Natural Science Report, Ochanomizu University, Vol. 27, No. 1, 1976
お茶の水女子大学 自然科学報告 第27巻 第1号

Two Electron Diatomic Molecules. I. The James-Coolidge Method and Its Programs

Naoko Ueda
Institute of Physics, College of General Education, University of Tokyo

Hiroshi Sato, Eiichi Ishiguro
Department of Physics, Faculty of Science, Ochanomizu University

and Terashi Takezawa
Department of Mathematics, Faculty of Science and Engineering, Nihon University

(Received April 10, 1976)

Computer programs which calculate energies and wave functions of two electron diatomic molecules by the James-Coolidge method are given. These programs can be applied to the singlet or the triplet Σ^+ states of two electron diatomic molecules of arbitrary nuclear charges.

§ 1. Introduction

To investigate various properties of molecules we need to know the electronic energies and the electronic wave functions of molecules in the adiabatic approximation scheme. Also to examine the atomic collision problems the knowledge on the energy correlation diagrams is very useful. For this purpose many methods, for example, Heitler-London method,[1] molecular orbital method,[2] and configuration mixing method[3] etc., have been developed. Especially, for two electron diatomic molecules, the James-Coolidge method is the most powerful and by this method James-Coolidge,[4] Kołos-Roothaan,[5] and Kołos-Wolniewicz[6] have examined the hydrogen molecule and obtained the successful results. To extend their studies, we constructed the computer programs which can obtain the electronic energies and the electronic wave functions of the Σ^+ states of two electron diatomic molecules.

In § 2 the James-Coolidge method is explained, and in § 3 simple explanation of the computer programs and how to use them are shown. In Appendix A the matrix components of the unity and the Hamiltonian operator are expressed by auxiliary functions $Z^\nu(m, n, j, k, p; 2\alpha)$ and in Appendix B the method of evaluating the functions $Z^\nu(m, n, j, k, p; 2\alpha)$ is shown. Furthermore, we give the lists of the computer programs in Appendix C.

§ 2. Method of calculation

In this paper we shall always use the atomic units. The non-relativistic Hamiltonian of two electron diatomic molecules in the adiabatic approximation is given by

$$\begin{aligned} H &= T + U\,, \\ T &= -\frac{1}{2}(\Delta_1 + \Delta_2)\,, \\ U &= -\frac{Z_a}{r_{a1}} - \frac{Z_b}{r_{b1}} - \frac{Z_a}{r_{a2}} - \frac{Z_b}{r_{b2}} + \frac{1}{r_{12}} + \frac{Z_a Z_b}{R}\,, \end{aligned} \tag{1}$$

with the notation of Fig. 1, Z_a and Z_b being the nuclear charges.

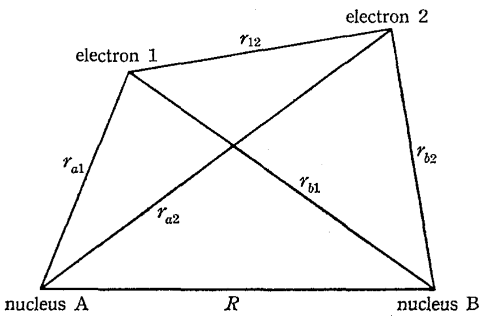

Fig. 1. Coordinates

Putting

$$\begin{aligned} \lambda_i &= (r_{ai} + r_{bi})/R\,, \\ \mu_i &= (r_{ai} - r_{bi})/R\,, \quad (i=1, 2)\,, \end{aligned} \tag{2a}$$

and

$$\rho = 2r_{12}/R\,, \tag{2b}$$

the Hamiltonian (1) is expressed as follows,

$$\begin{aligned} H &= T + \left(\frac{Z_a + Z_b}{2}\right)U^{(1)} + \left(\frac{Z_a - Z_b}{2}\right)U^{(2)} + U^{(3)} + \frac{Z_a Z_b}{R} \\ &= R^{-2}T' + \left(\frac{Z_a + Z_b}{2}\right)R^{-1}U'^{(1)} + \left(\frac{Z_a - Z_b}{2}\right)R^{-1}U'^{(2)} + R^{-1}U'^{(3)} + R^{-1}Z_a Z_b\,, \end{aligned} \tag{3a}$$

where

$$T' = -\frac{1}{2}(\Delta_1 + \Delta_2)R^2\,, \tag{3b}$$

$$U'^{(1)} = -4\lambda_1/(\lambda_1^2-\mu_1^2)-4\lambda_2/(\lambda_2^2-\mu_2^2)\ , \tag{3c}$$

$$U'^{(2)} = 4\mu_1/(\lambda_1^2-\mu_1^2)+4\mu_2/(\lambda_2^2-\mu_2^2)\ , \tag{3d}$$

$$U'^{(3)} = \frac{2}{\rho}\ . \tag{3e}$$

For the Σ^+ state, the James-Coolidge wave function is

$$\Psi = \Sigma C_{mnjkp}[mnjkp]\ , \tag{4}$$

where

$$[mnjkp] = N_{mnjkp} \exp[-\alpha(\lambda_1+\lambda_2)](\lambda_1^m\lambda_2^n\mu_1^j\mu_2^k \pm \lambda_1^n\lambda_2^m\mu_1^k\mu_2^j)\rho^p\ . \tag{5}$$

Here m, n, j, k, and p are a set of numbers each of which takes zero or positive integer, α is a variation parameter, and N_{mnjkp} is a normalization constant. The sign $+$ (or $-$) corresponds to the singlet state (or the triplet state). Furthermore, in the case of homonuclear molecules, the restriction $j+k=$ even (or odd) is added for the g state (or for the u state).

The electronic energy E and the coefficients C_{mnjkp} are determined by the following secular equation,

$$\det|H_{fg}-ES_{fg}|=0\ , \tag{6}$$

where, f and g represent sets of m, n, j, k, and p. The S_{fg} and H_{fg} are the matrix components of the unity and the Hamiltonian operator,

$$S_{fg} = \int [m_f n_f j_f k_f p_f]\,[m_g n_g j_g k_g p_g]\, \mathrm{d}V_1 \mathrm{d}V_2\ , \tag{7a}$$

$$H_{fg} = R^{-2}T'_{fg} + R^{-1}\left\{\left(\frac{Z_a+Z_b}{2}\right)U'^{(1)}_{fg} + \left(\frac{Z_a-Z_b}{2}\right)U'^{(2)}_{fg} + U'^{(3)}_{fg} + Z_aZ_bS_{fg}\right\}\ , \tag{7b}$$

$$T'_{fg} = \int [m_f n_f j_f k_f p_f]\, T'[m_g n_g j_g k_g p_g]\, \mathrm{d}V_1 \mathrm{d}V_2\ , \tag{7c}$$

$$U'^{(1)}_{fg} = \int [m_f n_f j_f k_f p_f]\, U'^{(1)}[m_g n_g j_g k_g p_g]\, \mathrm{d}V_1 \mathrm{d}V_2\ , \tag{7d}$$

$$U'^{(2)}_{fg} = \int [m_f n_f j_f k_f p_f]\, U'^{(2)}[m_g n_g j_g k_g p_g]\, \mathrm{d}V_1 \mathrm{d}V_2\ , \tag{7e}$$

and

$$U'^{(3)}_{fg} = \int [m_f n_f j_f k_f p_f]\, U'^{(3)}[m_g n_g j_g k_g p_g]\, \mathrm{d}V_1 \mathrm{d}V_2\ . \tag{7f}$$

The method of calculation of the matrix components will be given in Appendix A.

§ 3 Computer programs

We have constructed a set of programs which calculates the total energies and the wave functions of two electron diatomic molecules by the James-Coolidge method. The lists of programs are given in Appendix C. These programs are divided into seven parts. Each part forms one job and the relations among the jobs are shown in Fig. 2.

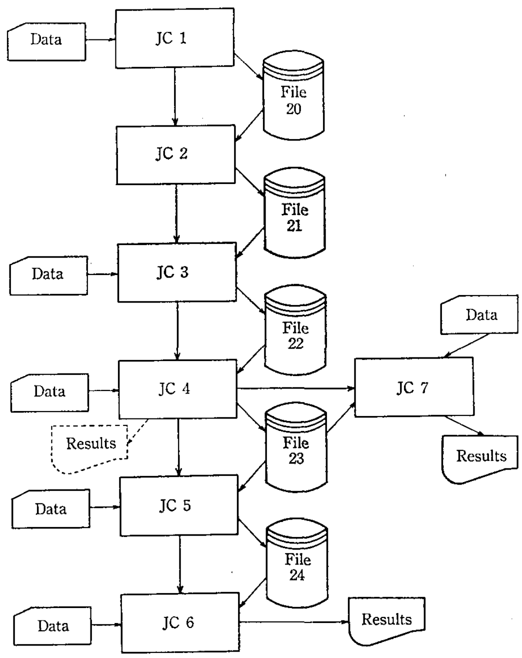

Fig. 2. Relations among the jobs.

1. Job JC1

This job calculates the auxiliary functions $Z^{\nu}(m, n, j, k, p; 2\alpha)$ for $p=-1$ and 0. The range of ν, m, n, j, k is

$$\begin{aligned} &0 \leqq \nu \leqq 3 , \\ &0 \leqq n \leqq m \leqq 20-2\nu , \\ &0 \leqq k \leqq j \leqq 20-2\nu . \end{aligned} \tag{8}$$

The quadruple precision calculation is used for the calculation of the auxiliary functions $W^{\nu}_{\tau}(m, n; 2\alpha)$ because of cancellation due to the recurrence formulas. For the remaining part, the double precision is used. As an input datum, the value of α is given. Then the values of the auxiliary functions $Z^{\nu}(m, n, j, k, p; 2\alpha)$ are written in the file (No. 20).

2. Job JC2

The values of $Z^{0}(m, n, j, k, p; 2\alpha)$ are evaluated from the results of the job JC1. The range of m, n, j, k, p is

$$0 \leqq n \leqq m \leqq 16 \;,$$
$$0 \leqq k \leqq j \leqq 16 \;, \qquad \text{for } -1 \leqq p \leqq 4 \;, \tag{9a}$$

and

$$0 \leqq n \leqq m \leqq 14 \;,$$
$$0 \leqq k \leqq j \leqq 14 \;, \qquad \text{for } p = 5, 6 \;. \tag{9b}$$

The double precision calculation is used. There are no imput data. The results of this job are written in the file (No. 21).

3. Job JC3

This job calculates the matrix components S_{fg}, T_{fg}, and U_{fg} for the unnormalized bases. The imput data are the nuclear charges Z_a and Z_b, the multiplicity, the number of bases adopted, and the sets of parameters determining the bases. The nuclear charges Z_a and Z_b must be given in real type and the multiplicity means $2S+1$, namely 1 for the singlet state and 3 for the triplet state. Moreover, the number of bases can't exceed 160. Of course, one can relax this limitation on the number of bases, if available core memories are increased. Since, however, then the necessary CPU time increases extraordinary, the practical merit is not expected. The base is defined by m, n, j, k, and p. The sets of m, n, j, k, and p are put in one after another according to the order of bases. The range of m, n, j, k, and p is

$$0 \leqq m,\, n,\, j,\, k \leqq 6 \;,$$
$$0 \leqq p \leqq 3 \;. \tag{10}$$

This program checks whether the base is allowed or not, evaluates the matrix components, and writes them in the file (No. 22).

4. Job JC4

The normalization of the bases and the rearrangement of the matrix components are very useful for the later calculation, and are performed by the use of the job JC4. The matrix components are written in the file (No. 23). Also they are printed, (or are not printed) in LP sheets if the input datum is not 0(or is 0).

5. Job JC5

In the calculation of the James-Coolidge method, it is desirable to select the important bases. But, the choice of the bases is very difficult and troublesome, because the nondiagonal matrix components of S are considerably large. If the orthonormalized bases are used, it seems that the bases which have a small coefficient in the wave function have small effect. Therefore, these bases can be dropped out. It is desirable that the bases which are taken off have large value of $m+n+j+k+p$, since the matrix components for the bases with small $m+n+j+k+p$ can be obtained with small error. For such reasons, we adopt Schmidt's orthonormalization. In the job JC5 Schmidt's orthonormalization and the transformation of matrix components are performed. In the job JC5, the quadruple precision calculation is used. If the input datum is 0, the unnecessary bases are taken off in the later calculation, otherwise not. The results are written in the file (No. 24).

6. Job JC6

This job calculates the total energies and the wave functions by the use of the results of the job JC5. If the internuclear distance and the number of order of the considered state counted from the lowest of the same symmetry are given, the total energy and the wave function are obtained. The coefficients of the bases are not for the orthonormalized ones. We can, however, obtain, as the output, the coefficients for the orthonormalized bases by the control of the input datum. The secular equation is solved with the quadruple precision.

7. Job JC7

This job is the additional ones to the job JC5. Because of the large overlapping of the bases, appreciable amounts of errors may arise when Schmidt's orthonormalization is performed. Then, it is necessary to check the order of this error. We use the job JC7 for this purpose. If the values of the matrix components are required for selecting the important bases, we can put out the matrix components H_{fg} by specifying the control number in the input datum.

Acknowledgement

This work is a part of the projects of the development of application program library for the HITAC 8700/8800 computer at the Computer Centre, University of Tokyo. Financial supports are partially afforded by the Grant-in-Aid for Scientific Researches from the Ministry of Education (No. 011604). We used the HITAC 8700/8800 computer at the Computer Centre, University of Tokyo.

References

1) W. Heitler and F. London: Z. Physik **44** (1927) 455.
2) F. Hund: Z. Physik **40** (1927) 742.
3) S. Weinbaum: J. chem. Phys. **1** (1933) 593.
4) H. M. James and A. S. Coolidge: J. chem. Phys. **1** (1933) 825.

5) W. Kołos and C. C. J. Roothaan: Rev. mod. Phys. **32** (1960) 219.
6) W. Kołos and L. Wolniewicz: J. chem. Phys. **49** (1968) 404.
7) M. Kotani, A. Amemiya, E. Ishiguro, and T. Kimura: *Tables of Molecular Integrals* (Maruzen, Tokyo, 1963).

Appendix A. The matrix components S_{fg}, T_{fg}, $U_{fg}^{(1)}$, $U_{fg}^{(2)}$, and $U_{fg}^{(3)}$.

$$\begin{aligned} S_{fg} &= N_f N_g \int \exp[-\alpha(\lambda_1+\lambda_2)](\lambda_1^{m_f}\lambda_2^{n_f}\mu_1^{j_f}\mu_2^{k_f}\rho^{p_f} \pm \lambda_1^{n_f}\lambda_2^{m_f}\mu_1^{k_f}\mu_2^{j_f}\rho^{p_f}) \\ &\quad \times \exp[-\alpha(\lambda_1+\lambda_2)](\lambda_1^{m_g}\lambda_2^{n_g}\mu_1^{j_g}\mu_2^{k_g}\rho^{p_g} \pm \lambda_1^{n_g}\lambda_2^{m_g}\mu_1^{k_g}\mu_2^{j_g}\rho^{p_g})\mathrm{d}V_1\mathrm{d}V_2 \\ &= 8\pi^2 N_f N_g \left(\frac{R}{2}\right)^6 \{s(m_f n_f j_f k_f p_f,\ m_g n_g j_g k_g p_g;\ \alpha) \\ &\quad \pm s(m_f n_f j_f k_f p_f,\ n_g m_g k_g j_g p_g;\ \alpha)\}\ , \end{aligned} \tag{A1}$$

where

$$\begin{aligned} & s(m_f n_f j_f k_f p_f,\ m_g n_g j_g k_g p_g;\ \alpha) \\ &\quad = \frac{1}{4\pi^2}\left(\frac{2}{R}\right)^6 \int \exp[-\alpha(\lambda_1+\lambda_2)]\lambda_1^{m_f}\lambda_2^{n_f}\mu_1^{j_f}\mu_2^{k_f}\rho^{p_f} \\ &\quad\quad \times \exp[-\alpha(\lambda_1+\lambda_2)]\lambda_1^{m_g}\lambda_2^{n_g}\mu_1^{j_g}\mu_2^{k_g}\rho^{p_g}\mathrm{d}V_1\mathrm{d}V_2\ . \end{aligned} \tag{A2}$$

Using

$$\mathrm{d}V = \left(\frac{R}{2}\right)^3(\lambda^2-\mu^2)\mathrm{d}\lambda\mathrm{d}\mu\mathrm{d}\phi\ , \tag{A3}$$

and the auxiliary functions

$$\begin{aligned} Z^\nu(m,\ n,\ j,\ k,\ p;\ 2\alpha) &= \frac{1}{4\pi^2}\int \exp[-2\alpha(\lambda_1+\lambda_2)]\lambda_1^m\lambda_2^n\mu_1^j\mu_2^k\rho^p \\ \times[(\lambda_1^2-1)(\lambda_2^2-1)(1-\mu_1^2)(1-\mu_2^2)]^{\nu/2} & \cos^\nu(\phi_1-\phi_2)\mathrm{d}\lambda_1\mathrm{d}\lambda_2\mathrm{d}\mu_1\mathrm{d}\mu_2\mathrm{d}\phi_1\mathrm{d}\phi_2\ , \end{aligned} \tag{A4}$$

$$\begin{aligned} & s(m_f n_f j_f k_f p_f,\ m_g n_g j_g k_g p_g;\ \alpha) \\ &\quad = Z^0(m_f+m_g+2,\ n_f+n_g+2,\ j_f+j_g,\ k_f+k_g,\ p_f+p_g;\ 2\alpha) \\ &\quad - Z^0(m_f+m_g+2,\ n_f+n_g,\ j_f+j_g,\ k_f+k_g+2,\ p_f+p_g;\ 2\alpha) \\ &\quad - Z^0(m_f+m_g,\ n_f+n_g+2,\ j_f+j_g+2,\ k_f+k_g,\ p_f+p_g;\ 2\alpha) \\ &\quad + Z^0(m_f+m_g,\ n_f+n_g,\ j_f+j_g+2,\ k_f+k_g+2,\ p_f+p_g;\ 2\alpha)\ . \end{aligned} \tag{A5}$$

For this expression the following abbreviation is used

$$\begin{aligned} & s(m_f n_f j_f k_f p_f,\ m_g n_g j_g k_g p_g;\ \alpha) \\ &\quad = Z(22000) - Z(20020) - Z(02200) + Z(00220)\ . \end{aligned} \tag{A6}$$

The method of calculation of the functions $Z^\nu(m,\ n,\ j,\ k,\ p;\ 2\alpha)$ will be given in Appendix B.

Similarly, the other matrix components are represented by using the same auxiliary functions $Z^\nu(m,\ n,\ j,\ k,\ p;\ 2\alpha)$. Namely,

$$\begin{aligned} T'_{fg} &= 8\pi^2 N_f N_g \left(\frac{R}{2}\right)^6 \{t(m_f n_f j_f k_f p_f,\ m_g n_g j_g k_g p_g;\ \alpha) \\ &\quad \pm t(m_f n_f j_f k_f p_f,\ n_g m_g k_g j_g p_g;\ \alpha)\}\ , \end{aligned} \tag{A7a}$$

$$U'^{(1)}_{fg} = 8\pi^2 N_f N_g \left(\frac{R}{2}\right)^6 \{u^{(1)}(m_f n_f j_f k_f p_f,\ m_g n_g j_g k_g p_g;\alpha)$$
$$\pm u^{(1)}(m_f n_f j_f k_f p_f,\ n_g m_g k_g j_g p_g;\alpha)\}\ , \tag{A7b}$$

$$U'^{(2)}_{fg} = 8\pi^2 N_f N_g \left(\frac{R}{2}\right)^6 \{u^{(2)}(m_f n_f j_f k_f p_f,\ m_g n_g j_g k_g p_g;\alpha)$$
$$\pm u^{(2)}(m_f n_f j_f k_f p_f,\ n_g m_g k_g j_g p_g;\alpha)\}\ , \tag{A7c}$$

and

$$U'^{(3)}_{fg} = 8\pi^2 N_f N_g \left(\frac{R}{2}\right)^6 \{u^{(3)}(m_f n_f j_f k_f p_f,\ m_g n_g j_g k_g p_g;\alpha)$$
$$\pm u^{(3)}(m_f n_f j_f k_f p_f,\ n_g m_g k_g j_g p_g;\alpha)\}\ . \tag{A7d}$$

Using the same abbreviation of Eq. (A6), the functions $t(m_f n_f j_f k_f p_f,\ m_g n_g j_g k_g p_g;\alpha)$ etc. are expressed as follows,

$$t(m_f n_f j_f k_f p_f,\ m_g n_g j_g k_g p_g;\alpha)$$
$$\begin{aligned}
= -\frac{1}{2}[& -4\alpha\{Z(12000)-Z(10020)+Z(21000)-Z(01200)\} \\
& +\{(m_f-m_g)^2+(m_f+m_g)-(j_f-j_g)^2-(j_f+j_g) \\
& +(p_f-p_g)(m_f-m_g-j_f+j_g)\}\{Z(02000)-Z(00020)\} \\
& +\{(n_f-n_g)^2+(n_f+n_g)-(k_f-k_g)^2-(k_f+k_g) \\
& +(p_f-p_g)(n_f-n_g-k_f+k_g)\}\{Z(20000)-Z(00200)\} \\
& -\{(m_f-m_g)^2-(m_f+m_g)\}\{Z(-22000)-Z(-20020)\} \\
& -\{(n_f-n_g)^2-(n_f+n_g)\}\{Z(2-2000)-Z(0-2200)\} \\
& +\{(j_f-j_g)^2-(j_f+j_g)\}\{Z(02-200)-Z(00-220)\} \\
& +\{(k_f-k_g)^2-(k_f+k_g)\}\{Z(200-20)-Z(002-20)\} \\
& +[2\{(p_f-p_g)^2+(p_f+p_g)\}+(p_f-p_g) \\
& \times(m_f-m_g+n_f-n_g+j_f-j_g+k_f-k_g)] \\
& \times\{Z(2200-2)-Z(2002-2)-Z(0220-2)+Z(0022-2)\} \\
& -(p_f-p_g)(m_f-m_g-j_f+j_g)\{Z(0400-2)-Z(0004-2)\} \\
& -(p_f-p_g)(n_f-n_g-k_f+k_g)\{Z(4000-2)-Z(0040-2)\} \\
& +2(p_f-p_g)(m_f-m_g)\{Z(-1311-2)-Z(-1113-2)\} \\
& +2(p_f-p_g)(n_f-n_g)\{Z(3-111-2)-Z(1-131-2)\} \\
& -2(p_f-p_g)(j_f-j_g)\{Z(13-11-2)-Z(11-13-2)\} \\
& -2(p_f-p_g)(k_f-k_g)\{Z(311-1-2)-Z(113-1-2)\}]\ ,
\end{aligned} \tag{A8a}$$

$$u^{(1)}(m_f n_f j_f k_f p_f,\ m_g n_g j_g k_g p_g;\alpha)$$
$$= -4\{Z(12000)-Z(10020)+Z(21000)-Z(01200)\}\ , \tag{A8b}$$

$$u^{(2)}(m_f n_f j_f k_f p_f,\ m_g n_g j_g k_g p_g;\alpha)$$
$$= 4\{Z(02100)-Z(00120)+Z(20010)-Z(00210)\}\ , \tag{A8c}$$

and

$$u^{(3)}(m_f n_f j_f k_f p_f,\ m_g n_g j_g k_g p_g\,;\ \alpha)$$
$$=2\{\{Z(2200-1)-Z(2002-1)-Z(0220-1)+Z(0022-1)\}\ . \quad \text{(A8d)}$$

Appendix B. Auxiliary functions $Z^{\nu}(m,\ n,\ j,\ k,\ p\,;\ 2\alpha)$.

The auxiliary functions $Z^{\nu}(m,\ n,\ j,\ k,\ p\,;\ 2\alpha)$ are defined by Eq. (A4). By the relation

$$\rho^2=\lambda_1^2+\lambda_2^2+\mu_1^2+\mu_2^2-2-2\lambda_1\lambda_2\mu_1\mu_2$$
$$-2[(\lambda_1^2-1)(\lambda_2^2-1)(1-\mu_1^2)(1-\mu_2^2)]^{1/2}\cos(\phi_1-\phi_2)\ , \quad \text{(B1)}$$

the following recurrence formula is obtained:

$$Z^{\nu}(m,\ n,\ j,\ k,\ p+2\,;\ 2\alpha)=Z^{\nu}(m+2,\ n,\ j,\ k,\ p\,;\ 2\alpha)$$
$$+Z^{\nu}(m,\ n+2,\ j,\ k,\ p\,;\ 2\alpha)+Z^{\nu}(m,\ n,\ j+2,\ k,\ p\,;\ 2\alpha)$$
$$+Z^{\nu}(m,\ n,\ j,\ k+2,\ p\,;\ 2\alpha)-2Z^{\nu}(m,\ n,\ j,\ k,\ p\,;\ 2\alpha)$$
$$-2Z^{\nu}(m+1,\ n+1,\ j+1,\ k+1,\ p\,;\ 2\alpha)-2Z^{\nu+1}(m,\ n,\ j,\ k,\ p\,;\ 2\alpha)\ . \quad \text{(B2)}$$

Therefore, $Z^{0}(m,\ n,\ j,\ k,\ p\,;\ 2\alpha)$ with $p\geqq 1$ can be obtained from $Z^{\nu}(m,\ n,\ j,\ k,\ -1\,;\ 2\alpha)$ and $Z^{\nu}(m,\ n,\ j,\ k,\ 0\,;\ 2\alpha)$.

For $p=0$, using the Eq. (A4),

$$Z^{0}(m,\ n,\ j,\ k,\ 0\,;\ 2\alpha)=4A_m(2\alpha)A_n(2\alpha)/[(j+1)(k+1)]\ , \quad \text{(B3a)}$$

$$Z^{1}(m,\ n,\ j,\ k,\ 0,\ 2\alpha)=0\ ,$$

$$Z^{2}(m,\ n,\ j,\ k,\ 0\,;\ 2\alpha)=8[A_{m+2}(2\alpha)-A_m(2\alpha)]$$
$$\times[A_{n+2}(2\alpha)-A_n(2\alpha)]/[(j+1)(j+3)(k+1)(k+3)] \quad \text{(B3c)}$$

and

$$Z^{3}(m,\ n,\ j,\ k,\ 0\,;\ 2\alpha)=0\ , \quad \text{(B3d)}$$

when j and k are both even; otherwise the values of these functions are zero. Here, the function $A_n(x)$[7] is defined by

$$A_n(x)\equiv\int_1^\infty \lambda^n e^{-x\lambda}d\lambda\ , \quad \text{(B4)}$$

and its value is calculated by the following recurrence formula,

$$A_n(x)=\frac{1}{x}\{e^{-x}+nA_{n-1}(x)\}\ , \quad \text{(B5a)}$$

$$A_0(x)=e^{-x}/x\ . \quad \text{(B5b)}$$

For $p=-1$, the Neumann expansion[8] of ρ^{-1},

$$\rho^{-1}=\sum_{\tau=0}^{\infty}\sum_{\nu=0}^{\tau} D_\tau^\nu P_\tau^\nu\begin{pmatrix}\lambda_1\\ \lambda_2\end{pmatrix} Q_\tau^\nu\begin{pmatrix}\lambda_2\\ \lambda_1\end{pmatrix} P_\tau^\nu(\mu_1)P_\tau^\nu(\mu_2)\cos\nu(\phi_1-\phi_2)\ ,\quad \begin{pmatrix}\lambda_1<\lambda_2\\ \lambda_2<\lambda_1\end{pmatrix}\ , \tag{B6a}$$

$$D_\tau^0=2\tau+1\ , \tag{B6b}$$

$$D_\tau^\nu=(-1)^\nu 2(2\tau+1)[(\tau-\nu)!/(\tau+\nu)!]^2\ ,\qquad \text{for } \nu>0\ , \tag{B6c}$$

is substituted into the Eq. (A4), and the following expressions are obtained.

$$Z^0(m,\ n,\ j,\ k,\ -1;\ 2\alpha)=\sum_{\tau=0}^{\infty}(2\tau+1)C_\tau^0(j)C_\tau^0(k)W_\tau^0(m,\ n;\ 2\alpha)\ , \tag{B7a}$$

$$Z^1(m,\ n,\ j,\ k,\ -1;\ 2\alpha)=-\sum_{\tau=1}^{\infty}[(2\tau+1)/\tau^2(\tau+1)^2]C_\tau^1(j)C_\tau^1(k)W_\tau^1(m,\ n;\ 2\alpha)\ , \tag{B7b}$$

$$\begin{aligned}Z^2(m,\ n,\ j,\ k,\ -1;\ 2\alpha)=&\frac{1}{2}\sum_{\tau=2}^{\infty}[(2\tau+1)/(\tau+2)^2(\tau+1)^2\tau^2(\tau-1)^2]\\ &\times C_\tau^2(j)C_\tau^2(k)W_\tau^2(m,\ n;\ 2\alpha)+\frac{1}{2}\sum_{\tau=0}^{\infty}(2\tau+1)\{C_\tau^0(j)-C_\tau^0(j+2)\}\\ &\times\{C_\tau^0(k)-C_\tau^0(k+2)\}\{W_\tau^0(m+2,\ n+2;\ 2\alpha)-W_\tau^0(m+2,\ n;\ 2\alpha)\\ &-W_\tau^0(m,\ n+2;\ 2\alpha)+W_\tau^0(m,\ n;\ 2\alpha)\}\ ,\end{aligned} \tag{B7c}$$

and

$$\begin{aligned}&Z^3(m,\ n,\ j,\ k,\ -1;\ 2\alpha)\\ &=-\frac{1}{4}\sum_{\tau=3}^{\infty}[(2\tau+1)/(\tau+3)^2(\tau+2)^2(\tau+1)^2\tau^2(\tau-1)^2(\tau-2)^2]\\ &\quad\times C_\tau^3(j)C_\tau^3(k)W_\tau^3(m,\ n;\ 2\alpha)-\frac{3}{4}\sum_{\tau=1}^{\infty}[(2\tau+1)/\tau^2(\tau+1)^2]\{C_\tau^1(j)-C_\tau^1(j+2)\}\\ &\quad\times\{C_\tau^1(k)-C_\tau^1(k+2)\}\{W_\tau^1(m+2,\ n+2;\ 2\alpha)-W_\tau^1(m+2,\ n;\ 2\alpha)\\ &\quad-W_\tau^1(m,\ n+2;\ 2\alpha)+W_\tau^1(m,\ n;\ 2\alpha)\}\ ,\end{aligned} \tag{B7d}$$

where $C_\tau^\nu(k)$[7] and $W_\tau^\nu(m,\ n;\ x)$[7] are

$$C_\tau^\nu(k)\equiv\int_{-1}^{1}d\mu(1-\mu^2)^{\nu/2}P_\tau^\nu(\mu)\mu^k\ , \tag{B8a}$$

$$=\frac{2^{\tau+1}k!(\tau+\nu)!\left(\dfrac{k+\tau+\nu}{2}\right)!}{(\tau-\nu)!\left(\dfrac{k+\nu-\tau}{2}\right)!(k+\tau+\nu+1)!}\ ,\qquad k+\tau+\nu=\text{even}\ ,\ \tau\geqq\nu\ ,\ k\geqq\tau-\nu\ , \tag{B8b}$$

$$=0\ ,\qquad \text{otherwise}\ , \tag{B8c}$$

and

$$W_\tau^\nu(m, n; x) \equiv \int_1^\infty d\lambda_1 \int_1^\infty d\lambda_2 [(\lambda_1^2-1)(\lambda_2^2-1)]^{\nu/2} P_\tau^\nu\begin{pmatrix}\lambda_1\\ \lambda_2\end{pmatrix} Q_\tau^\nu\begin{pmatrix}\lambda_2\\ \lambda_1\end{pmatrix} \lambda_1^m \lambda_2^n e^{-x(\lambda_1+\lambda_2)} . \quad \text{(B9)}$$

The auxiliary functions $W_\tau^\nu(m, n; x)$ are familiar in the literatures of molecular integrals and the method of their calculation is given by Kotani, Amemiya, Ishiguro, and Kimura.[7]

APPENDIX C   LISTS OF PROGRAMS   (1)

```
      PROGRAM JC1
C***********************************************************************
C*       PROGRAM JC1                                                   *
C*       TWO ELECTRON SYSTEM IN DIATOMIC MOLECULES                     *
C*             SIGMA PLUS STATE                                        *
C*             JAMES-COOLIDGE'S METHOD                                 *
C*       AUXILIARY FUNCTION Z(NU,M,N,J,K,P)                            *
C*             0.LE.NU.LE.3                                            *
C*             0.LE.N.LE.M.LE.20-NU*2                                  *
C*             0.LE.K.LE.J.LE.20-NU*2                                  *
C*             P=-1 AND 0                                              *
C*       INPUT DATA                                                    *
C*             ALPHA                                                   *
C***********************************************************************
      IMPLICIT REAL*8(A-H,O-Z)
      REAL*16 QALPH2
      DIMENSION Z(720)
      COMMON /FACTOR/FC,FD
      III=720
      FC=1.D-60
      FD=1.D0/DSQRT(FC)
C*****    INPUT      *************************************************
      READ(5,1000) ALPHA
      WRITE(6,2000) ALPHA
      ALPHA2=ALPHA+ALPHA
      QALPH2=ALPHA2
      CALL QW(17,3,14,QALPH2,QALPH2)
      CALL DA(ALPHA2)
      Z(1)=ALPHA
      I=1
C*****    Z(0,M,N,J,K,-1)         ***************************************
      DO 40 JP1=1,21
      J=JP1-1
      L=J.AND.1
      DO 40 KP1=1,JP1
      K=KP1-1
      IF(((J+K).AND.1).NE.0) GO TO 40
      DO 30 MP1=1,21
      M=MP1-1
      DO 30 NP1=1,MP1
      N=NP1-1
      I=I+1
      IF(I.LE.III) GO TO  10
      WRITE(20) Z
      I=1
   10 ZZ=0.D0
      DO 20 IT=L,K,2
      ZZ=ZZ+DFLOAT(IT+IT+1)*C(0,IT,J)*C(0,IT,K)*W(IT,0,M,N)
   20 CONTINUE
      Z(I)=ZZ
   30 CONTINUE
   40 CONTINUE
C*****    Z(1,M,N,J,K,-1)         ***************************************
      DO 80 JP1=1,19
      J=JP1-1
      L=(J.AND.1)+1
      DO 80 KP1=1,JP1
      K=KP1-1
      IF(((J+K).AND.1).NE.0) GO TO 80
      MM=K+1
      DO 70 MP1=1,19
      M=MP1-1
      DO 70 NP1=1,MP1
      N=NP1-1
      I=I+1
      IF(I.LE.III) GO TO  50
      WRITE(20) Z
      I=1
   50 ZZ=0.D0
      DO 60 IT=L,MM,2
      ZZ=ZZ-DFLOAT(IT+IT+1)*C(1,IT,J)*C(1,IT,K)*W(IT,1,M,N)
     2    /DFLOAT((IT*(IT+1))**2)
   60 CONTINUE
      Z(I)=ZZ
   70 CONTINUE
```

APPENDIX C LISTS OF PROGRAMS (2)

```
   80 CONTINUE
C*****     Z(2,M,N,J,K,-1)      ****************************************
      DO 130 JP1=1,17
      J=JP1-1
      L=J.AND.1
      DO 130 KP1=1,JP1
      K=KP1-1
      IF(((J+K).AND.1).NE.0) GO TO 130
      MM=K+2
      DO 120 MP1=1,17
      M=MP1-1
      DO 120 NP1=1,MP1
      N=NP1-1
      I=I+1
      IF(I.LE.III) GO TO 90
      WRITE(20) Z
      I=1
   90 ZZ=0.D0
      DO 100 IT=L,MM,2
      ZZ=ZZ+DFLOAT(IT+IT+1)*(C(0,IT,J+2)-C(0,IT,J))*(C(0,IT,K+2)
     2    -C(0,IT,K))*(W(IT,0,M+2,N+2)-W(IT,0,M+2,N)-W(IT,0,M,N+2)
     3    +W(IT,0,M,N))
  100 CONTINUE
      DO 110 IT=L+2,MM,2
      ZZ=ZZ+DFLOAT(IT+IT+1)*C(2,IT,J)*C(2,IT,K)*W(IT,2,M,N)
     2    /DFLOAT(((IT+2)*(IT+1)*IT*(IT-1))**2)
  110 CONTINUE
      Z(I)=ZZ*0.5D0
  120 CONTINUE
  130 CONTINUE
C*****     Z(3,M,N,J,K,-1)      ****************************************
      DO 180 JP1=1,15
      J=JP1-1
      L=(J.AND.1)+1
      DO 180 KP1=1,JP1
      K=KP1-1
      IF(((J+K).AND.1).NE.0) GO TO 180
      MM=K+3
      DO 170 MP1=1,15
      M=MP1-1
      DO 170 NP1=1,MP1
      N=NP1-1
      I=I+1
      IF(I.LE.III) GO TO 140
      WRITE(20) Z
      I=1
  140 ZZ=0.D0
      DO 150 IT=L,MM,2
      ZZ=ZZ-DFLOAT(IT+IT+1)*(C(1,IT,J+2)-C(1,IT,J))*(C(1,IT,K+2)
     2    -C(1,IT,K))*(W(IT,1,M+2,N+2)-W(IT,1,M+2,N)-W(IT,1,M,N+2)
     3    +W(IT,1,M,N))/DFLOAT(((IT+1)*IT)**2)
  150 CONTINUE
      ZZ=ZZ*3.D0
      DO 160 IT=L+2,MM,2
      ZZ=ZZ-DFLOAT(IT+IT+1)*C(3,IT,J)*C(3,IT,K)*W(IT,3,M,N)
     2    /DFLOAT(((IT+3)*(IT+2)*(IT+1)*IT*(IT-1)*(IT-2))**2)
  160 CONTINUE
      Z(I)=ZZ*0.25D0
  170 CONTINUE
  180 CONTINUE
C*****     Z(0,M,N,J,K,0)      *****************************************
      DO 200 JP1=1,21,2
      J=JP1-1
      DO 200 KP1=1,JP1,2
      K=KP1-1
      DO 200 MP1=1,21
      M=MP1-1
      DO 200 NP1=1,MP1
      N=NP1-1
      I=I+1
      IF(I.LE.III) GO TO 190
      WRITE(20) Z
      I=1
  190 Z(I)=4.D0*A(M)*A(N)/DFLOAT((J+1)*(K+1))
  200 CONTINUE
```

APPENDIX C LISTS OF PROGRAMS (3)

```
C*****    Z(2,M,N,J,K,0)       *****************************************
      DO 220 JP1=1,17,2
      J=JP1-1
      DO 220 KP1=1,JP1,2
      K=KP1-1
      DO 220 MP1=1,17
      M=MP1-1
      DO 220 NP1=1,MP1
      N=NP1-1
      I=I+1
      IF(I.LE.III) GO TO 210
      WRITE(20) Z
      I=1
  210 Z(I)=8.D0*(A(M+2)-A(M))*(A(N+2)-A(N))
     2    /DFLOAT((J+3)*(J+1)*(K+3)*(K+1))
  220 CONTINUE
      WRITE(20) Z
      STOP
 1000 FORMAT(D20.0)
 2000 FORMAT(1H1,10X,'AUXILIARY FUNCTION Z(NU,M,N,J,K,P)      P=-1,0',
     2//16X,'ALPHA =',F10.5,/1H1)
      END
      REAL FUNCTION A*8(N)
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /AD/AA(21)
      DIMENSION AAA(1)
      EQUIVALENCE (AAA(1),AA(2))
      A=AAA(N)
      RETURN
      END
      SUBROUTINE DA(ALPHA2)
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /AD/AA(21)
      ALI=1.D0/ALPHA2
      EXPMAL=DEXP(-ALPHA2)
      AA(1)=EXPMAL*ALI
      DO 10 N=1,20
      AA(N+1)=(EXPMAL+DFLOAT(N)*AA(N))*ALI
   10 CONTINUE
      RETURN
      END
      REAL FUNCTION C*8(NU,ITAU,M)
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /FACTOR/FC,FD
      COMMON /FCTRL/DFCT(57)
      DIMENSION DFCTRL(1)
      EQUIVALENCE(DFCTRL(1),DFCT(2))
      I=NU+ITAU+M
      J=M+NU-ITAU
      IF(((I.AND.1).NE.0).OR.(J.LT.0)) GO TO 20
      C=DFLOAT(2**(ITAU+1))*DFCTRL(M)/DFCTRL(I+1)
     2*DFCTRL(ITAU+NU)/DFCTRL(ITAU-NU)*DFCTRL(I/2)/DFCTRL(J/2)*FD
   10 RETURN
   20 C=0.D0
      GO TO 10
      END
      REAL FUNCTION W*8(/TAU/,/NU/,/M/,/N/)
C*****MOLECULAR INTEGRAL(KAS) AUXILIARY FUNCTION W ********************
C*               PICK OUT ROUTINE (W,DOUBLE PRECISION)                *
C***********************************************BY T.TAKEZAWA*****
      REAL*16 WTN
      COMMON/WC/WTN(42350),IAD(72),NT(5),ITNM1
      INTEGER TAU
      I=NT(NU+1)+TAU
      I=IAD(I)+(ITNM1-TAU-NU)*N+M
      W=WTN(I)
      RETURN
      END
      SUBROUTINE GW(ITX,INX,IMNX,ALPHA,BETA)
C*****MOLECULAR INTEGRAL(KAS) AUXILIARY FUNCTION W ********************
C*               W TAU,NU(M,N,ALPHA,BETA)  COVERING W0,0(0,0,ALPHA,BETA) *
C*                  TO W3,17(14,14,ALPHA,BETA)                         *
C*                  QUADRUPLE PRECISION                                *
C*                  MAX(TAU=20, MAX(NU)=3, MAX(M)=MAX(N)=34            *
C*                  MAX(TAU+NU)=20, MAX(TAU+NU+N)=34                   *
```

APPENDIX C LISTS OF PROGRAMS (4)

```
C*                   ALPHA.GT.0.0,  BETA.GT.0.0 ,  ALPHA+BETA.LE.88.7       *
C**************************************************************BY T.TAKEZAWA******
      IMPLICIT REAL*16(A-H,O-Z)
      REAL*16 NU
      LOGICAL NEZ
      COMMON/WC/WTN(42350),IAD(72),NT(5),ITNM1
      DIMENSION F0A(35),F0B(35),F1A(35),F1B(35),F0AB(70),F1AB(70),
     1     S1(35,35),S2(35,35),S3(35,35),S4(35,35),ZT(13896),IAW(77)
      EQUIVALENCE (WTN(4495),S1(1)),(WTN(5720),S2(1)),(WTN(6945),S3(1)),
     1 (WTN(8170),S4(1)),(WTN( 9395),F0A(1)),(WTN( 9430),F0B(1)),
     2 (WTN( 9465),F1A(1)),(WTN( 9500),F1B(1)),(WTN( 9535),F0AB(1)),
     3 (WTN( 9605),F1AB(1)),(WTN(13896),ZT(1))
      DATA ZERO,ONE,TWO,THREE,FIVE,C1,C2,C3,C4,KT,NN,NM,A0,B0 /
     1 0.Q0,1.Q0,2.Q0,3.Q0,5.Q0,.25Q0,.75Q0,.444444444444444444444444444
     2444444444Q0,.555555555555555555555555555555555555Q0,-1,-1,-1,0.,0.
     3 /
      DATA IAW/1,1226,2382,3471,4495,5456,6356,7197,7981,8710,9386,
     1      10011,10587,11116,11600,12041,12441,12802,13126,13415,13671,
     2      13896,14985,16009,16970,17870,18711,19495,20224,20900,21525,
     3      22101,22630,23114,23555,23955,24316,24640,24929,25185,25410,
     4      26371,27271,28112,28896,29625,30301,30926,31502,32031,32515,
     5      32956,33356,33717,34041,34330,34586,34811,35652,36436,37165,
     6      37841,38466,39042,39571,40055,40496,40896,41257,41581,41870,
     7      42126,1,21,39,55/
C   ***********************    CHECK ARGUMENTS    ***
      IF(ITX.LT.0.OR.ITX.GT.20.OR.INX.LT.0.OR.INX.GT.3.OR.IMNX.LT.0.
     2OR.IMNX.GT.34.OR.ITX+INX.GT.20.OR.ITX+INX+IMNX.GT.34) GO TO 500
      IF(ALPHA.LE.0.  .OR. BETA.LE.0.)GO TO 501
      IF(KT.GE.ITX .AND. NN.GE.INX .AND. NM.GE.IMNX .AND.
     '   A0.EQ.ALPHA .AND. B0.EQ.BETA) GO TO 200
      KT=ITX
      NN=INX
      NM=IMNX
      A0=ALPHA
      B0=BETA
      NEZ=INX.EQ.0
      DO 800 I=1,77
  800 IAD(I)=IAW(I)
C   ***********************    S0(M,N,ALPHA,BETA),S1(M,N,ALPHA,BETA)    ***
C   ***********************    S0(N,M,BETE,ALPHA),S1(N,M,BETA,ALPHA)    ***
      IWEND=ITX+INX+1
      MEND=IWEND+IMNX
      ITNM1=MEND
      CALL QF(MEND,ALPHA,F0A,F1A)
      CALL QF(MEND+MEND-1,ALPHA+BETA,F0AB,F1AB)
      BI=ONE/BETA
      EB= EXP(-BETA)
      H=ZERO
      IF(ALPHA.EQ.BETA)GO TO 300
      EA= EXP(-ALPHA)
      AI=ONE/ALPHA
      CALL QF(MEND,BETA,F0B,F1B)
      DO 15 N=1,MEND
      N1=N-1
      DO 10 M=1,MEND
      MN=M+N1
      S1(M,N)=(H*S1(M,N1)+EB*F0A(M)-F0AB(MN))*BI
      S2(M,N)=(H*S2(M,N1)+EB*F1A(M)-F1AB(MN))*BI
      S3(M,N)=(H*S3(M,N1)+EA*F0B(M)-F0AB(MN))*AI
   10 S4(M,N)=(H*S4(M,N1)+EA*F1B(M)-F1AB(MN))*AI
   15 H=ONE+H
C   ***********************    W0,0(M,N,ALPHA,BETA)    ***
  100 I=IAD(1)
      DO 20 N=1,MEND
      DO 20 M=1,MEND
      WTN(I)=S1(M,N)+S3(N,M)
   20 I=I+1
      IF(NEZ .AND. ITX.EQ.0) GO TO 200
C   ***********************    W1,0(M,N,ALPHA,BETA)    ***
      MEND=MEND-1
      I=IAD(2)
      DO 30 N=1,MEND
      NP1=N+1
      DO 30 M=1,MEND
      WTN( I)=S2(M,NP1)+S4(N,M+1)
```

APPENDIX C LISTS OF PROGRAMS (5)

```
   30 I=I+1
      IF(NEZ .AND. ITX.EQ.1) GO TO 200
C     **********************   Z0(M,N,ALPHA,BETA), W2,0(M,N,ALPHA,BETA)***
      I=IAD(3)
      K=IAD(1)
      L=IAD(2)+MEND
      MEND=MEND-1
      DO 45 N=1,MEND
      NP1=N+1
      NP2=NP1+1
      DO 40 M=1,MEND
      ZT(I)=S1(M,NP2)+S3(N,M+2)+S2(M+1,N)+S4(NP1,M)-THREE*WTN(L+1)
      WTN(I)=C1*WTN(K)-C2*ZT( I )
      I=I+1
      K=K+1
   40 L=L+1
      K=K+2
   45 L=L+1
      IF(NEZ .AND. ITX.EQ.2) GO TO 200
      MEND=MEND-1
C     ************************   Z1(M,N,ALPHA,BETA),W3,0(M,N,ALPHA,BETA)  **
      IR2A=IAD(2)
      IR1A=IR2A+(MEND+2)*2
      IR3A=IAD(1)+MEND+3
      IR4A=IAD(3)+MEND+1
      IR5A=IAD(4)
      DO 55 N=1,MEND
      DO 50 M=1,MEND
      ZT(IR5A)=THREE*(WTN(IR1A)+WTN(IR2A+2))-WTN(IR3A+1)-WTN(IR2A)
     ~         -FIVE*WTN(IR4A+1)
      WTN(IR5A)=C3*WTN(IR2A)-C4*ZT(IR5A)
      IR1A=IR1A+1
      IR2A=IR2A+1
      IR3A=IR3A+1
      IR4A=IR4A+1
   50 IR5A=IR5A+1
      IR1A=IR1A+2
      IR2A=IR2A+2
      IR3A=IR3A+3
   55 IR4A=IR4A+1
      IF(NEZ .AND. ITX.EQ.3) GO TO 200
C     **********************   W TAU,0 ,Z TAU   ***
      TAU=THREE
      DO 65 I=5,IWEND
      MEND=MEND-1
      TAU=TAU+ONE
      IR1A=IAD(I)
      IR2A=IAD(I-2)
      IR3A=IR2A+MEND+MEND+4
      IR4A=IAD(I-3)+MEND+3
      IR5A=IAD(I-1)+MEND+1
      T2I=ONE/(TAU*TAU)
      C01=TAU+TAU-THREE
      C02=C01-TWO
      C03=C01+TWO
      C04=TAU-ONE
      C04=C04*C04*T2I
      C05=(C01+TWO)*T2I
      DO 65 N=1,MEND
      DO 60 M=1,MEND
      ZT(IR1A)=ZT(IR2A)+C01*(WTN(IR2A+2)+WTN(IR3A))-C02*WTN(IR4A+1)
     1         -C03*WTN(IR5A+1)
      WTN(IR1A)=C04*WTN(IR2A)-C05*ZT(IR1A)
      IR1A=IR1A+1
      IR2A=IR2A+1
      IR3A=IR3A+1
      IR4A=IR4A+1
   60 IR5A=IR5A+1
      IR2A=IR2A+2
      IR3A=IR3A+2
      IR4A=IR4A+3
   65 IR5A=IR5A+1
      IF(NEZ) GO TO 200
C     **********************   W TAU,NU   ***
      L1=IWEND-2
```

APPENDIX C LISTS OF PROGRAMS (6)

```
      DO 85 JNU=1,INX
      NU=JNU
      IWEND=IWEND-2
      MEND=IWEND+IMNX
      DO 80 JTAU=JNU,L1
      TAU=JTAU
      IWI=NT(JNU+1)+JTAU
      IWIL=NT(JNU)+JTAU
      IR1A=IAD(IWI)
      IR2A=IAD(IWIL)+MEND+1
      IR3A=IAD(IWIL-1)
      IR4A=IAD(IWIL+1)
      C04=TAU+NU
      H=TAU+ONE
      C05=H-NU
      C06=H+TAU
      C01=-C04*C05
      C02=(C04-ONE)**2*C04/C06
      C03=(C05+ONE)**2*C05/C06
      DO 75 N=1,MEND
      DO 70 M=1,MEND
      WTN(IR1A)=C01*WTN(IR2A+1)+C02*WTN(IR3A)+C03*WTN(IR4A)
      IR1A=IR1A+1
      IR2A=IR2A+1
      IR3A=IR3A+1
   70 IR4A=IR4A+1
      IR2A=IR2A+1
   75 IR3A=IR3A+2
   80 MEND=MEND-1
   85 L1=L1-1
  200 RETURN
  300 DO 95 N=1,MEND
      NM1=N-1
      DO 90 M=1,MEND
      MN=M+NM1
      S1(M,N)=(H*S1(M,NM1)+EB*F0A(M)-F0AB(MN))*BI
      S2(M,N)=(H*S2(M,NM1)+EB*F1A(M)-F1AB(MN))*BI
      S3(M,N)=S1(M,N)
   90 S4(M,N)=S2(M,N)
   95 H=ONE+H
      GO TO 100
C  *********************     ERROR MESSAGE    ***
  500 WRITE(6,600) ITX,INX,IMNX
      STOP
  501 WRITE(6,601) ALPHA,BETA
      STOP
  600 FORMAT(1H0,15H(SUBR. QW) TAU=,I6,5X,3HNU=,I6,5X,2HM=,I6 /12X,60HTA
     1U AND NU AND M SHOULD BE SATISFY THE FOLLOWING INEQUATIONS /16X,
     2 73H0.GE.TAU.LE.20, 0.GE.NU.LE.3, 0.GE.M.LE.34, TAU+NU.LE.20
     3NU+M.LE.34, /)
  601 FORMAT(1H0,17H(SUBR. QW) ALPHA=,1PE23.15,5X,5HBETA=,E23.15,5X,
     1 25HARGUMENT OF W IS INVALID./)
      END
      SUBROUTINE QF(M,ALPHA,F0,F1)
C****MOLECULAR INTEGRAL(KAS) AUXILIARY FUNCTION F *******************
C*              F 0(M,ALPHA),  F 1(M,ALPHA)                          *
C*                 QUADRUPLE PRECISION (36 DIGITS)                   *
C*               M.GE.1, ALPHA.GT.0                                  *
C*********************************************************BY T.TAKEZAWA*****
      IMPLICIT REAL*16(A-H,O-Z)
      REAL*8 FC,FD
      DIMENSION F0(M),F1(M)
      COMMON /FACTOR/FC,FD
      DATA EULER/.577215664901532860606512090082402431/
      EM=EXP(-ALPHA)*FC
      X2=ALPHA+ALPHA
      EI2X=QEIE(-X2)
      EOGC=EULER+ LOG(X2)
      F0(1)=EM*(EOGC+EI2X)/X2
      XI=1.Q0/ALPHA
      H=2.Q0
      A0=EM*XI
      A1=(EM+A0)*XI
      A2=(EM+A1+A1)*XI
      EEX=EI2X*EM*XI
```

APPENDIX C LISTS OF PROGRAMS (7)

```
      F1(1)=(EDGC*A1-EEX+XI*EEX)/H-A0
      F1(2)=(EDGC*A2+EEX)/H-(EEX-XI*EEX+EM+H*XI*EM)*XI
      F1(3 )=F1(1)+(3.Q0*F1(2)-F0(1))*XI
      F0(2)=F1(1)+A0
      F0(3)=F1(2)+A1
      DO 10 I=4,M
      F0(I)=F1(I-1)+A2
      H=1.Q0+H
      A2=(EM+H*A2)*XI
   10 F1(I)=F1(I-2)+(3.Q0*F1(I-1)-F0(I-2)-(2.Q0-H)*(F1(I-1)-F1(I-3)))*XI
      RETURN
      END
      REAL FUNCTION QEIE*16(X)
C*****EXPONENTIAL INTEGRAL EI(X)*EXP(-X), -EI(-X)*EXP(X)****************
C*                   QUADRUPLE PRECISION (36 DIGITS)                   *
C*****************************************************BY T.TAKEZAWA*****
      IMPLICIT REAL*16(A-H,O-Z)
      DIMENSION X0(18),TABLEP(18),TABLEM(18)
C *** ZERO POINTS ***
      DATA X0/3.3Q0,4.Q0,4.8Q0,5.8Q0,7.1Q0,8.6Q0,10.4Q0,12.5Q0,15.2Q0,18
     1.4Q0,22.2Q0,26.9Q0,32.5Q0,39.4Q0,47.6Q0,57.6Q0,69.7Q0,83.Q0/,
     2C100/100.Q0/,C1,C2/1.Q0,2.Q0/
C *** MASTER VALUES OF EI(X)*EXP(-X) ***
      DATA TABLEP
     '            /44.8537724756737459666837503688703855,35.95520078636206
     1961774216758266074445,28.5554856795924443560201882968443875,22.2830
     213006931921405575831722085597,17.1508376212076528851381709984536159
     3,13.5265041139175554681803929873875053,10.8128297353874584470776091
     4433066202,8.7819020211785444084678027060768442 9,7.084719123528884
     5521098434637070440 4,
     6             5.7691153020385872666565239851254315 8,4.7287467102376741
     183837309792258654404,3.8673006094414383778382836153502167 5,3.1780
     20354772840819361996075400477 27,2.6060371294938505175312306283425761
     317,2.1469579435134789629974881859707721,1.767357146269297241281459
     489927502125,1.4559220991807543601507263870521299,1.21969824417642 0
     543671159294416866409/
C *** MASTER VALUES OF -EI(-X)*EXP(X) ***
      DATA TABLEM
     '            /24.2361032738517179842802903300138 46,20.63456499010558 3
     131020457588768578 09,17.6553899922275490844225877258125266,14.9678
     272506024076072575320065926 57,12.5040182387138624594210032550448208
     3,10.5136538324982612526442436790694708,8.83092443778196227673839951
     42840940368,7.4435275189775051256377627556441844 5,6.194083826443568
     523471326330680039114,
     6             5.1671872416555239680121590973557216 1,4.3177740433099171
     14476547464357425 2111,3.5885493954921341781086012050028467,2.98759
     241163658188489651049608223601,2.4766964747798062754343294276147264
     36,2.0584514873171346408106854456742809 2,1.706965828524471152442682
     480473572445,1.4147026000595521603693738100710450 2,1.190641095161811
     528733945913722 6850272/
C *** EULER CONSTANT ***
      DATA EULER/.577215664901532860606512090082402431/
      ABSX=ABS(X)
      IF( ABSX .GE. 90. ) GO TO 11
      IF( ABSX .GE.  3. ) GO TO 3
C ************* POWER SERIES EXPANSION ***
      IB=1
      QEIE=X
      TERM1=X
      TERM2=X
      H=C2
    1 TERM2=TERM2*X/H
      TERM1=TERM2/H
      GO TO 16
    2 QEIE= EXP(-X)*(QEIE+EULER+ LOG(ABSX))
      GO TO 14
C ************* INTERPOLATION (EXP(-X)*EI(X),EXP(X)*(-EI(-X))) ***
    3 IB=2
      DO 4 I=1,18
      IF( .5*(X0(I+1)+X0(I))-ABSX ) 4,4,5
    4 CONTINUE
      I=18
    5 IF(X)6,6,7
    6 XX=-X0(I)-X
      XXI=C1 +X/X0(I)
```

```
      APPENDIX C      LISTS OF PROGRAMS      (  8)

      TERM1=TABLEM(I)/C100                                              10005260
      GO TO 8                                                           10005270
    7 XX=X0(I)-X                                                        10005280
      XXI=XX/X0(I)                                                      10005290
      TERM1=TABLEP(I)/C100                                              10005300
    8 QEIE=TERM1                                                        10005310
      XI=C1 /X0(I)                                                      10005320
      H=C1                                                              10005330
      TERM2=H                                                           10005340
    9 TERM1=XX*(TERM1-TERM2*XI)/H                                       10005350
      GO TO 16                                                          10005360
   10 TERM2=TERM2*XXI                                                   10005370
      GO TO 9                                                           10005380
C ************ ASYMPTOTIC EXPANSION ***                                  10005390
   11 XI=C1/X                                                           10005400
      IB=3                                                              10005410
      QEIE=C1                                                           10005420
      H=C1                                                              10005430
      TERM1=C1                                                          10005440
   12 TERM1=TERM1*H*XI                                                  10005450
      GO TO 16                                                          10005460
   13 QEIE=QEIE/X                                                       10005470
   14 IF(X.LT.0.) QEIE=-QEIE                                            10005480
   15 RETURN                                                            10005490
   16 QEIN=QEIE+TERM1                                                   10005500
      IF(QEIE .EQ. QEIN) GO TO(2,15,13) , IB                            10005510
      QEIE=QEIN                                                         10005520
      H=H+C1                                                            10005530
      GO TO(1,10,12) , IB                                               10005540
      RETURN                                                            10005550
      END                                                               10005560
      BLOCK DATA                                                        10005570
      IMPLICIT REAL*8 (A-H,O-Z)                                         10005580
      COMMON /FCTRL/DFCT1(57)                                           10005590
      DATA DFCT1/                                                       10005600
     1        Z4110000000000000,  Z4110000000000000,  Z4120000000000000,10005610
     1        Z4160000000000000,  Z4218000000000000,  Z4278000000000000,10005620
     1        Z432D000000000000,  Z4413B00000000000,  Z449D800000000000,10005630
     2        Z4558980000000000,  Z46375F0000000000,  Z4726115000000000,10005640
     2        Z481C8CFC00000000,  Z4917328CC0000000,  Z4A144C3B28000000,10005650
     3        Z4B13077775800000,  Z4C13077775800000,  Z4D1437EEECD80000,10005660
     3        Z4E16BEECCA730000,  Z4F1B02B930689000,  Z5021C3677C82B400,10005670
     4        Z512C5077D36B8C40,  Z523CEEA4C2B3E0D8,  Z5357970CD7E29336,10005680
     4        Z5483629343D3DCD1,  Z55CD4A0619FB0906,  Z5714D9849EA37EEA,10005690
     5        Z58232F0FCBB3E62A,  Z593D925BA47AD2C9,  Z5A6F99461A1E9E0C,10005700
     5        Z5BD13F6370F96856,  Z5D1956AD0AAE33A2,  Z5E32AD5A155C6744,10005710
     6        Z5F688589CC0E94FC,  Z60DE1BC4D19EFC97,  Z621E5DCBE8A8BC88,10005720
     6        Z6344530ACB7BA832,  Z649E0008F68DF4F3,  Z661774015499125C,10005730
     7        Z67392AC33E351CC0,  Z688EEAE81B84C7E0,  Z6A16E39F2C684402,10005740
     7        Z6B3C1581D491B285,  Z6CA179CCEB478FC5,  Z6E1BC0EF38704CB5,10005750
     8        Z6F4E0EACCEBBD78D,  Z70E06A0E525C0C3F,  Z72293378A11EE63F,10005760
     8        Z737B9A69E35CB2BD,  Z7517A88E4484BE36,  Z7649EEBC961ED268,10005770
     9        Z77EBA8F91E823EAB,  Z792FDE529A3274BA,  Z7A9E90719EC722A8,10005780
     9        Z7C217277F77E014F,  Z7D72F97C62C1247F,  Z7F192693359A3FFB/10005790
      END                                                               10005800

      PROGRAM JC2                                                       20000010
C***********************************************************************20000020
C*        PROGRAM JC2                                                  *20000030
C*        TWO ELECTRON SYSTEM IN DIATOMIC MOLECULES                    *20000040
C*             SIGMA PLUS STATE                                        *20000050
C*             JAMES-COOLIDGE'S METHOD                                 *20000060
C*        AUXILIARY FUNCTION Z(NU,M,N,J,K,P)                           *20000070
C*             NU=0                                                    *20000080
C*             0.LE.N.LE.M.LE.16      FOR -1.LE.P.LE.4                 *20000090
C*             0.LE.N.LE.M.LE.14      FOR P=5 OR 6                     *20000100
C*             0.LE.K.LE.J.LE.16      FOR -1.LE.P.LE.4                 *20000110
C*             0.LE.K.LE.J.LE.14      FOR P=5 OR 6                     *20000120
C*        INPUT DATA                                                   *20000130
C*              NONE                                                   *20000140
C***********************************************************************20000150
      IMPLICIT REAL*8(A-H,Q-Z)                                          20000160
      IMPLICIT INTEGER(I-P)                                             20000170
      COMMON /ZD/Z(89155)                                               20000180
```

APPENDIX C LISTS OF PROGRAMS (9)

```
      DIMENSION Z0(720)
      III=720
C*****     INPUT FROM THE FILE 20        *********************************
      READ(20) Z0
      ALPHA=Z0(1)
      WRITE(6,1000) ALPHA
      J=1
      DO 20 I=1,89155
      J=J+1
      IF(J.LE.III) GO TO 10
      READ(20) Z0
      J=1
   10 Z(I)=Z0(J)
   20 CONTINUE
      Z0(1)=ALPHA
      I=1
      DO 90 PP2=1,8
      P=PP2-2
      GO TO (60,60,30,60,30,60,30,60),PP2
C*****     Z(0,M,N,J,K,P)        *****************************************
   30 II=0
      NUP1MX=3-P/2
      DO 50 PPP=1,2
      PP=P+PPP-1
      PPM2=PP-2
      DO 50 NUP1=1,NUP1MX
      NU=NUP1-1
      MP1MAX=20-P-NU*2
      DO 50 JP1=1,MP1MAX
      J=JP1-1
      DO 50 KP1=1,JP1
      K=KP1-1
      IF(((J+K).AND.1).NE.0) GO TO 50
      DO 40 MP1=1,MP1MAX
      M=MP1-1
      DO 40 NP1=1,MP1
      N=NP1-1
      II=II+1
      Z(II)=ZZ(NU,M+2,N,J,K,PPM2)+ZZ(NU,M,N+2,J,K,PPM2)
     2      +ZZ(NU,M,N,J+2,K,PPM2)+ZZ(NU,M,N,J,K+2,PPM2)
     3      -2.D0*(ZZ(NU,M,N,J,K,PPM2)+ZZ(NU,M+1,N+1,J+1,K+1,PPM2)
     4      +ZZ(NU+1,M,N,J,K,PPM2))
   40 CONTINUE
   50 CONTINUE
C*****     OUTPUT TO THE FILE 21       **********************************
   60 MP1MAX=17
      IF(P.GT.4) MP1MAX=15
      DO 90 JP1=1,MP1MAX
      J=JP1-1
      DO 90 KP1=1,JP1
      K=KP1-1
      IF(((J+K).AND.1).NE.0) GO TO 90
      DO 80 MP1=1,MP1MAX
      M=MP1-1
      DO 80 NP1=1,MP1
      N=NP1-1
      I=I+1
      IF(I.LE.III) GO TO 70
      WRITE(21) Z0
      I=1
   70 Z0(I)=ZZ(0,M,N,J,K,P)
   80 CONTINUE
   90 CONTINUE
      WRITE(21) Z0
      STOP
 1000 FORMAT(1H1,10X,'AUXILIARY FUNCTION Z(0,M,N,J,K,P)',
     2//16X,'ALPHA =',F10.5,/1H1)
      END
      REAL FUNCTION ZZ*8(NU,M,N,J,K,P)
      IMPLICIT REAL*8(A-H,Q-Z)
      IMPLICIT INTEGER(I-P)
      COMMON /ZD/Z(89155)
      DIMENSION L(4,8),LL(4)
      DATA L/1,27952,46952,59345,67025,0,82271,0,
     2       1,19001,31394,0,39074,58074,70467,0,
```

APPENDIX C LISTS OF PROGRAMS (10)

```
     3        1,12394,0,0,20074,32467,0,0,
     4        1,0,0,0,7681,0,0,0/
      DATA LL/231,190,153,120/
      ZZ=0.D0
      IF(M.GT.N) GO TO 10
      MM=N
      NN=M
      GO TO 20
   10 MM=M
      NN=N
   20 IF(J.GT.K) GO TO 30
      JJ=K
      KK=J
      GO TO 40
   30 JJ=J
      KK=K
   40 PP2=P+2
      NUP1=NU+1
      IF(P.NE.0) GO TO 50
      IF((NU.AND.1).NE.0) GO TO 70
      IF(((JJ.AND.1)+(KK.AND.1)).NE.0) GO TO 70
      I=(JJ*(JJ+2)/8+KK/2)*LL(NUP1)+MM*(MM+1)/2+NN+L(NUP1,PP2)
      GO TO 60
   50 IF(((J+K).AND.1).NE.0) GO TO 70
      I=((JJ*(JJ+2)+KK*2)/4)*LL(NUP1+(P+1)/2)+MM*(MM+1)/2+NN+L(NUP1,PP2)
   60 ZZ=Z(I)
   70 RETURN
      END
```

```
      PROGRAM JC3
C****************************************************************************
C*          PROGRAM JC3                                                     *
C*          TWO ELECTRON SYSTEM IN DIATOMIC MOLECULES                       *
C*                SIGMA PLUS STATE                                          *
C*                JAMES-COOLIDGE'S METHOD                                   *
C*          CALCULATION OF MATRIX ELEMENTS                                  *
C*                UNNORMALIZED BESES                                        *
C*          INPUT DATA                                                      *
C*                NUCLEAR CHARGE  ZA,ZB                                     *
C*                MULTIPLICITY                                              *
C*                NUMBER OF BASES                                           *
C*                BASE (M,N,J,K,P)                                          *
C*                      0.LE.M,N,J,K.LE.6                                   *
C*                      0.LE.P.LE.3                                         *
C****************************************************************************
      IMPLICIT REAL*8(A-H,Q-Z)
      IMPLICIT INTEGER(I-P)
      LOGICAL HMPLR,TRPLT
      DIMENSION MM(160),NN(160),JJ(160),KK(160),PP(160),FF(720)
      COMMON /Z00/ Z0(89718)
      COMMON /AZHT/ALPHA,Z1,Z2,HMPLR,TRPLT
      III=720
      NBSMAX=160
      MNMAX=6
      JKMAX=6
      PMAX=3
C*****    INPUT    **********************************************************
      READ(5,1000) ZA,ZB,MLTPLT,NBASES,
     2    (MM(I),NN(I),JJ(I),KK(I),PP(I),I=1,NBASES)
      Z1=(ZA+ZB)*0.5D0
      Z2=(ZA-ZB)*0.5D0
      HMPLR=Z2.EQ.0.D0
      TRPLT=MLTPLT.EQ.3
      IF(NBASES.GT.NBSMAX) GO TO 70
      DO 20 I=1,NBASES
      IF(MM(I)-NN(I)) 3,1,5
    1 IF(JJ(I)-KK(I)) 4,2,5
    2 IF(TRPLT) GO TO 70
      GO TO 5
    3 J=MM(I)
      MM(I)=NN(I)
      NN(I)=J
    4 J=JJ(I)
      JJ(I)=KK(I)
```

APPENDIX C LISTS OF PROGRAMS (11)

```
      KK(I)=J
    5 IF((NN(I).LT.0).OR.(MM(I).GT.MNMAX)) GO TO 70
      IF((KK(I).LT.0).OR.(JJ(I).GT.JKMAX)) GO TO 70
      IF((PP(I).LT.0).OR.(PP(I).GT.PMAX)) GO TO 70
      J=0
   10 J=J+1
      IF(J.EQ.I) GO TO 20
      IF((MM(I).EQ.MM(J)).AND.(NN(I).EQ.NN(J)).AND.(JJ(I).EQ.JJ(J))
     2     .AND.(KK(I).EQ.KK(J)).AND.(PP(I).EQ.PP(J))) GO TO 70
      GO TO 10
   20 CONTINUE
C*****     INPUT FROM THE FILE 21        ********************************
      READ(21) FF
      ALPHA=FF(1)
      WRITE(6,2000) ALPHA,ZA,ZB,MLTPLT,NBASES,
     2     (I,MM(I),NN(I),JJ(I),KK(I),PP(I),I=1,NBASES)
      J=1
      DO 40 I=1,89718
      J=J+1
      IF(J.LE.III) GO TO 30
      READ(21) FF
      J=1
   30 Z0(I)=FF(J)
   40 CONTINUE
C*****     CALCULATION OF MATRIX ELEMENTS AND OUTPUT TO THE FILE 22.      *
      WRITE(22) ALPHA,ZA,ZB,MLTPLT,NBASES,MM,NN,JJ,KK,PP
      L=-2
      DO 60 J=1,NBASES
      MB=MM(J)
      NB=NN(J)
      JB=JJ(J)
      KB=KK(J)
      PB=PP(J)
      DO 60 I=1,J
      MA=MM(I)
      NA=NN(I)
      JA=JJ(I)
      KA=KK(I)
      PA=PP(I)
      L=L+3
      IF(L.LE.III) GO TO 50
      WRITE(22) FF
      L=1
   50 FF(L  )=SSS(MA,NA,JA,KA,PA,MB,NB,JB,KB,PB)
      FF(L+1)=TTT(MA,NA,JA,KA,PA,MB,NB,JB,KB,PB)
      FF(L+2)=UUU(MA,NA,JA,KA,PA,MB,NB,JB,KB,PB)
   60 CONTINUE
      WRITE(22) FF
      GO TO 80
   70 WRITE(6,3000) NBASES,(MM(I),NN(I),JJ(I),KK(I),PP(I),I=1,NBASES)
   80 WRITE(6,4000)
      STOP
 1000 FORMAT(2D10.0,I5,/I5,/(5I5))
 2000 FORMAT(1H1,10X,'CALCULATION OF MATRIX ELEMENTS',
     2     //16X,'ALPHA =',F10.5,
     3     //16X,'THE CHARGE OF THE NUCLEUS A',F10.2,
     4     //16X,'THE CHARGE OF THE NUCLEUS B',F10.2,
     5     //16X,'MULTIPLICITY' I10,
     6     //16X,'NUMBER OF BASES',I10,
     7    ///21X,'NUMBER    M    N    J    K    P',/(21X,I5,1X,5I4))
 3000 FORMAT(1H0,10X,'ERROR IN BASES',/11X,I5,/(11X,5I5))
 4000 FORMAT(1H1)
      END
      REAL FUNCTION SSS*8(MA,NA,JA,KA,PA,MB,NB,JB,KB,PB)
      IMPLICIT REAL *8 (A-H,Q-Z)
      IMPLICIT INTEGER (I-P)
      LOGICAL HMPLR,TRPLT
      COMMON /AZHT/ALPHA,Z1,Z2,HMPLR,TRPLT
      SSS=SS(MA,NA,JA,KA,PA,NB,MB,KB,JB,PB)
      IF(TRPLT) SSS=-SSS
      SSS=SS(MA,NA,JA,KA,PA,MB,NB,JB,KB,PB)+SSS
      GO TO 10
      ENTRY TTT(MA,NA,JA,KA,PA,MB,NB,JB,KB,PB)
      SSS=TT(MA,NA,JA,KA,PA,NB,MB,KB,JB,PB)
      IF(TRPLT) SSS=-SSS
```

APPENDIX C LISTS OF PROGRAMS (12)

```
      SSS=TT(MA,NA,JA,KA,PA,MB,NB,JB,KB,PB)+SSS
      GO TO 10
      ENTRY UUU(MA,NA,JA,KA,PA,MB,NB,JB,KB,PB)
      SSS=U1(MA,NA,JA,KA,PA,NB,MB,KB,JB,PB)
      IF(TRPLT) SSS=-SSS
      SSS=Z1*(U1(MA,NA,JA,KA,PA,MB,NB,JB,KB,PB)+SSS)
      A=U3(MA,NA,JA,KA,PA,NB,MB,KB,JB,PB)
      IF(TRPLT) A=-A
      SSS=SSS+(U3(MA,NA,JA,KA,PA,MB,NB,JB,KB,PB)+A)
      IF(HMPLR) GO TO 10
      A=U2(MA,NA,JA,KA,PA,NB,MB,KB,JB,PB)
      IF(TRPLT) A=-A
      SSS=SSS+Z2*(U2(MA,NA,JA,KA,PA,MB,NB,JB,KB,PB)+A)
   10 RETURN
      END
      REAL FUNCTION SS*8(MA,NA,JA,KA,PA,MB,NB,JB,KB,PB)
      IMPLICIT REAL*8(A-H,O-Z)
      IMPLICIT INTEGER(I-P)
      LOGICAL HMPLR,TRPLT
      COMMON /AZHT/ALPHA,Z1,Z2,HMPLR,TRPLT
      I=1
      GO TO 10
      ENTRY TT(MA,NA,JA,KA,PA,MB,NB,JB,KB,PB)
      I=2
      GO TO 10
      ENTRY U1(MA,NA,JA,KA,PA,MB,NB,JB,KB,PB)
      I=3
      GO TO 10
      ENTRY U2(MA,NA,JA,KA,PA,MB,NB,JB,KB,PB)
      I=4
      GO TO 10
      ENTRY U3(MA,NA,JA,KA,PA,MB,NB,JB,KB,PB)
      I=5
   10 MAB=MA+MB
      NAB=NA+NB
      JAB=JA+JB
      KAB=KA+KB
      PAB=PA+PB
      MABP2=MAB+2
      NABP2=NAB+2
      JABP2=JAB+2
      KABP2=KAB+2
      IF(I.GT.1) GO TO 20
      SS=      Z(MABP2,NABP2,JAB,KAB,PAB)-Z(MABP2,NAB,JAB,KABP2,PAB)
     2        -Z(MAB,NABP2,JABP2,KAB,PAB)+Z(MAB,NAB,JABP2,KABP2,PAB)
      GO TO 70
   20 MABP1=MAB+1
      NABP1=NAB+1
      JABP1=JAB+1
      KABP1=KAB+1
      GO TO (70,30,40,50,60),I
   30 MABP4=MAB+4
      NABP4=NAB+4
      JABP4=JAB+4
      KABP4=KAB+4
      MABP3=MAB+3
      NABP3=NAB+3
      JABP3=JAB+3
      KABP3=KAB+3
      MABM1=MAB-1
      NABM1=NAB-1
      JABM1=JAB-1
      KABM1=KAB-1
      MABM2=MAB-2
      NABM2=NAB-2
      JABM2=JAB-2
      KABM2=KAB-2
      PABM2=PAB-2
      MMAB=MA-MB
      NNAB=NA-NB
      JJAB=JA-JB
      KKAB=KA-KB
      PPAB=PA-PB
      SS=-4.D0*ALPHA
     2       *(Z(MABP1,NABP2,JAB,KAB,PAB)-Z(MABP1,NAB,JAB,KABP2,PAB)
```

APPENDIX C LISTS OF PROGRAMS (13)

```
     3        +Z(MABP2,NABP1,JAB,KAB,PAB)-Z(MAB,NABP1,JABP2,KAB,PAB))
     4    +DFLOAT(MMAB*MMAB+MAB-JJAB*JJAB-JAB+PPAB*(MMAB-JJAB))
     5        *(Z(MAB,NABP2,JAB,KAB,PAB)-Z(MAB,NAB,JAB,KABP2,PAB))
     6    +DFLOAT(NNAB*NNAB+NAB-KKAB*KKAB-KAB+PPAB*(NNAB-KKAB))
     7        *(Z(MABP2,NAB,JAB,KAB,PAB)-Z(MAB,NAB,JABP2,KAB,PAB))
     I=MMAB*MMAB-MAB
      IF(I.NE.0) SS=SS-DFLOAT(I)
     2        *(Z(MABM2,NABP2,JAB,KAB,PAB)-Z(MABM2,NAB,JAB,KABP2,PAB))
     I=NNAB*NNAB-NAB
      IF(I.NE.0) SS=SS-DFLOAT(I)
     2        *(Z(MABP2,NABM2,JAB,KAB,PAB)-Z(MAB,NABM2,JABP2,KAB,PAB))
     I=JJAB*JJAB-JAB
      IF(I.NE.0) SS=SS+DFLOAT(I)
     2        *(Z(MAB,NABP2,JABM2,KAB,PAB)-Z(MAB,NAB,JABM2,KABP2,PAB))
     I=KKAB*KKAB-KAB
      IF(I.NE.0) SS=SS+DFLOAT(I)
     2        *(Z(MABP2,NAB,JAB,KABM2,PAB)-Z(MAB,NAB,JABP2,KABM2,PAB))
      I=2*(PPAB*PPAB+PAB)+PPAB*(MMAB+NNAB+JJAB+KKAB)
      IF(I.NE.0) SS=SS+DFLOAT(I)
     2 *(Z(MABP2,NABP2,JAB,KAB,PABM2)-Z(MABP2,NAB,JAB,KABP2,PABM2)
     3  -Z(MAB,NABP2,JABP2,KAB,PABM2)+Z(MAB,NAB,JABP2,KABP2,PABM2))
     I=PPAB*(MMAB-JJAB)
      IF(I.NE.0) SS=SS-DFLOAT(I)
     2       *(Z(MAB,NABP4,JAB,KAB,PABM2)-Z(MAB,NAB,JAB,KABP4,PABM2))
     I=PPAB*(NNAB-KKAB)
      IF(I.NE.0) SS=SS-DFLOAT(I)
     2       *(Z(MABP4,NAB,JAB,KAB,PABM2)-Z(MAB,NAB,JABP4,KAB,PABM2))
     I=2*PPAB*MMAB
      IF(I.NE.0) SS=SS+DFLOAT(I)
     2       *(Z(MABM1,NABP3,JABP1,KABP1,PABM2)
     3        -Z(MABM1,NABP1,JABP1,KABP3,PABM2))
     I=2*PPAB*NNAB
      IF(I.NE.0) SS=SS+DFLOAT(I)
     2       *(Z(MABP3,NABM1,JABP1,KABP1,PABM2)
     3        -Z(MABP1,NABM1,JABP3,KABP1,PABM2))
     I=2*PPAB*JJAB
      IF(I.NE.0) SS=SS-DFLOAT(I)
     2       *(Z(MABP1,NABP3,JABM1,KABP1,PABM2)
     3        -Z(MABP1,NABP1,JABM1,KABP3,PABM2))
     I=2*PPAB*KKAB
      IF(I.NE.0) SS=SS-DFLOAT(I)
     2       *(Z(MABP3,NABP1,JABP1,KABM1,PABM2)
     3        -Z(MABP1,NABP1,JABP3,KABM1,PABM2))
      SS=-SS*0.5D0
      GO TO 70
   40 SS=-4.D0
     2       *(Z(MABP1,NABP2,JAB,KAB,PAB)-Z(MABP1,NAB,JAB,KABP2,PAB)
     3        +Z(MABP2,NABP1,JAB,KAB,PAB)-Z(MAB,NABP1,JABP2,KAB,PAB))
      GO TO 70
   50 SS= 4.D0
     2       *(Z(MAB,NABP2,JABP1,KAB,PAB)-Z(MAB,NAB,JABP1,KABP2,PAB)
     3        +Z(MABP2,NAB,JAB,KABP1,PAB)-Z(MAB,NAB,JABP2,KABP1,PAB))
      GO TO 70
   60 PABM1=PAB-1
      SS=2.D0
     2       *(Z(MABP2,NABP2,JAB,KAB,PABM1)-Z(MABP2,NAB,JAB,KABP2,PABM1)
     3        -Z(MAB,NABP2,JABP2,KAB,PABM1)+Z(MAB,NAB,JABP2,KABP2,PABM1))
   70 RETURN
      END
      REAL FUNCTION Z*8(M,N,J,K,P)
      IMPLICIT REAL*8(A-H,Q-Z)
      IMPLICIT INTEGER(I-P)
      COMMON /Z0D/Z0(89718)
      DIMENSION L(8)
      DATA L/1,12394,24787,37180,49573,61966,74359,82039/
      IF(((J+K).AND.1).NE.0) GO TO 60
      IF(M.GT.N) GO TO 10
      MM=N
      NN=M
      GO TO 20
   10 MM=M
      NN=N
   20 IF(J.GT.K) GO TO 30
      JJ=K
      KK=J
```

APPENDIX C LISTS OF PROGRAMS (14)

```
      GO TO 40
   30 JJ=J
      KK=K
   40 LL=153
      IF(P.GT.4) LL=120
      I=((JJ*(JJ+2)+KK*2)/4)*LL+MM*(MM+1)/2+NN+L(P+2)
      Z=Z0(I)
   50 RETURN
   60 Z=0.D0
      GO TO 50
      END


      PROGRAM JC4
C**********************************************************************
C*        PROGRAM JC4                                                *
C*        TWO ELECTRON SYSTEM IN DIATOMIC MOLECULES                  *
C*             SIGMA PLUS STATE                                      *
C*             JAMES-COOLIDGE'S METHOD                               *
C*        NORMALIZATION AND REARRANGEMENT OF MATRIX ELEMENTS         *
C*        INPUT DATA                                                 *
C*             PRINT                                                 *
C*                    PRINT OUT WHEN PRINT.NE.0                      *
C**********************************************************************
      IMPLICIT REAL*8(A-H,Q-Z)
      IMPLICIT INTEGER(I-P)
      DIMENSION MM(160),NN(160),JJ(160),KK(160),PP(160),FN(160),
     2     S(12880),T(12880),U(12880),FF(720)
      III=720
C*****    INPUT       ************************************************
      READ(5,1000) PRINT
C*****    INPUT FROM THE FILE 22       ********************************
      READ(22) ALPHA,ZA,ZB,MLTPLT,NBASES,MM,NN,JJ,KK,PP
      WRITE(6,2000) ALPHA,ZA,ZB,MLTPLT,NBASES,
     2               (I,MM(I),NN(I),JJ(I),KK(I),PP(I),I=1,NBASES)
      NMX=NBASES*(NBASES+1)/2
      J=III
      DO 20 I=1,NMX
      J=J+3
      IF(J.LE.III) GO TO 10
      READ(22) FF
      J=1
   10 S(I)=FF(J)
      T(I)=FF(J+1)
      U(I)=FF(J+2)
   20 CONTINUE
C*****    NORMALIZATION     *******************************************
      DO 30 I=1,NBASES
      II=I*(I+1)/2
      FN(I)=1.D0/DSQRT(S(II))
   30 CONTINUE
      IJ=0
      DO 40 J=1,NBASES
      DO 40 I=1,J
      IJ=IJ+1
      ANF=FN(I)*FN(J)
      S(IJ)=S(IJ)*ANF
      T(IJ)=T(IJ)*ANF
      U(IJ)=U(IJ)*ANF
   40 CONTINUE
C*****    OUTPUT TO THE FILE 23       *********************************
      WRITE(23) ALPHA,ZA,ZB,MLTPLT,NBASES,MM,NN,JJ,KK,PP
      J=0
      DO 60 I=1,NMX
      J=J+1
      IF(J.LE.III) GO TO 50
      WRITE(23) FF
      J=1
   50 FF(J)=S(I)
   60 CONTINUE
      WRITE(23) FF
      J=0
      DO 80 I=1,NMX
      J=J+1
      IF(J.LE.III) GO TO 70
```

APPENDIX C LISTS OF PROGRAMS (15)

```
      WRITE(23) FF                                                      40000630
      J=1                                                               40000640
   70 FF(J)=T(I)                                                        40000650
   80 CONTINUE                                                          40000660
      WRITE(23) FF                                                      40000670
      J=0                                                               40000680
      DO 100 I=1,NMX                                                    40000690
      J=J+1                                                             40000700
      IF(J.LE.III) GO TO 90                                             40000710
      WRITE(23) FF                                                      40000720
      J=1                                                               40000730
   90 FF(J)=U(I)                                                        40000740
  100 CONTINUE                                                          40000750
      WRITE(23) FF                                                      40000760
      IF(PRINT.EQ.0) GO TO 120                                          40000770
      WRITE(6,3000)                                                     40000780
      IJ=0                                                              40000790
      DO 110 J=1,NBASES                                                 40000800
      DO 110 I=1,J                                                      40000810
      IJ=IJ+1                                                           40000820
      WRITE(6,4000) I,J,S(IJ),T(IJ),U(IJ)                               40000830
  110 CONTINUE                                                          40000840
      WRITE(6,5000)                                                     40000850
  120 STOP                                                              40000860
 1000 FORMAT(I5)                                                        40000870
 2000 FORMAT(1H1,10X,'NORMALIZATION AND REARRANGEMENT OF MATRIX ELEMENTS40000880
     2',//16X,'ALPHA =',F10.5,                                          40000890
     3      //16X,'THE CHARGE OF THE NUCLEUS A',F10.2,                  40000900
     4      //16X,'THE CHARGE OF THE NUCLEUS B',F10.2,                  40000910
     5      //16X,'MULTIPLICITY',I10,                                   40000920
     6      //16X,'NUMBER OF BASES',I10,                                40000930
     7    ///21X,'NUMBER    M    N    J    K    P',/(21X,I5,1X,5I4))     40000940
 3000 FORMAT(1H1,15X,'    I    J',11X,'S(I,J)',14X,'T(I,J)',14X,'U(I,J)') 40000950
 4000 FORMAT(16X,2I4,3F20.10)                                           40000960
 5000 FORMAT(1H1)                                                       40000970
      END                                                               40000980
```

```
      PROGRAM JC5                                                       50000010
C***********************************************************************50000020
C*        PROGRAM JC5                                                  *50000030
C*        TWO ELECTRON SYSTEM IN DIATOMIC MOLECULES                    *50000040
C*             SIGMA PLUS STATE                                        *50000050
C*             JAMES-COOLIDGE'S METHOD                                 *50000060
C*        SCHUMIDT'S ORTHONORMALIZATION                                *50000070
C*             AND TRANSFORMATION OF MATRIX ELEMENTS                   *50000080
C*        INPUT DATA                                                   *50000090
C*             NBS(I)                                                  *50000100
C*                  TAKE IN THE BASE I WHEN NBS(I)=1                   *50000110
C***********************************************************************50000120
      IMPLICIT REAL *16 (A-H,Q-Z)                                       50000130
      IMPLICIT INTEGER (I-P)                                            50000140
      DIMENSION MM(160),NN(160),JJ(160),KK(160),PP(160),NBS(160),       50000150
     2    A(12880),WW(160,160)                                          50000160
      REAL *8 S(12880),T(12880),U(12880),TA(12880),UA(12880),FF(720),   50000170
     2    ALPHA,ZA,ZB                                                   50000180
      EQUIVALENCE (S(1),T(1),U(1)),(TA(1),UA(1),FF(1))                  50000190
      III=720                                                           50000200
C*****    INPUT    *****************************************************50000210
      READ(5,1000) NBS                                                  50000220
C*****    INPUT FROM THE FILE 23      **********************************50000230
      READ(23) ALPHA,ZA,ZB,MLTPLT,NBASES,MM,NN,JJ,KK,PP                 50000240
      NBSOLD=NBASES                                                     50000250
      NMXOLD=NBASES*(NBASES+1)/2                                        50000260
      I=0                                                               50000270
      DO 10 J=1,NBSOLD                                                  50000280
      IF(NBS(J).LE.0) GO TO 10                                          50000290
      I=I+1                                                             50000300
      MM(I)=MM(J)                                                       50000310
      NN(I)=NN(J)                                                       50000320
      JJ(I)=JJ(J)                                                       50000330
      KK(I)=KK(J)                                                       50000340
      PP(I)=PP(J)                                                       50000350
   10 CONTINUE                                                          50000360
      NBASES=I                                                          50000370
```

APPENDIX C LISTS OF PROGRAMS (16)

```
      NMX=NBASES*(NBASES+1)/2
      WRITE(6,2000) ALPHA,ZA,ZB,MLTPLT,NBASES,
     2     (I,MM(I),NN(I),JJ(I),KK(I),PP(I),I=1,NBASES)
C*****     INPUT OF S      *********************************************
      J=III
      DO 30 I=1,NMXOLD
      J=J+1
      IF(J.LE.III) GO TO 20
      READ(23) FF
      J=1
   20 S(I)=FF(J)
   30 CONTINUE
      I=0
      J=0
      DO 40 L=1,NBSOLD
      DO 40 K=1,L
      J=J+1
      IF((NBS(K).LE.0).OR.(NBS(L).LE.0)) GO TO 40
      I=I+1
      S(I)=S(J)
   40 CONTINUE
C*****     SCHUMIDT'S ORTHONORMALIZATION       **************************
      DO 50 I=1,NMX
      A(I)=0.Q0
   50 CONTINUE
      A(1)=1.Q0
      DO 90 I=2,NBASES
      IIM1=I*(I-1)/2
      II=IIM1+I
      A(II)=1.Q0
      DO 70 J=1,I-1
      JJM1=J*(J-1)/2
      SS=0.Q0
      DO 60 K=1,J
      KI=IIM1+K
      KJ=JJM1+K
      SS=SS+A(KJ)*S(KI)
   60 CONTINUE
      DO 70 K=1,J
      KI=IIM1+K
      KJ=JJM1+K
      A(KI)=A(KI)-SS*A(KJ)
   70 CONTINUE
      SS=0.Q0
      KJ=0
      DO 80 J=1,I
      JI=IIM1+J
      DO 80 K=1,J
      KJ=KJ+1
      KI=IIM1+K
      W=A(KI)*A(JI)*S(KJ)
      IF(K.NE.J) W=W+W
      SS=SS+W
   80 CONTINUE
      W=1.Q0/QSQRT(SS)
      DO 90 J=1,I
      JI=IIM1+J
      A(JI)=A(JI)*W
   90 CONTINUE
C*****     OUTPUT TO THE FILE 24      *********************************
      WRITE(24) ALPHA,ZA,ZB,MLTPLT,NBASES,MM,NN,JJ,KK,PP
      J=0
      DO 110 I=1,NMX
      J=J+1
      IF(J.LE.III) GO TO 100
      WRITE(24) FF
      J=1
  100 FF(J)=A(I)
  110 CONTINUE
      WRITE(24) FF
C*****     INPUT OF T      *********************************************
      J=III
      DO 130 I=1,NMXOLD
      J=J+1
      IF(J.LE.III) GO TO 120
```

APPENDIX C   LISTS OF PROGRAMS   (17)

```
      READ(23) FF
      J=1
  120 T(I)=FF(J)
  130 CONTINUE
      I=0
      J=0
      DO 140 L=1,NBSOLD
      DO 140 K=1,L
      J=J+1
      IF((NBS(K).LE.0).OR.(NBS(L).LE.0)) GO TO 140
      I=I+1
      T(I)=T(J)
  140 CONTINUE
C*****    TRANSFORMATION OF T     **************************************
      DO 180 I=1,NBASES
      IIM1=I*(I-1)/2
      DO 180 J=1,NBASES
      W=0.Q0
      DO 170 K=1,I
      KI=IIM1+K
      IF(K.GT.J) GO TO 150
      KJ=J*(J-1)/2+K
      GO TO 160
  150 KJ=K*(K-1)/2+J
  160 W=W+A(KI)*T(KJ)
  170 CONTINUE
      WW(I,J)=W
  180 CONTINUE
      IJ=0
      DO 200 J=1,NBASES
      DO 200 I=1,J
      IIM1=I*(I-1)/2
      IJ=IJ+1
      W=0.Q0
      DO 190 K=1,I
      KI=IIM1+K
      W=W+A(KI)*WW(J,K)
  190 CONTINUE
      TA(IJ)=W
  200 CONTINUE
C*****    OUTPUT TO THE FILE 24     ************************************
      J=0
      DO 220 I=1,NMX
      J=J+1
      IF(J.LE.III) GO TO 210
      WRITE(24) FF
      J=1
  210 FF(J)=TA(I)
  220 CONTINUE
      WRITE(24) FF
C*****    INPUT OF U     ***********************************************
      J=III
      DO 240 I=1,NMXOLD
      J=J+1
      IF(J.LE.III) GO TO 230
      READ(23) FF
      J=1
  230 U(I)=FF(J)
  240 CONTINUE
      I=0
      J=0
      DO 250 L=1,NBSOLD
      DO 250 K=1,L
      J=J+1
      IF((NBS(K).LE.0).OR.(NBS(L).LE.0)) GO TO 250
      I=I+1
      U(I)=U(J)
  250 CONTINUE
C*****    TRANSFORMATION OF U     **************************************
      DO 290 I=1,NBASES
      IIM1=I*(I-1)/2
      DO 290 J=1,NBASES
      W=0.Q0
      DO 280 K=1,I
      KI=IIM1+K
```

APPENDIX C   LISTS OF PROGRAMS   (18)

```
      IF(K.GT.J) GO TO 260
      KJ=J*(J-1)/2+K
      GO TO 270
  260 KJ=K*(K-1)/2+J
  270 W=W+A(KI)*U(KJ)
  280 CONTINUE
      WW(I,J)=W
  290 CONTINUE
      IJ=0
      DO 310 J=1,NBASES
      DO 310 I=1,J
      IIM1=I*(I-1)/2
      IJ=IJ+1
      W=0.Q0
      DO 300 K=1,I
      KI=IIM1+K
      W=W+A(KI)*WW(J,K)
  300 CONTINUE
      UA(IJ)=W
  310 CONTINUE
C*****    OUTPUT TO THE FILE 24       **********************************
      J=0
      DO 330 I=1,NMX
      J=J+1
      IF(J.LE.III) GO TO 320
      WRITE(24) FF
      J=1
  320 FF(J)=UA(I)
  330 CONTINUE
      WRITE(24) FF
      WRITE(6,3000)
      STOP
 1000 FORMAT(80I1)
 2000 FORMAT(1H1,10X,'SCHUMIDT''S ORTHONORMALIZATION AND TRANSFORMATION
     2OF MATRIX ELEMENTS',//16X,'ALPHA =',F10.5,
     3    //16X,'THE CHARGE OF THE NUCLEUS A',F10.2,
     4    //16X,'THE CHARGE OF THE NUCLEUS B',F10.2,
     5    //16X,'MULTIPLICITY',I10,
     6    //16X,'NUMBER OF BASES',I10,
     7   ///21X,'NUMBER    M    N    J    K    P',/(21X,I5,1X,5I4))
 3000 FORMAT(1H1)
      END


      PROGRAM JC6
C************************************************************************
C*        PROGRAM JC6                                                   *
C*        TWO ELECTRON SYSTEM IN DIATOMIC MOLECULES                     *
C*             SIGMA PLUS STATE                                         *
C*             JAMES-COOLIDGE'S METHOD                                  *
C*        CALCULATION OF ENERGIES AND WAVE FUNCTIONS                    *
C*        INPUT DATA                                                    *
C*             PRINT                                                    *
C*                  PRINT OUT THE COEFFICIENTS OF THE ORTHONORMALIZED   *
C*                  BASES WHEN PRINT.NE.0                               *
C*             INTERNUCLEAR DISTANCE                                    *
C*             NUMBER OF THE STATE                                      *
C************************************************************************
      IMPLICIT REAL *16 (A-H,Q-Z)
      IMPLICIT INTEGER (I-P)
      REAL *8 A(12880),TA(12880),UA(12880),FF(720),
     2    ALPHA,ZA,ZB,R,RR
      DIMENSION MM(160),NN(160),JJ(160),KK(160),PP(160),X(160),Y(160),
     2    HH(160,160)
      III=720
C*****    INPUT      *******************************************************
      READ(5,1000) PRINT
C*****    INPUT FROM THE FILE 24       **********************************
      READ(24) ALPHA,ZA,ZB,MLTPLT,NBASES,MM,NN,JJ,KK,PP
      NMX=NBASES*(NBASES+1)/2
      J=III
      DO 20 I=1,NMX
      J=J+1
      IF(J.LE.III) GO TO 10
      READ(24) FF
```

APPENDIX C LISTS OF PROGRAMS (19)

```
      J=1
   10 A(I)=FF(J)
   20 CONTINUE
      J=III
      DO 40 I=1,NMX
      J=J+1
      IF(J.LE.III) GO TO 30
      READ(24) FF
      J=1
   30 TA(I)=FF(J)
   40 CONTINUE
      J=III
      DO 60 I=1,NMX
      J=J+1
      IF(J.LE.III) GO TO 50
      READ(24) FF
      J=1
   50 UA(I)=FF(J)
   60 CONTINUE
      WRITE(6,2000) ALPHA,ZA,ZB,MLTPLT,NBASES,
     2              (I,MM(I),NN(I),JJ(I),KK(I),PP(I),I=1,NBASES)
      IF(PRINT.EQ.0) GO TO 80
      WRITE(6,3000)
      L=0
      DO 70 I=1,NBASES
      K=L+1
      L=L+I
      WRITE(6,4000) I,(A(J),J=K,L)
   70 CONTINUE
C*****    CALCULATION OF TOTAL ENERGY      ****************************
   80 READ(5,5000,END=130) R,NSTATE
      RR=R*R
      K=0
      DO 90 J=1,NBASES
      DO 90 I=1,J
      K=K+1
      HH(I,J)=TA(K)/RR+UA(K)/R
      HH(J,I)=HH(I,J)
   90 CONTINUE
      CALL SQEQ(NBASES,HH,NSTATE,ENERGY)
      K=0
      DO 100 J=1,NBASES
      DO 100 I=1,J
      K=K+1
      HH(I,J)=TA(K)/RR+UA(K)/R
      HH(J,I)=HH(I,J)
  100 CONTINUE
      CALL VECTOR(NBASES,HH,ENERGY,X)
      TTLENR=ENERGY+ZA*ZB/R
      IF(PRINT.NE.0) WRITE(6,6000) (I,X(I),I=1,NBASES)
      DO 120 I=1,NBASES
      W=0.Q0
      DO 110 J=I,NBASES
      K=J*(J-1)/2+I
      W=W+X(J)*A(K)
  110 CONTINUE
      Y(I)=W
  120 CONTINUE
      WRITE(6,7000) ALPHA,ZA,ZB,MLTPLT,R,NSTATE,NBASES,TTLENR,
     2     (I,MM(I),NN(I),JJ(I),KK(I),PP(I),Y(I),I=1,NBASES)
      GO TO 80
  130 WRITE(6,8000)
      STOP
 1000 FORMAT(I5)
 2000 FORMAT(1H1,10X,'CALCULATION OF ENERGIES AND WAVE FUNCTIONS',
     2     //16X,'ALPHA =',F10.5,
     3     //16X,'THE CHARGE OF THE NUCLEUS A',F10.2,
     4     //16X,'THE CHARGE OF THE NUCLEUS B',F10.2,
     5     //16X,'MULTIPLICITY',I10,
     6     //16X,'NUMBER OF BASES',I10,
     7    ///21X,'NUMBER    M    N    J    K    P',/(21X,I5,1X,5I4))
 3000 FORMAT(1H1,10X,'ORTHONORMALIZED BASES BY SCHMITZ''S METHOD',
     2     //11X,'NUMBER',10X,'COEFFICIENTS')
 4000 FORMAT(//11X,I5,6X,5D20.10,/(22X,5D20.10))
 5000 FORMAT(D20.0,I5)
```

APPENDIX C LISTS OF PROGRAMS (20)

```
6000 FORMAT(1H1,10X,'THE WAVE FUNCTION',
    2     //11X,'ORTHONORMALIZED BASES BY SCHUMIDT''S METHOD',
    3     //11X,'NUMBER',10X,'COEFFICIENT',/(11X,I5,6X,D20.10))
7000 FORMAT(1H1,10X,'ALPHA =',F10.5,
    2     //11X,'THE CHARGE OF THE NUCLEUS A',F10.2,
    3     //11X,'THE CHARGE OF THE NUCLEUS B',F10.2,
    4     //11X,'MULTIPLICITY' I10,
    5     //11X,'INTERNUCLEAR DISTANCE',F10.5,
    6     //11X,'THE',I3,'TH STATE',
    7     //11X,'NUMBER OF BASES',I10,
    8     //11X,'TOTAL ENERGY',F15.10,
    9     ///11X,'THE WAVE FUNCTION',
    9     /11X,'NUMBER      M   N   J   K   P',10X,'COEFFICIENT',
    9     /(11X,I5,4X,5I3,5X,D20.10))
8000 FORMAT(1H1)
     END
     SUBROUTINE SQEQ(NN,A,NSTATE,EE)
     IMPLICIT REAL *16 (A-H,O-Z)
     DIMENSION A(160,160),P(160),U(160)
     COMMON /QSQD/AA(160),BB(160),E,DET,N,ND
     EQUIVALENCE (P(1),AA(1)),(U(1),BB(1))
     N=NN
     NM1=N-1
     NM2=N-2
     DO 90 K=1,NM2
     KP1=K+1
     DO 10 I=KP1,N
     U(I)=A(K,I)
  10 CONTINUE
     W=0.Q0
     DO 20 I=KP1,N
     W=W+U(I)**2
  20 CONTINUE
     S=QSQRT(W)
     IF(U(KP1).GT.0.D0) S=-S
     UUD2=W-U(KP1)*S
     U(KP1)=U(KP1)-S
     DO 50 I=KP1,N
     W=0.Q0
     DO 40 J=KP1,N
  40 W=W+A(I,J)*U(J)
     P(I)=W/UUD2
  50 CONTINUE
     W=0.Q0
     DO 60 I=KP1,N
     W=W+U(I)*P(I)
  60 CONTINUE
     W=W/(UUD2+UUD2)
     DO 70 I=KP1,N
     P(I)=P(I)-W*U(I)
  70 CONTINUE
     DO 80 J=KP1,N
     DO 80 I=KP1,J
     A(I,J)=A(I,J)-U(I)*P(J)-P(I)*U(J)
     A(J,I)=A(I,J)
  80 CONTINUE
     BB(KP1)=S
  90 CONTINUE
     DO 100 I=1,N
     AA(I)=A(I,I)
 100 CONTINUE
     BB(N)=A(N-1,N)
     EMIN=-(QABS(AA(1))+QABS(BB(2)))
     DO 110 I=2,NM1
     W=-(QABS(AA(I))+QABS(BB(I))+QABS(BB(I+1)))
     IF(EMIN.LT.W) GO TO 110
     EMIN=W
 110 CONTINUE
     W=-(QABS(AA(N))+QABS(BB(N)))
     IF(EMIN.GT.W) EMIN=W
     E=EMIN
     CALL SQSUB
     IF(DET.EQ.0.Q0) GO TO 170
     NDS=ND-NSTATE
     DELTA=QABS(EMIN)*0.3Q0
```

APPENDIX C LISTS OF PROGRAMS (21)

```
  120 E=E+DELTA
      CALL SQSUB
      IF(DET.EQ.0.Q0) GO TO 170
      IF(NDS-ND) 120,140,130
  130 E=E-DELTA
      DELTA=DELTA*0.5Q0
      GO TO 120
  140 ISND=ISN(DET)
  150 EE=E
  160 DELTA=DELTA*0.5Q0
      E=EE-DELTA
      IF(QABS(E-EE).LT.0.5Q-10) GO TO 170
      CALL SQSUB
      IF(ISN(DET)*ISND) 160,170,150
  170 EE=E
      RETURN
      END
      SUBROUTINE SQSUB
      IMPLICIT REAL *16 (A-H,O-Z)
      COMMON /QSQD/AA(160),BB(160),E,DET,N,ND
      ND=0
      IDFX=0
      D2=1.Q0
      DET=E-AA(1)
      ISND=ISN(DET)
      IF(ISND.LT.0) ND=1
      DO 30 K=2,N
      D1=D2
      D2=DET
      IF(ISND.NE.0) ISN2=ISND
      DET=(E-AA(K))*D2-BB(K)**2*D1
      ISND=ISN(DET)
      ADET=QABS(DET)
      IF(ADET.LT.1.Q10) GO TO 10
      D2=D2*1.Q-10
      DET=DET*1.Q-10
      IDFX=IDFX+10
      GO TO 20
   10 IF(ADET.GT.1.Q-10) GO TO 20
      D2=D2*1.Q10
      DET=DET*1.Q10
      IDFX=IDFX-10
   20 IF(ISND*ISN2.LT.0) ND=ND+1
   30 CONTINUE
      RETURN
      END
      FUNCTION ISN(X)
      IMPLICIT REAL *16 (A-H,O-Z)
      IF(X) 10,20,30
   10 ISN=-1
      GO TO 40
   20 ISN=0
      GO TO 40
   30 ISN=1
   40 RETURN
      END
      SUBROUTINE VECTOR(N,A,E,X)
      IMPLICIT REAL *16 (A-H,O-Z)
      DIMENSION A(160,160),X(160)
      NM1=N-1
      J=1
      A(1,1)=A(1,1)-E
      W=QABS(A(1,1))
      DO 10 I=2,N
      A(I,I)=A(I,I)-E
      Y=QABS(A(I,I))
      IF(Y.GE.W) GO TO 10
      J=I
      W=Y
   10 CONTINUE
      DO 20  I=1,N
      W=A(I,J)
      A(I,J)=A(I,N)
      A(I,N)=-W
   20 CONTINUE
```

APPENDIX C LISTS OF PROGRAMS (22)

```
      CALL CPLNEQ(NM1,N,A)
      DO 30 I=1,NM1
      X(I)=A(I,N)
   30 CONTINUE
      X(N)=X(J)
      X(J)=1.Q0
      W=0.Q0
      DO 40 I=1,N
      W=W+X(I)**2
   40 CONTINUE
      W=1.Q0/QSQRT(W)
      DO 50 I=1,N
      X(I)=X(I)*W
   50 CONTINUE
      RETURN
      END
      SUBROUTINE CPLNEQ(M,N,A)
      DIMENSION A(160,160)
      IMPLICIT REAL *16 (A-H,O-Z)
      DO 60 K=1,M
      L=0
      W=0.Q0
      DO 10 I=K,M
      Y=QABS(A(I,K))
      IF(Y.LE.W) GO TO 10
      L=I
      W=Y
   10 CONTINUE
      IF(L.EQ.K) GO TO 30
      DO 20 J=K,N
      W=A(K,J)
      A(K,J)=A(L,J)
      A(L,J)=W
   20 CONTINUE
   30 W=1.Q0/A(K,K)
      DO 40 J=K,N
      A(K,J)=A(K,J)*W
   40 CONTINUE
      DO 60 I=1,M
      IF(I.EQ.K) GO TO 60
      W=A(I,K)
      DO 50 J=K,N
      A(I,J)=A(I,J)-A(K,J)*W
   50 CONTINUE
   60 CONTINUE
      RETURN
      END

      PROGRAM JC7
C*********************************************************************
C*       PROGRAM JC7                                                 *
C*       TWO ELECTRON SYSTEM IN DIATOMIC MOLECULES                   *
C*            SIGMA PLUS STATE                                       *
C*            JAMES-COOLIDGE'S METHOD                                *
C*       SCHUMIDT'S ORTHONORMALIZATION AND PRINTING OUT              *
C*            THE MATRIX ELEMENTS                                    *
C*       INPUT DATA                                                  *
C*            PRINT                                                  *
C*                 PRINT OUT THE MATRIX ELEMENTS WHEN P.NE.0         *
C*            INTERNUCLEAR DISTANCE                                  *
C*********************************************************************
      IMPLICIT REAL *16 (A-H,Q-Z)
      IMPLICIT INTEGER(I-P)
      DIMENSION MM(160),NN(160),JJ(160),KK(160),PP(160),
     2    A(12880),WW(160,160)
      REAL *8 S(12880),T(12880),U(12880),H(12880),HA(12880),FF(720),
     2    ALPHA,ZA,ZB,R
      EQUIVALENCE (S(1),H(1)),(T(1),HA(1)),(FF(1),A(1))
      III=720
      SERR=1.Q-6
C*****   INPUT      **********************************************
      READ(5,1000) PRINT,R
C*****   INPUT FROM THE FILE 23    ********************************
      READ(23) ALPHA,ZA,ZB,MLTPLT,NBASES,MM,NN,JJ,KK,PP
```

```
      NMX=NBASES*(NBASES+1)/2
      J=III
      DO 20 I=1,NMX
      J=J+1
      IF(J.LE.III) GO TO 10
      READ(23) FF
      J=1
   10 S(I)=FF(J)
   20 CONTINUE
      J=III
      DO 40 I=1,NMX
      J=J+1
      IF(J.LE.III) GO TO 30
      READ(23) FF
      J=1
   30 T(I)=FF(J)
   40 CONTINUE
      J=III
      DO 60 I=1,NMX
      J=J+1
      IF(J.LE.III) GO TO 50
      READ(23) FF
      J=1
   50 U(I)=FF(J)
   60 CONTINUE
C*****    SCHUMIDT'S ORTHONORMALIZATION       ***************************
      WRITE(6,2000)
      DO 70 I=1,NMX
      A(I)=0.Q0
   70 CONTINUE
      A(1)=1.Q0
      DO 130 I=2,NBASES
      IIM1=I*(I-1)/2
      II=IIM1+I
      A(II)=1.Q0
      SSS=1.Q0
      DO 90 J=1,I-1
      JJM1=J*(J-1)/2
      SS=0.Q0
      DO 80 K=1,J
      KI=IIM1+K
      KJ=JJM1+K
      SS=SS+A(KJ)*S(KI)
   80 CONTINUE
      SSS=SSS-SS*SS
      DO 90 K=1,J
      KI=IIM1+K
      KJ=JJM1+K
      A(KI)=A(KI)-SS*A(KJ)
   90 CONTINUE
      SS=0.Q0
      KJ=0
      DO 100 J=1,I
      JI=IIM1+J
      DO 100 K=1,J
      KJ=KJ+1
      KI=IIM1+K
      W=A(KI)*A(JI)*S(KJ)
      IF(K.NE.J) W=W+W
      SS=SS+W
  100 CONTINUE
      W=QABS(SS/SSS-1.Q0)
      WRITE(6,3000) I,MM(I),NN(I),JJ(I),KK(I),PP(I),SS,SSS,W
      IF((SS.LT.0.D0).OR.(W.GT.SERR)) GO TO 110
      W=1.Q0/QSQRT(SS)
      GO TO 120
  110 WRITE(6,4000)
      W=0.Q0
  120 DO 130 J=1,I
      JI=IIM1+J
      A(JI)=A(JI)*W
  130 CONTINUE
      IF(PRINT.EQ.0) GO TO 230
C*****    TRNSFORMATION OF MATRIX ELEMENTS       ************************
      RR=R*R
```

APPENDIX C LISTS OF PROGRAMS (24)

```
      DO 140 I=1,NMX
      H(I)=T(I)/RR+U(I)/R
  140 CONTINUE
      DO 180 I=1,NBASES
      IIM1=I*(I-1)/2
      DO 180 J=1,NBASES
      W=0.Q0
      DO 170 K=1,I
      KI=IIM1+K
      IF(K.GT.J) GO TO 150
      KJ=J*(J-1)/2+K
      GO TO 160
  150 KJ=K*(K-1)/2+J
  160 W=W+A(KI)*H(KJ)
  170 CONTINUE
      WW(I,J)=W
  180 CONTINUE
      IJ=0
      DO 200 J=1,NBASES
      DO 200 I=1,J
      IIM1=I*(I-1)/2
      IJ=IJ+1
      W=0.Q0
      DO 190 K=1,I
      KI=IIM1+K
      W=W+A(KI)*WW(J,K)
  190 CONTINUE
      HA(IJ)=W
  200 CONTINUE
      WRITE(6,5000) ALPHA,ZA,ZB,MLTPLT,R,NBASES,
     2     (I,MM(I),NN(I),JJ(I),KK(I),PP(I),I=1,NBASES)
      WRITE(6,6000)
      L=0
      DO 210 I=1,NBASES
      K=L+1
      L=L+I
      WRITE(6,7000) I,(A(J),J=K,L)
  210 CONTINUE
      WRITE(6,8000)
      L=0
      K=0
      DO 220 J=1,NBASES
      DO 220 I=1,J
      K=K+1
      L=L+1
      MM(L)=I
      NN(L)=J
      S(L)=HA(K)
      IF(L.LT.5) GO TO 220
      WRITE(6,9000) (MM(L),NN(L),S(L),L=1,5)
      L=0
  220 CONTINUE
      IF(L.NE.0) WRITE(6,9000) (MM(I),NN(I),S(I),I=1,L)
      WRITE(6,10000)
  230 STOP
 1000 FORMAT(I5,/D20.0)
 2000 FORMAT(1H1,10X,'CHECK OF SCHUMIDT''S ORTHONORMALIZATION',
     2     //11X,'BASE',5X,'   M    N    J    K    P',16X,'(1/N)**2')
 3000 FORMAT(11X,I4,5X,5I4,10X,2D20.10,D20.1)
 4000 FORMAT(15X,'OMIT')
 5000 FORMAT(1H1,15X,'ALPHA =',F10.5,
     2     //16X,'THE CHARGE OF THE NUCLEUS A',F10.2,
     3     //16X,'THE CHARGE OF THE NUCLEUS B',F10.2,
     4     //16X,'MULTIPLICITY' I10,
     5     //16X,'INTERNUCLEAR DISTANCE',F10.5,
     6     //16X,'NUMBER OF BASES',I10,
     7    ///21X,'NUMBER    M    N    J    K    P',/(21X,I5,1X,5I4))
 6000 FORMAT(1H1,///11X,'ORTHONORMALIZED BASES',
     2        //11X,'NUMBER',10X,'COEFFICIENTS')
 7000 FORMAT(/11X,I5,5X,5D20.10,/(21X,5D20.10))
 8000 FORMAT(1H1,///11X,'MATRIX ELEMENTS OF HAMILTONIAN OPERATOR',
     2          //3X,5(3X,'   I    J       H(I,J)',3X))
 9000 FORMAT(3X,5(3X,2I4,F15.8))
10000 FORMAT(1H1)
      END
```