

Electronic Supplementary Information, Part B, to:

ACTIVITY COEFFICIENTS IN MIXED ELECTROLYTE SOLUTIONS

by Francesco Malatesta,^{*†} Chiara Fagiolini[†] and Roberto Franceschi[†].

* To whom correspondence should be addressed: E-mail franco@dccl.unipi.it (Francesco Malatesta). [†] Dipartimento di Chimica e Chimica Industriale dell'Università di Pisa, Via Risorgimento 35, 56236 Pisa - Italy

The IPBE algorithm – How to use the enclosed computation program.

The original program was written in FORTRAN IV ca. 20 years ago, thus it is quite rigid and not friendly. However, it works reasonably well, and despite the intrinsic limits of the model and statistical treatment that it is based on, it is still one of the most effective treatments capable of predicting the thermodynamic behavior of the different kinds of electrolytes (including high-charge electrolytes and some kinds of mixtures of electrolytes, see later) using –as its only parameter– the average distance of closest approach (*a*) of the ions to one another. Results for 2:2, as well as for 2:3, 3:2 and 3:3 salts since also the latter have finally become available, proved to be surprisingly faithful.^{Sb1, Sb2, Sb3} A previous version of the algorithm was used to study the kinetic salt effect in the reactions of ions of even sign,^{Sb4} and demonstrated once and for all the electrostatic nature of the Olson-Simonson salt effects. The relevant algorithm has always been made freely available to users on request. It is now provided once and for all as a part of the present supplementary information. We provide here some general remarks and the instructions for use. More details can be found in the original paper that described the algorithm.^{Sb5}

IPBE develops the same physical model as the Debye-Hückel theory (restricted primitive model),^{Sb6} following the same imperfect (and not completely consistent) thermodynamic approach known as the Poisson-Boltzmann equation (PBE). However, while the Debye-Hückel theory finds the solution of the approximate form of PBE which only retains the linear term of the series expansion of the Boltzmann exponential term, IPBE solves the complete equation by numerical methods (a mathematically improved version of Guggenheim's approach to numerical integration of PBE).^{Sb7} The differences between the Debye-Hückel theory and IPBE become very evident when dealing with high charge system (with which the linear approximation no longer holds and can even fail to predict the correct sign of the second derivative of $\ln \gamma_{\pm}$ at the limit of infinite dilution).

The input of data occurs via the keyboard in the form of a dialogue with a self-explaining menu selection. The possible options are:

(1) theoretical activity coefficients for single ions and the corresponding mean activity coefficient (as natural logarithms); (2) osmotic coefficients ϕ (as $1 - \phi$); (3) partial and apparent molar volume (for the part that exceeds the infinite dilution limiting value) in $\text{cm}^3 \text{mol}^{-1}$; (4) partial and apparent relative molar enthalpies (kJ mol^{-1}). Depending on the selected option, the program asks for:

- (i) the dielectric constant of the solvent, D (plus the derivative $D^{-1}\partial D/\partial p$ and the compressibility coefficient of the solvent $-V^{-1}\partial V/\partial p$, in bar^{-1} units, or $D^{-1}\partial D/\partial T$ and the thermal expansion coefficient of the solvent $V^{-1}\partial V/\partial T$, in K^{-1} units, if the volume or enthalpy options are selected). The values that the program suggests for water at $T = 298.15 \text{ K}$ are from B. B. Owen, R. C. Miller, C. E. Milner, H. L. Cogan, *J. Phys. Chem.*, 1961, **65**, 2065 (D , $D^{-1}\partial D/\partial p$, $D^{-1}\partial D/\partial T$); G. S. Kell, E. Whalley, *Phyl. Trans. A*, 1956, **258**, 565 ($-V^{-1}\partial V/\partial p$); F. T. Gucker, *J. Amer. Chem. Soc.*, 1934, **56**, 1017 ($V^{-1}\partial V/\partial T$).
- (ii) the distance of closest approach selected, a (in ångström units)
- (iii) the type of solute, i.e., the number of kinds of distinguishable cations and of anions (in two different steps, separated by more detailed questions concerning the multiplicity and charge of the ions, see item iv). In the hypothetical case of an electrolyte $\text{Na}_2\text{KLaCa}_0[\text{Co}(\text{CN})_6]_2(\text{SO}_4)_0$, the numbers to be answered are 3 for the cations and 2 for the anions, since the electrolyte contains three distinguishable kinds of cations ($M^+ = \text{Na}^+$ and K^+ together, undistinguishable;^{Sb8} Ca^{2+} , a fictitious component whose concentration has been set at zero; and La^{3+}) and two distinguishable anions, SO_4^{2-} (a fictitious component set at null concentration), and $[\text{Co}(\text{CN})_6]^{3-}$. Null species introduced in the formula of the electrolyte, like fictitious Ca_0 and $(\text{SO}_4)_0$ in the example, are a trick that forces the program to extend the calculation to the corresponding ions too, although the corresponding concentrations have been set at zero. (The situation recalls a trace ion in a supporting electrolyte.)

^{Sb1} F. Malatesta, S. Trombella, A. Giacomelli, M. Onor, *Polyhedron*, 2000, **19**, 2493.

^{Sb2} F. Malatesta, F. Bruni and N. Fanelli, *Phys. Chem. Chem. Phys.*, 2002, **4**, 121

^{Sb3} F. Malatesta and R. Zamboni, *J. Solution Chem.*, 1997, **26**, 791; F. Malatesta, L. Carbonaro, N. Fanelli, S. Ferrini, A. Giacomelli, *J. Solution Chem.*, 1999, **28**, 593.

^{Sb4} R. Zamboni, A. Giacomelli, F. Malatesta, A. Indelli, *J. Phys. Chem.*, 1976, **80**, 1418.

^{Sb5} F. Malatesta, T. Rotunno, *Gazz. Chim. Ital.*, 1983, **113**, 783.

^{Sb6} The ion-ion interactions – which are controlled by the ion-ion distances – make the natural variable of IPBE (like DHT and other ion interaction theories) the molar and not molal concentration. Substitution of molal concentrations for molar concentrations results in slight differences in the results, which, however, are nullified by the presence of the empiric parameter a . The same applies to the kind of activity coefficient that IPBE calculates, a rational scale (rather than molal scale) activity coefficient and – to all appearances – referring to MacMillan-Mayer's, rather than Gibbs' reference system; the empiric parameter, and model roughness, make such distinctions unimportant.

^{Sb7} E. A. Guggenheim, *Trans. Faraday Soc.*, 1960, **56**, 1152; *ibid.*, 1962, **58**, 86.

^{Sb8} As IPBE (like DHT) does not consistently admit different values of a for the different ions of a solution, all ions having the same charge are automatically treated as indistinguishable.

- (iv) For as many kinds of ions as declared in the previous step, the program asks for the stoichiometric coefficient (i.e., how many ions of that kind have to appear in the real or fictitious formula of the electrolyte), and the charge. In the example of $\text{Na}_2\text{KLaCa}_0[\text{Co}(\text{CN})_6]_2(\text{SO}_4)_0$, the answers are, for the first cation (Na^+ and K^+ together): 3, 1 ; second cation (La^{3+}): 1, 3; third cation (Ca^{2+}): 0, 2. The same questions, when asked for anions, are to be answered: 2, -3 (or 2, 3) for $[\text{Co}(\text{CN})_6]^{3-}$; 0, -2 (or 0, 2) for SO_4^{2-} . As a matter of fact, the program is capable of recognizing indistinguishable ions that have wrongly been declared as distinguishable, and automatically makes corrections. The formulae that do not correspond to electrically neutral systems are rejected, and requested again. Provided the electrical neutrality be preserved, the formulae that contain non integer instead of integer numbers of ions $n_1, n_2 \dots, n_i \dots$ are freely allowed, and can be used to simulate a mixture of different ionic species, whose different n_i are proportional to the respective ionic fractions $x_i = n_i / \sum_j n_j$. E.g., a mixed solution of 0.1 m Na_2SO_4 and 0.001 m $\text{La}_2(\text{SO}_4)_3$ can be treated as a 0.1 m solution of the fictitious salt $\text{Na}_{2.00}\text{La}_{0.02}(\text{SO}_4)_{1.03}$. When starting calculations, the program keeps the different n_i constant and varies the overall concentration, as if a mixture with a corresponding ionic composition were progressively diluted by adding a solvent. In such a case, the mean activity coefficients that the program still calculates lack significance, and the mean activity coefficients of the component electrolytes $-\text{Na}_2\text{SO}_4$ and $\text{La}_2(\text{SO}_4)_3$ in the example– are to be obtained by combining by hand the output values listed for single ions.
- (v) A selection is possible among a series of possible options of input units – molar concentrations or ionic strength or the corresponding square roots. (A further selection, not necessarily the same, is subsequently requested, between items vi and vii, for the output units.) Further, the program asks for the highest concentration (or ionic strength, square root of concentration..., as appropriate) at which calculations are to be performed (for activity coefficients, the program also asks for the lowest concentration limit; by entering the digit 0 one selects the default option).
- (vi) The computation chain automatically creates a natural set of concentrations to be treated. However, the user is allowed to introduce additional concentrations at which he wishes the calculations to be performed. The program asks for the number of these additional, user-defined concentrations, and their values, if any (enter the digit 0 for none; otherwise, enter the relevant number and their values, in decreasing order).
- (vii) The program then starts calculating, and results are written into a file (ipbout) which can be opened using e.g. Microsoft Word or any other file editors.

```

PROGRAM IPBE
C*****
C*****
C
C THIS PROGRAM CALCULATES SINGLE AND MEAN ACTIVITY COEFFICIENTS, OSMOTIC
C COEFFICIENTS, PARTIAL AND APPARENT MOLAR VOLUMES, PARTIAL AND APPARENT
C RELATIVE MOLAR ENTHALPIES, OF ELECTROLYTES, BY A NUMERICAL INTEGRATION
C OF THE POISSON-BOLTZMANN COMPLETE EQUATION IN THE ORIGINAL EXPONENTIAL
C FORM. THE ALGORITHMS WERE DESCRIBED IN THE PAPER
C
C "F.MALATESTA, T.ROTUNNO - GAZZ.CHIM.ITAL.,113(1983),783-787"
C
C CONCERNED WITH A PREVIOUS VERSION OF THE IPBE PROGRAM, IPBECF.
C IPBE DIFFERS FROM IPBECF IN THE LATTICE OF THE "X" AND "A" VALUES
C -SEE THE CITED PAPER FOR THE MEANING OF "X" AND "A"- GIVING IMPROVED
C PRECISION AND SPEEDINESS.
C
C
C LAST REVISION: MAY 20, 2003
C
C FRANCESCO MALATESTA ... DIPARTIMENTO DI CHIMICA E CHIMICA INDUSTRIALE
C UNIVERSITA' DI PISA
C VIA RISORGIMENTO 35 - 56126 PISA - ITALY
C EMAIL franco@dcci.unipi.it
C
C*****
C*****
C
C LOGIC UNIT TABLE :
C UNIT 5 = CONSOL INPUT (:CI:)
C UNIT 6 = CONSOL OUTPUT (:CO:)
C UNIT 7 = FILE IPBOUT
C
C*****
C*****
C
C IMPLICIT REAL*8 (A-H,O-Z)
C REAL*8 NZ(10),LNAPT,LNMACT(2,1000),NZZ(10),LAMBDA,L2
C CHARACTER*2 HEAD(10),NUMAB(50),ERLAB
C CHARACTER*13 TITLE(4)
C LOGICAL CNTDIO(10),CONTR1(10,1000),ERSIGN
C DIMENSION AZ(10),STIFR(50),PT(2),UTIL(2),ADEB(2),ANG1(2),PSI(2),
C 1S(2),XS(50),X(1000),SQRIF(1000),CILGAC(10,1000),ALAMB(1000),
C 2FDIF(1000),ELOGAC(1000),IIF(50),IID(50)
C COMMON NZZ,RNI(10),NTION
C DATA HEAD/10*' '/,NUMAB/'ST','ND','RD',47*'TH'//,TITLE/

```

```

1' ION STRENGTH',' SQRT(I.STR.)',' CONCENTRAT.',' SQRT(CONCN.)'/
2,ERSIGN/.FALSE./,STIMIN/0.D0/,XMIN/1.D-03/,FACDX/0.05D0/
C
C*****
C
C THE ASSIGNMENT "/,FACDX/0.05D0/" CAUSES ANY X TO BE FIVE PER CENT
C LOWER THAN PREVIOUS X. (SUBSTITUTE FOR 0.05D0 A NEW VALUE TO CHANGE
C THE DESCENT RATE: E.G., 0.025D0 MAKES X TO LOWER BY 2.5 INSTEAD OF
C 5 PER CENT ANY TIMES). THE RULE APPLIES TO ANY X>XMIN; FOR XMIN>X>0
C THE DESCENT STEP BECOME CONSTANT.
C.....
C THE ASSIGNMENT "/,XMIN/1.D-03/" STATES THE LOWEST IONIC STRENGTH THAT
C THE PROGRAM SELECTS TO CHANGE THE X-DESCENT RATE. IF EXTREME DILUTION
C LEVELS NEED BE TREATED WITH IMPROVED PRECISION, THE STATEMENT CAN BE
C CHANGED, E.G. SUBSTITUTING /1.D-04/ FOR /1.D-03/.
C.....
C WARNING: DO NOT SELECT "FACDX" TOO SMALL, THE NUMBER OF THE X VALUES
C RISKS TO EXCEED THE DIMENSIONS (1000) ASSIGNED TO VECTOR X(I) AND
C TO OTHER VARIABLES (SQRIF,..., ETC.), WITH FATAL EFFECTS.
C
C*****
C
C .....
C SELECTS OPTIONS AND READS INPUT PARAMETERS.
C
      OPEN(7,FILE='IPBOUT')
      WRITE(6,1000)
      READ(5,*)IFVE
1     WRITE(6,1010)
      READ(5,*)TO
      IF(TO.GT.0.D0)GO TO 2
      WRITE(6,1015)
      GO TO 1
2     WRITE(6,1020)
      READ(5,*)DIEL0
      IF(DIEL0.GT.0.D0)GO TO 3
      WRITE(6,1015)
      GO TO 2
3     WRITE(6,1030)
      READ(5,*)ANGS
      IF(ANGS.GT.0.D0)GO TO 4
      WRITE(6,1015)
      GO TO 3
4     P0=1.
5     VALNI=0.
      WW=0.
      ELNTR=0.
C
C .....
C READS IN SALT COMPOSITION.
6     WRITE(6,1040)
      READ(5,*)NC
      IF(NC.GT.0)GO TO 7
      WRITE(6,1015)
      GO TO 6
7     IF(NC.LE.9)GO TO 8
      WRITE(6,1066)
      READ(5,*)INTRP
      IF(INTRP.LE.1)GO TO 999
      GO TO 5
8     WRITE(6,1050)
      DO 10 I=1,NC
      WRITE(6,1055)I,NUMAB(I)
      READ(5,*)AZ(I),NZ(I)
      IF(AZ(I).GE.0.D0.AND.NZ(I).NE.0.D0)GO TO 9
      WRITE(6,1056)
      GO TO 8
9     VALNI=VALNI+AZ(I)
      ELNTR=ELNTR +AZ(I)*NZ(I)
10    WW=WW+AZ(I)*NZ(I)*NZ(I)/2.
11    WRITE(6,1060)
      READ(5,*)NA
      IF(NA.GT.0)GO TO 12
      WRITE(6,1015)
      GO TO 11
12    NTION=NC+NA
      IF(NTION.LE.10) GO TO 13
      WRITE(6,1066)

```

```

      READ(5,*)INTRP
      IF(INTRP.LE.1)GO TO 999
      GO TO 5
13  WRITE(6,1065)
      NC1=NC+1
      DO 20 I= NC1,NTION
      II=I-NC
14  WRITE(6,1070)II,NUMAB(II)
      READ(5,*)AZ(I),NZ(I)
      IF(AZ(I).GE.0.D0.AND.NZ(I).NE.0.D0)GO TO 15
      WRITE(6,1056)
      GO TO 14
15  IF(DSIGN(1.D0,NZ(I)).LT.0.)GO TO 17
      NZ(I)=-NZ(I)
17  VALNI=VALNI+AZ(I)
      ELNTR=ELNTR+AZ(I)*NZ(I)
20  WW=WW+AZ(I)*NZ(I)*NZ(I)/2.
      IF(DABS(ELNTR).LE.1.D-08)GO TO 30
      WRITE(6,1075)
      GO TO 5
30  IF(WW.GT.0.D0)GO TO 35
      WRITE(6,1076)
      GO TO 5
35  IF(IFVE-3) 70,50,60
C
C .....
C READS IN PARAMETERS FOR CALCULATING PARTIAL AND APPARENT MOLAR
C VOLUMES.
50 R=83.14475D0
      PT0=P0
      NP=2
      WRITE(6,1080)
      READ(5,*)LNAPT
      WRITE(6,1090)
      READ(5,*)DERD0
      WRITE(6,1100)
      READ(5,*)ALBE
      ALBE= -ALBE
C (VARPT=PRESSURE INTERVAL FOR THE NUMERICAL DERIVATIVE.)
      VARPT=0.01
      GO TO 100
C
C .....
C READS IN PARAMETERS FOR CALCULATING PARTIAL AND APPARENT
C MOLAR ENTHALPIES.
60 R =1.9872D0
      PT0=T0
      NP=2
      WRITE(6,1120)
      READ(5,*)LNAPT
      WRITE(6,1130)
      READ(5,*)DERD0
      WRITE(6,1135)
      READ(5,*)ALBE
C (VARPT=TEMPERATURE INTERVAL FOR THE NUMERICAL DERIVATIVE.)
      VARPT=0.01
      GO TO 100
C
C .....
C SETS PARAMETERS FOR CALCULATING ACTIVITY COEFFICIENTS
C AND OSMOTIC COEFFICIENTS.
70 NP=1
      VARPT=0.
      PT0=0.
      DERD0=0.
      LNAPT=0.
      ALBE=0.
C
C .....
C READS IN THE STATED IONIC STRENGTHS (SQRT OF, CONC.N.S,...) IN
C DECREASING ORDER
100 WRITE(6,1140)
      WRITE(6,1145)
      READ(5,*)IRADI
      IF(IRADI.LT.1.OR.IRADI.GT.4)THEN
      WRITE(6,1250)
      GO TO 100
      END IF

```

```

101 WRITE(6,1150)
    READ(5,*)STIFR(1)
    IF (IFVE.EQ.1) THEN
        WRITE(6,1151)
        READ(5,*)STIMIN
        GO TO(80,81,82,83),IRADI
81 STIMIN=STIMIN**2
    GO TO 80
82 STIMIN=STIMIN*WW
    GO TO 80
83 STIMIN=WW*STIMIN**2
80 CONTINUE
    END IF
    IF(STIFR(1).GT.0.D0)GO TO 102
    WRITE(6,1015)
    GO TO 101
102 NIF=1
    IID(1)=0
103 WRITE(6,1170)
    READ(5,*)NIF1
    IF(NIF1.LT.50)GO TO 104
    WRITE(6,1171)NIF1
    GO TO 103
104 IF(NIF1.LT.1)GO TO 108
    WRITE(6,1175)
    NIF=NIF1+1
    DO 107 I=2,NIF
105 WRITE(6,1180)I,NUMAB(I),TITLE(IRADI)
    READ(5,*)STIFR(I)
    IF(STIFR(I).LT.STIFR(I-1))GO TO 106
    WRITE(6,1185)STIFR(I),STIFR(I-1)
    GO TO 105
106 IF(STIFR(I).GT.0.D0)GO TO 107
    WRITE(6,1015)
    GO TO 105
107 IID(I)=0
108 DO 900 I=1,NIF
    GO TO(90,91,92,93),IRADI
91 STIFR(I)=STIFR(I)**2
    GO TO 90
92 STIFR(I)=STIFR(I)*WW
    GO TO 90
93 STIFR(I)=WW*STIFR(I)**2
90 IF(STIMIN.GT.STIFR(I))STIMIN=STIFR(I)
900 CONTINUE
94 WRITE(6,1200)
    WRITE(6,1145)
    READ(5,*)IRADI
    IF(IRADI.LT.1.OR.IRADI.GT.4)THEN
        WRITE(6,1250)
        GO TO 94
    END IF
    WRITE(6,1190)
    READ(5,*)IOOP
    WRITE(7,2000)
    WRITE(7,1500)T0,DIEL0,ANGS
    IF(IFVE-3)115,110,113
110 ALB=-ALBE
    WRITE(7,1510)LNAPT,DERD0,ALB,VARPT
    GO TO 115
113 WRITE(7,1520)LNAPT,DERD0,ALBE,VARPT
115 ADEBA=4.194942855D06/(T0*DIEL0)**1.5
    PSIA=0.329037348D0*ANGS*DSQRT(2.33632213D04/(T0*DIEL0))
    SAA= 1.671033458D05/(T0*DIEL0)
    DO 150 L=1,NP
    PT(L)=PT0+FLOAT(L/2-1/L)*VARPT
    IF (IFVE-3) 130,130,120
120 UTIL(L)=T0/PT(L)*DEXP(DERD0*(T0-PT(L)))
    GO TO 140
130 UTIL(L)=DEXP(DERD0*(P0-PT(L)))
140 ADEB(L)=ADEBA*UTIL(L)**1.5
    ANGL(L)=DEXP(LNAPT*(PT(L)-PT0))
    PSI(L)=PSIA*DSQRT(UTIL(L))*ANGL(L)
150 S(L)=SAA*UTIL(L)
C
C .....
C SETS UP THE VECTOR X(I), SEE REFERENCE PAPER, AND INSERTS
C THE STATED IONIC STRENGTH (SQRT OF, CONC., ...) IN THE RIGTH ORDER

```

```

C OF X(I)
DO 160 I=1,NIF
160 XS(I)=PSI(1)*DSQRT(STIFR(I))*DEXP(ALBE*(PT0-PT(1))/2.)
    XSMIN=PSI(1)*DSQRT(STIMIN)*DEXP(ALBE*(PT0-PT(1))/2.)
    ESLPE=1.
    IK=0
    I=1
    X(I)=10.
    DO 170 L=1,NIF
    IF(X(1).GT.XS(L)) GO TO 170
    XS(L)=X(1)
    IK=IK+1
    IID(L)=L
    IIF(L)=I
170 CONTINUE
180 I=I+1
    IF(X(I-1).GE.XMIN)DX=FACDX*X(I-1)
    X(I)=X(I-1)-DX
    IF(X(I).LE.0.D0.OR.X(I).LT.XSMIN)GO TO 210
    XTEMP=X(I)
    DO 200 L=1,NIF
    IF(IK.GT.0.AND.IID(L).EQ.L) GO TO 200
    IF(XTEMP.GT.XS(L)) GO TO 200
    IF(XTEMP.EQ.XS(L)) GO TO 190
    X(I)=XS(L)
    IK=IK+1
    IID(L)=L
    IIF(L)=I
    I=I+1
    GO TO 200
190 IK=IK+1
    IID(L)=L
    IIF(L)=I
200 CONTINUE
    X(I)=XTEMP
    GO TO 180
210 IF(IK.EQ.NIF) GO TO 230
    DO 220 L=1,NIF
    IF(IID(L).EQ.L) GO TO 220
    X(I)=XS(L)
    IK=IK+1
    IIF(L)=I
    I=I+1
220 CONTINUE
230 III=I-1
    DO 235 I=1,NTION
235 CNTDIO(I)=.FALSE.
    NTION1=NTION-1
    DO 240 I=1,NTION1
    IF(CNTDIO(I)) GO TO 241
    K1=I+1
C
C .....
C CHECKS FOR SALT SIMMETRY AND BUILDS UP THE EQUIVALENT SALT
C COMPOSITION I.E. NAKSO4 IS TREATED LIKE M2SO4 .
DO 242 K=K1,NTION
    IF(NZ(I).NE.NZ(K)) GO TO 243
    AZ(I)=AZ(I)+AZ(K)
    CNTDIO(K)=.TRUE.
243 CONTINUE
242 CONTINUE
241 CONTINUE
240 CONTINUE
    IAN=0
    JK=0
    DO 250 I=1,NTION
    IF(CNTDIO(I)) GO TO 251
    JK=JK+1
    NZ(JK)=NZ(I)
    AZ(JK)=AZ(I)
    IF(I.LE.NC) GO TO 251
    IAN=IAN+1
251 CONTINUE
250 CONTINUE
    NTION=JK
    NA=IAN
    NC=NTION-IAN
    JK=0

```

```

      IF(NC.NE.NA) GO TO 270
      NC1=NC+1
      DO 260 I=1,NC
      DO 260 J=NC1,NTION
      IF(DABS(NZ(I)+NZ(J)).GT.0.999D0) GO TO 260
      IF(DABS(AZ(I)-AZ(J)).GT.0.999D0) GO TO 260
      JK=JK+1
260 CONTINUE
270 ISIM=1
      IF(JK.EQ.NC) ISIM=2
      DO 280 I=1,NTION
280 RNI(I)=AZ(I)/(WW*2.)
      WRITE(7,1530)
      DO 285 I=1,NTION
285 WRITE(7,1540) I,NUMAB(I),NZ(I),AZ(I)
      NISIM= NTION/ISIM
C
C .....
C CALCULATES THE NATURAL LOGARITHMS OF ACTIVITY COEFFICIENTS FOR
C CENTRAL IONS AND LN'S OF MEAN ACTIVITY COEFFICIENTS.
      DO 555 KT=1,NP
      DO 330 I=1,III
      LNMACT(KT,I)=0.
330 SQRIF(I)=X(I)/PSI(KT)
      ANGPT=ANGS*ANG1(KT)
      DO 444 ION=1,NISIM
      ZSIGN=DSIGN(1.D0,NZ(ION))
      DO 333 L=1,NTION
333 NZZ(L)=ZSIGN*NZ(L)
      DO 340 I=1,III
      ALAMB(I)=0.
      FIDIF(I)=0.
      CONTR1(ION,I)=.FALSE.
      ELOGAC(I)=0.
      IF(IFVE-3) 335,340,340
335 ELOGAC(I)=-ADEB(KT)*NZZ(ION)*NZZ(ION)*SQRIF(I)/(1.+X(I))
340 CONTINUE
      AINCST=0.
      DELT0A=1.D-06
350 DELTAA=DELT0A+AINCST*1.0D-02
      AINCST=AINCST+DELTAA
      I=1
      Y=AINCST*DEXP(-X(I))
      Y1=-Y
      CALL DERIV(X(I),Y,Y1,Y2,Y3,Y4)
      IF(IIF(1).GT.1) GO TO 370
      FI=Y/X(I)
      LAMBDA=(FI-Y1)*ANGPT/(NZZ(ION)*S(KT))
      IF(LAMBDA.GT.1.D0) GO TO 360
      DEFI=FI-LAMBDA*NZZ(ION)*S(KT)/(ANGPT*(1.+X(I)))
      ELOGAC(I)=ELOGAC(I)+NZZ(ION)/2.*(FIDIF(I)+DEFI)*(LAMBDA-ALAMB(I))
      ALAMB(I)=LAMBDA
      FIDIF(I)=DEFI
      GO TO 370
360 CILGAC(ION,I)=ELOGAC(I)+NZZ(ION)*FIDIF(I)*(1.-ALAMB(I))
      CONTR1(ION,I)=.TRUE.
370 DO 400 I=2,III
      IF(CONTR1(ION,I)) GO TO 410
      H=X(I)-X(I-1)
      Y=((Y4*H/4.+Y3)*H/3.+Y2)*H/2.+Y1)*H+Y
      Y1=((Y4*H/3.+Y3)*H/2.+Y2)*H+Y1
      CALL DERIV(X(I),Y,Y1,Y2,Y3,Y4)
      IF(I.LT.IIF(1)) GO TO 400
      FI=Y/X(I)
      LAMBDA=(FI-Y1)*ANGPT/(NZZ(ION)*S(KT))
      IF(LAMBDA-1.) 391,391,390
390 CILGAC(ION,I)=ELOGAC(I)+NZZ(ION)*FIDIF(I)*(1.-ALAMB(I))
      CONTR1(ION,I)=.TRUE.
      GO TO 400
391 DEFI=FI-LAMBDA*NZZ(ION)*S(KT)/(ANGPT*(1.+X(I)))
      ELOGAC(I)=ELOGAC(I)+NZZ(ION)/2.*(FIDIF(I)+DEFI)*(LAMBDA-ALAMB(I))
      ALAMB(I)=LAMBDA
      FIDIF(I)=DEFI
400 CONTINUE
410 IF(I.GT.IIF(1)) GO TO 350
444 CONTINUE
420 IIF1=IIF(1)
      DO 430 I=IIF1,III

```

```

DO 430 ION=1,NISIM
LNMACT(KT,I)=LNMACT(KT,I)+CILGAC(ION,I)*AZ(ION)/VALNI*ISIM
430 CONTINUE
IF(KT.EQ.NP) GO TO 555
KT1=KT+1
COREL=(PSI(KT1)/PSI(KT))*DEXP(ALBE*(PT(KT)-PT(KT1))/2.)
DO 460 I=1,III
460 X(I)=X(I)*COREL
555 CONTINUE
C
C .....
C WRITES OUT THE CALCULATED LN'S OF ACTIVITY COEFFICIENTS.
WRITE(7,2000)
IF(IFVE.GT.2) GO TO 470
IF(IFVE.GT.1) GO TO 465
IIV=NISIM/7
IF(IIV.LT.1) GO TO 660
DO 654 K=1,IIV
IF=7*K
I1=(K-1)*7+1
WRITE(7,1580)TITLE(IRADI),(HEAD(I),I=1,IF)
WRITE(7,1585)(I,NUMAB(I),I=1,IF)
IF(ISIM.GT.1) GO TO 610
WRITE(7,1590)(HEAD(I),NZ(I),I=I1,IF)
GO TO 620
610 WRITE(7,1600)(HEAD(I),NZ(I),I=I1,IF)
620 IF(IOOP.NE.2) GO TO 640
DO 630 I=1,NIF
ASSX=FUNZ(STIFR(I),IRADI,WW)
630 WRITE(7,1620)ASSX,(CILGAC(ION,IIF(I)),ION=I1,IF)
GO TO 653
640 KI=1
KKI=IIF(1)
DO 652 I=KKI,III
STRGT=SQRIF(I)**2
ASSX=FUNZ(STRGT,IRADI,WW)
IF(I.EQ.IIF(KI))GO TO 645
WRITE(7,1620)ASSX,(CILGAC(ION,I),ION=I1,IF)
GO TO 651
645 WRITE(7,1625)ASSX,(CILGAC(ION,I),ION=I1,IF)
IF(KI.GE.NIF) GO TO 650
KI=KI+1
650 CONTINUE
651 CONTINUE
652 CONTINUE
653 CONTINUE
654 CONTINUE
WRITE(7,2000)
IF(IF.EQ.NISIM) GO TO 740
I1=IF+1
GO TO 670
660 I1=1
670 WRITE(7,1550)TITLE(IRADI),(HEAD(I),I=I1,NISIM)
WRITE(7,1555)(I,NUMAB(I),I=I1,NISIM)
IF(ISIM.GT.1)GO TO 680
WRITE(7,1560)(HEAD(I),NZ(I),I=I1,NISIM)
GO TO 690
680 WRITE(7,1570)(HEAD(I),NZ(I),I=I1,NISIM)
690 IF(IOOP.NE.2) GO TO 710
DO 700 I=1,NIF
ASSX=FUNZ(STIFR(I),IRADI,WW)
700 WRITE(7,1610)ASSX,LNMACT(1,IIF(I)),(CILGAC(ION,IIF(I)),ION=I1,
1,NISIM)
GO TO 999
710 KI=1
KKI=IIF(1)
DO 730 I=KKI,III
STRGT=SQRIF(I)**2
ASSX=FUNZ(STRGT,IRADI,WW)
IF(I.EQ.IIF(KI)) GO TO 720
WRITE(7,1610)ASSX,LNMACT(1,I),(CILGAC(ION,I),ION=I1,NISIM)
GO TO 730
720 WRITE(7,1615)ASSX,LNMACT(1,I),(CILGAC(ION,I),ION=I1,NISIM)
IF(KI.GE.NIF) GO TO 730
KI=KI+1
730 CONTINUE
GO TO 999
740 WRITE(7,1660)TITLE(IRADI)

```



```

        IF (IOOP.NE.2) GO TO 760
        DO 750 I=1,NIF
        ASSX=FUNZ (STIFR (I) , IRADI , WW)
750 WRITE (7,1670) ASSX, LNMACT (1, IIF (I))
        GO TO 999
760 KI=1
        KKI=IIF (1)
        DO 780 I=KKI, III
        STRGT=SQRIF (I) **2
        ASSX=FUNZ (STRGT, IRADI, WW)
        IF (I.EQ.IIF (KI)) GO TO 770
        WRITE (7,1670) ASSX, LNMACT (1, I)
        GO TO 780
770 WRITE (7,1675) ASSX, LNMACT (1, I)
        IF (KI.GE.NIF) GO TO 780
        KI=KI+1
780 CONTINUE
        GO TO 999
465 J=III
        KJ=NIF
        ORD=0.
        ABSC=0.
        OSM0=0.
        ERLAB=' '
        WRITE (7,1680) TITLE (IRADI)
472 IF (J.LT.IIF (1)) GO TO 999
C
C .....
C CALCULATES THE OSMOTIC COEFFICIENTS
        OSM0=(ABSC-LNMACT (1, J)) * (X (J) **2+ORD) /2.+OSM0
        OSMCF1=OSM0/X (J) **2
        STRGT=SQRIF (J) **2
        ASSX=FUNZ (STRGT, IRADI, WW)
        IF (IOOP.NE.2) GO TO 475
C WRITES OUT THE CALCULATED OSMOTIC COEFFICIENTS.
        IF (J.NE.IIF (KJ)) GO TO 478
        ASSX=FUNZ (STIFR (KJ) , IRADI, WW)
        WRITE (7,1640) ASSX, LNMACT (1, J) , ERLAB, OSMCF1
        KJ=KJ-1
        GO TO 478
475 IF (J.EQ.IIF (KJ)) GO TO 477
        WRITE (7,1640) ASSX, LNMACT (1, J) , ERLAB, OSMCF1
        GO TO 478
477 WRITE (7,1645) ASSX, LNMACT (1, J) , ERLAB, OSMCF1
        KJ=KJ-1
478 ABSC=LNMACT (1, J)
        ORD=X (J) **2
        J=J-1
        GO TO 472
470 RIFAS=DEXP (ALBE* (PT (NP) -PT0) /2.)
        EL2=0.
        ECS=0.
        I=III
        SUM=0.
        KI=NIF
        IF (IFVE.GT.3) GO TO 480
        WRITE (7,1630) TITLE (IRADI)
        GO TO 500
480 WRITE (7,1650) TITLE (IRADI)
500 IF (I.LT.IIF (1)) GO TO 999
        SQRIF (I)=SQRIF (I) *RIFAS
        STRGT=SQRIF (I) **2
        L2=(LNMACT (NP, I) -LNMACT (1, I)) / (PT (NP) -PT (1)) *VALNI*T0*R
        ERLAB=' '
        IF (I.EQ.IIF (1)) GO TO 505
        IDECR=I-1
        EL3=(LNMACT (NP, IDECR) -LNMACT (1, IDECR)) / (PT (NP) -PT (1)) *
1VALNI*T0*R
        CLMAX=(EL2+EL3) /2.D0+ABS (EL2-EL3)
        CLMIN=(EL2+EL3) /2.D0-ABS (EL2-EL3)
        IF (L2.GT.CLMAX.OR.L2.LT.CLMIN) THEN
        ERSIGN=.TRUE.
        ERLAB=' W'
        L2=EL2
        END IF
505 XX=PSIA*SQRIF (I)
        SIGMA=(3./XX**3) * (1.+XX-1./ (1.+XX) -2.*DLOG (1.+XX))
        IF (IFVE. GT.3) GO TO 510

```

```

C
C .....
C CALCULATES PARTIAL MOLAR VOLUMES
  PMVH=L2+WW*R*T0*ADEBA*SQRIF(I)/(1.+XX)*(3.*DERD0+ALBE-XX/
  1(1.+XX)*(DERD0+ALBE-2.*LNAPT))
  GO TO 520
510 L2=-L2*T0
C
C .....
C CALCULATES PARTIAL MOLAR ENTHALPIES.
  PMVH=L2-WW*R*T0**2*ADEBA*SQRIF(I)/(1.+XX)*(3.*(DERD0+1./T0)
  1+ALBE-XX/(1.+XX)*(DERD0+1./T0+ALBE-2.*LNAPT))
520 SUM=SUM+(EL2*ECS+L2*SQRIF(I))*(SQRIF(I)-ECS)
  FIL=SUM/STRGT
  IF(IFVE.GT.3)GO TO 530
C
C .....
C CALCULATES APPARENT MOLAR VOLUMES.
  AMVH=FIL+2.*R*T0*ADEBA*WW*SQRIF(I)*(ALBE*SIGMA/3.+DERD0/(1.+XX)
  1-2.*LNAPT/XX**3*(1.+2.*XX-XX**2/2.-1./(1.+XX)-3.*DLOG(1.+XX)))
  GO TO 540
C
C .....
C CALCULATES APPARENT MOLAR ENTHALPIES.
530 AMVH=FIL-2.*R*T0**2*ADEBA*WW*SQRIF(I)*(ALBE*SIGMA/3.+(DERD0+1./T0)
  1)/(1.+XX)-2.*LNAPT/XX**3*(1.+2.*XX-XX**2/2.-1./(1.+XX)-3.*DLOG(1.
  2+XX))
C
C .....
C WRITES OUT THE CALCULATED VOLUMES OR ENTHALPIES ACCORDING TO THE
C SELECTED OPTIONS.
C .....
540 IF(IOOP.NE.2) GO TO 550
  IF(I.NE.IIF(KI)) GO TO 570
  ASSX=FUNZ(STIFR(KI),IRADI,WW)
  WRITE(7,1640)ASSX,PMVH,ERLAB,AMVH
  KI=KI-1
  GO TO 570
550 IF(I.EQ.IIF(KI)) GO TO 560
  ASSX=FUNZ(STRGT,IRADI,WW)
  WRITE(7,1640)ASSX,PMVH,ERLAB,AMVH
  GO TO 570
560 ASSX=FUNZ(STRGT,IRADI,WW)
  WRITE(7,1645)ASSX,PMVH,ERLAB,AMVH
  KI=KI-1
570 EL2=L2
  ECS=SQRIF(I)
  I=I-1
  GO TO 500
999 IF(ERSIGN)WRITE(7,1685)
  STOP
C
C .....
C .....
C .....
1000 FORMAT(' DO YOU WANT:  1) NATURAL LN'S OF ACTIVITY COEFFICIENTS?
  1',15X,' 2) OSMOTIC COEFFICIENTS (1-FI)'/,15X,' 3) PARTIAL AND APP
  2ARENT MOLAR VOLUMES?'/,15X,' 4) PARTIAL AND APPARENT MOLAR ENTHALP
  3IES?'/,' (TYPE 1 OR 2 OR 3 OR 4) : ')
1010 FORMAT(' REFERENCE TEMPERATURE (T0) IN KELVIN? : ')
1015 FORMAT(' **** ERROR. ZERO AND NEGATIVE NUMBERS NOT ALLOWED./
  1' VALUE IGNORED. RE-ENTER.')
1020 FORMAT(' DIELECTRIC CONSTANT (E.G., 78.358 FOR WATER AT 298.15 K)
  1? : ')
1030 FORMAT(' CLOSEST DISTANCE OF IONIC APPROACH (IN ANGSTROM)? : ')
1040 FORMAT(' SALT (SALT MIXTURE) COMPOSITION/' HOW MANY TYPES OF CATI
  1ONS? : ')
1050 FORMAT(' STOICHIOMETRIC COEFFICIENT (=HOW MANY MOLES PER MOLE OF'/
  1' SALT, OR PER DEFINED MOLE OF MIXTURE) AND ELECTRIC CHARGE FOR:')
1055 FORMAT(' THE',I2,A2,' CATION? : ')
1056 FORMAT(' **** NEGATIVE STOICHIOM.COEFF.S AND/OR UNCHARGED IONS NO
  1T'/' ALLOWED. BOTH DATA DELETED. RE-ENTER.')
1060 FORMAT(' HOW MANY TYPES OF ANIONS ? : ')
1065 FORMAT(' STOICHIOMETRIC COEFFICIENT AND ELECTRIC CHARGE FOR:')
1066 FORMAT(' **** NOT ALLOWED. GLOBAL NUMBER OF ION TYPES EXCEEDS'/
  1' THE DIMENSION ASSIGNED TO THE ARRAYS NZ,NZZ,CNTDIO,CONTR1,'/
  2' AZ,CILGAC,,RNI,HEAD. YOU CAN :/' 1) EXIT (TYPE 1)'
  3 /' 2) PERFORM CALCULATIONS FOR A DIFFERENT SALT SOLUTION, TO BE

```

```

4/'/' DEFINED AGAIN (TYPE 2). '/' 1 OR 2 ? : ')
1070 FORMAT(' THE',I2,A2,' ANION? : ')
1075 FORMAT(' ****ERROR. NOT ELECTRONEUTRAL MIXTURE.'
1/' COMPOSITION IGNORED. RE-ENTER.')
1076 FORMAT(' **** WRONG COMPOSITION. '/'
1/' ELECTRONEUTRALITY WAS FULFILLED, BUT ALL STOICHIOM. '/'
2' COEFF.S ARE ZERO (=NO ION OF ANY TYPE IN THE ELECTROLYTE). '/'
3/' COMPOSITION IGNORED. RE-ENTER.')
1080 FORMAT(' DERIVATIVE OF LN OF THE DISTANCE OF CLOSEST APPROACH
1/'/' RESPECT TO PRESSURE (IN 1/BAR)? (USUAL ANSWER, 0.0) : ')
1090 FORMAT(' DERIVATIVE OF LN OF DIELECTRIC CONSTANT RESPECT TO PRESSU
1RE '/' (IN 1/BAR) (E.G. 47.10D-06 FOR WATER AT 298.15 K)? : ')
1100 FORMAT(' COMPRESSIBILITY COEFFICIENT OF THE SOLVENT (IN 1/BAR) '/'
1 (E.G., 45.24D-06 FOR WATER AT 298.15 K)? : ')
1120 FORMAT(' DERIVATIVE OF LN OF THE DISTANCE OF CLOSEST APPROACH
1/'/' RESPECT TO TEMPERATURE ? (USUAL ANSWER, 0.0) : ')
1130 FORMAT(' DERIVATIVE OF LN OF DIELECTRIC CONSTANT RESPECT TO TEMP',
1'ERATURE '/' (E.G. -45.88D-04 FOR WATER AT 298.15 K)? : ')
1135 FORMAT(' THERMAL EXPANSION COEFFICIENT OF THE SOLVENT (IN 1/KE',
1'LVIN) '/' (E.G., 2.55D-04 FOR WATER AT 298.15 K)? : ')
1140 FORMAT('/' DO YOU WANT TO INPUT : ')
1145 FORMAT(' 1) IONIC STRENGTHS ? '/' 2) SQUARE ROOTS OF IONIC STRENG
1THS? '/' 3) CONCENTRATIONS ? '/' 4) SQUARE ROOTS OF CONCENTRATIONS
2 ? '/' TYPE THE CORRESPONDING NUMBER: ')
1150 FORMAT(' HIGHEST IONIC STRENGTH (SQRT OF, CONC.N., ...) TO WHICH
1/'/' YOU WISH THE COMPUTATIONS ? : ')
1151 FORMAT(' LOWEST IONIC STRENGTH (SQRT OF, CONC.N., ...) TO WHICH
1/'/' YOU WISH THE COMPUTATIONS ? (TYPE 0 TO ATTAIN THE EXTREME '/'
2 DILUTION LIMIT OF THE AUTOMATIC PROCEDURE) : ')
1170 FORMAT(' HOW MANY USER-DEFINED IONIC STRENGTHS (SQRT OF, CONC.N.,
1...) '/' ARE TO BE ADDED TO THOSE OF THE AUTOMATIC PROCEDURE ?'
2/' (TYPE 0 FOR NONE. DO NOT EXCEED 49) : ')
1171 FORMAT(' **** ERROR.',I5,' EXCEEDS THE DIMENSION ASSIGNED TO '/'
1' THE ARRAYS STIFR,XS, NUMAB,IIF,IID. VALUE IGNORED. RE-ENTER.')
1175 FORMAT(' TYPE IN DECREASING CONCENTRATION ORDER:')
1180 FORMAT(I3,A2,A13,' ? : ')
1185 FORMAT(' **** ERROR.',1PD11.3,' GREATER THAN',1D11.3, /
1' NOT ALLOWED VALUE IGNORED. RE-ENTER.')
1190 FORMAT('/' THE OUTPUT WILL BE SEND TO FILE: <IPBOUT> '/' DO YOU WANT
1 THE RESULTS PRINTOUT AT '/',10X,'1) ALL IONIC STRENGTHS (SQRT OF,..
2.ETC.) ?' /,10X,'2) ONLY THE IONIC STRENGTHS (SQRT OF,..
3..ETC.) YOU ADDED ?' / (TYPE 1 OR 2) : ')
1200 FORMAT('/' DO YOU WANT THE RESULTS PRINTOUT VS : '/')
1250 FORMAT(' **** NUMBERS LOWER THAN 1 OR GREATER THAN 4 NOT ALLOWED.
1/'/' RE-ENTER. '/')
1500 FORMAT(' T0=',3PD11.2,' KELVIN'/' DIELECTRIC'
1,' CONSTANT=',2PD13.4,/' CLOSEST DISTANCE OF IONIC APPROACH (A) ='
2,1PD13.4,' ANGSTROM')
1510 FORMAT(' DERIVATIVE OF LN(A) RESP.TO P',1PD13.5,2X,'1/BAR'/' DERIV
1ATIVE OF LN(DIEL.CONST.) RESP. TO P=',D13.5,2X,'1/BAR'/' COMPRESSI
2BILITY COEF.OF THE SOLVENT=',D13.5,2X,'1/BAR'/' DELTA P=',D13.5)
1520 FORMAT(' DERIVATIVE OF LN(A) RESP.TO T',1PD13.5,2X,'1/KELVIN'
1/' DERIVATIVE OF LN(DIEL. CONST.) RESP. TO T=',D13.5,2X,'1/KEL',
2'VIN'/' THERMAL EXPANSION COEF.OF THE SOLVENT',D13.5,2X,'1/KEL',
3'VIN'/' DELTA T=',D13.5)
1530 FORMAT('/' EQUIVALENT SALT COMPOSITION: '/' ION',4X,'ELECTRIC CHARGE
1',3X,'STOICHIOMETRIC COEFF.'S')
1540 FORMAT(I2,A2,9X,F4.1,11X,1PD10.3)
1550 FORMAT(A13,2X,' LN(MEAN A.C.)',5X,6(A1,'LN(A.C.)',5X))
1555 FORMAT(34X,6(I2,A2,' ION',6X))
1560 FORMAT(33X,6(A2,' Z=',F4.1,5X))
1570 FORMAT(33X,6(A1,' Z+/-=',F4.1,3X))
1580 FORMAT(A13,4X,7(A1,' LN(A.C.)',4X))
1585 FORMAT(18X,7(I2,A2,' ION',6X))
1590 FORMAT(17X,7(A1,' Z=',F4.1,6X))
1600 FORMAT(17X,7(A1,' Z+/-=',F4.1,3X))
1610 FORMAT(2X,1PD12.5,2X,D12.5,5X,6(D12.5,2X))
1615 FORMAT(' *',1PD12.5,2X,D12.5,5X,6(D12.5,2X))
1620 FORMAT(2X,1PD12.5,3X,7(D12.5,2X))
1625 FORMAT(' *',1PD12.5,3X,7(D12.5,2X))
1630 FORMAT(2X,A13,11X,'V-V0',14X,'FIV-FIV0')
1640 FORMAT(2X,1PD12.5,8X,D12.5,A3,5X,D12.5)
1645 FORMAT(' *',1PD12.5,8X,D12.5,A3,5X,D12.5)
1650 FORMAT(2X,A13,13X,'L',18X,'FIL')
1660 FORMAT(A13,5X,'LN(MEAN ACT.COEF.)')
1670 FORMAT(2X,1PD12.5,7X,D12.5)
1675 FORMAT(' *',1PD12.5,7X,D12.5)
1680 FORMAT(/2X,A13,5X,'LN(MEAN ACT.COEF)',5X,

```

```

1'1-OSMOTIC COEF.')
1685 FORMAT(///' LABEL W = WARNING'/' BECAUSE OF A FAILURE OF THE DIFFE
1RENTIATION PROCEDURE'/' THE NON-DEBYE PART OF THE W LABELED VALUE
2WAS EQUATED'/' TO THE NON-DEBYE PART OF PREVIOUS UNLABELED VALUE'/'
3/)
2000 FORMAT(//)
CLOSE(7)
END

```

```

C
C *****
C *****
C *
C * THIS ROUTINE CALCULATES THE SECOND, THIRD, AND FORTH DERIVATIVES *
C * OF Y (SEE TEXT) RESPECT TO X(I). *
C *
C *****
C *****
C

```

```

SUBROUTINE DERIV(X,Y,Y1,Y2,Y3,Y4)
IMPLICIT REAL*8 (A-H,O-Z)
COMMON Z(10),RNI(10),NTION
Y2=0.
T3=0.
T4=0.
DO 1 I=1,NTION
T2=RNI(I)*Z(I)*DEXP(-Z(I)*Y/X)
Y2=Y2+T2
T3=T3+T2*Z(I)
1 T4=T4+T2*Z(I)*Z(I)
Y2=-X*Y2
Y3=Y2/X-(Y/X-Y1)*T3
Y4= Y2*T3-((Y/X-Y1)**2)/X)*T4
RETURN
END

```

```

C
C
C *****
C *****
C
C THIS ROUTINE SETS UP THE OUTPUT ABSCISSA PARAMETER AS IONIC
C STRENGTH, SQRT OF IONIC STRENGTH, CONCENTRATION, SQRT OF CONCN.,
C ACCORDING TO THE SELECTED OPTIONS.
C
C *****
C *****
C

```

```

FUNCTION FUNZ(X,I,W)
REAL*8 X,W,FUNZ
GO TO(1,2,3,4),I
1 FUNZ=X
GO TO 5
2 FUNZ=DSQRT(X)
GO TO 5
3 FUNZ=X/W
GO TO 5
4 FUNZ=DSQRT(X/W)
5 RETURN
END

```