Title
USE OF COMPUTERS IN THE EVALUATION OF THERMODYNAMIC DATA

Permalink
https://escholarship.org/uc/item/3db4n3d5

Author
Hawkins, Donald T.

Publication Date
1966-07-01

UCRL-16935

University of California

Ernest O. Lawrence Radiation Laboratory

USE OF COMPUTERS IN THE EVALUATION OF THERMODYNAMIC DATA

UCRL-16935

UNIVERSITY OF CALIFORNIA

Lawrence Radiation Laboratory
Berkeley, California

AEC Contract No. W-7405-eng-48

USE OF COMPUTERS IN THE EVALUATION OF THERMODYNAMIC DATA

Donald T. Hawkins

July 1966

USE OF COMPUTERS IN THE EVALUATION OF THERMODYNAMIC DATA

Donald T. Hawkins

Inorganic Materials Research Division
Lawrence Radiation Laboratory
and
Department of Mineral Technology
College of Engineering
University of California
Berkeley, California

July 1966

In 1955, a project for the evaluation of the thermodynamic properties of metals and binary metallic alloys was begun. The first edition of the results appeared in 1963,[1] and contained data on 67 elements and 180 alloy systems. The project is currently continuing; the scope of the evaluation has been expanded to include several nonmetallic elements (e.g. C, Si, and As) and alloy systems (e.g. carbides). It is proposed to expand the evaluation to include systems such as the oxides, nitrides, and borides.

At the outset of the project, all data were calculated by hand, but with the rapidly increasing volume of new data, this became a monumental task. Continual refinements of experimental techniques are leading to more precise data, which will necessitate further revision of the previous values. In order to cope with the production of data and to reduce the probabilities of error, an investigation into the possibility of utilizing high-speed digital computers in the data evaluation process was undertaken. Several programs for making some of the more routine calculations have been developed. The purpose of this report is to present these programs as an illustration of the application of computers to this problem.

Most of the programs were originally written in the FORTRAN IV language for use on an IBM 7044 computer. With the acquisition by the Lawrence Radiation Laboratory of a CDC 6600 machine, which uses Chippewa FORTRAN, the older programs were converted to the Chippewa language. Newer programs have been written in Chippewa FORTRAN.

The methods used in the evaluation have been described in detail previously,[1] and will be mentioned only briefly here. A short description and listing of each program are given in the Appendix of this report.

ELEMENTS

Low-Temperature Data. The only application of computers to the evaluation of low-temperature data is the integration of the selected Cp versus T curve by Simpson's Rule to obtain S°_{298}. One short program has been written to perform this integration, but in practice, no saving is gained by the time the cards are punched, verified, and read into the computer. Therefore, this program has not been used for some time, and no summary is included in this report.*

Gas Properties. Several compilations of tables of ideal gas thermodynamic properties of metals have been published. As each appeared,

---

*Most computer installations have library subroutines which can perform this integration.

a program was written to convert the values to a consistent system of units. The tables currently in use are those by Hilsenrath, Messina, and Evans.[3] Since these tables can be used for any system of units by multiplying by the appropriate value of R, no unit conversion is necessary. A program (HILSE), which has been incorporated as a subroutine into the master program for evaluating the data on an element (ELEMNT), calculates the table of selected gas phase values using the units recommended by the National Bureau of Standards.[4] A small, additive factor given by Hilsenrath, et al. to correct atomic weights to the values tabulated by Cameron and Wichers[5] is also used in the calculation.

High-Temperature Data. The first step is to obtain, plot, and select values for the function

$$Y = \frac{H^\circ_T - H^\circ_{298.15}}{T - 298.15}$$

Heat content measurements are often quite voluminous and are reported in a variety of ways. Considerable effort is often expended in the rather mundane tasks of unit conversion and calculation of Y. The first program listed in the Appendix (HEATCO) calculates values for Y from reported values of $H^\circ_T - H^\circ_{298.15}$ expressed in units of joules/gm, joules/gm-atom, cal/gm, or cal/gm-atom, and values of T expressed in °C or °K, and prints tables of T in °K and Y in cal/gm-atom degree. Values of $H^\circ_T - H^\circ_{298.15}$ in cal/gm-atom are also printed.

Once the selected Y curve has been drawn, calculation of the tables of selected values begins. Several shorter programs have been incorporated into one master program (ELEMNT), which completely calculates and prints the table in the standard format used in References 1 and 2. Values of Y are read in at 50°K intervals, and Cp values are calculated from the relation

$$Cp = Y + (T - 298.15)\frac{dY}{dT}.$$

The slope of the Y-function curve, $\frac{dY}{dT}$, is approximated by successively taking 100° intervals along the chosen curve, finding the slope of the line between the two end points, and assuming that the slope at the midpoint of the interval has the same value. Since the Y-function is generally a smooth curve, this assumption should not introduce a very large error. An excellent check is available because the Cp values thus obtained can be integrated to yield values of $H^\circ_T - H^\circ_{298.15}$, which are then compared with values obtained from the definition of the Y-function:

$$H^\circ_T - H^\circ_{298.15} = Y(T - 298.15).$$

In several trial cases, the discrepancy in the enthalpy values was less than 1 calorie, which is well within the experimental scatter. Should the occasion arise where an error of more than one calorie were introduced due to sharp inflections in the Y curve, Y values taken at closer intervals would reduce the error. Provision has been made in the program to use values at 25°K intervals, if needed. Once the Cp values are obtained, values of $S^\circ_T - S^\circ_{298.15}$ can be calculated by integration of Cp/T values. Combining the $S^\circ_T - S^\circ_{298.15}$ and $H^\circ_T - H^\circ_{298.15}$ values with the value of S°_{298} chosen from the low-temperature data yields values of the Gibbs energy function, $(G^\circ_T - H^\circ_{298.15})/T$.

For the liquid metals, a constant Cp is often assumed. Its value and the value of ΔS_m are read in as input data. Calculation of the remainder of the table is then straight forward. In cases where the Cp has been measured and is not constant, the data are at present calculated by hand, but these cases are well in the minority. When reliable data on many liquid metals become available, a program to calculate values in the liquid region when Cp is not constant can easily be written.

In the case of a polymorphic metal, provision has been made to include the values of the β-phase if $Cp_{(\beta)}$ is constant. Since the majority of polymorphic metals have only one additional phase, no provision was made for more than 1 additional phase.

Calculation of the gas properties is done by a subroutine (HILSE) as described in the section entitled Gas Properties.

Since values of the change in Gibbs energy function $\Delta\left[(G^\circ_T - H^\circ_{298.15})/T\right]$ for the reaction

$$M_{(c)} = M_{(g, 1atm)}$$

are useful in evaluating vapor pressure data, values of this function at each tabulation temperature are calculated from the gas and condensed phase tables and printed. These values are also punched on cards for use in other programs.

Vapor Pressure Data. The vapor pressure data are evaluated by the Third Law Method. If the data are numerous, as they often are, doing the Third Law calculation by computer to determine the value of $\Delta H^\circ_{v,298}$ from the measured vapor pressures saves much time. In this and all similar programs the value of the change in Gibbs energy function, $\Delta\left[(G^\circ_T - H^\circ_{298})/T\right]$, at temperatures other than the even 100° temperatures is found by a subroutine (LAGINT) which uses the method of second order Lagrangian interpolation. The error introduced by this procedure is of the order of 0.001, which is an order of magnitude less than if the calculation were done graphically. The program to do the Third Law calculation (S 3 LAWS) calculates $\Delta H^\circ_{v,298}$ from the relation

$$\Delta H^\circ_{v,298} = -RT\ell nP - T\Delta\left[\frac{(G^\circ_T - H^\circ_{298})}{T}\right]$$

The cards obtained as output from ELEMNT, giving values of T and the Gibbs energy function at even 100° intervals, are used as input, as well as cards containing measured temperatures and pressures. The output consists of a table of values for T, P, log P, and $\Delta H^\circ_{v,298}$. In addition, an average value of $\Delta H^\circ_{v,298}$ is calculated, as well as the deviation of each measurement from the average, and the average and standard deviations. An option is available for obtaining punched card output as well as printed output. If the pressure data are in millimeters of Hg, instead of atmospheres, provision has been made for unit conversion before the calculation of $\Delta H^\circ_{v,298}$ takes place.

If no thermal data are available, making impossible the calculation of the Gibbs energy function, a program (VAPOR) is used to calculate values of $R\ell nP$ and $\frac{1}{T}$ for use in the Second Law method of determining ΔH°_v over the measured temperature range. In this program also, conversion from millimeters Hg to atmospheres is available.

Once the selected value of $\Delta H^\circ_{v,298}$ is known, program VPCALC is used to obtain Table 3 (Vapor Pressure Data). The cards from ELEMNT containing Gibbs energy function values are used as input as in the case of S 3 LAWS. Values of $\Delta G^\circ_{v,T}$ and P are calculated from the relations

$$\Delta G^\circ_{v,T} = T\,\Delta\left[\frac{(G^\circ_T - H^\circ_{298})}{T}\right] + \Delta H^\circ_{v,298}$$

$$\Delta G^\circ_{v,T} = -RT\,\ell nP.$$

In calculating P, special provision had to be made for the extremely small values sometimes obtained (e.g. for W at 298.15°K, $P = 3.8\times10^{-142}$ atm.). If these values were calculated directly, an error would occur due to the limitation on the magnitude of numbers which can be stored in the computer. In order to avoid this, the value of the exponent and the mantissa are calculated separately. By setting P = 1, the normal boiling point can be found:

$$T_{bp} = \frac{-\Delta H^{\circ}_{v,298}}{\Delta\left[\frac{(G^{\circ}_{T} - H^{\circ}_{298})}{T}\right]}$$

However, since the Gibbs energy function is dependent on temperature, T_{bp} cannot be solved for explicitly. An iteration procedure using Subroutine LAGINT to interpolate the Gibbs energy function is employed to calculate T_{bp}. The limit of accuracy set on T_{bp} is 1 degree. In a similar way, values of T at even pressures (10^{-10}, 10^{-9}, ----, 10^{-1}) are also calculated:

$$T_{p} = \frac{\Delta H^{\circ}_{v,298}}{-R\ell nP - \Delta\left[\frac{(G^{\circ}_{T} - H^{\circ}_{298})}{T}\right]}$$

thus completing the evaluation.

In the previous programs it has been assumed that the vapor is an ideal monatomic gas. For several metals the vapor also contains higher atomic species. In these cases, several methods are available, but each case is somewhat specialized so that general programs for the complete evaluation cannot be written. Some programs for special cases where the calculations are lengthy have been written, and an example of one of these will serve as an illustration of what can be done.

The program (B3LAW) was written for selenium, in which both diatomic and hexatomic species exist. The program calculates $\Delta H^{\circ}_{v,298}$

for both species and averages the values for each species. The deviation of each measurement from the average as well as the average deviation are calculated for each species in an analogous manner to that of Program S3LAWS.

ALLOYS

The data for alloys are far more varied than the data for elements. Each system usually has to be treated as an individual case, and very few standard procedures are available. In addition, the data on alloys are much more sparse than the data on elements. This means that the application of computers to the evaluation of the thermodynamic data on alloys is very difficult. Thus far, only two programs have been proven useful.

The first is a rather short program (ACTIV) which calculates activity values from vapor pressure data. The punched cards from Program ELEMNT are used as input to calculate P°. Subroutine LAGINT is used to interpolate the Gibbs energy values.

$$P^\circ = \text{antilog}\left[\frac{1}{R}\left(\Delta\left[\frac{(G^\circ_T - H^\circ_{298})}{T}\right] - \frac{\Delta H^\circ_{v,298}}{T}\right)\right]$$

Vapor Pressures over the alloy are then read in and activities calculated:

$$a_i = \frac{P_i}{P^\circ_i}$$

From these values and the mole fraction of component i, x_i, other thermodynamic quantities follow:

$$\gamma_i = \frac{a_i}{x_i}$$

$$\Delta\overline{G}_i = RT\ell n\, a_i$$

$$\Delta\overline{G}_i^{xs} = RT\ell n\, \gamma_i$$

Output consists of tables of the above values plus an option to have them on punched cards.

The only other program that has been developed is useful when data exist over the entire composition range. This program (ALLOY) takes equilibrium data in the form of the α-function $\left(\alpha_1 = \Delta\overline{G}_1^{xs}/(1-x_1)^2\right)$ and performs the Gibbs-Duhem integration to obtain $\Delta\overline{G}_2^{xs}$. The integration is done by Simpson's Rule. Values for $\Delta\overline{G}_1$, $\Delta\overline{G}_1^{xs}$, γ_1, a_1, and the same quantities for component 2 are calculated and printed in the standard format used in Reference 1. The integral quantities ΔG and ΔG^{xs} are also calculated.

If heat or entropy data are available, subroutines are used to calculate the remaining quantities. Heat data are correlated using the Q-function ($Q = \Delta H/x_1x_2$). The partial molar heats for one component can be obtained graphically, and from these and the Gibbs energy values, the remaining partial molar and integral heats and entropies are calculated and printed in the standard format.

Entropy data are correlated with the β-function $\left(\beta_1 = \Delta\overline{S}_1^{xs}/(1-x_1)^2\right)$.

From this function, the Gibbs-Duhem integration will yield $\Delta\overline{S}_2^{xs}$, and all other quantities then follow as described above.

CONCLUSION

Programs for evaluating the thermodynamic data of metals and alloys have been described. Listings of these programs follow in the Appendix.

ACKNOWLEDGEMENTS

The assistance of Carl Quong and Bill Dempster of the Math and Computing Group, Lawrence Radiation Laboratory, in debugging the programs described in this paper is gratefully acknowledged. Subroutines LAGINT, LAGR[illegible]G , and BAINS were furnished by the Computer Library. Mrs. Marian Smith wrote several of the preliminary programs used in the evaluation project.

This work was performed under the auspices of the United States Atomic Energy Commission.

REFERENCES

1. R. Hultgren, R. L. Orr, P. D. Anderson, and K. K. Kelley, Selected Values of Thermodynamic Properties of Metals and Alloys, John Wiley and Sons, New York, 1963.
2. R. Hultgren, R. L. Orr, and K. K. Kelley, Supplement to Selected Values of Thermodynamic Properties of Metals and Alloys, Inorganic Materials Research Division, Lawrence Radiation Laboratory, University of California, Berkeley, California, 1964-present.
3. J. Hilsenrath, C. G. Messina, and W. H. Evans, Air Force Weapons Laboratory Report No. TDR-64-44, 1964.
4. Nat. Bur. Standards Technical News Bulletin, 47 (10), 175-7, (1963).
5. A. E. Cameron and E. Wichers, J. Am. Chem. Soc., 84, 4175-97, (1962).

APPENDIX

This section gives listings of all programs discussed in the preceeding section, as follows:

```
      PROGRAM HEATCO(INPUT,OUTPUT,TAPE 2=INPUT,TAPE 3=OUTPUT)
C     HIGH TEMPERATURE HEAT CONTENT CALCULATION
      DIMENSION IDENT(12),T(1000),H(1000),DIFF(1000),Y(1000),TT(1000),HH
     1(1000),P(1000)
C     READ INPUT DATA
   80 READ (2,1) IDENT
    1 FORMAT (12A6)
      IF(IDENT .EQ. 6H      )CALL EXIT
   82 READ (2,2) J,K,L,AW
    2 FORMAT(3I4,F10.5)
      I=0
    3 I=I+1
      READ (2,4) T(I),H(I)
    4 FORMAT (F8.2,F12.3)
      IF (T(I) .EQ. 0.0) GO TO 5
      GO TO 3
    5 NUMX = I-1
C     WRITE INPUT DATA
      WRITE (3,14) IDENT
   14 FORMAT (1H1,12A6)
      WRITE (3,2) J,K,L,AW
      WRITE (3,13)
   13 FORMAT (54H0 INPUT DATA                                       )
      WRITE (3,15)
   15 FORMAT (54H0     T        H                                   )
      DO 16 I=1,NUMX
      WRITE (3,17) T(I),H(I)
   17 FORMAT (1X,F8.2,4X,F8.2)
   16 CONTINUE
      IF (J .EQ. 1) GO TO 6
      GO TO 3
C     CONVERT DEGREES C TO DEGREES K
    6 DO 7 I=1,NUMX
      TT(I)=T(I)+273.15
    7 T(I)=TT(I)
    8 IF (K .EQ. 1) GO TO 9
      GO TO 31
C     CONVERT JOULES TO CALORIES
    9 DO 10 I=1,NUMX
   10 H(I)=(H(I))/(4.1840)
   31 IF (L.EQ. 1) GO TO 30
      GO TO 53
C     CONVERT GRAMS TO GRAM-ATOMS
   30 DO 32  I = 1,NUMX
   32 H(I) = H(I)*AW
   53 DO 12  I=1,NUMX
      DIFF(I)=T(I)-298.15
C     CALCULATE Y
   12 Y(I)=(H(I))/DIFF(I)
C     WRITE RESULTS
      WRITE (3,14) IDENT
      WRITE (3,18)
   18 FORMAT (54H0  CALCULATION RESULTS                             )
      WRITE (3,19)
   19 FORMAT(55H0    T(KELVIN)     H(T)-H(298)          Y             )
      DO 20 I=1,NUMX
      WRITE (3,21) T(I),H(I),Y(I)
```

```
21 FORMAT (5X,F8.2,6X,F8.2,8X,F10.4)
20 CONTINJE
   GO TO 80
   END
```

```
      PROGRAM ELEMNT(INPUT,OUTPUT,TAPE 2=INPUT,TAPE 3=OUTPUT,TAPE 14)
      DIMENSION NAME(2),IDENT(5),Y(500),T(500),H(500),SLOPE(500),CP(500)
     1,DS(500),DELTAH(500),DELTAS(500),GEF(500),DATE(2),EVAL(3),TT(100),
     2HLQ(100),SLQ(100),FLQ(100),GLQ(100),HHH(100),SS(100),GGEF(100),
     3DELTAF(500),DEV(100),F3(100),FEF(500),DG(500)
      COMMON TINC,N,NUM,T,NAME,NPAGE,LINES,L,MM,IK,TM,TF
C     READ INPUT DATA
    1 READ (2,100) NAME(1),NAME(2),DATE(1),DATE(2),(EVAL(J), J=1,3),
     1IDENT
  100 FORMAT(12A6)
      IF (NAME(1) .EQ. 6H            ) CALL EXIT
      READ(2,101) SST,TM,DSM,CPM,CPL,TINC,TF,TTR
  101 FORMAT(8F10.4)
      READ(2,1001) SLOPE(1),SLOPE(2),SLP
 1001 FORMAT(3F8.6)
C     IF THERE IS NO TRANSITION, TTR = 0
      IF(TTR .EQ. 0.0) GO TO 2
      READ(2,101) CPTR,STR,CPB
    2 I=0
    3 I=I+8
      N=I-7
      READ(2,101) Y(I-7),Y(I-6),Y(I-5),Y(I-4),Y(I-3),Y(I-2),Y(I-1),Y(I)
      DO 4  II=N,I
      IF(Y(II) .LE. 0.0) GO TO 5
    4 CONTINUE
      GO TO 3
    5 N=II-1
C     CALCULATE VALUES OF T
      T(1) = 298.15
      T(2) = 300.0
      DO 6  I=3,N
    6 T(I) = T(I-1) + TINC
      NPAGE = 1
C     WRITE TABLE HEADINGS
      WRITE(3,102)
  102 FORMAT(1H1)
  103 FORMAT(1H       )
      WRITE(3,104) NAME,NPAGE
  104 FORMAT(47H EVALUATION OF THE THERMODYNAMIC PROPERTIES OF   ,2A6,5X,
     16H PAGE  ,I2)
      WRITE(3,111) EVAL,DATE,IDENT
  111 FORMAT(4H BY ,3A6,10X,2A6,10X//5A6//)
      WRITE(3,105)
  105 FORMAT(35X,11H INPUT DATA    )
      WRITE(3,103)
C     WRITE INPUT DATA
      WRITE(3,106) SST,TM,CPM,DSM,CPL,SLOPE(1),SLP,TF
  106 FORMAT(7H SST = ,F6.3,3X,6H TM = ,F6.1,3X,12H CP AT MP = ,F5.3,3X,
     17H DSM = ,F5.2,3X,7H CPL = ,F5.3 / 23H DY/DT AT 298 DEG.K =  F8.6,
     23X,15H DY/DT AT MP = ,F8.6,3X,11H FINAL T = ,F6.1)
C     WRITE INPUT DATA FOR BETA PHASE IF THERE IS A TRANSITION
      IF (TTR .NE. 0.0) WRITE(3,107)TTR,CPB,STR,CPTR
  107 FORMAT (7H TTR = ,F6.1,3X,12H CP(BETA) = ,F5.3,3X,7H STR = ,F5.2,
     13X,13H CP AT TTR = ,F5.3)
      WRITE(3,103)
      WRITE(3,108)
```

```
  108 FORMAT(12X,2H T,11X,2H Y)
      WRITE(3,103)
      LINES=16
   14 DO 10  I=1,N
      WRITE(3,109) T(I),Y(I)
  109 FORMAT(10X,F7.2,6X,F7.4)
C     THE FOLLOWING PROGRAM SEGMENT, WHICH APPEARS SEVERAL TIMES IN THIS
C     PROGRAM, SPACES TO THE END OF THE PAGE AND PRINTS THE LAST LINE ON
C     EACH PAGE.  ONLY 55 LINES ARE PRINTED ON EACH PAGE.
      LINES=LINES+1
      IF(LINES .EQ. 55) CALL ENDPG
      IF (I .NE. N) GO TO 10
  151 WRITE(3,103)
      LINES=LINES+1
      IF(LINES .LT. 55) GO TO 151
      CALL ENDPG
   10 CONTINUE
C     CALCULATE HEAT VALUES DIRECTLY FROM THE Y-PLOT
   15 DO 7  I=1,N
    7 H(I)=Y(I)*(T(I)-298.15)
      NN=N-1
C     CALCULATE CP VALUES BY TAKING SLOPES ON THE Y-PLOT THEN INTEGRATE
C     THESE CP VALUES USING SUBROUTINE AREA TO OBTAIN THE HEAT.
      DO 8 I=3,NN
C     THE NEXT STATEMENT APPROXIMATES THE SLOPE OF THE Y- PLOT AT 100
C     DEGREE INTERVALS
    8 SLOPE(I) = (Y(I+1)-Y(I-1))/(T(I+1)-T(I-1))
      SLOPE(N) = SLP
      DO 9  I=1,N
      CP(I) = Y(I) + (T(I)-298.15)*SLOPE(I)
    9 DS(I) = CP(I)/T(I)
      CALL AREA(CP,DELTAH)
      CALL AREA(DS,DELTAS)
      DELTAH(1) = 0.0
      DELTAH(2) =((CP(2) + CP(1))/2.)*1.85
      DELTAS(1)=0.0
      DELTAS(2) = H(2)/299.10
C     NUM = NUMBER OF TEMPERATURES UP TO THE MELTING POINT
      DO 99  I=3,NUM
      DELTAH(I) = DELTAH(I) + DELTAH(2)
   99 DELTAS(I) = DELTAS(I) + DELTAS(2)
      GEF(1) = SST
C     REINDEX THE TEMPERATURE, CP, AND HEAT VALUES SO THAT ONLY THE EVEN
C     100 DEGREE VALUES ARE RETAINED IN THE CALCULATION
   16 IF(TINC .EQ. 25.) GO TO 500
      DO 501  I=4,N,2
      II=(I/2)+1
      CP(II)=CP(I)
      T(II)=T(I)
      H(II)=H(I)
  501 CONTINUE
      GO TO 503
  500 DO 502  I=6,N,4
      II=(I+6)/4
      CP(II)=CP(I)
      T(II)=T(I)
  502 H(II)=H(I)
```

```
  503 DO 12  I=2,NUM
      FEF(I) =       (H(I)/T(I))-DELTAS(I)-SST
      GEF(I) = -FEF(I)
      IF(T(I)-TF) 12,166,166
   12 CONTINUE
C     WRITE TABLE HEADINGS FOR CONDENSED PHASE
  166 WRITE(3,125)
  125 FORMAT(32H CONDENSED PHASE SELECTED VALUES             //)
      WRITE(3,112)
  112 FORMAT(5X,8H T DEG K,8X,3H CP,6X,11H H(T)-H(ST),2X,11H S(T)-S(ST),
     16X,4H GEF //)
      LINES=7
C     WRITE CONDENSED PHASE TABLE UP TO THE MELTING POINT
   19 DO 17  I=1,NUM
      WRITE(3,113) T(I),CP(I),H(I),DELTAS(I),GEF(I)
  113 FORMAT(6X,F7.2,7X,F6.3,7X,F6.0,7X,F6.3,7X,F6.3)
      LINES=LINES+1
      IF(LINES .EQ. 55) GO TO 18
   17 CONTINUE
      GO TO 188
   18 CALL ENDPG
  188 IF(T(I) .GE. TF) GO TO 1
C     THE FOLLOWING SECTION IS SKIPPED IF THERE IS NO TRANSITION OCCURRING
   20 IF (TTR .LE. 0.0) GO TO 21
C     CALCULATE AND WRITE VALUES AT TRANSITION TEMPERATURE (IF PRESENT)
      HT = ((CPTR + CP(NUM))/2.)*(TTR-T(NUM))+H(NUM)
      ST=((CPTR+CP(NUM))/2.)*ALOG(TTR/T(NUM))+DELTAS(NUM)
      FEFT=(HT/TTR)-ST-SST
      GEFT=-FEFT
      WRITE(3,113) TTR,CPTR,HT,ST,GEFT
      LINES=LINES+1
      IF(LINES .LT. 55) GO TO 45
      CALL ENDPG
   45 DHTR = TTR*STR+HT
      DSTR=STR+ST
      FFT=(DHTR/TTR)-DSTR-SST
      GFT=-FFT
      WRITE(3,113) TTR,CPB,DHTR,DSTR,GFT
      LINES=LINES+1
      IF(LINES .LT. 55) GO TO 22
      CALL ENDPG
C     CALCULATE AND WRITE VALUES FROM THE TRANSITION TEMPERATURE (IF PRESENT)
C     UP TO THE MELTING POINT
   22 TK = T(NUM) + 100.
      HK = CPB*(TK-TTR)+DHTR
      SK=CPB*ALOG(TK/TTR)+DSTR
      FK=(HK/TK)-SK-SST
      GK=-FK
      WRITE(3,113) TK,CPB,HK,SK,GK
      LINES=LINES+1
      IF(LINES .LT. 55) GO TO 46
      CALL ENDPG
   46 T(NUM+3)=TK
      JJ=NUM+4
      JJ20 = NUM+24
      DO 23  I=JJ,JJ20
      T(I) = T(I-1) + 100.
```

```
      IF(T(I)-TM) 23,23,24
   23 CONTINUE
   24 IK=I-1
      NM=NUM+4
      DO 25  I=NM,IK
      HHH(I)=CPB*(T(I)-TK)+HK
      SS(I) = CPB*ALOG(T(I)/TK)+SK
      FEF(I) = (HHH(I)/T(I))-SS(I)-SST
      GGEF(I) = -FEF(I)
      WRITE(3,113) T(I),CPB,HHH(I),SS(I),GGEF(I)
      LINES=LINES+1
      IF(LINES .LT. 55) GO TO 25
      CALL ENDPG
   25 CONTINUE
C     CALCULATE AND WRITE VALUES AT THE MELTING POINT FOR BOTH THE SOLID
C     AND LIQUID
      HH=CPB*(TM-T(IK))+HHH(IK)
      SSA=CPB*ALOG(TM/T(IK))+SS(IK)
      GO TO 26
   21 HH=((CPM+CP(NUM))/2.)*(TM-T(NUM))+H(NUM)
      SSA=((CPM+CP(NUM))/2.)*ALOG(TM/T(NUM))+DELTAS(NUM)
   26 FF=(HH/TM)-SSA-SST
      GG=-FF
      WRITE(3,113) TM,CPM,HH,SSA,GG
      LINES=LINES+1
      IF(LINES .LT. 55) GO TO 27
      CALL ENDPG
   27 HM=TM*DSM+HH
      SM=DSM+SSA
      FEFM=(HM/TM)-SM-SST
      GEFM=-FEFM
      WRITE(3,113) TM,CPL,HM,SM,GEFM
      LINES=LINES+1
      IF(LINES .LT. 55) GO TO 28
      CALL ENDPG
   28 IF (TTR .EQ. 0.0) GO TO 29
      TMM = T(IK) + 100.
      GO TO 30
   29 TMM = T(NUM) + 100.
   30 HMM = CPL*(TMM-TM) + HM
      SMM = CPL*ALOG(TMM/TM)+SM
      FMM=(HMM/TMM)-SMM-SST
      GMM = -FMM
      WRITE(3,113) TMM,CPL,HMM,SMM,GMM
      LINES=LINES+1
      IF(LINES .LT. 55) GO TO 31
      CALL ENDPG
   31 TT(1) = TMM + 100.
      DO 32  I=2,70
      TT(I) = TT(I-1) + 100.
      IF(TT(I)-TF) 32,33,33
   32 CONTINUE
   33 L=I
C     CALCULATE AND WRITE VALUES IN LIQUID FROM THE MELTING POINT UP TO THE
C     FINAL TEMPERATURE
      DO 34  I=1,L
      HLQ(I) = CPL*(TT(I)-TM)+HM
```

```
      SLQ(I) = CPL*ALOG(TT(I)/TM)+SM
      FLQ(I) = (HLQ(I)/TT(I))-SLQ(I)-SST
      GLQ(I)=-FLQ(I)
      WRITE(3,113) TT(I),CPL,HLQ(I),SLQ(I),GLQ(I)
      LINES=LINES+1
      IF(LINES .LT. 55) GO TO 34
      CALL ENDPG
   34 CONTINUE
      IF(LINES .EQ. 55) GO TO 36
   35 WRITE(3,103)
      LINES=LINES+1
      IF(LINES .LT. 55) GO TO 35
   36 CALL ENDPG
C     HILSE CALCULATES THE GAS PHASE TABLE
      CALL HILSE(F3)
C     THE FOLLOWING SECTION REINDEXES THE VALUES SO THAT THE INDICES RUN FROM
C     1 TO A FINAL VALUE.
      IF(TTR .GT. 0.0) GO TO 37
      GEF(NUM+1)=GG
      GEF(NUM+2)=GEFM
      GEF(NUM+3)=GMM
      M=NUM+4
      GO TO 38
   37 GEF(NUM+1)=GEFT
      GEF(NUM+2)=GFT
      GEF(NUM+3)=GK
      LL=NUM+4
      DO 377 I=LL,IK
      GEF(I)=GGEF(I)
  377 CONTINUE
      GEF(IK+1)=GG
      GEF(IK+2)=GEFM
      M=IK+3
   38 J1=0
      MM=M+L-1
      DO 39  I=M,MM
      J1=J1+1
      GEF(I)=GLQ(J1)
   39 CONTINUE
C     CALCULATE VALUES OF DELTA(GEF)
      DO 40  I=1,MM
      DELTAF(I)=F3(I)-GEF(I)
   40 CONTINUE
      WRITE(3,126)
  126 FORMAT(3X,2H T,4X,11H DELTA(GEF))
      WRITE(3,103)
      LINES=LINES+2
C     PUNCH VALUES OF DELTA(GEF) FOR USE IN OTHER PROGRAMS
      WRITE(14,100) NAME(1),NAME(2),IDENT
      DO 47  I=1,MM
      WRITE(3,127) T(I),DELTAF(I)
      WRITE(14,127) T(I),DELTAF(I)
  127 FORMAT(1X,F7.2,4X,F6.3)
      LINES=LINES+1
      IF(LINES .LT. 55) GO TO 47
      CALL ENDPG
   47 CONTINUE
```

```
   54 WRITE(3,103)
      LINES=LINES+1
      IF (LINES .LE. 55) GO TO 54
      CALL ENDPG
C     CALCULATE AND WRITE A COMPARISON OF HEAT VALUES OBTAINED DIRECTLY FROM
C     THE Y-PLOT WITH THOSE OBTAINED BY INTEGRATION OF THE CP CURVE
      WRITE(3,117)
  117 FORMAT(54H COMPARISON OF HEATS OBTAINED DIRECTLY FROM THE Y-PLOT
     1/53H WITH THOSE OBTAINED BY INTEGRATION OF THE CP VALUES.     //)
      WRITE(3,118)
  118 FORMAT(8X,2H T,5X,12H HEAT(YPLOT) ,2X,12H YPLOT SLOPE ,3X,17H HEAT
     1(INTEGRATED) ,3X,16H H(YPLOT)-H(INT)          //)
      LINES=9
      DO 43  I=1,NUM
      DEV(I)=H(I)-DELTAH(I)
      WRITE(3,119) T(I),H(I),SLOPE(I),DELTAH(I),DEV(I)
  119 FORMAT(6X,F7.2,6X,F6.0,6X,F9.6,11X,F6.0,12X,F8.1)
      LINES=LINES+1
      IF(LINES .LT. 55) GO TO 43
      CALL ENDPG
   43 CONTINUE
   44 WRITE(3,103)
      LINES=LINES+1
      IF(LINES .LE. 55) GO TO 44
      CALL ENDPG
      ENDFILE 14
      GO TO 1
      END
```

```
      SUBROUTINE ENDPG
C     WRITES THE LAST LINE ON EACH PAGE AND THE PAGE HEADINGS FOR THE
C     FOLLOWING PAGE.  THIS SUBROUTINE ALSO KEEPS TRACK OF THE PAGE NUMBERS.
      DIMENSION NAME(2), T(500)
      COMMON TINC,N,NUM,T,NAME,NPAGE,LINES,L,MM
      WRITE(3,110)
  110 FORMAT(1H   /50X,50H OFFICIAL NSRDS EVALUATION, CERTIFIED NO MISTE
     1AKS       )
      WRITE(3,1002)
 1002 FORMAT(1H1)
      NPAGE=NPAGE+1
      WRITE(3,1004)NAME,NPAGE
 1004 FORMAT(47H EVALUATION OF THE THERMODYNAMIC PROPERTIES OF  ,2A6,5X,
     16H PAGE  ,I2)
      WRITE(3,1003)
 1003 FORMAT(1H0)
      LINES=2
      RETURN
      END
```

```
      SUBROUTINE AREA(X,SUM)
C     INTEGRATES THE CP CURVE BY MEANS OF SIMPSON'S RULE
      DIMENSION A(500),X(500),SUM(500),T(500), NAME(2)
      COMMON TINC,N,NUM,T,NAME,NPAGE,LINES,L,MM
      SUM=0.0
      IF(TINC .EQ. 25.) GO TO 4
      DO 1  I=4,N,2
      A(I) = (100./6.)*(X(I-2)+4.*X(I-1)+X(I))
    1 SUM(I) = SUM(I-2)+A(I)
    2 DO 3  I=4,N,2
      II=(I/2)+1
    3 SUM(II) = SUM(I)
      NUM = II
      RETURN
    4 DO 5  I=6,N,4
      A(I) = (100./12.)*(X(I-4)+4.*X(I-3)+2.*X(I-2)+4.*X(I-1)+X(I))
    5 SUM(I) = SUM(I-4)+A(I)
      DO 6  I=6,N,4
      II=(I+5)/4
    6 SUM(II) = SUM(I)
      NUM=II
      RETURN
      END
```

```
      SUBROUTINE HILSE(F3)
C     CALCULATES THE GAS PHASE TABLE
      DIMENSION S(100),S2(100),C(100),C1(100),HA(100),H1(100),F2(100),
     1F3(100),T(500) ,NAME(2)
      COMMON TINC,N,NUM,T,NAME,NPAGE,LINES,L,MM
      DATA R/1.98717/
      READ(2,200) CORR
  200 FORMAT(F10.5)
  201 FORMAT(1H1)
      WRITE(3,203)
  203 FORMAT(1H            )
      WRITE(3,204)
  204 FORMAT(66H INPUT DATA TO SUBROUTINE HILSE. COLUMNS AS LISTED IN HI
     1LSENRATH.                    //)
      WRITE(3,211) CORR
  211 FORMAT(37H CORRECTION FACTOR FOR NEW AT. WT. =       ,F10.5//)
      WRITE(3,205)
  205 FORMAT(8X,2H 1,8X,2H 4,8X,2H 5,9X,2H 6  //)
      LINES=12
      I=0
    7 I=I+1
      READ(2,206) T(I),S(I),C(I),HA(I)
  206 FORMAT(1X,F7.2,5X,F7.4,5X,F7.4,4X,F8.2)
      IF(T(I) .EQ. 0.) GO TO 8
      GO TO 7
    8 MM=I-1
      DO 1  I=1,MM
      WRITE(3,207) T(I),S(I),C(I),HA(I)
  207 FORMAT(6X,F7.2,3X,F7.4,3X,F7.4,4X,F8.2)
      LINES=LINES+1
      IF(LINES .LT. 55) GO TO 1
      CALL ENDPG
    1 CONTINUE
    2 WRITE(3,203)
      LINES=LINES+1
      IF(LINES .LT. 55) GO TO 2
      CALL ENDPG
      WRITE(3,208)
  208 FORMAT(26H GAS PHASE SELECTED VALUES                          //)
      WRITE(3,209)
  209 FORMAT(5X,8H T DEG K,8X,3H CP,6X,11H H(T)-H(ST),2X,11H S(T)-S(ST),
     16X,4H GEF      //)
      LINES=7
      S298=R*(S(1)+CORR)
      S2(1) = 0.0
      DO 3  I=2,MM
    3 S2(I) = (R*S(I))-S298+R*CORR
      DO 4  I=1,MM
      C1(I) = R*C(I)
      H1(I) = R*(HA(I)-HA(1))
      F2(I) = (H1(I)/T(I))-S298-S2(I)
    4 F3(I) = -F2(I)
      DO 5  I=1,MM
      WRITE(3,210) T(I),C1(I),H1(I),S2(I),F3(I)
  210 FORMAT(6X,F7.2,7X,F6.3,7X,F6.0,7X,F6.3,7X,F6.3)
      LINES=LINES+1
```

```
      IF(LINES .LT. 55) GO TO 5
      CALL ENDPG
    5 CONTINUE
    6 WRITE(3,203)
      LINES = LINES +1
      IF(LINES .LT. 55) GO TO 6
      CALL ENDPG
      RETURN
      END
```

```
      PROGRAM HILSE (INPUT,OUTPUT,TAPE 2=INPUT,TAPE 3=OUTPUT)
CORRECTION PROGRAM FOR HILSENRATH ET AL TABLES OF IDEAL GAS THERMODY-
C   NAMIC FUNCTIONS.
      DIMENSION T(100),E(100),S(100),C(100),H(100),C1(100),S1(100),S2(10
     10),H1( 100),F1(100),F2(100),F3(100),IDENT (12)
      DATA R/1.98717/
C     READ INPUT DATA
    5 READ (2,100) IDENT
  100 FORMAT (12A6)
      IF (IDENT .EQ. 6H                  )CALL EXIT
      READ (2,102) CORR
  102 FORMAT (F10.5)
      I=0
    1 I=I+1
      READ ( 2,101) T(I),      S(I),C(I),H(I)
  101 FORMAT(F11.3,F10.4,F12.4,F12.2)
      IF (T(I) .EQ. 0.0) GO TO 2
      GO TO 1
    2 N=I-1
C     WRITE TABLE HEADINGS AND INPUT DATA
      WRITE (3,109) IDENT
  109 FORMAT (1H1,12A6)
      WRITE (3,103)
  103 FORMAT (50H0INPUT DATA. COLUMNS AS LISTED IN HILSERATH.           )
      WRITE (3,106)
  105 FORMAT ( 50H0      1       4           5           6           )
      WRITE (3,107)
  107 FORMAT (1H0)
      DO 3 I=1,N
      WRITE (3,101) T(I),      S(I),C(I),H(I)
    3 CONTINUE
C     CALCULATE CORRECTED VALUES
      S298 = R*(S(1)+CORR)
      S2(1)=0.0
      DO 10 I=2,N
      S2(I)=(R*S(I))-S298 +R*CORR
   10 CONTINUE
      DO 4 I=1,N
      C1(I)=R*C(I)
      H1(I)=(H(I)-H(1))*R
      F2(I)=(H1(I)/T(I))-S298-S2(I)
      F3(I)=-F2(I)
    4 CONTINUE
C     WRITE OUTPUT
      WRITE (3,109) IDENT
      WRITE (3,104)
  104 FORMAT (50H0SELECTED VALUES...                                     )
      WRITE (3,108)
  108 FORMAT (100H0       T          CP        HT-HST     ST-SST       -GFEF
     1                                                                    )
      WRITE (3,107)
      DO 5 I=1,N
      WRITE (3,105) T(I),C1(I),H1(I),S2(I),F3(I)
  105 FORMAT (2X,F8.2, 2X,F8.3,2X,F8.0,3X,F8.3,3X,F8.3)
    5 CONTINUE
      GO TO 5
      END
```

```
      PROGRAM  S3LAWS (INPUT,OUTPUT,TAPE2=INPUT,TAPE 3=OUTPUT,TAPE 14)
C  THIRD LAW CALCULATIONS FOR MONATOMIC SPECIES
      DIMENSION IDENT(12),TT(200),FEFD(200),P(200),X(200),XX(200),T(200)
     1,TQ(200),FEFQ(200),TR(200),FEFR(200),FF(200),PP(200),PPP(200),PD(2
     200),B(200),DHST(200),DHD(200),Q(200),RES(200)
      DATA R/1.98717/
C     READ INPUT DATA
    1 READ (2,100) IDENT
  100 FORMAT (12A6)
      IF(IDENT .EQ. 6H      )STOP
      READ(2,110) K,KK,M
  110 FORMAT(3I4)
      IF(M .EQ. 1) GO TO 155
C     READ T AND DELTA(GEF) AT EVEN TEMPERATURES
      I=0
    2 I=I+1
      READ(2,114) TT(I),FEFD(I)
  114 FORMAT(1X,F7.2,4X,F6.3)
      IF(I .EQ. 1) GO TO 14
C     FIND MELTING TEMPERATURE
      IF(TT(I) .EQ. TT(I-1)) GO TO 16
   14 IF(TT(I) .EQ. 0.0) GO TO 15
      GO TO 2
   16 TM=TT(I)
C     IMELT=NUMBER OF VALUES OF T AND DELTA(GEF) UP TO THE MELTING POINT
      IMELT=I-1
      GO TO 14
C     NN= TOTAL NUMBER OF VALUES OF T AND DELTA(GEF)
   15 NN=I-1
C     READ MEASURED PRESSURES.  X AND XX ARE DUMMIES NOT USED IN THIS PROGRAM.
  155 I=0
   17 I=I+1
      READ(2,101) I(I),P(I),X(I),XX(I)
  101 FORMAT(F8.2,E12.4,F10.2,F8.2)
      IF(T(I) .EQ. 0.0) GO TO 4
      GO TO 17
C     NUMX=NUMBER OF DATA POINTS
    4 NUMX=I-1
      IF(KK .EQ. 0) GO TO 18
C     CONVERT MILLIMETERS TO ATMOSPHERES.
      DO 19  I=1,NUMX
      P(I)=P(I)/760.
   19 CONTINUE
C     BREAK THE DELTA(GEF) CURVE INTO TWO PARTS AT THE MELTING POINT.
   18 DO 20  I=1,IMELT
      TQ(I)=TT(I)
      FEFQ(I)=FEFD(I)
   20 CONTINUE
      IMELT1=IMELT+1
      IA=1
      DO 21  I=IMELT1,NN
      TR(IA)=TT(I)
      FEFR(IA)=FEFD(I)
      IA=IA+1
   21 CONTINUE
      NA=NN-IMELT
```

```
C     FIND WHICH PART OF THE DELTA(GEF) CURVE T IS IN AND INTERPOLATE
      DO 22  I=1,NUMX
      TA=T(I)
      IF (TA .GT. TM) GO TO 23
      IF (TA .LE. TM) GO TO 200
  200 CALL LAGINT(IMELT,TQ,FEFQ,2,KX,TA,F)
      FF(I)=F
      GO TO 22
   23 CALL LAGINT(NA,TR,FEFR,2,KX,TA,F)
      FF(I)=F
   22 CONTINUE
C     WRITE INPUT DATA
      WRITE(3,102)
  102 FORMAT(1H1)
      WRITE (3,104)
  104 FORMAT (1H0)
      WRITE (3,103)
  103 FORMAT (50H0 INPUT DATA                                      )
C     NEXT STATEMENT AVOIDS PRINTING T AND DELTA(GEF) AT EVEN TEMPERATURES
C     MORE THAN ONCE FOR A GIVEN ELEMENT.
      IF(M .EQ. 1) GO TO 25
      WRITE (3,111)
  111 FORMAT(8H T DEG K,1X,11H DELTA(GEF)    )
      WRITE(3,104)
      DO 24  I=1,NN
      WRITE(3,114) TT(I),FEFD(I)
   24 CONTINUE
      WRITE(3,102)
   25 WRITE(3,100) IDENT
      WRITE(3,181)
  181 FORMAT(50H0    T         P     DELTA GEF                    )
      DO 89  I=1,NUMX
      WRITE(3,105) T(I),P(I),FF(I)
   89 CONTINUE
  105 FORMAT (1X,F8.2,2X,E12.4,2X,F8.3)
      WRITE(3,102)
C     THIRD LAW CALCULATION OF DELTA H
      SUM=0.0
      SUM1=0.0
      DO 6 I=1,NUMX
      PP(I)=ALOG10(P(I))
      PPP(I)=ALOG(P(I))
      PD(I)=-R*T(I)*PPP(I)
      B(I)=T(I)*FF(I)
      DHST(I)=PD(I)+B(I)
      SUM=SUM+DHST(I)
    6 CONTINUE
      ANUMX=NUMX
      DHAV=SUM/ANUMX
      NUMX=ANUMX
      DO 7 I=1,NUMX
      DHD(I)=DHST(I)-DHAV
      SUM1=SUM1+ABS(DHD(I))
    7 CONTINUE
      ANUMX=NUMX
      DEV=SUM1/ANUMX
      NUMX=ANUMX
```

```
C     PRINT RESULTS
      WRITE (3,106)
  106 FORMAT (50HOCALCULATION RESULTS                                   )
      WRITE (3,107)
  107 FORMAT (100H       T           P           LN P        LOG P    -RT
     1 LN P  -T(DELTAFEF)  DELTA HST       DEV                          )
      WRITE (3,104)
      L=0
      DO 8 I=1,NUMX
      WRITE (3,108) T(I),P(I),PPP(I),PP(I),PD(I),B(I),DHST(I) ,DHD(I)
  108 FORMAT (1X,F8.2,2X,E12.4,2X,F10.5,2X,F10.5,2X,F10.2,2X,F10.2,2X,F1
     10.2,2X,F8.2)
C     PUNCH OPTION
      IF (K-1) 8,9,9
C     TO AVOID PUNCHING IDENT BEFORE EACH VALUE
    9 IF (L .EQ. 0) GO TO 12
      GO TO 11
C     PUNCH RESULTS
   12 WRITE (14,100) IDENT
   11 WRITE(14,101) T(I),P(I),FF(I),DHST(I),DHD(I)
      L=1
    8 CONTINUE
      WRITE(3,109) DHAV
  109 FORMAT (20HOAVERAGE DELTA H298=  F10.2)
      WRITE(3,112) DEV
  112 FORMAT (19HOAVERAGE DEVIATION=      F6.2)
CALCULATION OF STANDARD DEVIATION
      SUM2 = 0.0
      DO 13  I=1,NUMX
      RES(I) = DHD(I)**2
      SUM2=SUM2+RES(I)
   13 CONTINUE
      ANUMX=NUMX
      SD = SQRT(SUM2/(ANUMX-1.0))
      NUMX=ANUMX
      WRITE(3,104)
      WRITE(3,113)NUMX,SD
  113 FORMAT(34H STANDARD DEVIATION FOR THE ABOVE      ,I4, 16H MEASUREME
     1NTS =      ,F6.2)
      ENDFILE 14
      GO TO 1
      END
```

```
      SUBROUTINE LAGINT( N, A, F, M, K, X, Y )
      DIMENSION A(1), F(1)
C  DETERMINE INDEX OF TABLE VALUE NEAREST THE INTERPOLATE.
      CALL BAINS( A, N, X, I )
      IF( I.EQ.0 ) GO TO 10
      IF( X.EQ.A(I) ) GO TO 9
C  DETERMINE INDEX OF THE FIRST POINT TO BE USED IN THE INTERPOLATION.
      J = MIN0( MAX0( I - M/2, 1 ), N - M )
      IF( M.EQ.1.AND.I.LE.N-1.AND.X.LT.A(I) ) J = J - 1
      CALL LAGRNG( A(J), F(J), M, X, Y )
   10 K = I
      RETURN
    9 Y = F(I)
      GO TO 10
      END
```

```
      SUBROUTINE LAGRNG( A, F, M, X, Y )
      DIMENSION A(11), F(11), DA(11), D(11,11)
C**94
      DATA D(1),D(13),D(25),D(37),D(49),D(61),D(73),D(85),D(97),D(109),
     1D(121)/11*1.0/
C**94
   15 L = M + 1
C  FORM TABLES.
      FN = 1.
      DO 1 KK = 1, L
      DA(KK) = A(KK) - X
      IF( DA(KK).EQ.0.0 ) GO TO 7
      FN = FN*DA(KK)
    1 CONTINUE
      KK = 1
      LL = 2
    2 DO 3 JJ = LL, L
      D(KK,JJ) = DA(KK) - DA(JJ)
    3 D(JJ,KK) = -D(KK,JJ)
      KK = KK + 1
      LL = KK + 1
      IF( LL.LE.L ) GO TO 2
C  COMPUTE INTERPOLATED VALUE.
      Y = 0.
      DO 6 KK = 1, L
      FD = 1.
      DO 5 JJ = 1, L
    5 FD = FD*D(JJ,KK)
    6 Y = Y + ( FN*F(KK) )/( DA(KK)*FD )
      RETURN
    7 Y = F(KK)
      RETURN
      END
```

```
      SUBROUTINE BAINS( SLIST, MM, Z, K )
      DIMENSION SLIST(1)
C     SLIST=TABLE, WHICH MUST BE MONOTONICALLY INCREASING
C     M= NUMBER OF ENTRIES IN SLIST
C     Z=VALUE TO BE FOUND IN TABLE
C     K=SUBSCRIPT OF VALUE IN TABLE NEAREST TO Z
C     PROGRAM RETURNS K = 0 IF Z IF OFF TABLE.
      M=MM
      L1=1
      L2=M
      K=1
      IF(Z-SLIST(1)) 1,15,3
    3 K=M
      IF(SLIST(M)-Z) 1,15,9
    9 K=M/2
      IF(Z-SLIST(K)) 20, 15, 29
   20 L2=K
      GO TO 23
   29 L1=K
   23 IF(L2-L1-1) 1,14,25
   25 M=L1+L2
      GO TO 9
   14 IF(2.*Z-SLIST(L1)-SLIST(L2)) 30,15,31
   30 K=L1
      GO TO 15
   31 K=L2
      GO TO 15
    1 K = 0
   15 RETURN
      END
```

```
      PROGRAM VAPOR (INPUT,OUTPUT,TAPE 2=INPUT,TAPE 3=OUTPUT)
      DIMENSION T(500),P(500),PL(500),PROD(500),DIV(500),IDENT(12),XAXIS
     1(50),YAXIS(50)
      DATA R/1.98717/
C     READ INPUT DATA
   13 READ (2,1) IDENT
    1 FORMAT (12A6)
      IF (IDENT .EQ. 6H          ) CALL EXIT
      READ (2,20) K
   20 FORMAT (I4)
      WRITE (3,8) IDENT
    8 FORMAT (1H1,12A6)
      WRITE (3,3)
    3 FORMAT (50H0  TEMP   PRESSURE    LOG P     RLNP      1/T      )
      WRITE (3,4)
    4 FORMAT (50H  (KELVIN)  (ATM)                                   )
      I=0
    9 I=I+1
      READ (2,5) T(I),P(I)
    5 FORMAT (F8.2,E12.3)
      IF (T(I) .EQ. 0.0) GO TO 10
      GO TO 9
   10 NUMX=I-1
      IF ( K .EQ. 0) GO TO 21
C     CONVERSION LOOP  MILLIMETERS--ATMOSPHERES
      DO 22 I=1,NUMX
   22 P(I)=P(I)/760.0
   21 DO 7 I=1,NUMX
      PL(I)=ALOG10(P(I))
      PROD(I)=R*ALOG(P(I))
      DIV(I)=1./T(I)
C     WRITE OUTPUT
      WRITE (3,6) T(I), P(I),PL(I),PROD(I),DIV(I)
    6 FORMAT (2X,F6.1,2X,E9.3,1X,F10.5,2X,F7.3,2X,E10.3)
    7 CONTINUE
      GO TO 13
      END
```

```
      PROGRAM VPCALC(INPUT,OUTPUT,TAPE 2=INPUT,TAPE 3=OUTPUT)
C     CALCULATION OF VAPOR PRESSURES FROM FREE ENERGY FUNCTIONS
      DIMENSION NAME(12),T(100),FEFD(100),X(100),DGT(100),Z(100),P(100),
     1MZ(100),BZ(100),SZ(100),TQ(100),FQ(100),A(20),TTQ(100),FFQ(100),
     2TQQ(100),FQQ(100),TP(20) ,GAPP(50),TAPP(50),GB(300),TB(300)
      DATA R/1.98717/
   46 READ (2,100) NAME
  100 FORMAT(12A6)
      IF (NAME .EQ. 6H           )STOP
      WRITE(3,101) NAME
  101 FORMAT (1H1,12A6)
      WRITE(3,102)
  102 FORMAT (1H0)
      WRITE(3,103)
  103 FORMAT(50H0                   INPUT DATA                          )
      READ(2,104) DHST
  104 FORMAT(F10.0)
      READ(2,116) TM,TTR
  116 FORMAT(2(2X,F7.2))
      WRITE (3,105) DHST
  105 FORMAT(22H0DELTA H(VAP),(298) = F10.0)
      I=0
    1 I=I+1
      READ(2,106) T(I),FEFD(I)
  106 FORMAT(1X,F7.2,4X,F6.3)
      IF (T(I) .EQ. 0.0) GO TO 2
      GO TO 1
C     N=TOTAL NUMBER OF POINTS
    2 N=I-1
      WRITE (3,107)
  107 FORMAT(8H T DEG K  1X,11H DELTA(GEF))
      WRITE(3,102)
      DO 3  I=1,N
      WRITE(3,1060) T(I),FEFD(I)
 1060 FORMAT(F12.2,2F12.3)
    3 CONTINUE
      WRITE (3,101) NAME
      WRITE (3,110)
  110 FORMAT(99H0         T             DGT                P
     1EXP                                         )
C     CALCULATE DELTA G
      DO 16  I=1,N
      X(I)=-FEFD(I)
      DGT(I)=X(I)*T(I)+DHST
C     Z(I) = ANTILOG P
      Z(I)=-DGT(I)/(R*T(I)*ALOG(10.0))
      IF(Z(I) .LT. 0.0) GO TO 4
      P(I)=10.0**Z(I)
C     MZ = EXPONENT OF P
      MZ(I)=0
      GO TO 15
C     CALCULATION OF P AND MZ IF Z IS NEGATIVE
    4 BZ(I)=AINT(Z(I))
      MZ(I)=INT(BZ(I))-1
      SZ(I)=Z(I)-BZ(I)+1.0
      P(I)=10.0**SZ(I)
```

```
C       PRINT RESULTS
   15  WRITE(3,111) T(I),DGT(I),P(I),MZ(I)
  111  FORMAT(6X,F7.2,9X,F8.0,9X,F7.4,9X,I5)
   16  CONTINUE
       WRITE(3,102)
C       FIND THE TWO TEMPERATURES BETWEEN WHICH DELTA G CHANGES SIGN
       DO 17  I=1,N
       IF (DGT(I) .LE. 0.0) GO TO 18
   17  CONTINUE
   18  ILOW=I-1
       IHIGH=I
C       TLOW AND THIGH ARE THE DESIRED TEMPERATURES.  THEY ARE 100 DEGREES
C       APART.
       TLOW=T(ILOW)
       THIGH=T(IHIGH)
C        GENERATE AN ARRAY OF DELTA(GEF) VALUES IN THE LIQUID
       DO 19  I=1,N
       IF(T(I) .EQ. TM) GO TO 20
   19  CONTINUE
   20  IMELT=I
       IMELT1=IMELT+1
       NLQ=N-IMELT
       IA=1
       DO 21  I=IMELT1,N
       TQ(IA)=T(I)
       FQ(IA)=FEFD(I)
       IA=IA+1
   21  CONTINUE
C       FIND T(BP) TO THE NEAREST 10 DEGREES
       IJ=1
   22  CALL LAGINT(NLQ,TQ,FQ,2,KX,TLOW,F)
       GAPP(IJ) = DHST-TLOW*F
       TAPP(IJ)=TLOW
       TLOW=TLOW+10.0
       IF(TLOW .GT. THIGH) GO TO 23
       IJ=IJ+1
       GO TO 22
C       FIND THE MINIMUM VALUE OF GAPP
   23  GMIN=1.E100
       DO 24  I=1,11
       IF(ABS(GAPP(I)) .GE. GMIN) GO TO 24
       GMIN=GAPP(I)
       IMIN=I
   24  CONTINUE
   25  TMIN=TAPP(IMIN) - 10.
       TMAX=TMIN+20.
       J=1
C       CALCULATE T(BP) TO THE NEAREST 0.1 DEGREE
   26  CALL LAGINT(NLQ,TQ,FQ,2,KX,TMIN,F)
       GB(J)=DHST-TMIN*F
       TB(J) = TMIN
       IF(TB(J) .GT. TMAX) GO TO 27
       J=J+1
       TMIN=TMIN+0.1
       GO TO 26
   27  JFINAL=J
       GBP=1.E100
```

```
      DO 28  I=1,JFINAL
      IF(ABS(GB(I)) .GE. GBP) GO TO 28
      GBP=GB(I)
      IBP=I
   28 CONTINUE
      WRITE(3,112) NAME(1),NAME(2),TB(IBP),GBP
  112 FORMAT(22H THE BOILING POINT OF   2A6,7H IS AT  F7.1,16H DEGREES K
     1ELVIN. /60H THIS VALUE OF T(BP) RESULTS IN A VALUE OF DELTA G EQUA
     2L TO   F4.1,1H.)
      WRITE(3,113) NAME(1),NAME(2)
  113 FORMAT(52H1 CALCULATION OF TEMPERATURES AT EVEN PRESSURES FOR  2A6
     1)
      WRITE(3,114)
  114 FORMAT(50H0  -LOG P       T,DEG K                              )
C     CALCULATION OF T AT EVEN PRESSURES
C     CALCULATE (-R LN P) AT EVEN PRESSURES FROM P = 10**-1 TO 10**-10
      Q=0.0
      DO 49  J=1,10
      Q=Q+1.0
      A(J)=R*ALOG(10.0)*Q
   49 CONTINUE
   50 IF(TTR .EQ. 0.0) GO TO 51
C     FOR A POLYMORPHIC METAL SPLIT THE DELTA(GEF) CURVE INTO 3 PARTS.
      DO 29  I=1,N
      IF(T(I) .EQ. TM) IM=I-1
      IF(T(I) .EQ. TTR) ITR=I-1
   29 CONTINUE
      ITR1=ITR+1
      IM1=IM+1
      DO 30  I=1,ITR
      TC(I)=T(I)
      FC(I)=FEFD(I)
   30 CONTINUE
      J=0
      DO 31  I=ITR1,IM
      J=J+1
      TTQ(J)=T(I)
      FFQ(J)=FEFD(I)
   31 CONTINUE
      JA=J
      J=0
      DO 32  I=IM1,N
      J=J+1
      TQQ(J)=T(I)
      FCQ(J)=FEFD(I)
   32 CONTINUE
      JJA=J
      GO TO 36
C     FOR A NON-POLYMORPHIC METAL SPLIT THE DELTA(GEF) CURVE INTO 2 PARTS.
   51 DO 33  I=1,N
      IF(T(I) .EQ. TM) IM=I-1
   33 CONTINUE
      IM1=IM+1
      DO 34  I=1,IM
      TC(I)=T(I)
      FC(I)=FEFD(I)
   34 CONTINUE
```

```
      J=0
      DO 35  I=IM1,N
      J=J+1
      TTQ(J)=T(I)
      FFQ(J)=FEFD(I)
   35 CONTINUE
      JA=J
C     FIND THE TWO TEMPERATURES BETWEEN WHICH THE DESIRED VALUE OF P LIES
   36 J=1
      Q=0.0
   47 Q=Q+1.0
   37 DO 52  I=1,N
      IF(Q .GE. (-MZ(I))) GO TO 38
   52 CONTINUE
   38 TL=T(I-1)
      TG=T(I)
C     INTERPOLATE
      IF(TTR .NE. 0.0) GO TO 43
   41 IF(TL .GE. TM) CALL LAGINT(JA,TTQ,FFQ,2,KX,TL,G)
      IF(TL .LT. TM) CALL LAGINT(IM,TQ,FQ,2,KX,TL,G)
C     CALCULATE T
      TEMP=DHST/(A(J)+G)
C     TEST TO SEE IF THE CALCULATED VALUE OF T IS WITHIN 1 DEGREE OF THE
C     INTERPOLATED VALUE
      IF(ABS(TEMP-TL) .LE. 1.0) GO TO 40
C     INCREMENT THE INTERPOLATED VALUE AND REPEAT THE PROCESS
      TL=TL+0.1
      GO TO 41
   40 TP(J)=TEMP
      J=J+1
      IF(J .EQ. 11) GO TO 42
      GO TO 47
   43 IF(TL .LE. TTR) CALL LAGINT(ITR,TQ,FQ,2,KX,TL,G)
      IF(TL .GT. TTR .AND. TL .LE. TM) CALL LAGINT(JA,TTQ,FFQ,2,KX,TL,G)
      IF(TL .GT. TM) CALL LAGINT(JJA,TQQ,FQQ,2,KX,TL,G)
      TEMP=DHST/(A(J)+G)
      IF(ABS(TEMP-TL) .LE. 1.0) GO TO 44
      TL=TL+0.1
      GO TO 43
   44 TP(J)=TEMP
      J=J+1
      IF(J .EQ. 11) GO TO 42
      GO TO 47
C     WRITE RESULTS
   42 DO 45  J=1,10
      WRITE(3,115) J,TP(J)
  115 FORMAT(4X,I3,8X,F6.1)
   45 CONTINUE
      GO TO 46
      END
```

```
      PROGRAM B3LAW (INPUT,OUTPUT,TAPE 2=INPUT,TAPE 3=OUTPUT)
C    THIRD LAW CALCULATION OF DELTA H FOR TWO VAPOR SPECIES I.E. SELENIUM
      DIMENSION IDENT (12),T(50),P2(50),P6(50),FEF2(50),FEF6(50),PSQ2(50
     1),PRT6(50),PSQ2L(50),PRT6L(50),PP(50),PPP(50),A(50),B(50),C(50),D(
     150),DH2(50),DH6(50),BB(50),DH2D(50),DH6D(50)
      DATA R/1.98717/
C     READ INPUT DATA
    1 READ (2,2) IDENT
    2 FORMAT (12A6)
      I=0
    3 I=I+1
      READ (2,4) T(I),P2(I),P6(I),FEF2(I),FEF6(I)
    4 FORMAT (F8.2,2E12.6,2F8.2)
      IF (T(I) .EQ. 0.0) GO TO 5
      GO TO 3
C     NUMX=NJMBER OF DATA POINTS
    5 NUMX=I-1
C     WRITE TABLE HEADINGS
      WRITE (3,6) IDENT
    6 FORMAT (1H1,12A6)
      WRITE (3,7)
    7 FORMAT (50H0INPUT DATA... T,P2,P6,FEF2,FEF6                        )
      WRITE (3,12)
   12 FORMAT (1H0)
C     WRITE INPUT DATA
      DO 8 I=1,NUMX
      WRITE (3,9) T(I),P2(I),P6(I),FEF2(I),FEF6(I)
    9 FORMAT (1X,F8.2,2E12.6,2F8.2)
    8 CONTINJE
C     3RD LAW CALCULATION
      SUM=0.0
      SUM1=0.0
      DO 10 I=1,NUMX
      PSQ2(I)=SQRT(P2(I))
      BB(I)=(1./6.)*ALOG(P6(I))
      PRT6(I)=EXP(BB(I))
      FEF2(I)=-FEF2(I)
      FEF6(I)=-FEF6(I)
      PSQ2L(I)=ALOG(PSQ2(I))
      PRT6L(I)=ALOG(PRT6(I))
      PP(I)=ALOG10(PSQ2(I))
      PPP(I)=ALOG10(PRT6(I))
      A(I)=-R*T(I)*PSQ2L(I)
      B(I)=-R*T(I)*PRT6L(I)
      C(I)=-T(I)*FEF2(I)
      D(I)=-T(I)*FEF6(I)
C     CALCULATE DELTA H VALUES
      DH2(I)=A(I)+C(I)
      SUM=SUM+DH2(I)
      DH6(I)=B(I)+D(I)
      SUM1=SJM1+DH6(I)
   10 CONTINJE
C     TO AVERAGE THE DELTA H298 VALUES FOR EACH TEMPERATURE.
      SUM2=0.0
      SUM3=0.0
      DO 17 I=1,NUMX
      ANJMX=NUMX
```

```
      DH2AV=SUM/ANUMX
      DH5AV=SUM1/ANUMX
      DH2D(I)=DH2(I)-DH2AV
      SUM2=SUM2+ABS(DH2D(I))
      DH5D(I)=DH6(I)-DH6AV
      SUM3=SUM3+ABS(DH6D(I))
   17 CONTINUE
      ANUMX=NUMX
      DEV2=SUM2/ANUMX
      DEV6=SUM3/ANUMX
C     WRITE RESULTS
      WRITE (3,6) IDENT
      WRITE (3,11)
   11 FORMAT(114HODIATOMIC RESULTS..T,P2,SQRT P2,LN(SQRT P2),LOG(SQRT P2
     1),-RT LN(SQRT P2),DELTA(FEF),-T*DELTA(FEF),DELTA H298,DIFF        )
      WRITE (3,12)
      DO 13 I=1,NUMX
      WRITE (3,14) T(I),P2(I),PSQ2(I),PSQ2L(I),PP(I),A(I),FEF2(I),C(I),D
     1H2(I),DH2D(I)
   14 FORMAT (F8.2,2X,2E12.4,2F10.5,2F8.2,2X,F8.2,2X,F8.2,2X,F8.2)
   13 CONTINUE
      WRITE (3,18) DH2AV,DEV2
   18 FORMAT (24HO AVERAGE DELTA H298 IS F8.2,26H  THE AVERAGE DEVIATION
     1 IS,F8.2)
      WRITE (3,6) IDENT
      WRITE (3,15)
   15 FORMAT (114HO HEXATOMIC RESULTS, T,P6,ROOT P6, LN ROOT P6,LOG ROOT
     1P6,-RT LN (RT P6), DELTA(FEF),-T*DELTA(FEF),DELTA H298,DIFF       )
      WRITE (3,12)
      DO 16 I=1,NUMX
      WRITE (3,14) T(I),P6(I),PRT6(I),PRT6L(I),PPP(I),B(I),FEF6(I),D(I),
     1DH5(I),DH6D(I)
   15 CONTINUE
      GO TO 1
      CALL EXIT
      END
```

```
      PROGRAM ACTIV (INPUT,OUTPUT,TAPE 2=INPUT,TAPE3=OUTPUT,TAPE 14)
CALCULATION OF ACTIVITIES FROM VAPOR PRESSURE MEASUREMENTS
      DIMENSION IDENT(12),TT(100),FEFD(100),T(100),P(100),TQ(100),FEFQ(1
     100),TR(100),FEFR(100),FF(100),A(100),PP(100),ACT(100),G(100),DF(10
     20),DFX(100)
      DATA R/1.98717/
C     READ INPUT DATA
    8 READ (2,100) IDENT
  100 FORMAT(12A6)
      IF(IDENT .EQ. 6H           )STOP
      READ (2,101) DHST
  101 FORMAT (F10.1)
      READ (2,102) X
  102 FORMAT(F10.7)
      READ(2,110) K,KK,M
  110 FORMAT(3I4)
      IF (M .EQ. 1) GO TO 151
      I=0
    1 I=I+1
      READ(2,114) TT(I),FEFD(I)
  114 FORMAT(1X,F7.2,4X,F6.3)
      IF(I .EQ. 1) GO TO 10
      IF(TT(I) .EQ. TT(I-1)) GO TO 2
   10 IF(TT(I) .EQ. 0.0) GO TO 11
      GO TO 1
    2 TM=TT(I)
      IMELT=I-1
      GO TO 10
   11 NN=I-1
  151 I=0
   12 I=I+1
      READ(2,103) T(I),P(I)
  103 FORMAT (F8.2,E12.4,F7.3)
      IF(T(I) .EQ. 0.0) GO TO 13
      GO TO 12
   13 N=I-1
      IF(KK .EQ. 0) GO TO 14
C     CONVERSION FROM MILLIMETERS TO ATMOSPHERES
      DO 15  I=1,N
      P(I)=P(I)/760.
   15 CONTINUE
   14 DO 16  I=1,IMELT
      TQ(I)=TT(I)
      FEFQ(I)=FEFD(I)
   16 CONTINUE
C     DIVIDE THE DELTA(GEF) CURVE INTO TWO PARTS AT THE MELTING POINT
      IMELT1=IMELT+1
      IA=1
      NA=NN-IMELT
      DO 17  I=IMELT1,NN
      TR(IA)=TT(I)
      FEFR(IA)=FEFD(I)
      IA=IA+1
   17 CONTINUE
C     INTERPOLATE IN THE PROPER PART OF THE CURVE TO OBTAIN THE VALUE OF
C     DELTA(GEF) AT THE MEASURED TEMPERATURE
```

```
      DO 20  I=1,N
      TA=T(I)
      IF (TA .GT. TM) GO TO 200
      IF (TA .LE. TM) GO TO 300
  300 CALL LAGINT(IMELT,TQ,FEFQ,2,KX,TA,F)
      FF(I)=F
      GO TO 20
  200 CALL LAGINT(NA,TR,FEFR,2,KX,TA,F)
      FF(I)=F
   20 CONTINJE
   21 WRITE(3,108)
  108 FORMAT (1H1)
C     WRITE INPUT DATA
      WRITE(3,100)IDENT
      WRITE(3,104)
  104 FORMAT (1H0)
      WRITE(3,105)
  105 FORMAT(50H0          INPUT DATA                                      )
      IF(M .EQ. 1) GO TO 39
      WRITE(3,115)
  115 FORMAT(8H T DEG K,1X,11H DELTA(GEF)  )
      WRITE(3,104)
      DO 22 I=1,NN
      WRITE(3,114) TT(I),FEFD(I)
   22 CONTINJE
      WRITE(3,108)
   39 WRITE(3,104)
      WRITE (3,113) DHST, X
  113 FORMAT(15H0DELTA H298 =  ,F10.1, 10H       X =       ,F10.7)
      WRITE (3,106)
  106 FORMAT (50H0    T DEG K      P (ALLOY)      DELTA GEF             )
      WRITE (3,104)
      DO 3  I=1,N
      WRITE(3,107) T(I),P(I),FF(I)
  107 FORMAT(4X,F7.2,5X,E12.4,5X,F7.3)
    3 CONTINJE
      WRITE (3,104)
C     WRITE TABLE HEADINGS
      WRITE (3,112)
  112 FORMAT(90H0  T DEG K      P (ALLOY)        P (METAL)    ACTIVITY   G
     1AMMA       DGB         DGBXS                                     )
      WRITE (3,104)
      L=0
C     CALCULATE ACTIVITIES, ACTIVITY COEFFICIENTS, AND GIBBS ENERGIES
      DO 4  I=1,N
      A(I)=(FF(I)-DHST/T(I))/(R*ALOG(10.0))
      PP(I)=10.0**A(I)
      ACT(I) = P(I)/PP(I)
      G(I) = ACT(I)/X
      DF(I) = R*T(I)*ALOG(ACT(I))
      DFX(I) = R*T(I)*ALOG(G(I))
C     WRITE RESULTS
      WRITE (3,109) T(I),P(I),PP(I),ACT(I),G(I),DF(I),DFX(I)
  109 FORMAT (2X,F7.2,4X,E12.4,4X,E12.4,4X,F5.3,4X,F7.3,4X,F9.1,4X,F9.1)
C     PUNCH OPTION
      IF (K-1) 4,5,5
C     NEXT STATEMENT AVOIDS WRITING THE TITLE BEFORE EACH VALUE OF T, P, ETC.
```

```
  5 IF (L .EQ. 0) GO TO 6
    GO TO 7
  6 WRITE (14,100) IDENT
  7 WRITE(14,111) T(I),P(I),PP(I),ACT(I),G(I),DF(I),DFX(I)
111   FORMAT(1X,F7.2,1X,E12.4,1X, E12.4,1X,F5.3,1X,F7.3,1X,F9.1,1X,F9.
   11)
    L=1
  4 CONTINUE
    ENDFILE 14
    GO TO 8
    END
```

```
      PROGRAM ALLOY (INPUT,OUTPUT,TAPE 2=INPUT,TAPE 3=OUTPUT)
      DIMENSION IDENT(12),NAME(2),ALPH1(30),DGX1(30),AA(30),GAM1(30),
     1ACT1(30),DGX2(30),           GAM2(30),ACT2(30),XAREA(30),X1(30),X2(30
     2),DH(30),DH1(30),DH2(30),S(30),SX(30),DS1(30),DS2(30),DSX1(30),
     3DSX2(30),GX(30),G(30),DG1(30),DG2(30),B(30),YAREA(30),ALPH(30),
     4Q(30)
      COMMON XAREA,X1,X2,DH,DH1,DH2,S,SX,DS1,DS2,DSX1,DSX2,GX,G,DG1,
     1DGX1,DG2,DGX2,B
      DATA R/1.98717/
C     READ INPUT DATA
   13 READ(2,100) IDENT
  100 FORMAT(12A6)
      READ(2,131) NAME(1),NAME(2),NSTAT
  131 FORMAT(2A3,A6)
      READ (2,101) T,L
  101 FORMAT(F10.0,I10)
      DO 1  I=1,15,7
      READ (2,102) ALPH1(I), ALPH1(I+1), ALPH1(I+2),ALPH1(I+3),ALPH1(I+4
     1),ALPH1(I+5) , ALPH1(I+6)
    1 CONTINUE
  102 FORMAT(7F10.1)
C     CALCULATE X VALUES IN INTERVALS OF 0.05 FROM X = 0 TO X = 1
      X1(1) = 0.0
      DO 2  I = 2,21
      X1(I) = X1(I-1) + 0.05
    2 CONTINUE
      DO 3 I = 1,15,7
      READ (2,102) B(I), B(I+1), B(I+2), B(I+3), B(I+4), B(I+5), B(I+6)
    3 CONTINUE
C     WRITE TABLE HEADINGS
      WRITE(3,103)
  103 FORMAT(1H1)
      WRITE(3,104) IDENT
  104 FORMAT(12A6)
      WRITE(3,105)
  105 FORMAT(1H0)
      WRITE(3,106) NAME(1),NAME(2),NSTAT,T
  106 FORMAT(15H INPUT DATA FOR       ,A3,2H -,A3,15H ALLOYS IN THE    ,
     1A6,9H STATE AT    ,F6.0,15H DEGREES KELVIN               )
      WRITE(3,105)
      WRITE(3,107)
  107 FORMAT(50H   X1             ALPHA                B             X2   )
      WRITE(3,105)
C     WRITE INPUT DATA
      DO 4  I=1,21
      X2(I) = 1.0-X1(I)
    4 WRITE (3,108)  X1(I), ALPH1(I), B(I)    ,X2(I)
  108 FORMAT(1X,F4.2,8X,F8.0, 6X,F12.4,6X,F4.2)
      WRITE(3,105)
      WRITE (3,116)      NAME(1),NAME(2)
  116 FORMAT (89H NOTE.  'FIRST' ELEMENT IS THE ONE GIVEN FIRST IN THE T
     1ABLE HEADINGS, I.E., ELEMENT 1 IS           ,A3,17H AND ELEMENT 2 IS
     2,A3)
      WRITE(3,105)
C     L IS A CONTROL NUMBER TELLING WHICH TYPE OF DATA IS AVAILABLE
      IF(L-1) 41,42,43
```

```
   41 WRITE(3,128) L
  128 FORMAT(5H L =    ,I1,19H   Q PLOT DATA USED                    )
      GO TO 44
   42 WRITE(3,129) L
  129 FORMAT(5H L =    ,I1,22H   BETA PLOT DATA USED                   )
      GO TO 44
   43 WRITE(3,130) L
  130 FORMAT(5H L =         ,I1,24H   FREE EVERGY DATA ONLY          )
   44 WRITE(3,105)
      WRITE(3,132)
  132 FORMAT(40H B IN ABOVE TABLE IS EITHER Q OR BETA.            )
C     AREA INTEGRATES UNDER THE SELECTED CURVE
      CALL AREA(ALPH1,21)
C     CALCULATE TABLES
      DO 5  I=1,21,2
      DGX1(I) = ALPH1(I)*X2(I)**2
      IF(X1(I) - 0.05)  52,51,51
   51 DG1(I)=DGX1(I) + R*T*ALOG(X1(I))
   52 AA(I) = DGX1(I)/(R*T)
      GAM1(I)=EXP(AA(I))
      ACT1(I) = X1(I)*GAM1(I)
      DGX2(I) = -(ALPH1(I)*X1(I)*X2(I))+XAREA(I)
      IF(X2(I)-0.05) 54,53,53
   53 DG2(I) = DGX2(I) + R*T*ALOG (X2(I))
   54 AA(I) = DGX2(I)/(R*T)
      GAM2(I)=EXP(AA(I))
      ACT2(I) = X2(I)*GAM2(I)
      GX(I) = X1(I)*DGX1(I)+ X2(I)*DGX2(I)
    5 CONTINJE
      DO 6  I=3,19,2
    6 G(I) = X1(I)*DG1(I)+X2(I)*DG2(I)
      IF (L-1) 7,8,9
C     QPLOT IS USED FOR QPLOT DATA (HEATS)
    7 CALL QPLOT(B,T,21)
      GO TO 9
C     BETA IS USED FOR BETA-PLOT DATA (ENTROPIES)
    8 CALL BETA (B,T)
    9 IF (L-2) 15,10,15
C     WRITE TABLE HEADINGS FOR INTEGRAL QUANTITIES (NO HEAT DATA)
   10 WRITE (3,103)
      WRITE(3,104) IDENT
      WRITE(3,105)
      WRITE(3,105)
      WRITE (3,109)
  109 FORMAT(35H                              TABLE 1               )
      WRITE (3,110) NSTAT,T
  110 FORMAT(25H0INTEGRAL QUANTITIES FOR  ,A6,10H ALLOYS AT, F6.0,15H DE
     1GREES KELVIN                    )
      WRITE(3,105)
      WRITE(3,111)
  111 FORMAT (34H  X2         DELTA G        DELTA GXS            )
      WRITE(3,105)
      DO 11 I=3,19,2
      J=22-I
C     WRITE TABLE FOR INTEGRAL QUANTITIES (NO HEAT DATA)
   11  WRITE (3,112) X2(J),G(J),GX(J)
  112 FORMAT(1X,F3.1,6X,F8.0,6X,F8.0)
```

```
      WRITE(3,105)
      WRITE(3,105)
C     WRITE PARTIAL MOLAR QUANTITIES TABLE HEADINGS (NO HEAT DATA)
      WRITE (3,117)
  117 FORMAT(35H                              TABLE 2            )
      WRITE(3,105)
      WRITE (3,113) NSTAT,T
  113 FORMAT (29H PARTIAL MOLAR QUANTITIES FOR, A6,10H ALLOYS AT, F6.0,
     115H DEGREES KELVIN                          )
      WRITE(3,105)
      WRITE (3,114) NAME(1),NAME(2)
  114 FORMAT(20X,A3,47H  COMPONENT                                      ,A
     13,11H  COMPONENT)
      WRITE(3,105)
      WRITE(3,115)
  115 FORMAT (102H X2       ACTIV      GAMMA    DELTA GBAR    DELTA GBARXS   /
     1/  ACTIV      GAMMA      DELTA GBAR    DELTA GBARXS              )
      WRITE(3,105)
      XX=0.00
      XA=1.00
C     WRITE FIRST ROW OF PARTIAL MOLAR TABLE (X = 0)
      I=21
      WRITE(3,118)XX,XA,XA,XX,XX,XX,GAM2(I ),DGX2(I )
  118 FORMAT(1X,F3.1,3X,F7.5,3X,F7.5,3X,F7.0,7X,F7.0,8X,F7.5,3X,F7.5,3X,
     18H -INFIN.,6X,F7.0)
      DO 12 I=3,19,2
C     WRITE PARTIAL MOLAR TABLE
      J=22-I
   12 WRITE (3,119) X2(J),ACT1(J),GAM1(J),DG1(J),DGX1(J),ACT2(J),GAM2(J)
     1,DG2(J),DGX2(J)
  119 FORMAT(1X,F3.1,3X,F7.5,3X,F7.5,3X,F7.0,7X,F7.0,8X,F7.5,3X,F7.5,3X,
     1F7.0,7X,F7.0)
C     WRITE LAST LINE OF PARTIAL MOLAR TABLE (X = 1)
      I=1
      WRITE(3,120) XA,XX,GAM1( I),DGX1( I),XA,XA,XX,XX
  120 FORMAT(1X,F3.1,3X,F7.5,3X,F7.5,3X, 8H -INFIN.,6X,F7.0,8X,F7.5,3X,
     1F7.5,3X,F7.0,7X,F7.0)
      GO TO 13
C     WRITE INTEGRAL TABLE HEADINGS IF HEAT OR BETA DATA ARE AVAILABLE
   15 WRITE(3,103)
      WRITE(3,104) IDENT
      WRITE(3,105)
      WRITE(3,109)
      WRITE(3,105)
      WRITE(3,110)NSTAT,T
      WRITE(3,105)
      WRITE(3,121)
  121 FORMAT(71H  X2       DELTA G       DELTA H       DELTA S       DELTA GX
     1S       DELTA SXS                 )
      WRITE(3,105)
      DO 16  I=3,19,2
C     WRITE INTEGRAL TABLE IF HEAT OR BETA DATA ARE AVAILABLE
      J=22-I
   16 WRITE(3,122) X2(J),G(J),DH(J),S(J),GX(J),SX(J)
  122 FORMAT(1X,F3.1,4X,F8.0,5X,F8.0,5X,F7.4,7X,F8.0,7X,F7.4)
C     WRITE PARTIAL MOLAR TABLE HEADINGS FOR COMPONENT 1 IF HEAT OR BETA
C     DATA ARE AVAILABLE
```

```
      WRITE(3,103)
      WRITE(3,117)
      WRITE(3,105)
      WRITE(3,113) NSTAT,T
      WRITE(3,105)
      WRITE(3,123) NAME(1)
  123 FORMAT(27X,A3,11H  COMPONENT)
      WRITE(3,105)
      WRITE(3,124)
  124 FORMAT(94H  X1    ACTIV     GAMMA    DELTA GBAR  DELTA GBARXS    DE
     1LTA HBAR    DELTA SBAR   DELTA SBARXS                   )
      WRITE(3,105)
      XX=0.0
      XA=1.00
C     WRITE PARTIAL MOLAR TABLE FOR COMPONENT 1 IF HEAT OR BETA DATA ARE
C     AVAILABLE
      WRITE(3,126) XA,XA,XA,XX,XX,XX,XX,XX
  125 FORMAT(1X,F3.1,3X,F7.5,3X,F7.5,3X,8H -INFIN.,5X,F8.0,7X,F8.0,7X,7H
     1 INFIN.,7X,F7.4)
   18 DO 19  I=3,19,2
      J=22-I
   19 WRITE(3,126) X1(J),ACT1(J),GAM1(J),DG1(J),DGX1(J),DH1(J),DS1(J),
     1DSX1(J)
  126 FORMAT(1X,F3.1,3X,F7.5,3X,F7.5,3X,F8.0,5X,F8.0,7X,F8.0,7X,F7.4,7X,
     1F7.4)
      I=1
   17 WRITE(3,125) XX,XX,GAM1(I),DGX1(I),DH1(I),DSX1(I)
      WRITE(3,105)
      WRITE(3,105)
C     WRITE PARTIAL MOLAR TABLE AND HEADINGS FOR COMPONENT 2 IF HEAT OR
C     BETA DATA ARE AVAILABLE
      WRITE(3,123) NAME(2)
      WRITE(3,105)
      WRITE(3,133)
  133 FORMAT(94H  X2    ACTIV     GAMMA    DELTA GBAR  DELTA GBARXS    DE
     1LTA HBAR    DELTA SBAR   DELTA SBARXS                   )
      WRITE(3,105)
      I=21
      WRITE(3,127) XX,XX,GAM2(I ),DGX2(I ),DH2(I ),DSX2(I )
      DO 20  I=3,19,2
      J=22-I
   20 WRITE(3,126) X2(J),ACT2(J),GAM2(J),DG2(J),DGX2(J),DH2(J),DS2(J),
     1DSX2(J)
      XX=0.0
      XA=1.00
      WRITE(3,126) XA,XA,XA,XX,XX,XX,XX,XX
  127 FORMAT(1X,F3.1,3X,F7.5,3X,F7.5,3X,8H -INFIN.,5X,F8.0,7X,F8.0,7X,7H
     1 INFIN.,7X,F7.4)
      GO TO 13
      CALL EXIT
      END
```

```
      SUBROUTINE AREA(ALPH,N)
C     INTEGRATES THE SELECTED CURVE TO OBTAIN THE AREA UNDER IT BY MEANS
C     OF SIMPSON'S RULE
      DIMENSION IDENT(12),NAME(2),ALPH1(30),DGX1(30),AA(30),GAM1(30),
     1ACT1(30),DGX2(30),          GAM2(30),ACT2(30),XAREA(30),X1(30),X2(30
     2),DH(30),DH1(30),DH2(30),S(30),SX(30),DS1(30),DS2(30),DSX1(30),
     3DSX2(30),GX(30),G(30),DG1(30),DG2(30),B(30),YAREA(30),ALPH(30),
     4Q(30)
      COMMON XAREA,X1,X2,DH,DH1,DH2,S,SX,DS1,DS2,DSX1,DSX2,GX,G,DG1,
     1DGX1,DG2,DGX2,B
      SUM=0.0
      XAREA(1) = 0.0
      DO 1  I=3,N,2
      A=I-1
      YAREA(I) = ((X1(I)-X1(1))/(3.*A))*(ALPH(I)+4.*ALPH(I-1)+ALPH(I-2))
      SUM = SUM +YAREA(I)
      XAREA(I) = SUM
    1 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE QPLOT(B,T,N)
C     CALCULATES TABLES IF HEAT DATA ARE AVAILABLE
      DIMENSION IDENT(12),NAME(2),ALPH1(30),DGX1(30),AA(30),GAM1(30),
     1ACT1(30),DGX2(30),        GAM2(30),ACT2(30),XAREA(30),X1(30),X2(30
     2),DH(30),DH1(30),DH2(30),S(30),SX(30),DS1(30),DS2(30),DSX1(30),
     3DSX2(30),GX(30),G(30),DG1(30),DG2(30),B(30),YAREA(30),ALPH(30),
     4Q(30)
      COMMON XAREA,X1,X2,DH,DH1,DH2,S,SX,DS1,DS2,DSX1,DSX2,GX,G,DG1,
     1DGX1,DG2,DGX2
      DO 3 I=1,15,7
    3 READ (2,102) DH1(I),DH1(I+1),DH1(I+2),DH1(I+3),DH1(I+4),DH1(I+5),
     1DH1(I+5)
      DO 1I=1,N,2
      DH(I) = B(I)*X1(I)*X2(I)
      IF (X2(I) -0.05) 5,6,6
    5 DH2(I) = (DH(I)-X1(I)*DH1(I))/X2(I)
    5 S(I) = (DH(I)-G(I))/T
      IF (X1(I)-0.05) 7,8,8
    7 DH1(1) = B(1)
    8 IF(X2(I) -0.05) 9,10,10
    9 DH2(21) = B(21)
   10 IF(X1(I) -0.95) 12,12,11
   11 DH1(21) = 0.0
   12 IF (X2(I) - 0.95) 14,14,13
   13 DH2(1) = 0.0
   14 SX(I) =  (DH(I) -GX(I))/T
      DS1(I) =(DH1(I) - DG1(I))/T
      DSX1(I) = (DH1(I) - DGX1(I))/T
      DS2(I) = (DH2(I) - DG2(I))/T
      DSX2(I) = (DH2(I) - DGX2(I))/T
    1 CONTINJE
  102 FORMAT(7F10.1)
      RETURN
      END
```

```
      SUBROUTINE BETA (B,T)
C     CALCULATES TABLES IF BETA DATA ARE AVAILABLE
      DIMENSION IDENT(12),NAME(2),ALPH1(30),DGX1(30),AA(30),GAM1(30),
     1ACT1(30),DGX2(30),          GAM2(30),ACT2(30),XAREA(30),X1(30),X2(30
     2),DH(30),DH1(30),DH2(30),S(30),SX(30),DS1(30),DS2(30),DSX1(30),
     3DSX2(30),GX(30),G(30),DG1(30),DG2(30),B(30),YAREA(30),ALPH(30),
     4Q(30)
      COMMON XAREA,X1,X2,DH,DH1,DH2,S,SX,DS1,DS2,DSX1,DSX2,GX,G,DG1,
     1DGX1,DG2,DGX2
      DATA R/1.98717/
      CALL AREA(B,21)
      DO 1  I=1,21,2
      DSX1(I) = B(I)*X2(I)**2
      IF(X1(I) - 0.05) 2,3,3
    3 DS1(I) = DSX1(I) -R*ALOG(X1(I))
    2 DSX2(I) = -B(I)*X1(I)*X2(I)+XAREA(I)
      IF(X2(I) -0.05) 4,5,5
    5 DS2(I) = DSX2(I) -R*ALOG(X2(I))
    4 S(I)=X1(I)*DS1(I)+X2(I)*DS2(I)
      SX(I) = X1(I)*DSX1(I)+X2(I)*DSX2(I)
      DH1(I) = DGX1(I)+T*DSX1(I)
      DH2(I) = DGX2(I)+T*DSX2(I)
    1 DH(I) = X1(I)*DH1(I) + X2(I)*DH2(I)
      RETURN
      END
```