10280		6 FORMAT (5X,I2,5X,I2,4X,3F20.9)	VPR10860	
DO 16 I=1,NR	VPR10290		4 IVSUB=IVSUB+I	VPR10870	
I1=IR1*(IR1-1)/2	VPR10300		2 CONTINUE	VPR10880	
DO 17 IR2=1,IR1	VPR10310		IF (MARK.EQ.0.AND.ND3.GT.NDMAX) GO TO 9	VPR10890	
IND=IR2*11	VPR10320		MARK=0	VPR10900	
DO 15 I=1,NR	VPR10330		GO TO 7	VPR10910	
15 IDP(1)=0	VPR10340		9 MARK=1	VPR10920	
ISEL=0	VPR10350		GO TO 8	VPR10930	
DO 18 I=1,NR	VPR10360		7 CONTINUE	VPR10940	
IR3=1	VPR10370		RETURN	VPR10950	
IF (LARK.EQ.1.AND.I.GT.NR) IR3=IR3+NLARK	VPR10380		END	VPR10960	
IDP1=VCCB(IND,IR3)	VPR10390				
IF (IDP1.EQ.0) GO TO 18	VPR10400				
IDP(1)=1	VPR10410				
IF (IDP1.LT.0) IDP(1)=-1	VPR10420				
ISEL=1	VPR10430				
IF (LARK.EQ.0.OR.NDEGAI(1).E.NDMAX) GO TO 18	VPR10440				
IDP1=VCCB(IND,IR3+NLARK)	VPR10450				
IF (IDP1.NE.0) IDP(1)=2	VPR10460				
IF (IDP1.LT.0) IDP(1)=-2	VPR10470				
18 CONTINUE	VPR10480				
IF (ISEL.EQ.0) GO TO 17	VPR10490				
WRITE (IOUT,19) IR1,IRTP(1R),IRPAR(1R),IR2,IRTP(2R),IRPAR(2R)	VPR10500				
19, (IDP(1),I=1,NR)	VPR10510				
19 FORMAT (1X,I2,1H,A4,A3,4H) X ,I2,1H,A4,A3,1H),2X,I4,13I7)	VPR10520				
17 CONTINUE	VPR10530				
16 CONTINUE	VPR10540				
WRITE (IOUT,1)	VPR10550				
1 FORMAT (1H,' VECTOR COUPLING COEFFICIENTS',1X,20(1H-1//)	VPR10560				
MARK=0	VPR10570				
DO 7 I3=1,NR	VPR10580				

PROGRAM: GPINERY	ROUTINE: VIBROT	PAGE: 75	PROGRAM: GPINERY	ROUTINE: FILEVI	PAGE: 77
SUBROUTINE VIBROT	VIBR0010		SUBROUTINE FILEVI	FILE00010	
RETURN	VIBR0020	C***	DUMP OUTPUT COMMON BLOCK INFORMATION ONTO TAPE OR DISK	FILE00020	
END	VIBR0030		IMPLICIT REAL*8 (A-H,O-Z)	FILE00030	
			COMMON/OUT1/DUMB1(3911)	FILE00040	
			COMMON/OUT18/DUMB18(2046)	FILE00050	
			COMMON/OUT22/DUMD2(7692)	FILE00060	
			COMMON/OUT3/DUMD3(811,10UMD3)	FILE00070	
			COMMON/OUT4/DUMD4(2970)	FILE00075	
			LDUMP=20	FILE00080	
			WRITE(LDUMP,DUMD1)	FILE00090	
			WRITE(LDUMP,DUMD6)	FILE00100	
			WRITE(LDUMP,DUMD2)	FILE00110	
			WRITE(LDUMP,DUMD3,LDUMD3)	FILE00120	
			WRITE(LDUMP,DUMD4)	FILE00130	
			RETURN	FILE00140	
			END	FILE00150	

PROGRAM: GPINERY	ROUTINE: APPLY	PAGE: 76
SUBROUTINE APPLY	APPL0010	
RETURN	APPL0020	
END	APPL0030	

ACKNOWLEDGMENTS

We are grateful to the Chemistry Division and the Applied Mathematics Division for their support and cooperation in the implementation of this method. We thank Dr. A. C. Wahl for helpful comments and suggestions, and are particularly indebted to Dr. A. L. H. Chung, whose concurrent work at this laboratory had considerable influence on the development of this work. The algorithms for "unscrambling" sets of vectors having the same symmetry and for their phase alignment are due to her.

REFERENCES

1. E. P. Wigner, *Group Theory and Its Applications to the Quantum Mechanics of Atomic Spectra*, Academic Press, New York (1959).
2. H. Bethe, Ann. Physik [5] 3, 133 (1929).
3. R. S. Mulliken, Phys. Rev. 40, 55; 41, 49, 751 (1932); 43, 279 (1933).
4. E. U. Condon and G. H. Shortley, *The Theory of Atomic Spectra*, Cambridge University Press (1935).
5. C. J. Ballhausen, *Introduction to Ligand Field Theory*, McGraw-Hill Book Co., New York (1962).
6. C. C. J. Roothaan, Rev. Modern Phys. 23, 69 (1951).
7. G. Racah, Phys. Rev. 62, 438 (1942); 63, 367 (1943); 76, 1352 (1949).
8. A. R. Edmonds, *Angular Momentum in Quantum Mechanics*, Princeton University Press (1957).
9. Y. Tanabe and S. Sugano, J. Phys. Soc. Japan 9, 753 (1954).
10. Y. Tanabe and H. Kamimura, J. Phys. Soc. Japan 13, 394 (1958).
11. J. S. Griffith, *The Theory of Transition Metal Ions*, Cambridge University Press (1961).
12. J. S. Griffith, *The Irreducible Tensor Method for Molecular Symmetry Groups*, Prentice-Hall, Inc., Englewood Cliffs, N. J. (1962).
13. B. D. Joshi, J. Chem. Phys. 43, 840 (1965).
14. A. C. Wahl, J. Chem. Phys. 41, 2600 (1964).
15. A. D. McLean and M. Yoshimine, IBM J. Res. Devel. 12, 206 (1968).
16. E. Clementi and D. R. Davis, J. Comp. Phys. 2, 223 (1967).
17. I. G. Csizmadia, M. C. Harrison, J. W. Moscovitz, and B. T. Sutcliffe, Theoret. Chim. Acta 6, 191 (1966).
18. Z. Gershgorin and I. Shavitt, Int. J. Quantum Chem. 15, 403 (1967).
19. S. H. Glarum, private communication.
20. P. Pulay, Gy. Borossay, and F. Török, J. Mol. Struct. 2, 336 (1968), and references therein.
21. S. Flodmark and E. Blokker, Int. J. Quantum Chem. 15, 703 (1967).
22. J. R. Gabriel, J. Math. Phys. 5, 494 (1964); 9, 973 (1968).
23. J. R. Gabriel, J. Math. Phys. 10, 1932 (1969).
24. R. Moccia, Theoret. Chem. Acta 7, 85 (1967).
25. R. McWeeny, *Symmetry - An Introduction to Group Theory*, Pergamon Press, Oxford, p. 195 (1963).
26. A. L. H. Chung and G. L. Goodman, to be published.
27. T. D. Bouman, A. L. H. Chung, and G. L. Goodman, in *Sigma Molecular Orbital Theory*, edited by O. Sinanoglu and K. Wiberg, Yale University Press, New Haven, pp. 333-349 (1970).

28. M. Gussoni and G. Zerbi, *J. Mol. Spectry.* 26, 185 (1968).
29. S. L. Altmann, *Proc. Cambridge Phil. Soc.* 53, 343 (1957).
30. S. L. Altmann and C. J. Bradley, *Proc. Roy. Soc. A* 255, 199 (1962).