ORNL/TM 11923

ornl

OAK RIDGE NATIONAL LABORATORY

MARTIN MARIETTA

Validation of a PC Based Program for Single Stage Absorption Heat Pump

A. Zaltash
M. R. Ally

MANAGED BY
MARTIN MARIETTA ENERGY SYSTEMS, INC.
FOR THE UNITED STATES
DEPARTMENT OF ENERGY

ORNL/TM-11923

VALIDATION OF A PC BASED PROGRAM FOR SINGLE STAGE ABSORPTION HEAT PUMP

A. Zaltash
Energy Division
Oak Ridge National Laboratory

and

M. R. Ally
Energy Division
Oak Ridge National Laboratory

Final Report - September 1991

Prepared by the
Oak Ridge National Laboratory
Oak Ridge, Tennessee 37831
managed by
MARTIN MARIETTA ENERGY SYSTEMS, INC.
for the
U.S. DEPARTMENT OF ENERGY
under contract DE-AC05-84OR21400

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

NOMENCLATURE

A	heat transfer surface area	(ft^2)
COP	coefficient of performance	
COP_{nm}	net measured coefficient of performance	
[LiBr]	concentration of LiBr	(wt%)
[(Li, K, Na) NO_3]	concentration of LiBr	(wt%)
$\Delta\dot{m}_{conc.}$	concentrated solution mass flow rate	(lb/hr)
P	pressure	(psia)
Q	total heat transfer	(Btu/hr)
$[salt]_{abs.\ out}$	concentration of solution leaving the absorber	(wt%)
$T_{abs.\ out}$	temperature of solution leaving the absorber	(°F)
T	temperature	(°F)
U	overall heat transfer coefficient	(Btu/hr-°F-ft^2)
X	operating parameter under consideration	

PREFACE

This document was prepared for the U.S. Department of Energy (DOE) Office of Industrial Technologies (OIT) as part of the OIT Chemical Heat Pump Study.

ABSTRACT

An interactive computer code was developed to evaluate single stage absorption heat pump performance for temperature amplifier and heat amplifier modes using water as the refrigerant. This program performs the cycle calculations for single stage cycles based on the polynomial expressions developed to correlate experimental vapor-liquid-equilibrium (VLE) and specific enthalpy-concentration data for LiBr/water and (Li, K, Na)NO_3/water systems as well as the properties of pure water. The operating parameters obtained by this program were tested against mass and energy balances in documented cases and the results show that the maximum deviation between coefficient of performance (COP) values obtained by this software and the ones previously calculated is less than 3%. In addition, this program was used to study the effect of solution temperature leaving the absorber on the other operating parameters. This type of analysis could be used to improve and optimize cycle design.

EXECUTIVE SUMMARY

This report contains the results obtained by an interactive computer code developed to evaluate single stage absorption heat pump performance for temperature amplifier and heat amplifier modes using water as the refrigerant. This program performs the cycle calculations for single stage cycles based on the polynomial expressions developed to correlate experimental vapor-liquid-equilibrium (VLE) and specific enthalpy-concentration data for LiBr/water and (Li, K, Na)NO_3/water systems as well as the properties of pure water. The operating parameters obtained by this program were tested against mass and energy balances in documented cases and the results show that the maximum deviation between coefficient of performance (COP) values obtained by this software and the ones previously calculated is less than 3%. In addition, this program was used to study the effect of solution temperature leaving the absorber on the other operating parameters such as net measured COP, total heat transfer area (UA), and solution mass flow rate. Results show that the required UA as a function of solution temperature exhibits a minimum which could be used to determine the optimum cycle design.

1. INTRODUCTION

An interactive computer code was developed to evaluate heat pump performance for a particular absorption heat pump configuration (single stage in temperature amplifier and heat amplifier modes) and working fluid (using water as the refrigerant). The performance of a heat pump containing LiBr/water solution or (Li, K, Na)NO_3 /water mixture can be calculated using data contained in the Duhring charts and specific enthalpy versus concentration diagrams which were presented algebraically, by polynomial equations.[1,2,3] In addition, this program uses the polynomial expressions obtained for the properties of pure water (refrigerant).

The results of this program were checked against the cycle analysis (mass and energy balances) given by Suciu and Rebello[4] and Davidson and Erickson.[1] In addition, this program is used to study the effect of the absorber outlet temperature on the other cycle parameters.

2. DESCRIPTION OF THE COMPUTER CODE

The computer code is written in QuickBasic 4.5 for cycle analysis (mass and energy balances) on the liquid-vapor, (Li, K, Na) NO_3 /water and LiBr/water, single stage absorption heat pump. Mass and energy balances can be performed in temperature amplifier (TA) and heat amplifier (HA) modes. The program is divided into three modules (MODULEI.BAS, MODULEII.BAS, and MODULIII.BAS). The listing of these modules is given in Appendix A. The code listing appended to the report is useful to the users interested in studying and/or expanding this computer code to include more fluids with the help of the documentation supplied in the modules (main program and subroutines).

The executable form of this program is HEATPUMP.EXE which can be run on a regular PC with VGA or EGA graphics card; a copy of which is provided with this report for your use. If the appropiate graphics card is not available, this program will bypass the logos provided with this code. These logos include block diagrams of single stage cycle in TA and HA modes. To run the program, type HEATPUMP and hit <ENTER>, the rest should be self explanatory.

This program is written with the assumption that the user is familiar with the single stage absorption heat pump. A brief description of the input parameters for the TA and HA modes is provided to prepare the user for the type of inputs required to perform the desired cycle analysis. The program accepts the inputs in several units by toggling between them, however, the outputs are given in English units. The program gives the user the option of entering the inputs in the individual screen format or in the tabular format. In either case, the user has the option of changing the units of each input and/or going back to change any of the inputs. This program uses the approach temperatures to calculate the required heat transfer area (UA) in each of the units (absorber, condenser, evaporator, desorber, and solution heat exchanger).

The computer code is furnished with a routine which checks each stream to make sure that the solution in the streams is above its corresponding crystallization temperature. If this condition should arise, the program gives the user a warning for possible crystallization and the stream where the crystallization may occur. In this case, the user has to re-evaluate and re-enter the inputs.

3. CYCLE ANALYSIS ON THE LIQUID-VAPOR ABSORPTION HEAT PUMP

This program was tested in the TA mode against the mass and energy balances given by Suciu and Rebello[4] and Davidson and Erickson.[1] Tables 1 and 2 show the results from HEATPUMP program and Suciu and Rebello[4] respectively (70 wt% LiBr/water solution, [LiBr] = 70 wt%). The absorber output capacity is 2 MM Btu/hr (2×10^6 Btu/hr). The deviation between parameters is very small (Tables 1 and 2) except in the pressure and concentration of the solution entering the generator. This is due to a more accurate throttle calculation in the computer code. Throttling is a constant enthalpy (isenthalpic) process with a decrease in temperature and pressure which results in a mixture of liquid solution and water vapor. The coefficient of performance (COP) obtained by this software is 0.483 which compares well with the value of 0.481 calculated by Suciu and Rebello.[4] Tables 3 and 4 show the results obtained from the program and Davidson and Erickson[1] respectively, 90 wt% (Li, K, Na)NO_3 /water mixture. The absorber output capacity is 2.47 MM Btu/hr. The deviation between parameters is very small (Tables 3 and 4). However, the solution mass flow rates obtained by the program are about 33% less than the values obtained by Davidson and Erickson[1] which is due to the concentration difference in the solution leaving the absorber. The COP values obtained by the software and Davidson and Erickson[1] were 0.428 and 0.420 respectively.

This program was tested in the HA mode against the mass and energy balances given by Davidson and Erickson.[1] Tables 5 and 6 show the results from HEATPUMP program and Davidson and Erickson[1] respectively (52.8 wt% LiBr/water solution). The absorber output capacity is 3.75 MM Btu/hr. The deviation between parameters is very small (Tables 5 and 6). The heating COP obtained by this software is 1.78 which compares well with the value of 1.81 calculated by Davidson and Erickson.[1] Tables 7 and 8 show the results obtained from the program and Davidson and Erickson[1] respectively, 73 wt% (Li, K, Na)NO_3 /water mixture. The absorber ouput capacity is 3.75 MM Btu/hr. The heating COP values obtained by the software and Davidson and Erickson[1] were 1.86 and 1.87 respectively.

The effect of solution temperature leaving the absorber was simulated on two base cases (TA and HA) for both LiBr/water solution and (Li, K, Na)NO_3 /water mixture. The summary of base values used for (Li, K, Na)NO_3 /water and LiBr/water cycle calculations is given in Tables 9 and 10. Samples of the input and the output for these base values are given in Appendix B.

For absorption heat pump the heating COP is defined as the ratio of useful heat output divided by the high grade heat input. To account for parasitic pumping power, a net measured COP (COP_{nm}) is defined by Davidson and Erickson[1] as follows:
For temperature amplifier:

$$COP_{nm} = \frac{Q_{absorber}}{(Q_{evaporator} + Q_{desorber} + 4 \times \text{Pumping Power})} \quad (1)$$

For heat amplifier:

$$COP_{nm} = \frac{(Q_{absorber} + Q_{condenser})}{(Q_{desorber} + 4 \times \text{Pumping Power})} \quad (2)$$

The factor of four in these equations is to convert the pumping power to the prime energy, assuming

25% efficiency at the point of use. The pumping power is calculated by applying a 60% efficiency (mechanical efficiency) to the calculated ideal pump work. It should be noted that in the calculation of ideal pump work all frictional losses are neglected (the only friction is that occuring in the pump itself and is accounted for by the mechanical efficiency).

Table 1. Operating parameters in the TA mode for 70 wt% LiBr/water system from HEATPUMP program

Streams	T (°F)	P (psia)	[LiBr] (wt%)	Enthalpy (Btu/lb)	Mass Flow rate (lb/hr)
Steam from gen.	225.0	0.94	0.0	1160.6	1960.1
Water from con.	99.6	0.94	0.0	67.6	1960.1
Steam from evap.	215.0	15.59	0.0	1151.5	1960.1
Solution from abs.	348.0	15.59	66.0	168.3	34371.5
Solution to gen.	230.0	0.94	66.4	119.4	34371.5*
Solution from gen.	225.0	0.94	70.0	125.2	32411.4
Solution to abs.	338.0	15.59	70.0	170.6	32411.4

* A mixture of solution and water vapor.

Table 2. Operating parameters in the TA mode for 70 wt% LiBr/water system from Suciu and Rebello[4]

Streams	T (°F)	P (psia)	[LiBr] (wt%)	Enthalpy (Btu/lb)	Mass Flow rate (lb/hr)
Steam from gen.	230.0	0.95	0.0	1164.0	1970.0
Water from con.	100.0	0.95	0.0	68.0	1970.0
Steam from evap.	215.0	15.50	0.0	1151.5	1970.0
Solution from abs.	348.0	15.50	65.8	167.0	32840.0
Solution to gen.	230.0	1.28	68.6	125.6	32840.0*
Solution from gen.	225.0	0.95	70.0	125.0	30870.0
Solution to abs.	338.0	15.50	70.0	169.0	30870.0

* A mixture of solution and water vapor.

Table 3. Operating parameters in the TA mode for 90 wt% (Li, K, Na)NO_3/water system from HEATPUMP program

Streams	T (°F)	P (psia)	[(Li, K, Na)NO_3] (wt%)	Enthalpy (Btu/lb)	Mass Flow rate (lb/hr)
Steam from gen.	327.9	8.13	0.0	1205.2	3064.1
Water from con.	183.6	8.13	0.0	151.5	3064.1
Steam from evap.	327.9	100.75	0.0	1187.3	3064.1
Solution from abs.	500.0	100.75	85.8	299.6	66336.7
Solution to gen.	329.0	8.13	86.8	237.5	66336.7*
Solution from gen.	327.9	8.13	90.0	241.2	63272.6
Solution to abs.	482.0	100.75	90.0	295.6	63272.6

* A mixture of solution and water vapor.

Table 4. Operating parameters in the TA mode for 90 wt% (Li, K, Na)NO_3/water system from Davidson and Erickson[1]

Streams	T (°F)	P (psia)	[(Li, K, Na)NO_3] (wt%)	Enthalpy (Btu/lb)	Mass Flow rate (lb/hr)
Steam from gen.	327.9	8.72	0.0	1207.9	3243.2
Water from con.	186.8	8.72	0.0	154.9	3243.2
Steam from evap.	327.9	99.96	0.0	1188.9	3243.2
Solution from abs.	500.0	99.96	87.0	302.1	97215.9
Solution to gen.	345.9	8.72	87.0	251.7	97215.9
Solution from gen.	327.9	8.72	90.0	245.7	93964.8
Solution to abs.	482.0	99.96	90.0	297.3	93964.8

Table 5. Operating parameters in the HA mode for 52.8 wt% LiBr/water system from HEATPUMP program

Streams	T (°F)	P (psia)	[LiBr] (wt%)	Enthalpy (Btu/lb)	Mass Flow rate (lb/hr)
Steam from gen.	257.1	9.60	0.0	1172.3	3171.5
Water from con.	191.3	9.60	0.0	159.2	3171.5
Steam from evap.	142.2	3.05	0.0	1122.4	3171.5
Solution from abs.	191.3	3.05	49.8	81.1	55924.0
Solution to gen.	237.3	9.60	49.8	105.5	55924.0
Solution from gen.	257.1	9.60	52.8	115.5	52752.5
Solution to abs.	201.4	3.05	52.9	87.1	52752.5*

* A mixture of solution and water vapor.

Table 6. Operating parameters in the HA mode for 52.8 wt% LiBr/water system from Davidson and Erickson[1]

Streams	T (°F)	P (psia)	[LiBr] (wt%)	Enthalpy (Btu/lb)	Mass Flow rate (lb/hr)
Steam from gen.	257.2	9.61	0.0	1175.2	3520.7
Water from con.	191.3	9.61	0.0	159.1	3520.7
Steam from evap.	142.2	3.06	0.0	1124.0	3520.7
Solution from abs.	191.3	3.06	49.8	51.2	61929.5
Solution to gen.	242.1	9.61	49.8	108.3	61929.5
Solution from gen.	257.2	9.61	52.8	115.8	58440.5
Solution to abs.	201.4	3.06	52.8	87.2	58440.5

Table 7. Operating parameters in the HA mode for 73 wt% (Li, K, Na)NO_3/water system from HEATPUMP program

Streams	T (°F)	P (psia)	[(Li, K, Na)NO_3] (wt%)	Enthalpy (Btu/lb)	Mass Flow rate (lb/hr)
Steam from gen.	259.2	9.60	0.0	1173.2	3511.6
Water from con.	191.3	9.60	0.0	159.2	3511.6
Steam from evap.	142.2	3.05	0.0	1122.4	3511.6
Solution from abs.	192.2	3.05	69.7	176.8	78665.1
Solution to gen.	240.8	9.60	69.7	197.0	78665.1
Solution from gen.	259.2	9.60	73.0	203.7	75153.6
Solution to abs.	202.3	3.05	73.2	180.5	75153.6*

* A mixture of solution and water vapor.

Table 8. Operating parameters in the HA mode for 73 wt% (Li, K, Na)NO_3/water system Davidson and Erickson[1]

Streams	T (°F)	P (psia)	[(Li, K, Na)NO_3] (wt%)	Enthalpy (Btu/lb)	Mass Flow rate (lb/hr)
Steam from gen.	257.0	8.56	0.0	1174.7	3520.7
Water from con.	191.3	8.56	0.0	154.1	3520.7
Steam from evap.	142.2	3.06	0.0	1123.5	3520.7
Solution from abs.	191.3	3.06	70.0	171.0	85638.8
Solution to gen.	246.7	8.56	70.0	197.7	85638.8
Solution from gen.	257.0	8.56	73.0	203.6	82149.8
Solution to abs.	201.4	3.06	73.0	175.8	82149.8

Table 9. Summary of Base Values of Operating Parameters for (Li, K, Na)NO_3 /water system

Parameters	TA	HA
Inputs:		
Concentration of solution leaving the desorber (wt%)	90.0	73.0
Waste heat inlet temperature entering the evaporator (°F)	345.9	151.2
Absorber capacity (MM Btu/hr)	2.47	3.75
Cooling water temperature entering the absorber (°F)	482.0	181.4
Desorber temperature (°F)	329.0	259.2
Waste heat inlet temperature entering the generator (°F)	345.9	277.2
Cooling water temperature entering the condenser (°F)	150.8	181.4
Outputs:		
Generator Pressure (psia)	8.13	9.60
Condenser Pressure (psia)	8.13	9.60
Absorber Pressure (psia)	100.75	3.05
Evaporator Pressure (psia)	100.75	3.05
Net measured COP	0.4383	1.8810
Heating COP	0.4427	1.8836
Concentrated solution flow rate (lb/hr)	47473.51	61678.82
Total (UA) required (Btu/hr-°F)	554144.8	925188.8
Concentration of solution leaving the absorber (wt%)	84.7	69.0
Temperature of solution leaving the absorber (°F)	491.0	190.4

Table 10. Summary of Base Values of Operating Parameters for LiBr/water system

Parameters	TA	HA
Inputs:		
Concentration of solution leaving the desorber (wt%)	70.0	52.8
Waste heat inlet temperature entering the evaporator (°F)	235.0	151.2
Absorber capacity (MM Btu/hr)	2.0	3.75
Cooling water temperature entering the absorber (°F)	338.0	181.4
Desorber temperature (°F)	230.0	257.1
Waste heat inlet temperature entering the generator (°F)	235.0	275.1
Cooling water temperature entering the condenser (°F)	80.0	181.4
Outputs:		
Generator Pressure (psia)	0.94	9.60
Condenser Pressure (psia)	0.94	9.60
Absorber Pressure (psia)	15.59	3.05
Evaporator Pressure (psia)	15.59	3.05
Net measured COP	0.4843	1.7795
Heating COP	0.4849	1.7817
Concentrated solution flow rate (lb/hr)	28448.4	52752.5
Total (UA) required (Btu/hr-°F)	597427.1	856097.4
Concentration of solution leaving the absorber (wt%)	65.5	49.8
Temperature of solution leaving the absorber (°F)	345.0	191.3

The simulated results show that increasing the temperature ($T_{abs.\ out}$) of solution leaving the absorber would decrease the COP and increase the solution flow rate ($\dot{m}_{conc.}$). The total (UA) required for heat transfer decreases with increasing temperature and concentration of solution leaving the absorber. However, a point is reached where the (UA) starts to increase with the temperature and concentration of the solution leaving the absorber (Tables 11, 12, 13, and 14). The results are based on a percent relative deviation from the base values of each parameter. Tables 11 and 13 show the effect of $T_{abs.\ out}$ for (Li, K, Na)NO_3/water system on various parameters ($[salt]_{abs.\ out}$, COP_{nm}, UA, and $\dot{m}_{conc.}$) while Figures 1 and 2 show the effect of $T_{abs.\ out}$ on COP_{nm} and UA. Results show that for (Li, K, Na)NO_3 in the TA mode an increase of 0.9% from the base value of $T_{abs.\ out}$ would cause a decrease in the COP and UA of 1.0% and 8.2% respectively and an increase of 14.6% in the solution flow rate (amount of required salt). The simulated results show that for (Li, K, Na)NO_3 in the HA mode an increase of 0.9% from the base value of $T_{abs.\ out}$ would cause a decrease in the COP

and UA of 1.2% and 3.0% respectively and an increase of 21.8% in the solution flow rate. It should be noted that the concentration of (Li, K, Na)NO_3 solution in the TA mode is close to 90 wt% as compared to about 73 wt% for the HA mode.

The effect of $T_{abs.\ out}$ for LiBr/water system on various parameters ($[salt]_{abs.\ out}$, COP_{nm}, UA, and $\dot{m}_{conc.}$) are shown in Tables 12 and 14 while Figures 3 and 4 show the effect of $T_{abs.\ out}$ on COP_{nm} and UA. Results show that for LiBr in the TA mode an increase of 0.9% from the base value of $T_{abs.\ out}$ would cause a decrease in the COP and UA of 0.4% and 3.0% respectively and an increase of 13.9% in the solution flow rate. The simulated results show that for LiBr in the HA mode an increase of 1.4% from the base value of $T_{abs.\ out}$ would cause a decrease in the COP and UA of 1.8% and 3.6% respectively and an increase of 33.6% in the solution flow rate.

Table 11. Effect of solution temperature leaving the absorber in TA mode for (Li, K, Na)NO_3 /water system

% $\Delta T_{abs.\ out}$	% $\Delta[salt]_{abs.\ out}$	% ΔCOP_{nm}	% $\Delta(UA)_{total}$	% $\Delta\dot{m}_{conc.}$
0.92	0.71	-1.05	-8.21	14.61
1.47	1.06	-1.78	-9.68	25.22
1.65	1.18	-2.05	-9.78	29.14
1.83	1.30	-2.35	-9.70	33.28
2.02	1.53	-2.65	-9.47	37.66
3.67	2.60	-6.21	-0.04	92.32

where: $\% \Delta X = \dfrac{(X - X_{base\ value})}{X_{base\ value}} \times 100\%$

Table 12. Effect of solution temperature leaving the absorber in TA mode for LiBr/water system

% $\Delta T_{abs.\ out}$	% $\Delta[salt]_{abs.\ out}$	% ΔCOP_{nm}	% $\Delta(UA)_{total}$	% $\Delta\dot{m}_{conc.}$
0.58	0.46	-0.29	-2.40	8.93
0.87	0.76	-0.43	-2.99	13.93
1.16	1.07	-0.60	-3.28	19.34
1.30	1.07	-0.70	-3.32	22.22
1.45	1.22	-0.78	-3.27	25.22
2.32	1.98	-1.42	-2.80	46.32

where: $\% \Delta X = \dfrac{(X - X_{base\ value})}{X_{base\ value}} \times 100\%$

Table 13. Effect of solution temperature leaving the absorber in HA mode for (Li, K, Na)NO_3/water system

% $\Delta T_{abs.\ out}$	% $\Delta[salt]_{abs.\ out}$	% ΔCOP_{nm}	% $\Delta(UA)_{total}$	% $\Delta \dot{m}_{conc.}$
0.95	1.01	-1.23	-2.96	21.85
2.36	2.75	-4.14	-5.39	75.70
2.84	3.19	-5.66	-5.65	104.30
3.78	4.20	-10.58	-4.95	197.74
4.73	5.22	-22.75	-0.43	428.31
5.20	5.65	-39.70	7.06	744.24

$$\text{where: } \% \Delta X = \frac{(X - X_{base\ value})}{X_{base\ value}} \times 100\%$$

Table 14. Effect of solution temperature leaving the absorber in HA mode for LiBr/water system

% $\Delta T_{abs.\ out}$	% $\Delta[salt]_{abs.\ out}$	% ΔCOP_{nm}	% $\Delta(UA)_{total}$	% $\Delta \dot{m}_{conc.}$
1.41	1.61	-1.82	-3.55	33.64
1.88	2.21	-2.69	-4.24	49.89
2.35	2.81	-3.77	-4.67	70.26
2.82	3.21	-5.17	-4.82	96.54
3.29	3.82	-7.04	-4.61	131.72
4.23	4.82	-13.67	-2.35	256.23

$$\text{where: } \% \Delta X = \frac{(X - X_{base\ value})}{X_{base\ value}} \times 100\%$$

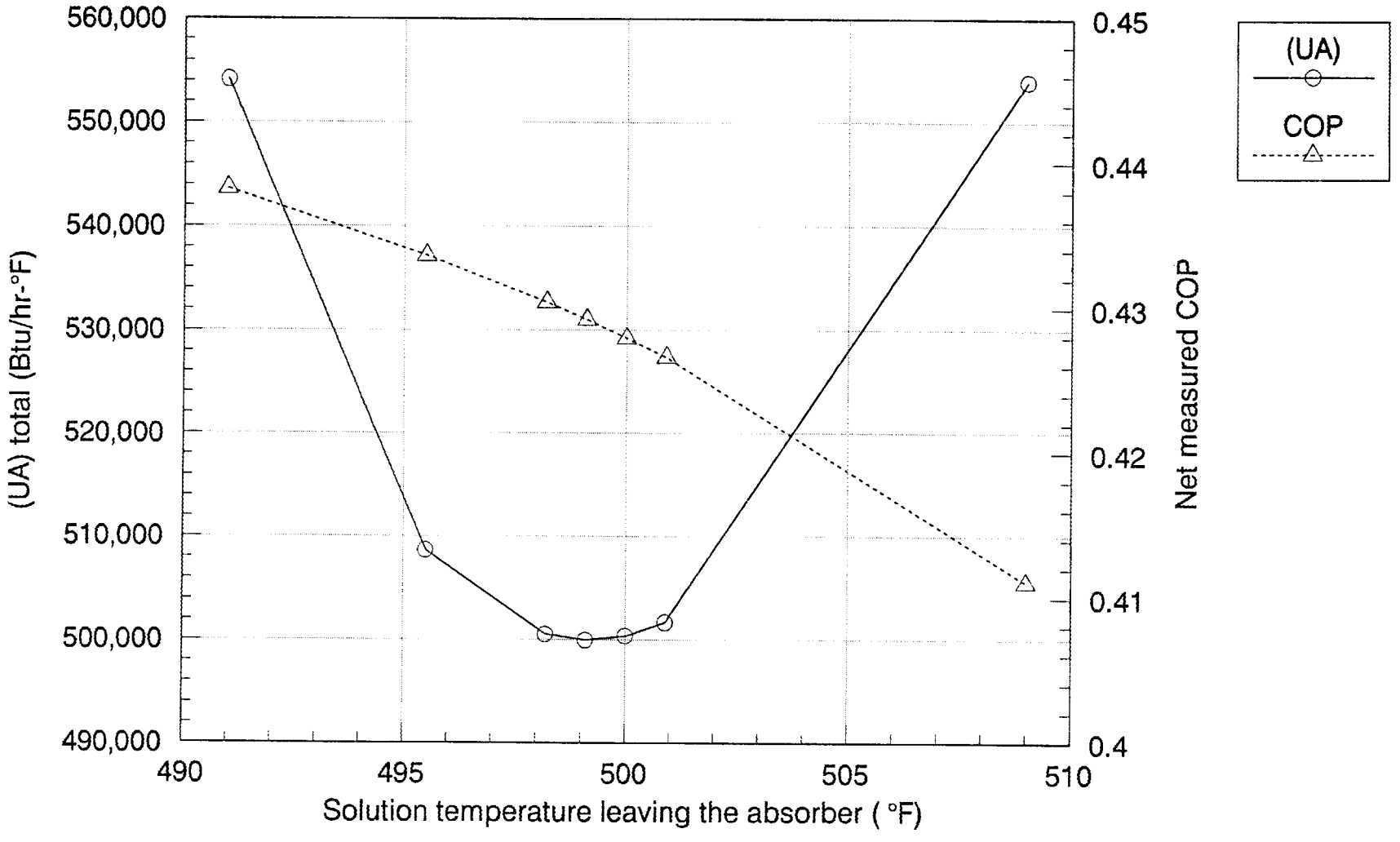

Figure 1. Total (UA) and COP values (TA mode) as a function of solution temperature leaving the absorber for 90 wt% (Li, K, Na)NO_3 mixture leaving the desorber

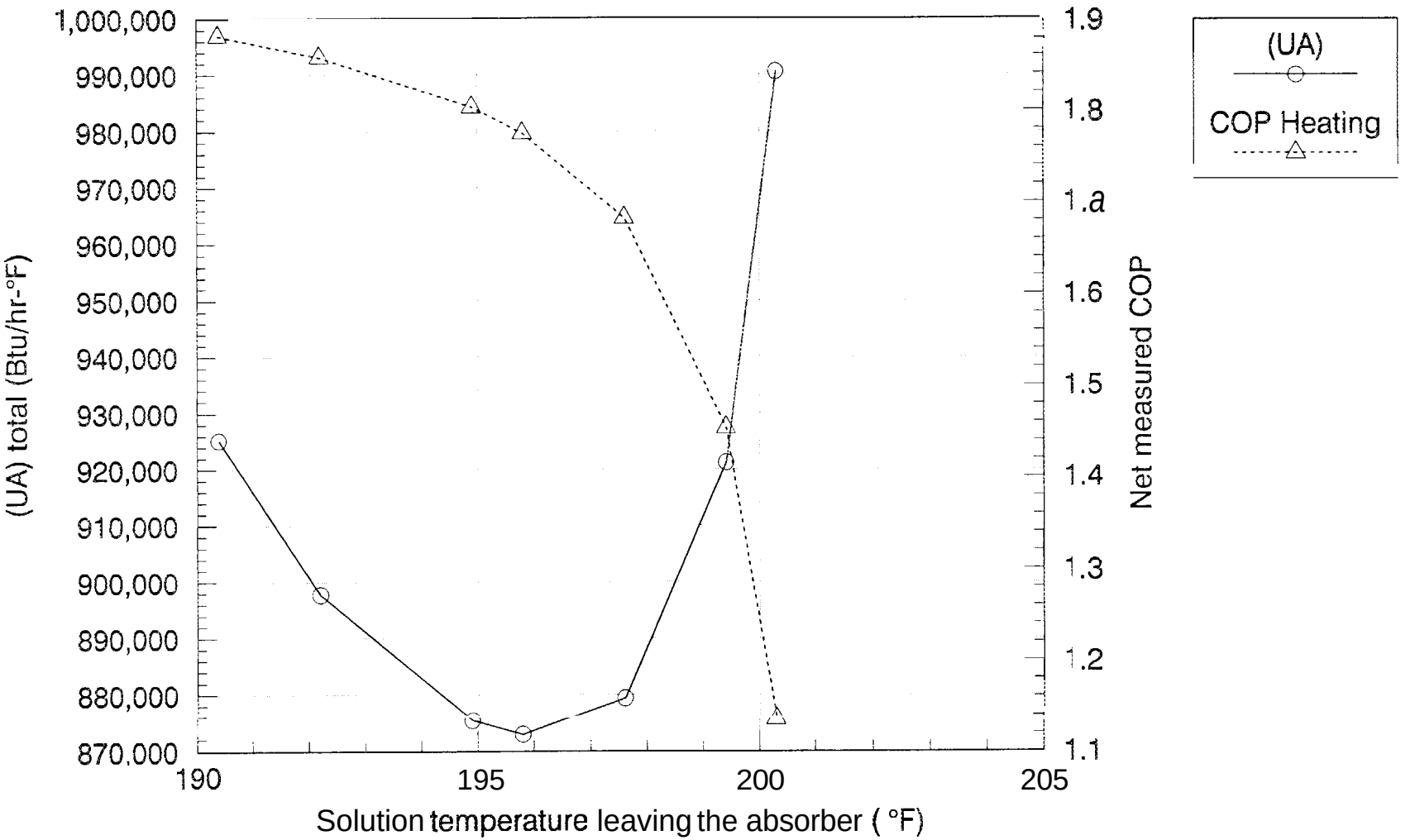

Figure 2. Total (UA) and COP values (HA mode) as a function of solution temperature leaving the absorber for 73 wt% (Li, K, Na)NO_3 mixture leaving the desorber

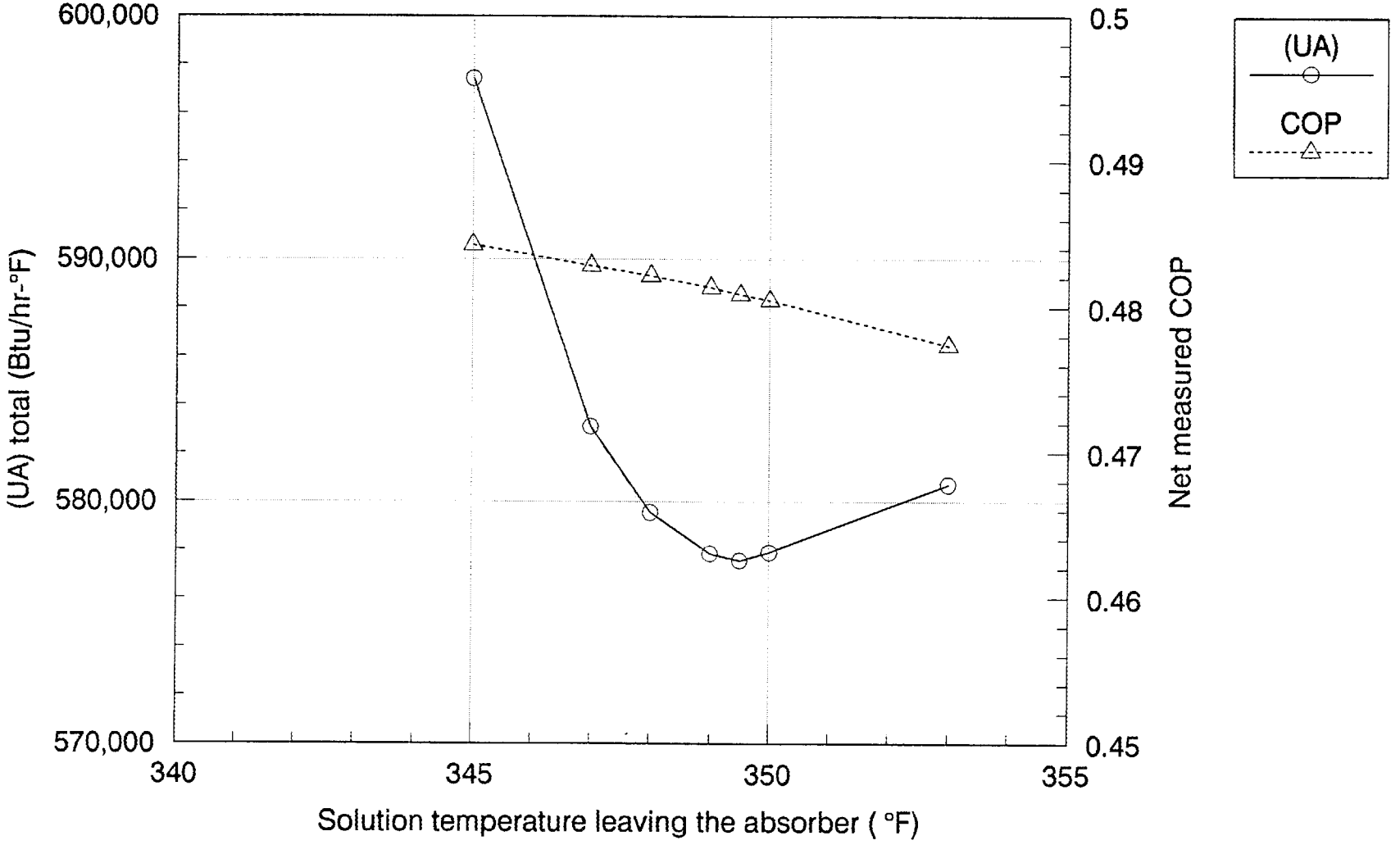

Figure 3. Total (UA) and COP values (TA mode) as a function of solution temperature leaving the absorber for 70 wt% LiBr mixture leaving the desorber

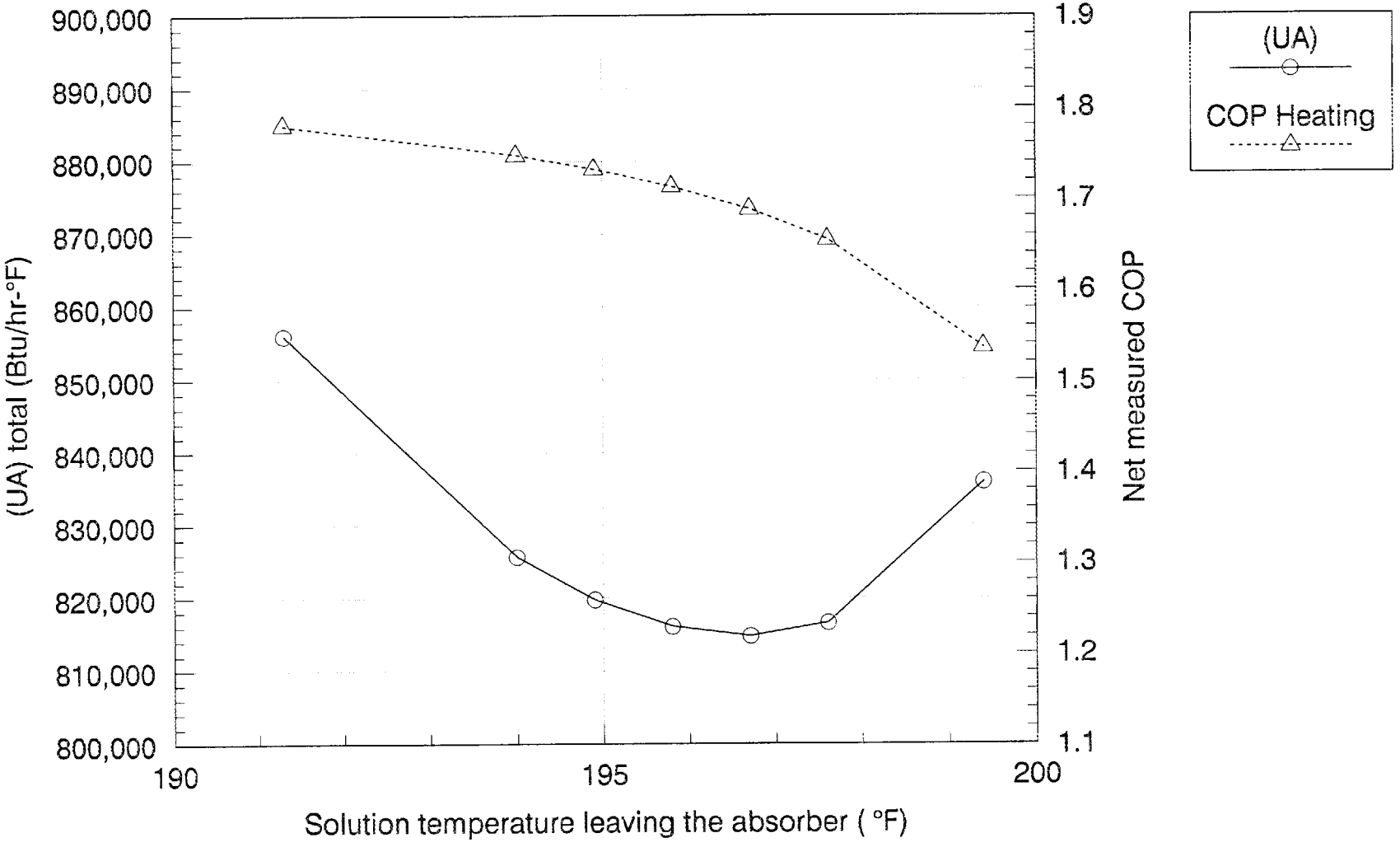

Figure 4. Total (UA) and COP values (HA mode) as a function of solution temperature leaving the absorber for 52.8 wt% LiBr mixture leaving the desorber

4. CONCLUSIONS

An interactive computer code was developed to evaluate single stage absorption heat pump performance for temperature amplifier and heat amplifier modes using water as the refrigerant. This program was tested against documented cases. The results show that the maximum deviation between the coefficient of performance (COP) values obtained by this software and the ones previously calculated is less than 3%. In addition, this program was used to study the effect of solution temperature leaving the absorber on the other operating parameters such as net measured COP, total UA required, and solution mass flow rate. Results show that the required UA as a function of solution temperature exhibits a minimum which could be used to determine the optimum cycle design.

5. REFERENCES

1. W. F. Davidson and D. C. Erickson, *New High Temperature Absorbent for Absorption Heat Pumps*, ORNL/Sub/85-22013/1, Oak Ridge National Laboratory, July 1986.

2. M. R. Ally, *Thermodynamic Properties of Aqueous Ternary Solutions Relevant to Chemical Heat Pumps*, ORNL/TM-10258, Oak Ridge National Laboratory, March 1987.

3. ASHRAE, *1989 Fundamentals*, American Society of Heating, Refrigeration, and Air-Conditioning Engineers, pp. 17.71-17.72.

4. D. F. Suciu and W. J. Rebello, *Assessment of the BCL and ORNL Chemical Heat Pumps*, EGG-CSH-6223, Idaho National Engineering Laboratory, April 1983.

Appendix A

Listing of Computer Code

MODULE1.BAS

```
DECLARE SUB denalk (tf!, wt!, rhosoln!)
DECLARE SUB denlibr (tf!, wt!, rhosoln!)
DECLARE SUB denh2o (tf!, rhoh2o!)
DECLARE SUB condabsin (tcwin!)
DECLARE SUB xhtcalalkHA (hxt!, T!, wtout!, wt!, hvapor!, wtabsout!)
DECLARE SUB xhtcalLiBrHA (hxt!, T!, wtout!, wt!, hvapor!, wtabsout!)
DECLARE SUB mainchoiceii ()
DECLARE SUB tabdata2 (ESTID$, mat$, con$, qout!, tcwin!, twastedesin!, twasteevapin!, tcapp!, teapp!, tdapp!, taapp!, thxapp!)
DECLARE SUB INPTWASTEDES (twastedesin!, twastedesin$, twastedesinUNIT$, trap$, aaaa!)
DECLARE SUB wasteheatevapin (twasteevapin!)
DECLARE SUB wasteheatdesin (twastedesin!)
DECLARE SUB logoTA ()
DECLARE SUB logoHA ()
DECLARE SUB hxtLiBr (tf!, wt, hxt1!)
DECLARE SUB BoostTalk (P!, wt!, tmax!)
DECLARE SUB abswtoutalk (T!, P!, wtout!, wt!)
DECLARE SUB thxoutcalalk (hxt!, wt!, thxout!)
DECLARE SUB xhtcalalk (hxt!, T!, wtout!, wt!, hvapor, wtabsin)
DECLARE SUB hxtalk (T1!, wt!, hxt!)
DECLARE SUB prealk (tf!, wt!, psat!)
DECLARE SUB hgsuper (psat!, Hg!, tsat!, T!, hsuperheat!)
DECLARE SUB TSATH2O (P!, T1!)
DECLARE SUB abswtoutLiBr (T!, P!, wtout!, wt!)
DECLARE SUB thxoutcalLiBr (hxt!, wt!, thxout!)
DECLARE SUB xhtcalLiBr (hxt!, T!, wtout!, wt!, hvapor, wtabsin)
DECLARE SUB preLiBr (tf!, wt!, psat!)
DECLARE SUB BoostTLiBr (P!, wt!, tmax!)
DECLARE SUB hgwater (tf!, Hg!)
DECLARE SUB hfwater (tf!, hf!)
DECLARE SUB preh2o (tf!, P!)
DECLARE SUB TCABSIN (tcain!)
DECLARE SUB Tdesorber (tdeso!)
DECLARE SUB inptcain (tcain!, tcain$, tcainUNIT$, trap$)
DECLARE SUB inptdeso (tdeso!, tdeso$, tdesoUNIT$, trap$)
DECLARE SUB wtabs (wt$, wt!)
DECLARE SUB desTapp (tdapp!)
DECLARE SUB AbsTapp (taapp!)
DECLARE SUB HXTapp (thxapp!)
DECLARE SUB inptcapp (tcapp!, tcapp$, tcappUNIT$, trap$)
DECLARE SUB inpteapp (teapp!, teapp$, teappUNIT$, trap$)
DECLARE SUB inptdapp (tdapp!, tdapp$, tdappUNIT$, trap$)
DECLARE SUB inptaapp (taapp!, taapp$, taappUNIT$, trap$)
DECLARE SUB inpthxapp (thxapp!, thxapp$, thxappUNIT$, trap$)
DECLARE SUB condtapp (tcapp!)
DECLARE SUB evaTapp (teapp!)
DECLARE SUB watecpre (tf!, hf!, Hg!, psat!, T1!)
DECLARE SUB inptcwin (tcwin!, tcwin$, tcwinUNIT$, trap$, ccc!)
DECLARE SUB INPTWASTE (twaste!, twaste$, twasteUNIT$, trap$, aaa!)
DECLARE SUB wasteheatin (twaste!)
DECLARE SUB INPqout (qout!, qout$, qoutUNIT$, trap$, bbb)
DECLARE SUB RUNID (ESTID$, trap$)
DECLARE SUB condh2oin (tcwin)
DECLARE SUB qtout (qout)
DECLARE SUB tabdata (ESTID$, mat$, con$, qout!, tcwin!, twaste, tcapp, teapp, tdapp, taapp, thxapp, tcain, tdeso)
DECLARE SUB MAINMENU ()
DECLARE SUB TITLESCR ()
DECLARE SUB mainchoicei ()
DECLARE SUB mixture ()
DECLARE SUB noise ()
DECLARE SUB border ()
COMMON SHARED frg, bac, escflag
frg = 14: bac = 1
' SETTING THE FUNCTION KEYS TO A VARIABLE - FUNCTION KEY 1 <F1> = "A",
' FUNCTION KEY 2 <F2> = "B" ETC.
                KEY 1, "A"
                KEY 2, "B"
                KEY 3, "C"
                KEY 4, "D"
                KEY 5, "E"
                KEY 6, "F"
                KEY 7, "G"
                KEY 8, "H"
                KEY 9, "I"
```

```
            KEY 10, "J"

PRGSTART:
' ***************TITLE PAGE OF THE PROGRAM***************

CALL TITLESCR

' ************************************************************

' ************************************************************************
' THE PROGRAM REMAINS AT THIS SPOT UNTIL THE USER INPUTS A KEY  <F1> - <F6>
' THIS WILL BE UTILIZED FREQUENTLY AND WILL BE REFERRED TO AS "MAIN MENU"
' IN THE COMMENTING OF THE PROGRAM
' ************************************************************************

TITLE:
        trap$ = INKEY$: IF trap$ = "" GOTO TITLE:
MAINMENU:
CALL MAINMENU: ' moved screen to sub


' ************************RESPONSE NEEDED*********************************

SELMAIN:
    trap$ = INKEY$: IF trap$ = "" GOTO SELMAIN:
    IF trap$ = "A" THEN GOTO mainchoicei:
    IF trap$ = "B" THEN GOTO mainchoiceii:
    IF trap$ = "F" OR ASC(trap$) = 27 THEN GOTO MAINCHOICEVI:
    CALL noise: COLOR 31, bac: LOCATE 23, 12: PRINT " O N L Y   K E Y S   <F1>, <F2> & <F6>   A R E   V A L I D "; : COLOR frg, bac
    GOTO SELMAIN:

' ************************************************************************

' ****************** START OF OPTION 1 ********************

mainchoicei:

' ****************** Temperature Amplifier (TA) **********************

CALL logoTA

' ********************DESIRED WORKING FLUID*****************
' ************************************************************

    CALL mixture:          'Working fluid
mix1:
    trap$ = INKEY$: IF trap$ = "" GOTO mix1:
    IF trap$ = "A" THEN mat$ = "LiBr": GOTO mix2:
    IF trap$ = "B" THEN mat$ = "(Li, K, Na) NO3": GOTO mix2:
    IF trap$ = "C" THEN mat$ = "Other": GOTO MAINMENU:
    IF trap$ = "D" OR ASC(trap$) = 27 THEN GOTO MAINMENU:
    CALL noise: COLOR 31, bac: LOCATE 23, 12: PRINT " O N L Y   K E Y S   <F1> - <F4>   A R E   V A L I D "; : COLOR frg, bac
    GOTO mix1:
mix2:
        CALL wtabs(con$, wt)  'Wt% of solution leaving the desorber (generator)
        IF escflag = 1 GOTO mainchoicei: 'ESCAPE ROUTE

' ************************************************************

' *************TO PROCEED OR RETURN TO MAIN MENU*************

CALL mainchoicei:   REM moved screen to sub

INIDIS:
        trap$ = INKEY$: IF trap$ = "" GOTO INIDIS:

' ************************************************************************
' IF USER DECIDES TO ESCAPE AND GO BACK TO PREVIOUS MENU (MAIN MENU)
' ************************************************************************

        IF ASC(trap$) = 27 GOTO MAINMENU:
        IF trap$ = "A" GOTO PROCEED:

PROCEED:
        CLS : CALL border:  LOCATE 5, 5: PRINT "DO YOU WISH TO ENTER THE DATA IN TABULAR FORM OR BY INDIVIDUAL SCREENS?"

' ************************************************************************************************
' THE USER CAN ENTER THE DATA EITHER BY INDIVIDUAL SCREENS OR AT ONE TIME TOGGLING BETWEEN DATA INPUT
```

```
' THE USER CAN ESCAPE BACK TO PREVIOUS MENU (ESCAPE ROUTE)
' ************************************************************************************************

NEEDCOR:
        LOCATE 23, 2: PRINT "<F1>-TABULAR INPUT  <F2>-INDIVIDUAL SCREEN INPUT  <ESC>-BACK TO PREVIOUS MENU"

' **********************RESPONSE NEEDED**************************

        trap$ = ""
        WHILE trap$ = ""
                trap$ = INKEY$
        WEND
        IF trap$ = "A" THEN
                CLS

CALL tabdata(ESTID$, mat$, con$, qout, tcwin, twaste, tcapp, teapp, tdapp, taapp, thxapp, tcain, tdeso)

                IF escflag = 1 THEN
                GOTO mainchoicei:
                END IF
' ************************************************************************
' IF USER WANTS TO GO BACK ESCFLAG BECOMES 1 FROM SUBROUTINE TABDATA
' ************************************************************************

        IF tdeso >= twaste THEN
        CALL noise
        CLS : CALL border
        LOCATE 15, 10: PRINT "You have entered a temperature for the desorber which is "
        LOCATE 16, 10: PRINT "greater than waste heat temperature."
        LOCATE 19, 26: PRINT "    PLEASE RE-ENTER"
        LOCATE 23, 26: PRINT "PRESS ANY KEY TO CONTINUE"
DO WHILE INKEY$ = "": LOOP
        CALL tabdata(ESTID$, mat$, con$, qout, tcwin, twaste, tcapp, teapp, tdapp, taapp, thxapp, tcain, tdeso)
        END IF



' ************************************************
' ESCAPE ROUTE BACK TO THE BEGINNING OF CHOICE #1
' ************************************************

        ELSEIF ASC(trap$) = 27 THEN
                GOTO mainchoicei:

' ***********ENTERING THE DATA BY INDIVIDUAL SCREENS***********

        ELSEIF trap$ = "B" THEN
        CLS : CALL border: : LOCATE 15, 10: PRINT "Enter your Run identification and press  <ENTER>"; : INPUT ESTID$
TOINWS:
        CALL wasteheatin(twaste)  '  Waste heat inlet temperature
        IF escflag = 1 GOTO mainchoicei: 'ESCAPE ROUTE
TOINDPART:
         CALL qtout(qout)     'Heat ouput
         IF escflag = 1 GOTO TOINWS: 'ESCAPE ROUTE
TOINRHOP:
         CALL condh2oin(tcwin)   ' Condenser water inlet temperature
        IF escflag = 1 GOTO TOINDPART: 'ESCAPE ROUTE
TOTCapp:
         CALL condtapp(tcapp)   ' Approach temperature in condenser
        IF escflag = 1 GOTO TOINRHOP: 'ESCAPE ROUTE

TOTEapp:
         CALL evaTapp(teapp)   ' Approach temperature in evaporator
        IF escflag = 1 GOTO TOTCapp: 'ESCAPE ROUTE
TOTDapp:
         CALL desTapp(tdapp)   ' Approach temperature in desorber
        IF escflag = 1 GOTO TOTEapp: 'ESCAPE ROUTE
TOTAapp:
         CALL AbsTapp(taapp)   ' Approach temperature in absorber
        IF escflag = 1 GOTO TOTDapp: 'ESCAPE ROUTE
TOTHXapp:
         CALL HXTapp(thxapp)   ' Approach temperature in solution heat exchanger
        IF escflag = 1 GOTO TOTAapp: 'ESCAPE ROUTE
TOTCAIN:
        CALL TCABSIN(tcain)   ' Cooling water temperature entering the absorber
        IF escflag = 1 GOTO TOTHXapp: 'ESCAPE ROUTE
TOTdesorber:
        CALL Tdesorber(tdeso)   ' Desorber temperature
```

```
            IF escflag = 1 GOTO TOTCAIN: 'ESCAPE ROUTE
            IF tdeso >= twaste THEN
            CALL noise
            CLS : CALL border
            LOCATE 15, 10: PRINT "You have entered a temperature for the desorber which is"
            LOCATE 16, 10: PRINT "greater than waste heat temperature"
            LOCATE 19, 26: PRINT "PLEASE ENTER ANOTHER VALUE"
            LOCATE 23, 26: PRINT "PRESS ANY KEY TO CONTINUE"
DO WHILE INKEY$ = "": LOOP
            GOTO TOTdesorber:
            END IF
PREVMUF:

            ELSE

' ************************************************************************
' END OF RESPONSE ALGORITHM FOR EITHER INPUT THE DATA BY SCREENS OR TABULAR
' ************************************************************************

                                GOTO NEEDCOR:
            END IF

' ***********************************************************
' END OF IF BLOCK FOR ENTERING DATA BY SCREENS OR TABULAR
' ***********************************************************


' ***********************************************
 twasteout = twaste
 tcaout = tcain
' ***********************************************

            CALL hfwater(tcwin, hcwin)
            tdesout = twaste - tdapp

' ***********************************************
            tevapout = (twaste - teapp) - teapp
' ***********************************************

            CALL preh2o(tevapout, pevapout)
            CALL hgwater(tevapout, hgevapout)
            CALL hfwater(tcain, hfcain)
            CALL hgwater(tcaout, hgcaout)
            CALL hfwater(twasteout, hwasteout)
            CALL hgwater(twaste, hwastein)
            tabsout = tcaout + taapp
            tabsin = tabsout - thxapp

' ***************************LiBr solution**********************

IF mat$ = "LiBr" THEN
CALL hxtLiBr(tdesout, wt, hdesout)
CALL preLiBr(tdesout, wt, pdesin)
prefconout = pdesin
CALL TSATH2O(prefconout, trefconout)
tcwout = trefconout - tcapp
CALL hfwater(trefconout, hfrefconout)
            CALL hfwater(tcwout, hcwout)

CALL BoostTLiBr(pevapout, wt, tmax)
CALL hxtLiBr(tabsin, wt, habsin)
CALL abswtoutLiBr(tabsout, pevapout, wtout, wt)
CALL hxtLiBr(tabsout, wtout, habsout)
beta = wt / (wt - wtout)
alpha = wtout / (wt - wtout)

hhxout = habsout - alpha / beta * (habsin - hdesout)

CALL hgwater(trefconout, Hg)
CALL hgsuper(pdesin, Hg, trefconout, tdeso, hdesvapout)

CALL thxoutcalLiBr(hhxout, wtout, thxout)

CALL xhtcalLiBr(hhxout, tdeso, wt1, wtout, hdesvapout, wt)
IF escflag = 1 THEN GOTO MAINMENU:

CALL hgwater(trefconout, Hg)
CALL hgsuper(pdesin, Hg, trefconout, tdesout, hdesvapout)
```

```
CALL hxtLiBr(tdeso, wt1, hdesin)

CALL denlibr(tdesout, wt, rhosoln)

' ******* Checking the crystallization temperature of the streams ******
CALL TcrysLiBr(wtout, tabsout, tcrysabsout, tcryscodeabsout)
CALL TcrysLiBr(wt1, tdeso, tcrysdesin, tcryscodedesin)
CALL TcrysLiBr(wt, tdesout, tcrysdesout, tcryscodedesout)
CALL TcrysLiBr(wt, tabsin, tcrysabsin, tcryscodeabsin)
' ***********************************************************************


' **********************(Li, K, Na) NO3 mixture**********************

ELSEIF mat$ = "(Li, K, Na) NO3" THEN
CALL hxtalk(tdesout, wt, hdesout)
CALL prealk(tdesout, wt, pdesin)
prefconout = pdesin
CALL TSATH2O(prefconout, trefconout)
tcwout = trefconout - tcapp
CALL hfwater(trefconout, hfrefconout)
            CALL hfwater(tcwout, hcwout)

CALL BoostTalk(pevapout, wt, tmax)
CALL hxtalk(tabsin, wt, habsin)
CALL abswtoutalk(tabsout, pevapout, wtout, wt)
CALL hxtalk(tabsout, wtout, habsout)
beta = wt / (wt - wtout)
alpha = wtout / (wt - wtout)
hhxout = habsout - alpha / beta * (habsin - hdesout)

CALL hgwater(trefconout, Hg)
CALL hgsuper(pdesin, Hg, trefconout, tdeso, hdesvapout)

CALL thxoutcalalk(hhxout, wtout, thxout)

CALL xhtcalalk(hhxout, tdeso, wt1, wtout, hdesvapout, wt)
IF escflag = 1 THEN GOTO MAINMENU:

CALL hgwater(trefconout, Hg)
CALL hgsuper(pdesin, Hg, trefconout, tdesout, hdesvapout)

CALL hxtalk(tdeso, wt1, hdesin)

CALL denalk(tdesout, wt, rhosoln)

' ******* Checking the crystallization temperature of the streams ******
CALL Tcrysalk(wtout, tabsout, tcrysabsout, tcryscodeabsout)
CALL Tcrysalk(wt1, tdeso, tcrysdesin, tcryscodedesin)
CALL Tcrysalk(wt, tdesout, tcrysdesout, tcryscodedesout)
CALL Tcrysalk(wt, tabsin, tcrysabsin, tcryscodeabsin)
' ***********************************************************************

END IF

' ************Determination of COP and mass flow rates**************

QA = hgevapout + alpha * habsin - beta * habsout
gamma = QA / (hgeaout - hfesin)
QE = hgevapout - hfrefconout
delta = QE / (hwastein - hwasteout)

QD = alpha * hdesout + hdesvapout - beta * hhxout
phi = QD / (hwastein - hwasteout)
QC = hdesvapout - hfrefconout
sy = QC / (hcwout - hcwin)
COP = QA / (QE + QD)
msteam = qout / (hgeaout - hfcain)
mw = msteam / gamma
msolconc = alpha * mw
msoldilut = beta * mw
mevawaste = delta * mw
mdeswaste = phi * mw
mconh2o = sy * mw

CALL denh2o(trefconout, rhoh2o)

' ********** assuming 60% efficiency for the pumps *************
```

```
WPabs = (pevapout - pdesin) * 144 / rhosoln / .6    '144 converts from (1/in^2) to (1/ft^2)
PBabs = WPabs * msolconc * .0012851                 '1.2851E-3 converts ft-lb to Btu/hr
WPcon = (pevapout - pdesin) * 144 / rhoh2o / .6
PBcon = WPcon * mw * .0012851

' ************************************************************************

' ****************** Warning for possible Crystallization ******************

IF tcryscodeabsout = 1 OR tcryscodedesin = 1 OR tcryscodedesout = 1 OR tcryscodeabsin = 1 THEN
CLS : CALL border
iill = 0
LOCATE 3, 3: PRINT DATE$
LOCATE 3, 26: PRINT "Run Identity "; ESTID$
LOCATE 5, 3: PRINT "Working fluid "; con$; " wt% "; mat$
IF tcryscodeabsout = 1 THEN
iill = iill + 1: LOCATE (6 + iill), 3: PRINT "Sol'n from absorber is below crystallization temperature of"; tcrysabsout; " Deg. F"
iill = iill + 1: LOCATE (6 + iill), 3: PRINT wtout; " wt% Sol'n from absorber at "; tabsout; " Deg. F"
END IF
IF tcryscodedesin = 1 THEN
iill = iill + 1: LOCATE (7 + iill), 3: PRINT "Sol'n to generator is below crystallization temperature of"; tcrysdesin; " Deg. F"
iill = iill + 1: LOCATE (7 + iill), 3: PRINT wt1; " wt% Sol'n to generator at "; tdeso; " Deg. F"
END IF
IF tcryscodedesout = 1 THEN
iill = iill + 1: LOCATE (8 + iill), 3: PRINT "Sol'n from generator is below crystallization temperature of"; tcrysdesout; " Deg. F"
iill = iill + 1: LOCATE (8 + iill), 3: PRINT wt; " wt% Sol'n from generator at "; tdesout; " Deg. F"
END IF
IF tcryscodeabsin = 1 THEN
iill = iill + 1: LOCATE (9 + iill), 3: PRINT "Sol'n to absorber is below crystallization temperature of"; tcrysabsin; " Deg. F"
iill = iill + 1: LOCATE (9 + iill), 3: PRINT wt; " wt% Sol'n to absorber at "; tabsin; " Deg. F"
END IF
CALL noise: COLOR 31, bac: LOCATE 19, 24: PRINT "WARNING: Freeze-up is Possible"
LOCATE 21, 26: PRINT "PLEASE CHECK YOUR INPUTS": COLOR frg, bac
LOCATE 23, 26: PRINT "PRESS ANY KEY TO CONTINUE"
DO WHILE INKEY$ = "": LOOP
GOTO mainchoice1:
END IF

' ************************************************************************************

CLS : CALL border
LOCATE 3, 3: PRINT DATE$
LOCATE 3, 26: PRINT "Run Identity "; ESTID$
LOCATE 5, 3: PRINT "Working fluid "; con$; " wt% "; mat$
LOCATE 6, 3: PRINT "Maximum possible boost temperature "; tmax; " Deg. F"
LOCATE 7, 3: PRINT "Concentration of solution entering the absorber"; wt; " wt%"
LOCATE 8, 3: PRINT "Concentration of solution leaving the absorber"; wtout; " wt%"
LOCATE 9, 3: PRINT "Temperature of solution leaving the absorber "; tabsout; " Deg. F "
LOCATE 10, 3: PRINT "Temperature of solution leaving the desorber "; tdesout; " Deg. F"
LOCATE 11, 3: PRINT "Lift temperature "; (tabsout - tdesout); " Deg. F"
LOCATE 12, 3: PRINT "Concentration of solution entering the desorber "; wt1; " wt%"
LOCATE 13, 3: PRINT "COP "; COP
LOCATE 14, 3: PRINT "Steam flow rate "; msteam; " lb/hr for a capacity of "; qout; " Btu/hr"
LOCATE 15, 3: PRINT "Refrigerant (water) flow rate "; mw; " lb/hr"
LOCATE 16, 3: PRINT "Concentrated solution flow rate "; msolconc; " lb/hr"
LOCATE 17, 3: PRINT "Dilute solution flow rate "; msoldilut; " lb/hr"
LOCATE 18, 3: PRINT "Evaporator waste flow rate "; mevawaste; " lb/hr"
LOCATE 19, 3: PRINT "Desorber waste flow rate "; mdeswaste; " lb/hr"
LOCATE 20, 3: PRINT "Condenser water flow rate "; mconh2o; " lb/hr"
LOCATE 23, 26: PRINT "PRESS ANY KEY TO CONTINUE"
DO WHILE INKEY$ = "": LOOP

' ************************************************************************
' Calculation of UA for absorber, desorber, evaporator, condenser,
' and solution heat exchanger.
' ************************************************************************
Qabsorber = mw * QA
Qdesorber = mw * QD
Qevaporator = mw * QE
Qcondenser = mw * QC
QHX = (habsout - hhxout) * msoldilut
DeltaTlmEVA = ((twaste - tevapout) - (twasteout - trefconout)) / LOG((twaste - tevapout) / (twasteout - trefconout))
UAevaporator = Qevaporator / DeltaTlmEVA
DeltaTlmCON = ((trefconout - tcwin) - (tdesout - tcwout)) / LOG((trefconout - tcwin) / (tdesout - tcwout))
UAcondenser = Qcondenser / DeltaTlmCON
DeltaTlmDES = ((twaste - tdesout) - (twasteout - tdeso)) / LOG((twaste - tdesout) / (twasteout - tdeso))
UAdesorber = Qdesorber / DeltaTlmDES
```

```
DeltaTlmHX = ((tabsout - tabsin) - (thxout - tdesout)) / LOG((tabsout - tabsin) / (thxout - tdesout))
UAHX = QHX / DeltaTlmHX

' Assuming temperature of solution after entering the absorber is 22 degrees F
' higher than the concentrated solution leaving the heat exchanger.
t3x = tabsin + 22

DeltaTlmABS = ((tabsout - tcain) - (t3x - tcaout)) / LOG((tabsout - tcain) / (t3x - tcaout))
UAabsorber = Qabsorber / DeltaTlmABS

CLS : CALL border
LOCATE 3, 3: PRINT DATE$
LOCATE 3, 26: PRINT "Run Identity "; ESTID$
LOCATE 5, 3: PRINT "Working fluid "; con$; " wt% "; mat$
LOCATE 6, 3: PRINT "Maximum possible boost temperature "; tmax; " Deg. F"
LOCATE 7, 3: PRINT "Total heat transferred in the absorber "; Qabsorber; " Btu/hr"
LOCATE 8, 3: PRINT "Total heat transferred in the desorber "; Qdesorber; " Btu/hr"
LOCATE 9, 3: PRINT "Total heat transferred in the condenser "; Qcondenser; " Btu/hr"
LOCATE 10, 3: PRINT "Total heat transferred in the evaporator "; Qevaporator; " Btu/hr"
LOCATE 11, 3: PRINT "Total heat transferred in the solution HX "; QHX; " Btu/hr"
LOCATE 12, 3: PRINT "Required UA for the absorber "; UAabsorber; " Btu/hr-Deg. F"
LOCATE 13, 3: PRINT "Required UA for the desorber "; UAdesorber; " Btu/hr-Deg. F"
LOCATE 14, 3: PRINT "Required UA for the condenser "; UAcondenser; " Btu/hr-Deg. F"
LOCATE 15, 3: PRINT "Required UA for the evaporator "; UAevaporator; " Btu/hr-Deg. F"
LOCATE 16, 3: PRINT "Required UA for the solution HX "; UAHX; " Btu/hr-Deg. F"
LOCATE 17, 3: PRINT "Total UA required "; (UAabsorber + UAdesorber + UAcondenser + UAevaporator + UAHX); " Btu/hr-Deg. F"
LOCATE 18, 3: PRINT "Power supplied to pump after desorber "; PBabs; " Btu/hr"
LOCATE 19, 3: PRINT "Power supplied to pump after condenser "; PBcon; " Btu/hr"
LOCATE 20, 3: PRINT "Total power supplied to both pumps "; (PBabs + PBcon); " Btu/hr"
LOCATE 21, 3: PRINT "COP corrected for parasitic pumping power "; Qabsorber / (Qevaporator + Qdesorber + 4 * (PBabs + PBcon))
LOCATE 23, 26: PRINT "PRESS ANY KEY TO CONTINUE"
DO WHILE INKEY$ = "": LOOP

CLS : CALL border
LOCATE 3, 3: PRINT DATE$
LOCATE 3, 30: PRINT "Run Identity "; ESTID$
LOCATE 5, 29: PRINT "Operating parameters"
LOCATE 7, 22: PRINT "Working fluid "; con$; " wt% "; mat$
FOR jjii = 1 TO 38: LOCATE 8, (20 + jjii): PRINT CHR$(205): NEXT jjii
LOCATE 10, 5: PRINT "  Stream              T          P       wt%     Enthalpy     Flow"
LOCATE 11, 5: PRINT "Description       (Deg. F)    (psia)          (Btu/lb)    (lb/hr)"
FOR jjii = 1 TO 74: LOCATE 12, (3 + jjii): PRINT CHR$(196): NEXT jjii
abc = 0!
LOCATE 13, 5: PRINT USING "Steam from Gen.    ####.#    ####.##    ###.#    ######.#    #####.#"; tdesout; pdesin; abc; hdesvapout; mw
LOCATE 14, 5: PRINT USING "Water from Con.    ####.#    ####.##    ###.#    ######.#    #####.#"; trefconout; prefconout; abc; hfrefconout; mw
LOCATE 15, 5: PRINT USING "Steam from Evp.    ####.#    ####.##    ###.#    ######.#    #####.#"; tevapout; pevapout; abc; hgevapout; mw
LOCATE 16, 5: PRINT USING "Sol.  from Abs.    ####.#    ####.##    ###.#    ######.#    #####.#"; tabsout; pevapout; wtout; habsout; msoldilut
LOCATE 17, 5: PRINT USING "Sol.  to  Gen.    ####.#    ####.##    ###.#    ######.#    #####.#"; tdeso; pdesin; wt1; hdesin; msoldilut
LOCATE 18, 5: PRINT USING "Sol.  from Gen.    ####.#    ####.##    ###.#    ######.#    #####.#"; tdesout; pdesin; wt; hdesout; msolconc
LOCATE 19, 5: PRINT USING "Sol.  to  Abs.    ####.#    ####.##    ###.#    ######.#    #####.#"; tabsin; pevapout; wt; habsin; msolconc
FOR jjii = 1 TO 74: LOCATE 20, (3 + jjii): PRINT CHR$(196): NEXT jjii
LOCATE 23, 28: PRINT "PRESS ANY KEY TO CONTINUE"
DO WHILE INKEY$ = "": LOOP


GOTO MAINMENU:

ENDop1:

'**********************BACK TO THE MAIN MENU************************

'****************** END OF OPTION 1 ******************

'****************** START OPTION 2 ******************

mainchoiceii:

'****************** Heat Amplifier (HA) ******************

CALL logoHA

'********************DESIRED WORKING FLUID*****************
'*************************************************************

            CALL mixture:   "Working fluid
mix1ii:
    trap$ = INKEY$: IF trap$ = "" GOTO mix1ii:
    IF trap$ = "A" THEN mat$ = "LiBr": GOTO mix2ii:
```

```
    IF trap$ = "B" THEN mat$ = "(Li, K, Na) NO3": GOTO mix2ii:
    IF trap$ = "C" THEN mat$ = "Other": GOTO MAINMENU:
    IF trap$ = "D" OR ASC(trap$) = 27 THEN GOTO MAINMENU:
    CALL noise: COLOR 31, bac: LOCATE 23, 12: PRINT " O N L Y   K E Y S   <F1> - <F4>   A R E   V A L I D "; : COLOR frg, bac
    GOTO mix1ii:
mix2ii:
        CALL wtabs(con$, wt)   ' Wt% of solution leaving the desorber (generator)
        IF escflag = 1 GOTO mainchoiceii: 'ESCAPE ROUTE

' ************************************************************

' ************TO PROCEED OR RETURN TO MAIN MENU************

CALL mainchoiceii:   REM moved screen to sub
INIDISii:
        trap$ = INKEY$: IF trap$ = "" GOTO INIDISii:

' ********************************************************************
' IF USER DECIDES TO ESCAPE AND GO BACK TO PREVIOUS MENU (MAIN MENU)
' ********************************************************************

        IF ASC(trap$) = 27 GOTO MAINMENU:
        IF trap$ = "A" GOTO PROCEEDii:

PROCEEDii:
        CLS : CALL border:   LOCATE 5, 5: PRINT "DO YOU WISH TO ENTER THE DATA IN TABULAR FORM OR BY INDIVIDUAL SCREENS?"

' ************************************************************************************************
' THE USER CAN ENTER THE DATA EITHER BY INDIVIDUAL SCREENS OR AT ONE TIME TOGGLING BETWEEN DATA INPUT
' THE USER CAN ESCAPE BACK TO PREVIOUS MENU (ESCAPE ROUTE)
' ************************************************************************************************

NEEDCORii:
        LOCATE 23, 2: PRINT "<F1>-TABULAR INPUT  <F2>-INDIVIDUAL SCREEN INPUT  <ESC>-BACK TO PREVIOUS MENU"

' **********************RESPONSE NEEDED**************************

        trap$ = ""
        WHILE trap$ = ""
                trap$ = INKEY$
        WEND
        IF trap$ = "A" THEN
                CLS

CALL tabdata2(ESTID$, mat$, con$, qout, tcwin, twastedesin, twasteevapin, tcapp, teapp, tdapp, taapp, thxapp)

' ********************************************************************
' IF USER WANTS TO GO BACK ESCFLAG BECOMES 1 FROM SUBROUTINE TA8DATA
' ********************************************************************

                IF escflag = 1 THEN
                GOTO mainchoiceii:
                END IF

' ************************************************
' ESCAPE ROUTE BACK TO THE BEGINNING OF CHOICE #2
' ************************************************

        ELSEIF ASC(trap$) = 27 THEN
                GOTO mainchoiceii:

' ***********ENTERING THE DATA BY INDIVIDUAL SCREENS***********

        ELSEIF trap$ = "B" THEN
        CLS : CALL border: : LOCATE 15, 10: PRINT "Enter your Run identification and press  <ENTER>"; : INPUT ESTID$
TOINWSii:
        CALL wasteheatevapin(twasteevapin)  '  Waste heat inlet temperature (evaporator)
        IF escflag = 1 GOTO mainchoiceii: 'ESCAPE ROUTE
TOINDPARTii:
         CALL qtout(qout)     'Heat ouput
         IF escflag = 1 GOTO TOINWSii: 'ESCAPE ROUTE
TOINRHOPii:
         CALL condabsin(tcwin)   ' Condenser and absorber cooling water temperature
        IF escflag = 1 GOTO TOINDPARTii: 'ESCAPE ROUTE
TOTCappii:
         CALL condtapp(tcapp)   ' Approach temperature in condenser
        IF escflag = 1 GOTO TOINRHOPii: 'ESCAPE ROUTE
```

```
TOTEappii:
            CALL evaTapp(teapp)   ' Approach temperature in evaporator
            IF escflag = 1 GOTO TOTCappii: 'ESCAPE ROUTE
TOTDappii:
            CALL desTapp(tdapp)   ' Approach temperature in desorber
            IF escflag = 1 GOTO TOTEappii: 'ESCAPE ROUTE
TOTAappii:
            CALL AbsTapp(taapp)   ' Approach temperature in absorber
            IF escflag = 1 GOTO TOTDappii: 'ESCAPE ROUTE
TOTHXappii:
            CALL HXTapp(thxapp)   ' Approach temperature in solution heat exchanger
            IF escflag = 1 GOTO TOTAappii: 'ESCAPE ROUTE

PREVMUFii:

            ELSE
' ************************************************************************
' END OF RESPONSE ALGORITHM FOR EITHER INPUT THE DATA BY SCREENS OR TABULAR
' ************************************************************************

                        GOTO NEEDCORii:
            END IF

' ************************************************************
' END OF IF BLOCK FOR ENTERING DATA BY SCREENS OR TABULAR
' ************************************************************


' ************************************************
  tcwout = tcwin
  tcain = tcwin
  tcaout = tcain
  twasteevapout = twasteevapin
' ************************************************

            trefconout = tcwout + tcapp
            CALL preh2o(trefconout, prefconout)
            CALL hfwater(trefconout, hfrefconout)
            CALL hgwater(tcwout, hcwout)
            CALL hfwater(tcwin, hcwin)

' ************************************************
            tevapout = twasteevapin - teapp
' ************************************************

            CALL preh2o(tevapout, pevapout)
            CALL hgwater(tevapout, hgevapout)
            CALL hfwater(tcain, hfcain)
            CALL hgwater(tcaout, hgcaout)
            CALL hfwater(twasteevapout, hwasteevapout)
            CALL hgwater(twasteevapin, hwasteevapin)

            pdesin = prefconout

            tabsout = tcaout + taapp
            tabsin = tabsout + thxapp
            thxabsin = tabsout + (1.5 * thxapp)

' ***************************LiBr solution**********************

IF mat$ = "LiBr" THEN
CALL BoostTLiBr(pdesin, wt, tdesout)
CALL hxtLiBr(tdesout, wt, hdesout)
tdeso = tdesout
            twastedesin = tdesout + tdapp
            twastedesout = twastedesin
            CALL hfwater(twastedesout, hwastedesout)
            CALL hgwater(twastedesin, hwastedesin)


CALL hxtLiBr(thxabsin, wt, hhxabsin)
CALL abswtoutLiBr(tabsout, pevapout, wtout, wt)
CALL hxtLiBr(tabsout, wtout, habsout)
beta = wt / (wt - wtout)
alpha = wtout / (wt - wtout)

hhxabsout = habsout - alpha / beta * (hhxabsin - hdesout)
```

```
CALL TSATH2O(pevapout, tsat)
CALL hgwater(tsat, Hg)
CALL hgsuper(pevapout, Hg, tsat, tabsin, habsvapin)

CALL thxoutcalLiBr(hhxabsout, wtout, thxout)
hdesin = hhxabsout

CALL TSATH2O(pdesin, tsat1)
CALL hgwater(tsat1, Hg1)
CALL hgsuper(pdesin, Hg1, tsat1, tdeso, hdesvapout)


CALL xhtcalLiBrHA(hhxabsin, tabsin, wt1, wt, habsvapin, wtout)
IF escflag = 1 THEN GOTO MAINMENU:


CALL hxtLiBr(tabsin, wt1, habsin)

CALL denlibr(tabsout, wtout, rhosoln)
' ******* Checking the crystallization temperature of the streams ******
CALL TcrysLiBr(wtout, tabsout, tcrysabsout, tcryscodeabsout)
CALL TcrysLiBr(wtout, thxout, tcrysdesin, tcryscodedesin)
CALL TcrysLiBr(wt, tdesout, tcrysdesout, tcryscodedesout)
CALL TcrysLiBr(wt, tabsin, tcrysabsin, tcryscodeabsin)
' ********************************************************************

' **********************(Li, K, Na) NO3 mixture**********************

ELSEIF mat$ = "(Li, K, Na) NO3" THEN
CALL BoostTalk(pdesin, wt, tdesout)
CALL hxtalk(tdesout, wt, hdesout)
tdeso = tdesout
        twastedesin = tdesout + tdapp
        twastedesout = twastedesin
        CALL hfwater(twastedesout, hwastedesout)
        CALL hgwater(twastedesin, hwastedesin)


CALL hxtalk(thxabsin, wt, hhxabsin)
CALL abswtoutalk(tabsout, pevapout, wtout, wt)
CALL hxtalk(tabsout, wtout, habsout)
beta = wt / (wt - wtout)
alpha = wtout / (wt - wtout)

hhxabsout = habsout - alpha / beta * (hhxabsin - hdesout)

CALL TSATH2O(pevapout, tsat)
CALL hgwater(tsat, Hg)
CALL hgsuper(pevapout, Hg, tsat, tabsin, habsvapin)

CALL thxoutcalalk(hhxabsout, wtout, thxout)
hdesin = hhxabsout

CALL TSATH2O(pdesin, tsat1)
CALL hgwater(tsat1, Hg1)
CALL hgsuper(pdesin, Hg1, tsat1, tdeso, hdesvapout)

CALL xhtcalalkHA(hhxabsin, tabsin, wt1, wt, habsvapin, wtout)
IF escflag = 1 THEN GOTO MAINMENU:

CALL hxtalk(tabsin, wt1, habsin)

CALL denalk(tabsout, wtout, rhosoln)

' ******* Checking the crystallization temperature of the streams ******
CALL Tcrysalk(wtout, tabsout, tcrysabsout, tcryscodeabsout)
CALL Tcrysalk(wtout, thxout, tcrysdesin, tcryscodedesin)
CALL Tcrysalk(wt, tdesout, tcrysdesout, tcryscodedesout)
CALL Tcrysalk(wt, tabsin, tcrysabsin, tcryscodeabsin)
' ********************************************************************

END IF

' ************Determination of COP and mass flow rates*************

QA = hgevapout + alpha * hhxabsin - beta * habsout
gamma = QA / (hgcaout - hfcain)
QE = hgevapout - hfrefconout
```

```
delta = QE / (hwasteevapin - hwasteevapout)

QD = alpha * hdesout + hdesvapout - beta * hdesin
phi = QD / (hwastedesin - hwastedesout)
QC = hdesvapout - hfrefconout
sy = QC / (hcwout - hcwin)
COPH = (QA + QC) / QD
COPC = QE / QD
msteam = qout / (hgcaout - hfcain)
mw = msteam / gamma
msolconc = alpha * mw
msoldilut = beta * mw
mevawaste = delta * mw
mdeswaste = phi * mw
mconh2o = sy * mw

efficiency = .6     'assuming 60% efficiency for the pumps
WPabs = (pdesin - pevapout) * 144 / rhosoln / .6     '144 converts from (1/in^2) to (1/ft^2)
PBabs = WPabs * msoldilut * .0012851                 '1.2851E-3 converts ft-lb to Btu/hr


' ****************** Warning for possible Crystallization ******************
IF tcryscodeabsout = 1 OR tcryscodedesin = 1 OR tcryscodedesout = 1 OR tcryscodeabsin = 1 THEN
CLS : CALL border
iill = 0
LOCATE 3, 3: PRINT DATE$
LOCATE 3, 26: PRINT "Run Identity "; ESTID$
LOCATE 5, 3: PRINT "Working fluid "; con$; " wt% "; mat$
IF tcryscodeabsout = 1 THEN
iill = iill + 1: LOCATE (6 + iill), 3: PRINT "Sol'n from absorber is below crystallization temperature of"; tcrysabsout; " Deg. F"
iill = iill + 1: LOCATE (6 + iill), 3: PRINT wtout; " wt% Sol'n from absorber at "; tabsout; " Deg. F"
END IF
IF tcryscodedesin = 1 THEN
iill = iill + 1: LOCATE (7 + iill), 3: PRINT "Sol'n to generator is below crystallization temperature of"; tcrysdesin; " Deg. F"
iill = iill + 1: LOCATE (7 + iill), 3: PRINT wtout; " wt% Sol'n to generator at "; thxout; " Deg. F"
END IF
IF tcryscodedesout = 1 THEN
iill = iill + 1: LOCATE (8 + iill), 3: PRINT "Sol'n from generator is below crystallization temperature of"; tcrysdesout; " Deg. F"
iill = iill + 1: LOCATE (8 + iill), 3: PRINT wt; " wt% Sol'n from generator at "; tdesout; " Deg. F"
END IF
IF tcryscodeabsin = 1 THEN
iill = iill + 1: LOCATE (9 + iill), 3: PRINT "Sol'n to absorber is below crystallization temperature of"; tcrysabsin; " Deg. F"
iill = iill + 1: LOCATE (9 + iill), 3: PRINT wt; " wt% Sol'n to absorber at "; tabsin; " Deg. F"
END IF
CALL noise: COLOR 31, bac: LOCATE 19, 24: PRINT "WARNING: Freeze-up is Possible"
LOCATE 21, 26: PRINT "PLEASE CHECK YOUR INPUTS": COLOR frg, bac
LOCATE 23, 26: PRINT "PRESS ANY KEY TO CONTINUE"
DO WHILE INKEY$ = "": LOOP
GOTO mainchoiceii:
END IF

' ************************************************************************************



' ********************************************************************

CLS : CALL border
LOCATE 3, 3: PRINT DATE$
LOCATE 3, 26: PRINT "Run Identity "; ESTID$
LOCATE 5, 3: PRINT "Working fluid "; con$; " wt% "; mat$
LOCATE 7, 3: PRINT "Concentration of solution leaving the desorber"; wt; " wt%"
LOCATE 8, 3: PRINT "Concentration of solution leaving the absorber"; wtout; " wt%"
LOCATE 9, 3: PRINT "Temperature of solution leaving the absorber "; tabsout; " Deg. F "
LOCATE 10, 3: PRINT "Temperature of solution leaving the desorber "; tdesout; " Deg. F"
LOCATE 11, 3: PRINT "Lift temperature "; (tabsout - tevapout); " Deg. F"
LOCATE 12, 3: PRINT "Concentration of solution entering the absorber "; wt1; " wt%"
LOCATE 13, 3: PRINT "COP heating and cooling "; COPH; " and  "; COPC
LOCATE 14, 3: PRINT "Steam flow rate "; msteam; " lb/hr for a capacity of "; qout; " Btu/hr"
LOCATE 15, 3: PRINT "Refrigerant (water) flow rate "; mw; " lb/hr"
LOCATE 16, 3: PRINT "Concentrated solution flow rate "; msolconc; " lb/hr"
LOCATE 17, 3: PRINT "Dilute solution flow rate "; msoldilut; " lb/hr"
LOCATE 18, 3: PRINT "Evaporator waste flow rate "; mevawaste; " lb/hr"
LOCATE 19, 3: PRINT "Desorber waste flow rate "; mdeswaste; " lb/hr"
LOCATE 20, 3: PRINT "Condenser water flow rate "; mconh2o; " lb/hr"
LOCATE 23, 26: PRINT "PRESS ANY KEY TO CONTINUE"
DO WHILE INKEY$ = "": LOOP
```

```
' ************************************************************************
' Calculation of UA for absorber, desorber, evaporator, condenser,
' and solution heat exchanger.
' ************************************************************************

CALL TSATH2O(pevapout, tevapin)
tevapin = tevapin - 5


Qabsorber = mw * QA
Qdesorber = mw * QD
Qevaporator = mw * QE
Qcondenser = mw * QC
QHX = (hhxabsout - hafeout) * msoldilut
DeltaTlmEVA = ((twasteevapin - tevapout) - (twasteevapout - tevapin)) / LOG((twasteevapin - tevapout) / (twasteevapout - tevapin))
UAevaporator = Qevaporator / DeltaTlmEVA
DeltaTlmCON = ((trefconout - tcwin) - (tdeso - tcwout)) / LOG((trefconout - tcwin) / (tdeso - tcwout))
UAcondenser = Qcondenser / DeltaTlmCON
DeltaTlmDES = ((twastedesout - thxout) - (twastedesin - tdesout)) / LOG((twastedesout - thxout) / (twastedesin - tdesout))
UAdesorber = Qdesorber / DeltaTlmDES

DeltaTlmHX = ((tdesout - thxout) - (thxabsin - tabsout)) / LOG((tdesout - thxout) / (thxabsin - tabsout))
UAHX = QHX / DeltaTlmHX

DeltaTlmABS = ((tabsout - tcain) - (tabsin - tcaout)) / LOG((tabsout - tcain) / (tabsin - tcaout))
UAabsorber = Qabsorber / DeltaTlmABS

CLS : CALL border
LOCATE 3, 3: PRINT DATE$
LOCATE 3, 26: PRINT "Run Identity "; ESTID$
LOCATE 5, 3: PRINT "Working fluid "; con$; " wt% "; mat$
LOCATE 7, 3: PRINT "Total heat transferred in the absorber "; Qabsorber; " Btu/hr"
LOCATE 8, 3: PRINT "Total heat transferred in the desorber "; Qdesorber; " Btu/hr"
LOCATE 9, 3: PRINT "Total heat transferred in the condenser "; Qcondenser; " Btu/hr"
LOCATE 10, 3: PRINT "Total heat transferred in the evaporator "; Qevaporator; " Btu/hr"
LOCATE 11, 3: PRINT "Total heat transferred in the solution HX "; QHX; " Btu/hr"
LOCATE 12, 3: PRINT "Required UA for the absorber "; UAabsorber; " Btu/hr-Deg. F"
LOCATE 13, 3: PRINT "Required UA for the desorber "; UAdesorber; " Btu/hr-Deg. F"
LOCATE 14, 3: PRINT "Required UA for the condenser "; UAcondenser; " Btu/hr-Deg. F"
LOCATE 15, 3: PRINT "Required UA for the evaporator "; UAevaporator; " Btu/hr-Deg. F"
LOCATE 16, 3: PRINT "Required UA for the solution HX "; UAHX; " Btu/hr-Deg. F"
LOCATE 17, 3: PRINT "Total UA required "; (UAabsorber + UAdesorber + UAcondenser + UAevaporator + UAHX); " Btu/hr-Deg. F"
LOCATE 18, 3: PRINT "Power supplied to pump after absorber "; PBabs; " Btu/hr"
LOCATE 19, 3: PRINT "COP corrected for parasitic pumping power "; (Qabsorber + Qcondenser) / (Qdesorber + 4 * PBabs)

LOCATE 23, 26: PRINT "PRESS ANY KEY TO CONTINUE"
DO WHILE INKEY$ = "": LOOP

CLS : CALL border
LOCATE 3, 3: PRINT DATE$
LOCATE 3, 30: PRINT "Run Identity "; ESTID$
LOCATE 5, 29: PRINT "Operating parameters"
LOCATE 7, 22: PRINT "Working fluid "; con$; " wt% "; mat$
FOR jjii = 1 TO 36: LOCATE 8, (20 + jjii): PRINT CHR$(205): NEXT jjii
LOCATE 10, 5: PRINT "  Stream              T        P      wt%    Enthalpy     Flow"
LOCATE 11, 5: PRINT "Description       (Deg. F)   (psia)          (Btu/lb)    (lb/hr)"
FOR jjii = 1 TO 74: LOCATE 12, (3 + jjii): PRINT CHR$(196): NEXT jjii
abc = 0!
LOCATE 13, 5: PRINT USING "Steam from Gen.   ####.#    ####.##   ###.#   ######.#    #####.#"; tdesout; pdesin; abc; hdesvapout; mw
LOCATE 14, 5: PRINT USING "Water from Con.   ####.#    ####.##   ###.#   ######.#    #####.#"; trefconout; prefconout; abc; hfrefconout; mw
LOCATE 15, 5: PRINT USING "Steam from Evp.   ####.#    ####.##   ###.#   ######.#    #####.#"; tevapout; pevapout; abc; hgevapout; mw
LOCATE 16, 5: PRINT USING "Sol. from Abs.    ####.#    ####.##   ###.#   ######.#    #####.#"; tabsout; pevapout; wtout; habsout; msoldilut
LOCATE 17, 5: PRINT USING "Sol. to Gen.      ####.#    ####.##   ###.#   ######.#    #####.#"; thxout; pdesin; wtout; hhxabsout; msoldilut
LOCATE 18, 5: PRINT USING "Sol. from Gen.    ####.#    ####.##   ###.#   ######.#    #####.#"; tdesout; pdesin; wt; hdesout; msolconc
LOCATE 19, 5: PRINT USING "Sol. to Abs.      ####.#    ####.##   ###.#   ######.#    #####.#"; tabsin; pevapout; wt1; habsin; msolconc
FOR jjii = 1 TO 74: LOCATE 20, (3 + jjii): PRINT CHR$(196): NEXT jjii
LOCATE 23, 28: PRINT "PRESS ANY KEY TO CONTINUE"
DO WHILE INKEY$ = "": LOOP


GOTO MAINMENU:

ENDop2:


' ***********************BACK TO THE MAIN MENU**************************
```

```
' ****************** END OF OPTION 2 ******************

' ********** OPTION SIX **********

' ****** EXIT THE PROGRAM **********

MAINCHOICEVI:
            CLS
    END

SUB AbsTapp (taapp) STATIC

' ************************************************************************
' THIS PROGRAM IS USED TO ENTER THE APPROACH TEMPERATURE IN ABSORBER
' (taapp) BY INDIVIDUAL SCREENS
' ************************************************************************

            escflag = 0
CH1INDP6:
ENTPART6:

' ****************************************************************
' USER COULD CHANGE THE UNITS BY PRESSING <F1> KEY OR
' COULD ESCAPE TO PREVIOUS MENU BY PRESSING <ESC> KEY OR
' ACCEPT THE INPUT BY PRESSING <F9>
'****************************************************************

            n = 1: taappCF = 1: taappUNIT$ = "Degrees F     ": taappUNIT1$ = taappUNIT$: taapp = 0: taapp$ = ""
            CLS : CALL border: : LOCATE 6, 6: PRINT "Type in the approach temperature in absorber. Use Function key <F1>"
            LOCATE 7, 6: PRINT "to select proper units for your entry. Press <F9> when finished."
            COLOR 15, bac: LOCATE 23, 7: PRINT "<F1>-TOGGLE UNITS       <F9>-CONFIRM ENTRY       <ESC>-PREVIOUS MENU"; : COLOR frg, bac
            LOCATE 19, 7: PRINT "Approach temperature in absorber    ";
PRTUNIT6:
            LOCATE 19, 56: PRINT taappUNIT$;
20006   LOCATE 19, 40, 0: trap$ = INKEY$: IF trap$ = "" GOTO 20006
            IF trap$ = "I" GOTO 21006
            IF ASC(trap$) = 8 THEN taapp$ = "": taapp = 0: LOCATE 19, 45: PRINT "       ": GOTO 20006
            IF trap$ = "A" GOTO CHANGEUNIT6:
    IF ASC(trap$) = 27 THEN
    escflag = 1
    GOTO 21506
    END IF
            IF ASC(trap$) < 46 OR ASC(trap$) > 57 OR ASC(trap$) = 47 GOTO 20006
            taapp$ = taapp$ + trap$
            LOCATE 19, 45: PRINT VAL(taapp$); : taapp = VAL(taapp$): GOTO 20006

' ************************************************************************
' CONFIRM YOUR TEMPERATURE INPUT PRESS <F1> FOR CORRECT INPUT OR <F2> FOR
' INCORRECT INPUT
' ************************************************************************

21006   CLS : CALL border: : LOCATE 12, 8: PRINT "Your chosen approach temperature in absorber is  "; taapp; "  "; taappUNIT$
            LOCATE 23, 12: PRINT "  <F1>- Correct                <F2>-Incorrect     ";
21106   DTCHECK$ = INKEY$: IF DTCHECK$ = "A" GOTO 21506
            IF DTCHECK$ = "B" THEN taapp$ = "": GOTO ENTPART6:
            GOTO 21106

' ********************CHANGING UNITS OF TEMPERATURE**********************

CHANGEUNIT6:
            n = n + 1: IF n = 3 THEN n = 1
            ON n GOTO 21206, 21216
21206   taappCF = 1: taappUNIT$ = "Degrees F     ": GOTO PRTUNIT6:
21216   taappCF = 2: taappUNIT$ = "Degrees C     ": GOTO PRTUNIT6:

' ************************************************************************

21506   CLS : IF taappCF = 2 THEN taapp = (taapp * 1.8)

' ********************taapp IS IN DEGREES F.**************************


END SUB

SUB abswtoutalk (T, P, wtout, wt) STATIC

' ************************************************************************
```

```
' This program is used to calculate the wt% of (Li, K, Na) NO3 mixture
' (wtout) leaving the absorber from its pressure (P in psia) and
' Temperature (T in Deg. F)
' ********************************************************************

            i = 0: wtout = wt
            C = 6.21147: D = -2886.373: E = -337269.46#
            LogP = LOG(P) / 2.302585
            T1 = (-D + SQR(D ^ 2 - 4 * E * (C - LogP))) / (2 * (C - LogP)) - 459.72
3001    a = -1.06427 + .102088 * wtout - .0016266 * wtout ^ 2 + 8.93054E-06 * wtout ^ 3
            B = 291.113 - 13.0305 * wtout + .187146 * wtout ^ 2 - 7.87688E-04 * wtout ^ 3
            fx = T - (a * T1) - B
            aprime = .102088 - (2 * .0016266) * wtout + (3 * 8.93054E-06) * wtout ^ 2
            bprime = (-13.0305) + (2 * .187146 * wtout) - (3 * 7.87688E-04) * wtout ^ 2
            dfx = (-aprime * T1) - bprime
            IF i > 200 GOTO 6001
            wtnew = wtout - (fx / dfx)
            IF ABS(fx) > .0001 THEN wtout = wtnew: i = i + 1: GOTO 3001
6001    IF ABS(fx) > .0001 AND i > 200 THEN PRINT "Did not converge": END
            wtout = wtnew

            END SUB

SUB abswtoutLiBr (T, P, wtout, wt) STATIC

' ********************************************************************
' This program is used to calculate the wt% of LiBr solution (wtout)
' leaving the absorber from its pressure (P in psia) and Temperature
' (T in Deg. F)
' ********************************************************************

            i = 0: wtout = wt
            C = 6.21147: D = -2886.373: E = -337269.46#
            LogP = LOG(P) / 2.302585
            T1 = (-D + SQR(D ^ 2 - 4 * E * (C - LogP))) / (2 * (C - LogP)) - 459.72
30      a = -2.00755 + .16976 * wtout - 3.133362E-03 * wtout ^ 2 + 1.97668E-05 * wtout ^ 3
            B = 321.128 - 19.322 * wtout + .374382 * wtout ^ 2 - .0020637 * wtout ^ 3
            fx = T - (a * T1) - B
            aprime = .16976 - (2 * 3.133362E-03) * wtout + (3 * 1.97668E-05) * wtout ^ 2
            bprime = (-19.322) + (2 * .374382) * wtout - (3 * .0020637) * wtout ^ 2
            dfx = (-aprime * T1) - bprime
            IF i > 200 GOTO 600
            wtnew = wtout - (fx / dfx)
            IF ABS(fx) > .0001 THEN wtout = wtnew: i = i + 1: GOTO 30
600     IF ABS(fx) > .0001 AND i > 200 THEN PRINT "Did not converge"
            wtout = wtnew

END SUB

SUB BoostTalk (P, wt, tmax) STATIC

' ********************************************************************
' Maximum Boost Temperature of Saturated (Li, K, Na) NO3 /water (tmax)
' Input P sat. (psia) and wt (wt%) to calculate T sat (Deg. F).
' ********************************************************************

            a = -1.06427 + .102088 * wt - .0016266 * wt ^ 2 + 8.93054E-06 * wt ^ 3
            B = 291.113 - 13.0305 * wt + .187146 * wt ^ 2 - 7.87688E-04 * wt ^ 3
            C = 6.21147: D = -2886.373: E = -337269.46#
            LogP = LOG(P) / 2.302585
            T1 = (-D + SQR(D ^ 2 - 4 * E * (C - LogP))) / (2 * (C - LogP)) - 459.72
            tmax = a * T1 + B

END SUB

SUB BoostTLiBr (P, wt, tmax) STATIC

' ********************************************************************
' Maximum Boost Temperature of Saturated LiBr (tmax)
' Input P sat. (psia) and wt (wt%) to calculate T sat (Deg. F).
' ********************************************************************

            a = -2.00755 + .16976 * wt - 3.133362E-03 * wt ^ 2 + 1.97668E-05 * wt ^ 3
            B = 321.128 - 19.322 * wt + .374382 * wt ^ 2 - .0020637 * wt ^ 3
            C = 6.21147: D = -2886.373: E = -337269.46#
            LogP = LOG(P) / 2.302585
            T1 = (-D + SQR(D ^ 2 - 4 * E * (C - LogP))) / (2 * (C - LogP)) - 459.72
            tmax = a * T1 + B
```

```
END SUB

SUB border STATIC

' *****************************************************
' THIS PROGRAM DRAWS A BOX AROUND DESIRED SCREEN INPUTS
' *****************************************************

LOCATE 2, 1: PRINT "╔══════════════════════════════════════════════════════════════════════════════╗";
FOR i = 1 TO 21
LOCATE (2 + i), 1: PRINT "║";
LOCATE (2 + i), 80: PRINT "║";
NEXT i
LOCATE 24, 1: PRINT "╚══════════════════════════════════════════════════════════════════════════════╝";
END SUB

        SUB condabsin (tcwin)

' ************************************************************************
' THIS PROGRAM IS USED TO INPUT THE CONDENSER WATER INLET TEMPERATURE (tcwin)
' BY INDIVIDUAL SCREENS
' ************************************************************************

        escflag = 0
CH1INDPiii:
ENTPARTiii:

' ************************************************************
' USER COULD CHANGE THE UNITS BY PRESSING <F1> KEY OR
' COULD ESCAPE TO PREVIOUS MENU BY PRESSING <ESC> KEY OR
' ACCEPT THE INPUT BY PRESSING <F9>
'************************************************************

        n = 1: tcwinCF = 1: tcwinUNIT$ = " Degrees F ": tcwinUNIT1$ = tcwinUNIT$: tcwin = 0: tcwin$ = ""
        CLS : CALL border: : LOCATE 6, 3: PRINT "Type in the condenser and absorber cooling water temperature. Use Function"
        LOCATE 7, 3: PRINT "key <F1> to select proper units for your entry. Press <F9> when finished."
        COLOR 15, bac: LOCATE 23, 7: PRINT "<F1>-TOGGLE UNITS      <F9>-CONFIRM ENTRY      <ESC>-PREVIOUS MENU"; : COLOR frg, bac
        LOCATE 19, 3: PRINT "Condenser and absorber cooling inlet temperature    ";
PRTUNITiii:
        LOCATE 19, 64: PRINT tcwinUNIT$;
20033   LOCATE 19, 40, 0: trap$ = INKEY$: IF trap$ = "" GOTO 20033
        IF trap$ = "I" GOTO 21033
        IF ASC(trap$) = 8 THEN tcwin$ = "": tcwin = 0: LOCATE 19, 55: PRINT "       ": GOTO 20033
        IF trap$ = "A" GOTO CHANGEUNITiii:
    IF ASC(trap$) = 27 THEN
    escflag = 1
    GOTO 21533
    END IF
        IF ASC(trap$) < 46 OR ASC(trap$) > 57 OR ASC(trap$) = 47 GOTO 20033
        tcwin$ = tcwin$ + trap$
        LOCATE 19, 55: PRINT VAL(tcwin$); : tcwin = VAL(tcwin$): GOTO 20033

' ************************************************************************
' CONFIRM YOUR TEMPERATURE INPUT PRESS <F1> FOR CORRECT INPUT OR <F2> FOR
' INCORRECT INPUT
' ************************************************************************

21033   CLS : CALL border: : LOCATE 12, 15: PRINT "Your chosen condenser and absorber"
         LOCATE 13, 15: PRINT "cooling water temperature is   "; tcwin; " "; tcwinUNIT$
        LOCATE 23, 12: PRINT " <F1>- Correct               <F2>-Incorrect     ";
21133   DTCHECK$ = INKEY$: IF DTCHECK$ = "A" GOTO 21533
        IF DTCHECK$ = "B" THEN tcwin$ = "": GOTO ENTPARTiii:
        GOTO 21133

' *******************CHANGING UNITS OF TEMPERATURE********************

CHANGEUNITiii:
        n = n + 1: IF n = 3 THEN n = 1
        ON n GOTO 21233, 21234
21233   tcwinCF = 1: tcwinUNIT$ = " Degrees F ": GOTO PRTUNITiii:
21234   tcwinCF = 2: tcwinUNIT$ = " Degrees C ": GOTO PRTUNITiii:

' ****************************************************************
```

```
21533    CLS : IF tcwinCF = 2 THEN tcwin = (tcwin * 1.8) + 32

' ********************tcwin IS IN DEGREES F.*************************

END SUB

            SUB condh2oin (tcwin) STATIC

' ************************************************************************
' THIS PROGRAM IS USED TO INPUT THE CONDENSER WATER INLET TEMPERATURE (tcwin)
' BY INDIVIDUAL SCREENS
' ************************************************************************

            escflag = 0
CH1INDP:
ENTPART:

' ****************************************************************
' USER COULD CHANGE THE UNITS BY PRESSING <F1> KEY OR
' COULD ESCAPE TO PREVIOUS MENU BY PRESSING <ESC> KEY OR
' ACCEPT THE INPUT BY PRESSING <F9>
'****************************************************************

            n = 1: tcwinCF = 1: tcwinUNIT$ = "Degrees F     ": tcwinUNIT1$ = tcwinUNIT$: tcwin = 0: tcwin$ = ""
            CLS : CALL border: : LOCATE 6, 6: PRINT "Type in the condenser water inlet temperature. Use Function key <F1>"
            LOCATE 7, 6: PRINT "to select proper units for your entry. Press <F9> when finished."
            COLOR 15, bac: LOCATE 23, 7: PRINT "<F1>-TOGGLE UNITS        <F9>-CONFIRM ENTRY        <ESC>-PREVIOUS MENU"; : COLOR frg, bac
            LOCATE 19, 7: PRINT "Condenser water inlet temperature    ";
PRTUNIT:
            LOCATE 19, 56: PRINT tcwinUNIT$;
2000    LOCATE 19, 40, 0: trap$ = INKEY$: IF trap$ = "" GOTO 2000
            IF trap$ = "I" GOTO 2100
            IF ASC(trap$) = 8 THEN tcwin$ = "": tcwin = 0: LOCATE 19, 45: PRINT "        ": GOTO 2000
            IF trap$ = "A" GOTO CHANGEUNIT:
    IF ASC(trap$) = 27 THEN
    escflag = 1
    GOTO 2150
    END IF
            IF ASC(trap$) < 46 OR ASC(trap$) > 57 OR ASC(trap$) = 47 GOTO 2000
            tcwin$ = tcwin$ + trap$
            LOCATE 19, 45: PRINT VAL(tcwin$); : tcwin = VAL(tcwin$): GOTO 2000

' ************************************************************************
' CONFIRM YOUR TEMPERATURE INPUT PRESS <F1> FOR CORRECT INPUT OR <F2> FOR
' INCORRECT INPUT
' ************************************************************************

2100    CLS : CALL border: : LOCATE 12, 8: PRINT "Your chosen condenser water inlet temperature is   "; tcwin; "  "; tcwinUNIT$
            LOCATE 23, 12: PRINT "  <F1>- Correct                <F2>-Incorrect     ";
2110    DTCHECK$ = INKEY$: IF DTCHECK$ = "A" GOTO 2150
            IF DTCHECK$ = "B" THEN tcwin$ = "": GOTO ENTPART:
            GOTO 2110

' ********************CHANGING UNITS OF TEMPERATURE*********************

CHANGEUNIT:
            n = n + 1: IF n = 3 THEN n = 1
            ON n GOTO 2120, 2121
2120    tcwinCF = 1: tcwinUNIT$ = "Degrees F     ": GOTO PRTUNIT:
2121    tcwinCF = 2: tcwinUNIT$ = "Degrees C     ": GOTO PRTUNIT:

' ************************************************************************

2150    CLS : IF tcwinCF = 2 THEN tcwin = (tcwin * 1.8) + 32

' ********************tcwin IS IN DEGREES F.*************************

END SUB
```

MODULEILBAS

```
DECLARE SUB denalk (tfl, wt!, rhosoln!)
DECLARE SUB denlibr (tfl, wt!, rhosoln!)
DECLARE SUB denh2o (tfl, rhoh2o!)
DECLARE SUB condabsin (tcwin!)
DECLARE SUB xhtcalalkHA (hxt!, Tl, wtout!, wt!, hvapor!, wtabsout!)
DECLARE SUB xhtcalLiBrHA (hxt!, Tl, wtout!, wt!, hvapor!, wtabsout!)
DECLARE SUB mainchoiceii ()
DECLARE SUB tabdata2 (ESTID$, mat$, con$, qout!, tcwin!, twastedesin!, twasteevapin!, tcapp!, teapp!, tdapp!, taapp!, thxapp!)
DECLARE SUB INP1WASTEDES (twastedesin!, twastedesin$, twastedesinUNIT$, trap$, aaaa!)
DECLARE SUB wasteheatevapin (twasteevapin!)
DECLARE SUB wasteheatdesin (twastedesin!)
DECLARE SUB logoTA ()
DECLARE SUB logoHA ()
DECLARE SUB hxtLiBr (tfl, wt, hxt1!)
DECLARE SUB BoostTalk (P!, wt!, tmax!)
DECLARE SUB abswtoutalk (Tl, P!, wtout!, wt!)
DECLARE SUB thxoutcalalk (hxt!, wt!, thxout!)
DECLARE SUB xhtcalalk (hxt!, Tl, wtout!, wt!, hvapor, wtabsin)
DECLARE SUB hxtalk (Tl!, wt!, hxt!)
DECLARE SUB prealk (tfl, wt!, psat!)
DECLARE SUB hgsuper (psat!, Hg!, tsat!, Tl, hsuperheat!)
DECLARE SUB TSATH2O (P!, Tl!)
DECLARE SUB abswtoutLiBr (Tl, P!, wtout!, wt!)
DECLARE SUB thxoutcalLiBr (hxt!, wt!, thxout!)
DECLARE SUB xhtcalLiBr (hxt!, Tl, wtout!, wt!, hvapor, wtabsin)
DECLARE SUB preLiBr (tfl, wt!, psat!)
DECLARE SUB BoostTLiBr (P!, wt!, tmax!)
DECLARE SUB hgwater (tfl, Hg!)
DECLARE SUB hfwater (tfl, hf!)
DECLARE SUB preh2o (tfl, P!)
DECLARE SUB TCABSIN (tcain!)
DECLARE SUB Tdesorber (tdeso!)
DECLARE SUB inptcain (tcain!, tcain$, tcainUNIT$, trap$)
DECLARE SUB inptdeso (tdeso!, tdeso$, tdesoUNIT$, trap$)
DECLARE SUB wtabs (wt$, wt!)
DECLARE SUB desTapp (tdapp!)
DECLARE SUB AbsTapp (taapp!)
DECLARE SUB HXTapp (thxapp!)
DECLARE SUB inptcapp (tcapp!, tcapp$, tcappUNIT$, trap$)
DECLARE SUB inpteapp (teapp!, teapp$, teappUNIT$, trap$)
DECLARE SUB inptdapp (tdapp!, tdapp$, tdappUNIT$, trap$)
DECLARE SUB inptaapp (taapp!, taapp$, taappUNIT$, trap$)
DECLARE SUB inpthxapp (thxapp!, thxapp$, thxappUNIT$, trap$)
DECLARE SUB condtapp (tcapp!)
DECLARE SUB evaTapp (tcapp!)
DECLARE SUB waterpre (tfl, hfl, Hg!, psat!, Tl!)
DECLARE SUB inptcwin (tcwin!, tcwin$, tcwinUNIT$, trap$, ccc!)
DECLARE SUB INP1WASTE (twaste!, twaste$, twasteUNIT$, trap$, aaa!)
DECLARE SUB wasteheatin (twaste!)
DECLARE SUB INPqout (qout!, qout$, qoutUNIT$, trap$, bbb)
DECLARE SUB RUNID (ESTID$, trap$)
DECLARE SUB condh2oin (tcwin)
DECLARE SUB qtout (qout)
DECLARE SUB tabdata (ESTID$, mat$, con$, qout!, tcwin!, twaste, tcapp, teapp, tdapp, taapp, thxapp, tcain, tdeso)
DECLARE SUB MAINMENU ()
DECLARE SUB TITLESCR ()
DECLARE SUB mainchoicei ()
DECLARE SUB mixture ()
DECLARE SUB noise ()
DECLARE SUB border ()
COMMON SHARED frg, bac, escflag
frg = 14: bac = 1

SUB condtapp (tcapp) STATIC

' ************************************************************************
' THIS PROGRAM IS USED TO INPUT THE APPROACH TEMPERATURE IN CONDENSER (tcapp)
' BY INDIVIDUAL SCREENS
' ************************************************************************

        escflag = 0
CH1INDP3:
ENTPART3:
```

```
' ***************************************************************
' USER COULD CHANGE THE UNITS BY PRESSING <F1> KEY OR
' COULD ESCAPE TO PREVIOUS MENU BY PRESSING <ESC> KEY OR
' ACCEPT THE INPUT BY PRESSING <F9>
'****************************************************************

            n = 1: tcappCF = 1: tcappUNIT$ = "Degrees F     ": tcappUNIT1$ = tcappUNIT$: tcapp = 0: tcapp$ = ""
            CLS : CALL border: : LOCATE 6, 6: PRINT "Type in the approach temperature in condenser. Use Function key <F1>"
            LOCATE 7, 6: PRINT "to select proper units for your entry. Press <F9> when finished."
            COLOR 15, bac: LOCATE 23, 7: PRINT "<F1>-TOGGLE UNITS        <F9>-CONFIRM ENTRY        <ESC>-PREVIOUS MENU"; : COLOR frg, bac
            LOCATE 19, 7: PRINT "Approach temperature in condenser    ";
PRTUNIT3:
            LOCATE 19, 56: PRINT tcappUNIT$;
20003   LOCATE 19, 40, 0: trap$ = INKEY$: IF trap$ = "" GOTO 20003
            IF trap$ = "I" GOTO 21003
            IF ASC(trap$) = 8 THEN tcapp$ = "": tcapp = 0: LOCATE 19, 45: PRINT "        ": GOTO 20003
            IF trap$ = "A" GOTO CHANGEUNIT3:
    IF ASC(trap$) = 27 THEN
    escflag = 1
    GOTO 21503
    END IF
            IF ASC(trap$) < 46 OR ASC(trap$) > 57 OR ASC(trap$) = 47 GOTO 20003
            tcapp$ = tcapp$ + trap$
            LOCATE 19, 45: PRINT VAL(tcapp$); : tcapp = VAL(tcapp$): GOTO 20003

' ***************************************************************************
' CONFIRM YOUR TEMPERATURE INPUT PRESS <F1> FOR CORRECT INPUT OR <F2> FOR
' INCORRECT INPUT
' ***************************************************************************

21003   CLS : CALL border: : LOCATE 12, 8: PRINT "Your chosen approach temperature in condenser is  "; tcapp; "  "; tcappUNIT$
            LOCATE 23, 12: PRINT "  <F1>- Correct                <F2>-Incorrect    ";
21103   DTCHECK$ = INKEY$: IF DTCHECK$ = "A" GOTO 21503
            IF DTCHECK$ = "B" THEN tcapp$ = "": GOTO ENTPART3:
            GOTO 21103

' ********************CHANGING UNITS OF TEMPERATURE*********************

CHANGEUNIT3:
            n = n + 1: IF n = 3 THEN n = 1
            ON n GOTO 21203, 21213
21203   tcappCF = 1: tcappUNIT$ = "Degrees F     ": GOTO PRTUNIT3:
21213   tcappCF = 2: tcappUNIT$ = "Degrees C     ": GOTO PRTUNIT3:

' ***************************************************************************

21503   CLS : IF tcappCF = 2 THEN tcapp = (tcapp * 1.8)
' ********************tcapp IS IN DEGREES F.*************************


END SUB

            SUB denslt (tf, wt, rhosoln) STATIC

' **********DENSITY OF (Li, K, Na) NO3 MIXTURES (rhosoln)************

' ********************** tf is in Degrees F ************************

            a11 = 1.0980812353#: a21 = -.00127501810157#
            a12 = .0024334711#: a22 = .000011473879 74#
            a13 = 7.47538E-05: a23 = -9.640613E-08
            T = (tf - 32) / 1.8
            rhosoln = a11 + a12 * wt + a13 * wt ^ 2 + (a21 + a22 * wt + a23 * wt ^ 2) * T
            rhosoln = 62.3719 * rhosoln

' ***********************rhosoln IS IN lbm/ft3*******************************

' ***************wt IS IN WT% AND T IS IN DEGREES C*****************


END SUB

SUB denh2o (tf, rhoh2o)

' ***************DENSITY OF WATER (rhoh2o)*****************

            T = tf
            a11 = 62.366496#: a21 = .005687 61#
```

```
            a12 = -1.035203E-04: a22 = 9.5E-08
            rhoh2o = a11 + a21 * T + a12 * T ^ 2 + a22 * T ^ 3

' ************************rhoh2o IS IN lbm/ft3************************

' ************************ t IS IN DEGREES F ************************


END SUB

            SUB denlibr (tf, wt, rhosoln) STATIC

' ****************DENSITY OF LiBr SOLUTIONS (rhosoln)*****************

            T = tf
            x = wt / 100
            a11 = 1.2925: a21 = -5.9576E-04
            a12 = -.64382: a22 = .0013353
            a13 = 2.3251: a23 = -.0015455
            rhosoln = a11 + a12 * x + a13 * x ^ 2 + (a21 + a22 * x + a23 * x ^ 2) * T
            rhosoln = 62.3719 * rhosoln

' ************************rhosoln IS IN lbm/ft3************************

' ***************wt IS IN WT% AND t IS IN DEGREES F*****************

END SUB

SUB desTapp (tdapp) STATIC

' ****************************************************************************
' THIS PROGRAM IS USED TO INPUT THE APPROACH TEMPERATURE IN DESORBER (tdapp)
' BY INDIVIDUAL SCREENS
' ****************************************************************************

            escflag = 0
CH1INDP5:
ENTPART5:

' ****************************************************************
' USER COULD CHANGE THE UNITS BY PRESSING <F1> KEY OR
' COULD ESCAPE TO PREVIOUS MENU BY PRESSING <ESC> KEY OR
' ACCEPT THE INPUT BY PRESSING <F9>
'****************************************************************

            n = 1: tdappCF = 1: tdappUNIT$ = "Degrees F     ": tdappUNIT1$ = tdappUNIT$: tdapp = 0: tdapp$ = ""
            CLS : CALL border: : LOCATE 6, 6: PRINT "Type in the approach temperature in desorber. Use Function key <F1>"
            LOCATE 7, 6: PRINT "to select proper units for your entry. Press <F9> when finished."
            COLOR 15, bac: LOCATE 23, 7: PRINT "<F1>-TOGGLE UNITS        <F9>-CONFIRM ENTRY        <ESC>-PREVIOUS MENU"; : COLOR frg, bac
            LOCATE 19, 7: PRINT "Approach temperature in desorber    ";
PRTUNIT5:
            LOCATE 19, 56: PRINT tdappUNIT$;
20005    LOCATE 19, 40, 0: trap$ = INKEY$: IF trap$ = "" GOTO 20005
            IF trap$ = "I" GOTO 21005
            IF ASC(trap$) = 8 THEN tdapp$ = "": tdapp = 0: LOCATE 19, 45: PRINT "       ": GOTO 20005
            IF trap$ = "A" GOTO CHANGEUNITS:
    IF ASC(trap$) = 27 THEN
    escflag = 1
    GOTO 21505
    END IF
            IF ASC(trap$) < 46 OR ASC(trap$) > 57 OR ASC(trap$) = 47 GOTO 20005
            tdapp$ = tdapp$ + trap$
            LOCATE 19, 45: PRINT VAL(tdapp$); : tdapp = VAL(tdapp$): GOTO 20005

' ****************************************************************************
' CONFIRM YOUR TEMPERATURE INPUT PRESS <F1> FOR CORRECT INPUT OR <F2> FOR
' INCORRECT INPUT
' ****************************************************************************

21005    CLS : CALL border: : LOCATE 12, 8: PRINT "Your chosen approach temperature in desorber is  "; tdapp; "  "; tdappUNIT$
            LOCATE 23, 12: PRINT "  <F1>- Correct                <F2>-Incorrect     ";
21105    DTCHECK$ = INKEY$: IF DTCHECK$ = "" GOTO 21505
            IF DTCHECK$ = "B" THEN tdapp$ = "": GOTO ENTPART5:
            GOTO 21105

' ********************CHANGING UNITS OF TEMPERATURE*********************

CHANGEUNIT5:
```

```
            n = n + 1: IF n = 3 THEN n = 1
            ON n GOTO 21205, 21215
21205   tdappCF = 1: tdappUNIT$ = "Degrees F     ": GOTO PRTUNITS:
21215   tdappCF = 2: tdappUNIT$ = "Degrees C     ": GOTO PRTUNITS:

' ***************************************************************

21505   CLS : IF tdappCF = 2 THEN tdapp = (tdapp * 1.8)

' ******************tdapp IS IN DEGREES F.*************************


END SUB

SUB evaTapp (teapp) STATIC

' ***************************************************************************
' THIS PROGRAM IS USED TO INPUT THE APPROACH TEMPERATURE IN EVAPORATOR (teapp)
' BY INDIVIDUAL SCREENS
' ***************************************************************************

            escflag = 0
CH1INDP4:
ENTPART4:

' ***************************************************************
' USER COULD CHANGE THE UNITS BY PRESSING <F1> KEY OR
' COULD ESCAPE TO PREVIOUS MENU BY PRESSING <ESC> KEY OR
' ACCEPT THE INPUT BY PRESSING <F9>
'***************************************************************

            n = 1: teappCF = 1: teappUNIT$ = "Degrees F     ": teappUNIT1$ = teappUNIT$: teapp = 0: teapp$ = ""
            CLS : CALL border: : LOCATE 6, 6: PRINT "Type in the approach temperature in evaporator. Use Function key <F1>"
            LOCATE 7, 6: PRINT "to select proper units for your entry. Press <F9> when finished."
            COLOR 15, bac: LOCATE 23, 7: PRINT "<F1>-TOGGLE UNITS       <F9>-CONFIRM ENTRY       <ESC>-PREVIOUS MENU"; : COLOR frg, bac
            LOCATE 19, 7: PRINT "Approach temperature in evaporator    ";
PRTUNIT4:
            LOCATE 19, 56: PRINT teappUNIT$;
20004   LOCATE 19, 40, 0: trap$ = INKEY$: IF trap$ = "" GOTO 20004
            IF trap$ = "I" GOTO 21004
            IF ASC(trap$) = 8 THEN teapp$ = "": teapp = 0: LOCATE 19, 45: PRINT "       ": GOTO 20004
            IF trap$ = "A" GOTO CHANGEUNIT4:
    IF ASC(trap$) = 27 THEN
    escflag = 1
    GOTO 21504
    END IF
            IF ASC(trap$) < 46 OR ASC(trap$) > 57 OR ASC(trap$) = 47 GOTO 20004
            teapp$ = teapp$ + trap$
            LOCATE 19, 45: PRINT VAL(teapp$); : teapp = VAL(teapp$): GOTO 20004

' ***************************************************************************
' CONFIRM YOUR TEMPERATURE INPUT PRESS <F1> FOR CORRECT INPUT OR <F2> FOR
' INCORRECT INPUT
' ***************************************************************************

21004   CLS : CALL border: : LOCATE 12, 8: PRINT "Your chosen approach temperature in evaporator is  "; teapp; "  "; teappUNIT$
            LOCATE 23, 12: PRINT "  <F1> - Correct                <F2> -Incorrect    ";
21104   DTCHECK$ = INKEY$: IF DTCHECK$ = "A" GOTO 21504
            IF DTCHECK$ = "B" THEN teapp$ = "": GOTO ENTPART4:
            GOTO 21104

' ******************CHANGING UNITS OF TEMPERATURE*******************
CHANGEUNIT4:
            n = n + 1: IF n = 3 THEN n = 1
            ON n GOTO 21204, 21214
21204   teappCF = 1: teappUNIT$ = "Degrees F     ": GOTO PRTUNIT4:
21214   teappCF = 2: teappUNIT$ = "Degrees C     ": GOTO PRTUNIT4:

' ***************************************************************

21504   CLS : IF teappCF = 2 THEN teapp = (teapp * 1.8)

' ******************teapp IS IN DEGREES F.*************************

END SUB

SUB hfwater (tf, hf)
```

```
' ****************************************************************
' Specific Enthalpy of water (hf) at T sat. (Deg. F) and P sat.
' Temperature range 32 to 670 Deg. F
' ****************************************************************

        T = tf
        IF T >= 32 AND T < 460 THEN
        hf = -32.3350204958# + 1.0129480405# * T - 1.296821E-04 * T ^ 2 + 3.594E-07 * T ^ 3
        IF hf < 0 THEN hf = 0
        ELSEIF T >= 460 AND T <= 670 THEN
        hf = -1013.3309563909# + 6.76439975 94# * T - .0113990799# * T ^ 2 + 7.7541E-06 * T ^ 3
        ELSEIF T > 670 THEN
        hf = -1013.3309563909# + 6.76439975 94# * T - .0113990799# * T ^ 2 + 7.7541E-06 * T ^ 3
        PRINT "WARNING: T is outside the range"
        ELSEIF T < 32 THEN
        PRINT "WARNING: T is outside the range and is less than 32 Degrees F."
        END IF

END SUB

SUB hgsuper (psat, Hg, tsat, T, hsuperheat)

' ****************************************************************************
' This program calculates the specific enthalpy of superheated steam
' (hsuperheat) at psat (psia) and T (Deg. F).
' The saturation temperature is tsat (Deg. F).
' ****************************************************************************

        tK = (T - 32) / 1.8 + 273.15
        tsatK = (tsat - 32) / 1.8 + 273.15
        a = 7.219
        B = .002374
        C = 2.67E-07
        delhsuperheat = a * (tK - tsatK) + B / 2 * (tK ^ 2 - tsatK ^ 2) + C / 3 * (tK ^ 3 - tsatK ^ 3)

' ****************delhsuperheat is in cal/gmole of steam*******************

        hsuperheat = Hg + (delhsuperheat * .1001)

' ****************************************************************************
' delhsuperheat is converted to Btu/lbm of steam by the 0.1001 factor.
'****************************************************************************

END SUB

SUB hgwater (tf, Hg)

' ****************************************************************
' Specific Enthalpy of Steam (Hg) at T sat. (Deg. F) and P sat.
' Temperature range 32 to 670 Deg. F
' ****************************************************************

        T = tf
        IF T >= 32 AND T < 460 THEN
        Hg = 1061.45416538 82# + .4207217919# * T + 1.889358E-04 * T ^ 2 - 9.194E-07 * T ^ 3
        ELSEIF T >= 460 AND T <= 670 THEN
        Hg = 2625.4135464707# - 8.7571662962999999# * T + .0181972403# * T ^ 2 - 1.27609E-05 * T ^ 3
        ELSEIF T > 670 THEN
        Hg = 2625.4135464707# - 8.7571662962999999# * T + .0181972403# * T ^ 2 - 1.27609E-05 * T ^ 3
        PRINT "WARNING: T is outside the range"
        ELSEIF T < 32 THEN
        PRINT "WARNING: T is outside the range and is less than 32 Degrees F."
        END IF

END SUB
SUB hxtalk (T1, wt, hxt)

' ****************************************************************
' Specific Enthalpy of (Li, K, Na) NO3 Mixtures (hxt)
' CORRELATION BETWEEN 50 WT% TO 94.1 WT%
' IF OUTSIDE THE RANGE OF WT%, THIS PROGRAM CALCULATES THE VALUE
' USING THE CORRELATION WITH A WARNING
' ****************************************************************

        T = T1
        IF wt >= 50 AND wt <= 94.1 THEN
        alpha = 394.516 - 8.64996 * wt + .0628787 * wt ^ 2
        beta = .388691 + 3.02719E-03 * wt - 3.80068E-05 * wt ^ 2
```

```
            ELSE
            PRINT "WARNING: wt is outside the range Cp and HXT could be inaccurate"
            alpha = 394.516 - 8.64996 * wt + .0628787 * wt ^ 2
            beta = .388691 + 3.02719E-03 * wt - 3.80068E-05 * wt ^ 2
            END IF
            hxt = alpha + beta * T

' **********************hxt IS IN Btu/lbm**********************

' ************wt IS IN WT% AND T IS IN DEGREES F*************

END SUB

SUB HXTapp (thxapp) STATIC

' ************************************************************************
' THIS PROGRAM IS USED TO INPUT THE APPROACH TEMPERATURE IN SOLUTION HEAT
' EXCHANGER (thxapp) BY INDIVIDUAL SCREENS
' ************************************************************************

            escflag = 0
CHIINDP7:
ENTPART7:

' ************************************************************
' USER COULD CHANGE THE UNITS BY PRESSING <F1> KEY OR
' COULD ESCAPE TO PREVIOUS MENU BY PRESSING <ESC> KEY OR
' ACCEPT THE INPUT BY PRESSING <F9>
'************************************************************

            n = 1: thxappCF = 1: thxappUNIT$ = "Degrees F     ": thxappUNIT1$ = thxappUNIT$: thxapp = 0: thxapp$ = ""
            CLS : CALL border: : LOCATE 6, 6: PRINT "Type in the approach temperature in solution heat exchanger. Use Function "
            LOCATE 7, 6: PRINT "key <F1> to select proper units for your entry. Press <F9> when finished."
            COLOR 15, bac: LOCATE 23, 7: PRINT "<F1>-TOGGLE UNITS        <F9>-CONFIRM ENTRY        <ESC>-PREVIOUS MENU"; : COLOR frg, bac
            LOCATE 19, 7: PRINT "Approach temperature in solution heat exchanger    ";
PRTUNIT7:
            LOCATE 19, 65: PRINT thxappUNIT$;
20007   LOCATE 19, 40, 0: trap$ = INKEY$: IF trap$ = "" GOTO 20007
            IF trap$ = "1" GOTO 21007
            IF ASC(trap$) = 8 THEN thxapp$ = "": thxapp = 0: LOCATE 19, 54: PRINT "        ": GOTO 20007
            IF trap$ = "A" GOTO CHANGEUNIT7:
    IF ASC(trap$) = 27 THEN
    escflag = 1
    GOTO 21507
    END IF
            IF ASC(trap$) < 46 OR ASC(trap$) > 57 OR ASC(trap$) = 47 GOTO 20007
            thxapp$ = thxapp$ + trap$
            LOCATE 19, 55: PRINT VAL(thxapp$); : thxapp = VAL(thxapp$): GOTO 20007

' ************************************************************************
' CONFIRM YOUR TEMPERATURE INPUT PRESS <F1> FOR CORRECT INPUT OR <F2> FOR
' INCORRECT INPUT
' ************************************************************************

21007   CLS : CALL border: : LOCATE 12, 3: PRINT "Your chosen approach temperature in heat exchanger is "; thxapp; " "; thxappUNIT$
            LOCATE 23, 12: PRINT "  <F1>- Correct                <F2>-Incorrect    ";
21107   DTCHECK$ = INKEY$: IF DTCHECK$ = "A" GOTO 21507
            IF DTCHECK$ = "B" THEN thxapp$ = "": GOTO ENTPART7:
            GOTO 21107

' **********************CHANGING UNITS OF TEMPERATURE**********************

CHANGEUNIT7:
            n = n + 1: IF n = 3 THEN n = 1
            ON n GOTO 21207, 21217
21207   thxappCF = 1: thxappUNIT$ = "Degrees F     ": GOTO PRTUNIT7:
21217   thxappCF = 2: thxappUNIT$ = "Degrees C     ": GOTO PRTUNIT7:

' ************************************************************

21507   CLS : IF thxappCF = 2 THEN thxapp = (thxapp * 1.8)

' ********************thxapp IS IN DEGREES F.**********************



END SUB
```

```
SUB hxtLiBr (tf, wt, hxt1) STATIC

' ***********SPECIFIC ENTHALPY OF LiBr SOLUTIONS (hxt1)***********

            a0 = -1015.07 + 79.5387 * wt - 2.35602 * wt ^ 2 + .0303158 * wt ^ 3 - 1.4003E-04 * wt ^ 4
            a1 = 4.68108 - .303777 * wt + 8.44845E-03 * wt ^ 2 - 1.04772E-04 * wt ^ 3 + 4.801E-07 * wt ^ 4
            a2 = -.0049107 + 3.83184E-04 * wt - 1.079E-05 * wt ^ 2 + 1.3152E-07 * wt ^ 3 - 5.897E-10 * wt ^ 4
            T = tf
            hxt1 = (a0 + a1 * T + a2 * T ^ 2)

' ************************hxt1 IS IN Btu/lbm************************

' ************************t IS IN DEGREES F*********************

END SUB

SUB INPqout (qout, qout$, qoutUNIT$, trap$, bbb) STATIC

' ********************************************************************
' THIS PROGRAM IS USED TO INPUT CAPACITY (qout) IN TABULAR FORMAT
' ********************************************************************

LOCATE 6, (3 + bbb + 1): COLOR bac, frg: PRINT SPACE$(8)
LOCATE 6, (3 + bbb + 1): PRINT qout$
qoutCF = 1
LOCATE 6, 35: COLOR frg, bac
25    LOCATE 6, (3 + bbb + 12): PRINT qoutUNIT$
TRAPP:
            trap$ = ""
WHILE trap$ = ""
            trap$ = INKEY$
WEND
 IF ASC(trap$) = 8 THEN
            qout$ = "": qout = 0: LOCATE 6, (3 + bbb + 1): COLOR bac, frg: PRINT SPACE$(8)
            GOTO TRAPP:
            END IF
IF ASC(trap$) = 9 OR ASC(trap$) = 13 OR ASC(trap$) = 27 THEN GOTO DATAF:
IF trap$ = "A" GOTO CHANGEU:
            IF ASC(trap$) < 46 OR ASC(trap$) > 57 OR ASC(trap$) = 47 GOTO TRAPP:
IF LEN(qout$) < 5 THEN
            qout$ = qout$ + trap$
ELSE
            CALL noise
            GOTO TRAPP:
END IF
            LOCATE 6, (3 + bbb + 1): COLOR bac, frg: PRINT qout$; : qout = VAL(qout$): GOTO TRAPP:
CHANGEU:

' ***************************CHANGING UNITS**********************

LOCATE 6, 35: COLOR bac, frg
            n = n + 1: IF n = 5 THEN n = 1
            ON n GOTO 26, 27, 28, 29
26    qoutCF = 1: qoutUNIT$ = " Btu/hr    ": GOTO 25
27    qoutCF = 1 / 1055: qoutUNIT$ = " J/hr      ": GOTO 25
28    qoutCF = 1000000!: qoutUNIT$ = " MM Btu/hr ": GOTO 25
29    qoutCF = 3412.32: qoutUNIT$ = "  kW       ": GOTO 25
DATAF:
            qout = qout * qoutCF

' ***********************qout IS IN Btu/hr**********************

LOCATE 6, (3 + bbb + 1): COLOR frg, bac: PRINT SPACE$(8)
LOCATE 6, (3 + bbb + 1): PRINT qout$
LOCATE 6, (3 + bbb + 12): PRINT qoutUNIT$

END SUB

SUB inptaapp (taapp, taapp$, taappUNIT$, trap$) STATIC
' ********************************************************************
' THIS PROGRAM IS USED TO INPUT APPROACH TEMPERATURE IN ABSORBER (taapp)
' IN TABULAR FORMAT
' ********************************************************************

LOCATE 11, 38: COLOR bac, frg: PRINT SPACE$(8)
LOCATE 11, 38: PRINT taapp$
LOCATE 11, 51: COLOR frg, bac
```

```
taappCF = 1
388 LOCATE 11, 51: PRINT taappUNIT$
398   trap$ = ""
WHILE trap$ = ""
            trap$ = INKEY$
WEND

            IF ASC(trap$) = 8 THEN
            taapp$ = "": taapp = 0: LOCATE 11, 38: COLOR bac, frg: PRINT SPACE$(8)
            GOTO 398
            END IF
IF ASC(trap$) = 9 OR ASC(trap$) = 13 OR ASC(trap$) = 27 THEN GOTO DATAH8:
IF trap$ = "A" GOTO CHANGEUNI8:
            IF ASC(trap$) < 46 OR ASC(trap$) > 57 OR ASC(trap$) = 47 GOTO 398
IF LEN(taapp$) < 5 THEN
            taapp$ = taapp$ + trap$
            LOCATE 11, 38: COLOR bac, frg: PRINT taapp$; : taapp = VAL(taapp$): GOTO 398
ELSE
            CALL noise
            GOTO 398
END IF
CHANGEUNI8:

' **********************CHANGING UNITS*************************

LOCATE 11, 51: COLOR bac, frg
            n = n + 1: IF n = 3 THEN n = 1
            ON n GOTO 418, 428
418   taappCF = 1: taappUNIT$ = " Deg. F": GOTO 388
428   taappCF = 2: taappUNIT$ = " Deg. C": GOTO 388
DATAH8:

' ***************taapp IS IN DEGREES F.**********************

            IF taappCF = 2 THEN taapp = (taapp * 1.8)

LOCATE 11, 38: COLOR frg, bac: PRINT SPACE$(8)
LOCATE 11, 38: PRINT taapp$
LOCATE 11, 51: PRINT taappUNIT$

END SUB

SUB inptcain (tcain, tcain$, tcainUNIT$, trap$) STATIC

' ************************************************************************
' THIS PROGRAM IS USED TO INPUT COOLING WATER TEMPERATURE ENTERING THE
' ABSORBER (tcain) IN TABULAR FORMAT
' ************************************************************************

LOCATE 13, 53: COLOR bac, frg: PRINT SPACE$(8)
LOCATE 13, 53: PRINT tcain$
LOCATE 13, 66: COLOR frg, bac
tcainCF = 1
3899 LOCATE 13, 66: PRINT tcainUNIT$
3999   trap$ = ""
WHILE trap$ = ""
            trap$ = INKEY$
WEND

            IF ASC(trap$) = 8 THEN
            tcain$ = "": tcain = 0: LOCATE 13, 53: COLOR bac, frg: PRINT SPACE$(8)
            GOTO 3999
            END IF
IF ASC(trap$) = 9 OR ASC(trap$) = 13 OR ASC(trap$) = 27 THEN GOTO DATAH99:
IF trap$ = "A" GOTO CHANGEUNI99:
            IF ASC(trap$) < 46 OR ASC(trap$) > 57 OR ASC(trap$) = 47 GOTO 3999
IF LEN(tcain$) < 5 THEN
            tcain$ = tcain$ + trap$
            LOCATE 13, 53: COLOR bac, frg: PRINT tcain$; : tcain = VAL(tcain$): GOTO 3999
ELSE
            CALL noise
            GOTO 3999
END IF
CHANGEUNI99:

' **********************CHANGING UNITS*************************

LOCATE 13, 66: COLOR bac, frg
```

```
            n = n + 1: IF n = 3 THEN n = 1
            ON n GOTO 4199, 4299
4199    tcainCF = 1: tcainUNIT$ = " Deg. F": GOTO 3899
4299    tcainCF = 2: tcainUNIT$ = " Deg. C": GOTO 3899
DATAH99:

' ***************tcain IS IN DEGREES F.**********************

            IF tcainCF = 2 THEN tcain = (tcain * 1.8) + 32

LOCATE 13, 53: COLOR frg, bac: PRINT SPACE$(8)
LOCATE 13, 53: PRINT tcain$
LOCATE 13, 66: PRINT tcainUNIT$

END SUB

SUB inptcapp (tcapp, tcapp$, tcappUNIT$, trap$) STATIC

' ************************************************************************
' THIS PROGRAM IS USED TO INPUT APPROACH TEMPERATURE IN CONDENSER (tcapp)
' IN TABULAR FORMAT
' ************************************************************************

LOCATE 8, 38: COLOR bac, frg: PRINT SPACE$(8)
LOCATE 8, 38: PRINT tcapp$
LOCATE 8, 51: COLOR frg, bac
tcappCF = 1
385 LOCATE 8, 51: PRINT tcappUNIT$
395   trap$ = ""
WHILE trap$ = ""
            trap$ = INKEY$
WEND

            IF ASC(trap$) = 8 THEN
            tcapp$ = "": tcapp = 0: LOCATE 8, 38: COLOR bac, frg: PRINT SPACE$(8)
            GOTO 395
            END IF
IF ASC(trap$) = 9 OR ASC(trap$) = 13 OR ASC(trap$) = 27 THEN GOTO DATAH5:
IF trap$ = "A" GOTO CHANGEUNI5:
            IF ASC(trap$) < 46 OR ASC(trap$) > 57 OR ASC(trap$) = 47 GOTO 395
IF LEN(tcapp$) < 5 THEN
            tcapp$ = tcapp$ + trap$
            LOCATE 8, 38: COLOR bac, frg: PRINT tcapp$; : tcapp = VAL(tcapp$): GOTO 395
ELSE
            CALL noise
            GOTO 395
END IF
CHANGEUNI5:

' **********************CHANGING UNITS************************

LOCATE 8, 51: COLOR bac, frg
            n = n + 1: IF n = 3 THEN n = 1
            ON n GOTO 415, 425

415   tcappCF = 1: tcappUNIT$ = " Deg. F": GOTO 385
425   tcappCF = 2: tcappUNIT$ = " Deg. C": GOTO 385
DATAH5:

' ***************tcapp IS IN DEGREES F.**********************

            IF tcappCF = 2 THEN tcapp = (tcapp * 1.8)

LOCATE 8, 38: COLOR frg, bac: PRINT SPACE$(8)
LOCATE 8, 38: PRINT tcapp$
LOCATE 8, 51: PRINT tcappUNIT$

END SUB

SUB inptcwin (tcwin, tcwin$, tcwinUNIT$, trap$, ccc) STATIC

' ************************************************************************
' THIS PROGRAM IS USED TO INPUT CONDENSER WATER INLET TEMPERATURE (tcwin)
' IN TABULAR FORMAT
' ************************************************************************

LOCATE 7, (3 + ccc + 1): COLOR bac, frg: PRINT SPACE$(8)
LOCATE 7, (3 + ccc + 1): PRINT tcwin$
```

```
LOCATE 7, (3 + ccc + 12): COLOR frg, bac
tcwinCF = 1
38 LOCATE 7, (3 + ccc + 12): PRINT tcwinUNIT$
39  trap$ = ""
WHILE trap$ = ""
        trap$ = INKEY$
WEND

        IF ASC(trap$) = 8 THEN
        tcwin$ = "": tcwin = 0: LOCATE 7, (3 + ccc + 1): COLOR bac, frg: PRINT SPACE$(8)
        GOTO 39
        END IF
IF ASC(trap$) = 9 OR ASC(trap$) = 13 OR ASC(trap$) = 27 THEN GOTO DATAH:
IF trap$ = "A" GOTO CHANGEUNI:
        IF ASC(trap$) < 46 OR ASC(trap$) > 57 OR ASC(trap$) = 47 GOTO 39
IF LEN(tcwin$) < 5 THEN
        tcwin$ = tcwin$ + trap$
        LOCATE 7, (3 + ccc + 1): COLOR bac, frg: PRINT tcwin$; : tcwin = VAL(tcwin$): GOTO 39
ELSE
        CALL noise
        GOTO 39
END IF
CHANGEUNI:

' **********************CHANGING UNITS*************************

LOCATE 7, (3 + ccc + 12): COLOR bac, frg
        n = n + 1: IF n = 3 THEN n = 1
        ON n GOTO 41, 42

41   tcwinCF = 1: tcwinUNIT$ = " Deg. F": GOTO 38
42   tcwinCF = 2: tcwinUNIT$ = " Deg. C": GOTO 38
DATAH:

' ***************tcwin IS IN DEGREES F.**********************

        IF tcwinCF = 2 THEN tcwin = (tcwin * 1.8) + 32

LOCATE 7, (3 + ccc + 1): COLOR frg, bac: PRINT SPACE$(8)
LOCATE 7, (3 + ccc + 1): PRINT tcwin$
LOCATE 7, (3 + ccc + 12): PRINT tcwinUNIT$

END SUB

SUB inptdapp (tdapp, tdapp$, tdappUNIT$, trap$) STATIC

' ************************************************************************
' THIS PROGRAM IS USED TO INPUT APPROACH TEMPERATURE IN DESORBER (tdapp)
' IN TABULAR FORMAT
' ************************************************************************

LOCATE 10, 38: COLOR bac, frg: PRINT SPACE$(8)
LOCATE 10, 38: PRINT tdapp$
LOCATE 10, 51: COLOR frg, bac
tdappCF = 1
387 LOCATE 10, 51: PRINT tdappUNIT$
397  trap$ = ""
WHILE trap$ = ""
        trap$ = INKEY$
WEND

        IF ASC(trap$) = 8 THEN
        tdapp$ = "": tdapp = 0: LOCATE 10, 38: COLOR bac, frg: PRINT SPACE$(8)
        GOTO 397
        END IF
IF ASC(trap$) = 9 OR ASC(trap$) = 13 OR ASC(trap$) = 27 THEN GOTO DATAH7:
IF trap$ = "A" GOTO CHANGEUNI7:
        IF ASC(trap$) < 46 OR ASC(trap$) > 57 OR ASC(trap$) = 47 GOTO 397
IF LEN(tdapp$) < 5 THEN
        tdapp$ = tdapp$ + trap$
        LOCATE 10, 38: COLOR bac, frg: PRINT tdapp$; : tdapp = VAL(tdapp$): GOTO 397
ELSE
        CALL noise
        GOTO 397
END IF
CHANGEUNI7:

' **********************CHANGING UNITS*************************
```

```
LOCATE 10, 51: COLOR bac, frg
        n = n + 1: IF n = 3 THEN n = 1
        ON n GOTO 417, 427

417   tdappCF = 1: tdappUNIT$ = " Deg. F": GOTO 387
427   tdappCF = 2: tdappUNIT$ = " Deg. C": GOTO 387
DATAH7:

' ***************tdapp IS IN DEGREES F.**********************

        IF tdappCF = 2 THEN tdapp = (tdapp * 1.8)

LOCATE 10, 38: COLOR frg, bac: PRINT SPACE$(8)
LOCATE 10, 38: PRINT tdapp$
LOCATE 10, 51: PRINT tdappUNIT$

END SUB

SUB inptdeso (tdeso, tdeso$, tdesoUNIT$, trap$) STATIC

' ************************************************************
' THIS PROGRAM IS USED TO INPUT DESORBER TEMPERATURE (tdeso)
' IN TABULAR FORMAT
' ************************************************************
entertdeso:

LOCATE 14, 27: COLOR bac, frg: PRINT SPACE$(8)
LOCATE 14, 27: PRINT tdeso$
LOCATE 14, 40: COLOR frg, bac
tdesoCF = 1
3898 LOCATE 14, 40: PRINT tdesoUNIT$
3998  trap$ = ""
WHILE trap$ = ""
        trap$ = INKEY$
WEND

        IF ASC(trap$) = 8 THEN
        tdeso$ = "": tdeso = 0: LOCATE 14, 27: COLOR bac, frg: PRINT SPACE$(8)
        GOTO 3998
        END IF
IF ASC(trap$) = 9 OR ASC(trap$) = 13 OR ASC(trap$) = 27 THEN GOTO DATAH98:
IF trap$ = "A" GOTO CHANGEUNI98:
        IF ASC(trap$) < 46 OR ASC(trap$) > 57 OR ASC(trap$) = 47 GOTO 3998
IF LEN(tdeso$) < 5 THEN
        tdeso$ = tdeso$ + trap$
        LOCATE 14, 27: COLOR bac, frg: PRINT tdeso$; : tdeso = VAL(tdeso$): GOTO 3998
ELSE
        CALL noise
        GOTO 3998
END IF
CHANGEUNI98:

' **********************CHANGING UNITS**************************

LOCATE 14, 40: COLOR bac, frg
        n = n + 1: IF n = 3 THEN n = 1
        ON n GOTO 4198, 4298
4198   tdesoCF = 1: tdesoUNIT$ = " Deg. F": GOTO 3898
4298   tdesoCF = 2: tdesoUNIT$ = " Deg. C": GOTO 3898
DATAH98:

' ****************tdeso IS IN DEGREES F.**********************

        IF tdesoCF = 2 THEN tdeso = (tdeso * 1.8) + 32

LOCATE 14, 27: COLOR frg, bac: PRINT SPACE$(8)
LOCATE 14, 27: PRINT tdeso$
LOCATE 14, 40: PRINT tdesoUNIT$

END SUB

SUB inpteapp (teapp, teapp$, teappUNIT$, trap$) STATIC

' ************************************************************************
' THIS PROGRAM IS USED TO INPUT APPROACH TEMPERATURE IN EVAPORATOR (teapp)
' IN TABULAR FORMAT
' ************************************************************************
```

```
LOCATE 9, 40: COLOR bac, frg: PRINT SPACE$(8)
LOCATE 9, 40: PRINT teapp$
LOCATE 9, 53: COLOR frg, bac
teappCF = 1
386 LOCATE 9, 53: PRINT teappUNIT$
396   trap$ = ""
WHILE trap$ = ""
            trap$ = INKEY$
WEND

            IF ASC(trap$) = 8 THEN
            teapp$ = "": teapp = 0: LOCATE 9, 40: COLOR bac, frg: PRINT SPACE$(8)
            GOTO 396
            END IF
IF ASC(trap$) = 9 OR ASC(trap$) = 13 OR ASC(trap$) = 27 THEN GOTO DATAH6:
IF trap$ = "A" GOTO CHANGEUNI6:
            IF ASC(trap$) < 46 OR ASC(trap$) > 57 OR ASC(trap$) = 47 GOTO 396
IF LEN(teapp$) < 5 THEN
            teapp$ = teapp$ + trap$
            LOCATE 9, 40: COLOR bac, frg: PRINT teapp$; : teapp = VAL(teapp$): GOTO 396
ELSE
            CALL noise
            GOTO 396
END IF
CHANGEUNI6:

' **********************CHANGING UNITS************************

LOCATE 9, 53: COLOR bac, frg
            n = n + 1: IF n = 3 THEN n = 1
            ON n GOTO 416, 426

416   teappCF = 1: teappUNIT$ = " Deg. F": GOTO 386
426   teappCF = 2: teappUNIT$ = " Deg. C": GOTO 386
DATAH6:

' ****************teapp IS IN DEGREES F.************************

            IF teappCF = 2 THEN teapp = (teapp * 1.8)

LOCATE 9, 40: COLOR frg, bac: PRINT SPACE$(8)
LOCATE 9, 40: PRINT teapp$
LOCATE 9, 53: PRINT teappUNIT$

END SUB

SUB inpthxapp (thxapp, thxapp$, thxappUNIT$, trap$) STATIC

' ************************************************************************
' THIS PROGRAM IS USED TO INPUT APPROACH TEMPERATURE IN SOLUTION HEAT
' EXCHANGER (thxapp) IN TABULAR FORMAT
' ************************************************************************

LOCATE 12, 53: COLOR bac, frg: PRINT SPACE$(8)
LOCATE 12, 53: PRINT thxapp$
LOCATE 12, 66: COLOR frg, bac
thxappCF = 1
389 LOCATE 12, 66: PRINT thxappUNIT$
399   trap$ = ""
WHILE trap$ = ""
            trap$ = INKEY$
WEND

            IF ASC(trap$) = 8 THEN
            thxapp$ = "": thxapp = 0: LOCATE 12, 53: COLOR bac, frg: PRINT SPACE$(8)
            GOTO 399
            END IF
IF ASC(trap$) = 9 OR ASC(trap$) = 13 OR ASC(trap$) = 27 THEN GOTO DATAH9:
IF trap$ = "A" GOTO CHANGEUNI9:
            IF ASC(trap$) < 46 OR ASC(trap$) > 57 OR ASC(trap$) = 47 GOTO 399
IF LEN(thxapp$) < 5 THEN
            thxapp$ = thxapp$ + trap$
            LOCATE 12, 53: COLOR bac, frg: PRINT thxapp$; : thxapp = VAL(thxapp$): GOTO 399
ELSE
            CALL noise
            GOTO 399
END IF
CHANGEUNI9:
```

```
' **********************CHANGING UNITS**************************

LOCATE 12, 66: COLOR bac, frg
            n = n + 1: IF n = 3 THEN n = 1
            ON n GOTO 419, 429
419   thxappCF = 1: thxappUNIT$ = " Deg. F": GOTO 389
429   thxappCF = 2: thxappUNIT$ = " Deg. C": GOTO 389
DATAH9:

' ***************thxapp IS IN DEGREES F.**********************

            IF thxappCF = 2 THEN thxapp = (thxapp * 1.8)

LOCATE 12, 53: COLOR frg, bac: PRINT SPACE$(8)
LOCATE 12, 53: PRINT thxapp$
LOCATE 12, 66: PRINT thxappUNIT$

END SUB

SUB INPTWASTE (twaste, twaste$, twasteUNIT$, trap$, aaa) STATIC

' ************************************************************************
' THIS PROGRAM IS USED TO INPUT THE WASTE INLET TEMPERATURE (twaste)
' IN TABULAR FORM
' ************************************************************************

LOCATE 5, (3 + aaa + 1): COLOR bac, frg: PRINT SPACE$(8)
LOCATE 5, (3 + aaa + 1): PRINT twaste$
LOCATE 5, (3 + aaa + 12): COLOR frg, bac
twasteCF = 1
3800  LOCATE 5, (3 + aaa + 12): PRINT twasteUNIT$
3900  trap$ = ""
WHILE trap$ = ""
            trap$ = INKEY$
WEND

            IF ASC(trap$) = 8 THEN
            twaste$ = "": twaste = 0: LOCATE 5, (3 + aaa + 1): COLOR bac, frg: PRINT SPACE$(8)
            GOTO 3900
            END IF
IF ASC(trap$) = 9 OR ASC(trap$) = 13 OR ASC(trap$) = 27 THEN GOTO DATAH2:
IF trap$ = "A" GOTO CHANGEUNI2:
            IF ASC(trap$) < 46 OR ASC(trap$) > 57 OR ASC(trap$) = 47 GOTO 3900
IF LEN(twaste$) < 5 THEN
            twaste$ = twaste$ + trap$
            LOCATE 5, (3 + aaa + 1): COLOR bac, frg: PRINT twaste$; : twaste = VAL(twaste$): GOTO 3900
ELSE
            CALL noise
            GOTO 3900
END IF
CHANGEUNI2:

' **********************CHANGING UNITS**************************

LOCATE 5, (3 + ccc + 12): COLOR bac, frg
            n = n + 1: IF n = 3 THEN n = 1
            ON n GOTO 4100, 4200

4100  twasteCF = 1: twasteUNIT$ = " Deg. F": GOTO 3800
4200  twasteCF = 2: twasteUNIT$ = " Deg. C": GOTO 3800
DATAH2:

' ****************twaste IS IN DEGREES F.**********************

            IF twasteCF = 2 THEN twaste = (twaste * 1.8) + 32

LOCATE 5, (3 + aaa + 1): COLOR frg, bac: PRINT SPACE$(8)
LOCATE 5, (3 + aaa + 1): PRINT twaste$
LOCATE 5, (3 + aaa + 12): PRINT twasteUNIT$

END SUB

SUB logoHA

SCREEN 9
COLOR 14, 1
CLS
```

```
' ************************** Heat Amplifier (HA) ******************************

' Define a viewport and draw a border around it:
VIEW (20, 20)-(639, 199), , 1

' Redefine the coordinates of the viewport with logical
' coordinates:

WINDOW (0, 0)-(10, 10)

' ***************** Desorber Coolant Flow Direction*****************
LINE (5.1, 8.9)-(5.4, 8.9)
LINE -(5.25, 8.8)
LINE (5.4, 8.9)-(5.25, 9)

LINE (5.35, 8.6)-(5.05, 8.6)
LINE -(5.2, 8.7)
LINE (5.05, 8.6)-(5.2, 8.5)

' ***************** Absorber Waste Heat Flow Direction*****************
LINE (5.35, 4.2)-(5.65, 4.2)
LINE -(5.5, 4.1)
LINE (5.65, 4.2)-(5.5, 4.3)

LINE (5.6, 3.9)-(5.3, 3.9)
LINE -(5.45, 3.8)
LINE (5.3, 3.9)-(5.45, 4)

' ***************** Condenser Waste Heat Flow Direction*****************
LINE (2.9, 3.8)-(2.6, 3.8)
LINE -(2.75, 3.7)
LINE (2.6, 3.8)-(2.75, 3.9)

LINE (2.5, 3.5)-(2.8, 3.5)
LINE -(2.65, 3.4)
LINE (2.8, 3.5)-(2.65, 3.6)

' ***************** Evaporator Coolant Flow Direction *****************
LINE (3.4, 2.1)-(3.1, 2.1)
LINE -(3.25, 2.25)
LINE (3.1, 2.1)-(3.25, 2)

LINE (3.05, 1.8)-(3.35, 1.8)
LINE -(3.2, 1.7)
LINE (3.35, 1.8)-(3.2, 1.9)

' ************ Flow Direction of Refrigerant leaving the Condenser *********
LINE (1.25, 2.4)-(1.25, 2.7)
LINE -(1.29, 2.9)
LINE (1.25, 2.7)-(1.21, 2.9)

LINE (1.25, .7)-(1.25, 1)
LINE -(1.29, 1.2)
LINE (1.25, 1)-(1.21, 1.2)

' ************ Flow Direction of Refrigerant Leaving the Desorber ************
LINE (2.25, 6.3)-(2.25, 6)
LINE -(2.29, 6.2)
LINE (2.25, 6)-(2.21, 6.2)

LINE (4.3, 9.5)-(4, 9.5)
LINE -(4.15, 9.4)
LINE (4, 9.5)-(4.15, 9.6)

' ************ Flow Direction of Solution Entering the Desorber ************
LINE (8, 7.7)-(8, 8)
LINE -(8.04, 7.8)
LINE (8, 8)-(7.96, 7.8)

LINE (8, 4.7)-(8, 5)
LINE -(8.04, 4.8)
LINE (8, 5)-(7.96, 4.8)

' ************ Flow Direction of Solution Leaving the Desorber *************
LINE (7.2, 7.7)-(7.2, 7.4)
LINE -(7.24, 7.6)
LINE (7.2, 7.4)-(7.16, 7.6)
```

```
' ********** Flow Direction of Refrigerant entering the Absorber **********
LINE (4.5, 3)-(4.5, 3.3)
LINE -(4.54, 3.2)
LINE (4.5, 3.3)-(4.46, 3.2)

LINE (4.7, 4.7)-(5, 4.7)
LINE -(4.85, 4.6)
LINE (5, 4.7)-(4.85, 4.8)

' ********** Flow Direction of Solution Leaving the Absorber **********
LINE (6.75, 2.5)-(6.75, 2.2)
LINE -(6.79, 2.4)
LINE (6.75, 2.2)-(6.71, 2.4)

' ********** Desorber **********
LINE (5.5, 9.9)-(7.5, 9.9), , B
LINE -(7.5, 7.9), , B
LINE -(5.5, 7.9), , B
LINE -(5.5, 9.9), , B

' ********** Desorber Coolant **********
LINE (5, 8.6)-(7, 8.6), , B
LINE -(7, 8.9), , B
LINE -(5, 8.9), , B

' ********** Solution in Desorber **********
LINE (5.5, 8.3)-(7.5, 8.3), , B
PAINT (6.5, 8.1), 2, 14

' ********** Refrigerant Leaving the Desorber **********
LINE (2.25, 4.75)-(2.25, 9.5), , B
LINE -(5.5, 9.5), , B
LINE (2.25, 4.75)-(2.25, 4.2)
LINE -(2, 4.2)
' ********** Solution Entering the Desorber **********
LINE (5.6, 9.7)-(8, 9.7), , B
LINE -(8, 1.65), , B
LINE -(7.125, 1.65), , B

LINE (5.6, 9.7)-(5.65, 9.6)
LINE (5.6, 9.7)-(5.55, 9.6)

LINE (6.1, 9.7)-(6.15, 9.6)
LINE (6.1, 9.7)-(6.05, 9.6)

LINE (6.6, 9.7)-(6.65, 9.6)
LINE (6.6, 9.7)-(6.55, 9.6)

LINE (7.1, 9.7)-(7.15, 9.6)
LINE (7.1, 9.7)-(7.05, 9.6)

' ********** Solution Leaving the Absorber **********
LINE (6.75, 3.2)-(6.75, 1.25), , B
LINE -(7.125, 1.25), , B

' ********** Solution Leaving the Desorber **********
LINE (7.2, 7.9)-(7.2, 5.2), , B

' ********** Throttled Valve (Absorber) **********
LINE (7.1, 5.2)-(7.3, 5.2), , B
LINE -(7.2, 4.9)
LINE -(7.1, 5.2)
PAINT (7.2, 4.99), 6, 14

LINE (7.1, 4.6)-(7.3, 4.6), , B
LINE -(7.2, 4.9)
LINE -(7.1, 4.6)
PAINT (7.2, 4.7), 6, 14

' ********** Throttled Valve (Evaporator) **********
LINE (2.4, 1)-(2.6, 1), , B
LINE -(2.5, .7)
LINE -(2.4, 1)
PAINT (2.5, .89), 6, 14

LINE (2.4, .4)-(2.6, .4), , B
LINE -(2.5, .7)
LINE -(2.4, .4)
```

```
PAINT (2.5, .5), 6, 14

' ********** Solution Heat Exchanger **********
LINE (7, 6.9)-(8.2, 6.9), , B
LINE -(8.2, 5.9), , B
LINE -(7, 5.9), , B
LINE -(7, 6.9), , B

' ********** Tubes Shown in the Solution Heat Exchanger **********
LINE (7.125, 6.65)-(8.075, 6.65), , B
LINE (7.125, 6.4)-(8.075, 6.4), , B
LINE (7.125, 6.2)-(8.075, 6.2), , B

' ********** Absorber **********
LINE (5.75, 4.9)-(7.75, 4.9), , B
LINE -(7.75, 3.2), , B
LINE -(5.75, 3.2), , B
LINE -(5.75, 4.9), , B
CIRCLE (7.125, 1.25), .175

' ********** Solution in the Absorber **********
LINE (5.75, 3.6)-(7.75, 3.6), , B
PAINT (6.5, 3.3), 2, 14

' ********** Waste Heat to the Absorber **********
LINE (5.25, 4.2)-(7.2, 4.2), , B
LINE -(7.2, 3.9), , B
LINE -(5.25, 3.9), , B

' ********** Refrigerant entering the Absorber **********
LINE (5.75, 4.7)-(4.5, 4.7), , B
LINE -(4.5, 2.4), , B
LINE -(3, 2.4), , B

' ********** Evaporator **********
LINE (1.5, 2.6)-(3, 2.6), , B
LINE -(3, 1), , B
LINE -(1.5, 1), , B
LINE -(1.5, 2.6), , B

' ********** Refrigerant in the Evaporator **********
LINE (1.5, 2.3)-(3, 2.3), , B
PAINT (2.5, 2), 7, 14

' ********** Coolant in the Evaporator **********
LINE (3.5, 2.1)-(1.75, 2.1), , B
LINE -(1.75, 1.8), , B
LINE -(3.5, 1.8), , B

' ********** Refrigerant Entering the Evaporator **********
LINE (2.5, .4)-(2.5, .075), , B

' ********** Refrigerant Entering the Condenser **********
LINE (2.5, .075)-(1.25, .075), , B
LINE -(1.25, 3.2), , B

' ********** Condenser **********
LINE (1, 4.4)-(2, 4.4), , B
LINE -(2, 3.2), , B
LINE -(1, 3.2), , B
LINE -(2, 4.4), , B

' ********** Refrigerant in the Condenser **********
LINE (1, 3.4)-(2, 3.4), , B
PAINT (1.5, 3.3), 7, 14

' ********** Waste Heat in the Condenser **********
LINE (3, 3.8)-(1.5, 3.8), , B
LINE -(1.5, 3.5), , B
LINE -(3, 3.5), , B

' ********** Legend of the Figure **********
LINE (.05, 9.99)-(.25, 9.99), , B
LINE -(.25, 9.79), , B
LINE -(.05, 9.79), , B
LINE -(.05, 9.99), , B
PAINT (.15, 9.89), 2, 14
```

```
LINE (.05, 9.3)-(.25, 9.3), , B
LINE -(.25, 9.1), , B
LINE -(.05, 9.1), , B
LINE -(.05, 9.3), , B
PAINT (.15, 9.2), 7, 14

LINE (.05, 8.05)-(.25, 8.05), , B
LINE -(.15, 8.35)
LINE -(.05, 8.05)
PAINT (.15, 8.15), 6, 14

LINE (.05, 8.65)-(.25, 8.65), , B
LINE -(.15, 8.35)
LINE -(.05, 8.65)
PAINT (.15, 8.5), 6, 14

LOCATE 1, 2: PRINT CHR$(218)
LOCATE 5, 2: PRINT CHR$(192)
FOR j = 1 TO 15: LOCATE 1, 2 + j: PRINT CHR$(196): NEXT j
FOR j = 1 TO 15: LOCATE 5, 2 + j: PRINT CHR$(196): NEXT j
FOR j = 1 TO 3: LOCATE 1 + j, 17: PRINT CHR$(179): NEXT j
FOR j = 1 TO 3: LOCATE 1 + j, 2: PRINT CHR$(179): NEXT j
LOCATE 1, 17: PRINT CHR$(191)
LOCATE 5, 17: PRINT CHR$(217)
LOCATE 2, 6
PRINT "Solution"
LOCATE 3, 6
PRINT "Refrigerant"
LOCATE 4, 6
PRINT "Throttle"
LOCATE 1, 50
PRINT "Desorber"
LOCATE 8, 10
PRINT "Condenser"
LOCATE 14, 25
PRINT "Evaporator"
LOCATE 7, 47
PRINT "Absorber"
LOCATE 5, 69
PRINT "Solution"
LOCATE 6, 69:
PRINT "  Heat"
LOCATE 7, 69
PRINT "Exchanger"
LOCATE 18, 24
PRINT "Schematic of Chemical Heat Pump (HA)"
LOCATE 23, 27
PRINT "Press any key to continue"
pause$ = INPUT$(1)
CLS : SCREEN 0
END SUB

 SUB logoTA
SCREEN 9
COLOR frg, bac
CLS

' ********** Temperature Amplifier (TA) **********

' Define a viewport and draw a border around it:
VIEW (20, 20)-(639, 199), , 1

' Redefine the coordinates of the viewport with logical
' coordinates:

WINDOW (0, 0)-(10, 10)

' ********** Absorber Coolant Flow Direction**********
LINE (5.1, 8.9)-(5.4, 8.9)
LINE -(5.25, 8.8)
LINE (5.4, 8.9)-(5.25, 9)

LINE (5.35, 8.6)-(5.05, 8.6)
LINE -(5.2, 8.7)
LINE (5.05, 8.6)-(5.2, 8.5)

' ********** Desorber Waste Heat Flow Direction**********
LINE (5.35, 4.2)-(5.65, 4.2)
```

```
LINE -(5.5, 4.1)
LINE (5.65, 4.2)-(5.5, 4.3)

LINE (5.6, 3.9)-(5.3, 3.9)
LINE -(5.45, 3.8)
LINE (5.3, 3.9)-(5.45, 4)

' ********** Evaporator Waste Heat Flow Direction**********
LINE (2.9, 3.8)-(2.6, 3.8)
LINE -(2.75, 3.7)
LINE (2.6, 3.8)-(2.75, 3.9)

LINE (2.5, 3.5)-(2.8, 3.5)
LINE -(2.65, 3.4)
LINE (2.8, 3.5)-(2.65, 3.6)

' ********** Condenser Coolant Flow Direction**********
LINE (3.4, 2.1)-(3.1, 2.1)
LINE -(3.25, 2.25)
LINE (3.1, 2.1)-(3.25, 2)
LINE (3.05, 1.8)-(3.35, 1.8)
LINE -(3.2, 1.7)
LINE (3.35, 1.8)-(3.2, 1.9)

' ********** Flow Direction of Refrigerant Leaving the Condenser **********
LINE (2.5, 1.35)-(2.5, 1.05)
LINE -(2.54, 1.25)
LINE (2.5, 1.05)-(2.46, 1.25)

' ********** Flow Direction of Refrigerant Entering the Evaporator *********
LINE (1.25, 2.6)-(1.25, 2.9)
LINE -(1.29, 2.7)
LINE (1.25, 2.9)-(1.21, 2.7)

LINE (1.25, .7)-(1.25, 1)
LINE -(1.29, .8)
LINE (1.25, 1)-(1.21, .8)

' ********** Flow Direction of Refrigerant Entering the Absorber **********
LINE (2.25, 6)-(2.25, 6.3)
LINE -(2.29, 6.1)
LINE (2.25, 6.3)-(2.21, 6.1)

LINE (4, 9.5)-(4.3, 9.5)
LINE -(4.15, 9.4)
LINE (4.3, 9.5)-(4.15, 9.6)

' ********** Flow Direction of Solution Entering the Absorber **********
LINE (8, 7.7)-(8, 8)
LINE -(8.04, 7.8)
LINE (8, 8)-(7.96, 7.8)

LINE (8, 4.7)-(8, 5)
LINE -(8.04, 4.8)
LINE (8, 5)-(7.96, 4.8)

' ********** Flow Direction of Solution Leaving the Absorber **********
LINE (7.2, 7.7)-(7.2, 7.4)
LINE -(7.24, 7.6)
LINE (7.2, 7.4)-(7.16, 7.6)

' ********** Flow Direction of Refrigerant Leaving the Desorber **********
LINE (4.5, 3.3)-(4.5, 3)
LINE -(4.54, 3.2)
LINE (4.5, 3)-(4.46, 3.2)

LINE (5, 4.7)-(4.7, 4.7)
LINE -(4.85, 4.6)
LINE (4.7, 4.7)-(4.85, 4.8)

' ********** Flow Direction of Solution Leaving the Desorber **********
LINE (6.75, 2.5)-(6.75, 2.2)
LINE -(6.79, 2.4)
LINE (6.75, 2.2)-(6.71, 2.4)

' ********** Absorber **********
LINE (5.5, 9.9)-(7.5, 9.9), , B
LINE -(7.5, 7.9), , B
```

```
LINE -(5.5, 7.9), , B
LINE -(5.5, 9.9), , B

' ********** Absorber Coolant **********
LINE (5, 8.6)-(7, 8.6), , B
LINE -(7, 8.9), , B
LINE -(5, 8.9), , B

' ********** Solution in Absorber **********
LINE (5.5, 8.3)-(7.5, 8.3), , B
PAINT (6.5, 8.1), 2, 14

' ********** Refrigerant Entering the Absorber **********
LINE (2.25, 4.75)-(2.25, 9.5), , B
LINE -(5.5, 9.5), , B

' ********** Solution Entering the Absorber **********
LINE (5.6, 9.7)-(8, 9.7), , B
LINE -(8, 1.65), , B
LINE -(7.125, 1.65), , B

LINE (5.6, 9.7)-(5.65, 9.6)
LINE (5.6, 9.7)-(5.55, 9.6)
LINE (6.1, 9.7)-(6.15, 9.6)
LINE (6.1, 9.7)-(6.05, 9.6)

LINE (6.6, 9.7)-(6.65, 9.6)
LINE (6.6, 9.7)-(6.55, 9.6)

LINE (7.1, 9.7)-(7.15, 9.6)
LINE (7.1, 9.7)-(7.05, 9.6)

' ********** Solution Leaving the Desorber **********
LINE (6.75, 3.2)-(6.75, 1.25), , B
LINE -(7.125, 1.25), , B

' ********** Solution Leaving the Absorber **********
LINE (7.2, 7.9)-(7.2, 5.2), , B

' ********** Throttled Valve **********
LINE (7.1, 5.2)-(7.3, 5.2), , B
LINE -(7.2, 4.9)
LINE -(7.1, 5.2)
PAINT (7.2, 4.99), 6, 14

LINE (7.1, 4.6)-(7.3, 4.6), , B
LINE -(7.2, 4.9)
LINE -(7.1, 4.6)
PAINT (7.2, 4.7), 6, 14

' ********** Solution Heat Exchanger **********
LINE (7, 6.9)-(8.2, 6.9), , B
LINE -(8.2, 5.9), , B
LINE -(7, 5.9), , B
LINE -(7, 6.9), , B

' ********** Tubes Shown in the Solution Heat Exchanger **********
LINE (7.125, 6.65)-(8.075, 6.65), , B
LINE (7.125, 6.4)-(8.075, 6.4), , B
LINE (7.125, 6.2)-(8.075, 6.2), , B

' ********** Desorber **********
LINE (5.75, 4.9)-(7.75, 4.9), , B
LINE -(7.75, 3.2), , B
LINE -(5.75, 3.2), , B
LINE -(5.75, 4.9), , B
CIRCLE (7.125, 1.25), .175

' ********** Solution in the Desorber **********
LINE (5.75, 3.6)-(7.75, 3.6), , B
PAINT (6.5, 3.3), 2, 14

' ********** Waste Heat to the Desorber **********
LINE (5.25, 4.2)-(7.2, 4.2), , B
LINE -(7.2, 3.9), , B
LINE -(5.25, 3.9), , B

' ********** Refrigerant Leaving the Desorber **********
```

```
LINE (5.75, 4.7)-(4.5, 4.7), , B
LINE -(4.5, 2.4), , B
LINE -(3, 2.4), , B

' ********** Condenser **********
LINE (2, 2.6)-(3, 2.6), , B
LINE -(3, 1.4), , B
LINE -(2, 1.4), , B
LINE -(2, 2.6), , B
CIRCLE (2.5, .5), .175

' ********** Refrigerant in the Condenser **********
LINE (2, 1.6)-(3, 1.6), , B
PAINT (2.5, 1.5), 7, 14

' ********** Coolant in the Condenser **********
LINE (3.5, 2.1)-(2.25, 2.1), , B
LINE -(2.25, 1.8), , B
LINE -(3.5, 1.8), , B

' ********** Refrigerant Leaving the Condenser **********
LINE (2.5, 1.4)-(2.5, .5), , B

' ********** Refrigerant Entering the Evaporator **********
LINE (2.5, .075)-(1.25, .075), , B
LINE -(1.25, 3.2), , B
' ********** Evaporator **********
LINE (1, 4.75)-(2.5, 4.75), , B
LINE -(2.5, 3.2), , B
LINE -(1, 3.2), , B
LINE -(2.5, 4.75), , B

' ********** Refrigerant in the Evaporator **********
LINE (1, 4.5)-(2.5, 4.5), , B
PAINT (1.5, 4), 7, 14

' ********** Waste Heat in the Evaporator **********
LINE (3, 3.8)-(1.25, 3.8), , B
LINE -(1.25, 3.5), , B
LINE -(3, 3.5), , B

' ********** Legend of the Figure **********
LINE (.05, 9.99)-(.25, 9.99), , B
LINE -(.25, 9.79), , B
LINE -(.05, 9.79), , B
LINE -(.05, 9.99), , B
PAINT (.15, 9.89), 2, 14

LINE (.05, 9.3)-(.25, 9.3), , B
LINE -(.25, 9.1), , B
LINE -(.05, 9.1), , B
LINE -(.05, 9.3), , B
PAINT (.15, 9.2), 7, 14

LINE (.05, 8.05)-(.25, 8.05), , B
LINE -(.15, 8.35)
LINE -(.05, 8.05)
PAINT (.15, 8.15), 6, 14

LINE (.05, 8.65)-(.25, 8.65), , B
LINE -(.15, 8.35)
LINE -(.05, 8.65)
PAINT (.15, 8.5), 6, 14

LOCATE 1, 2: PRINT CHR$(218)
LOCATE 5, 2: PRINT CHR$(192)
FOR j = 1 TO 15: LOCATE 1, 2 + j: PRINT CHR$(196): NEXT j
FOR j = 1 TO 15: LOCATE 5, 2 + j: PRINT CHR$(196): NEXT j
FOR j = 1 TO 3: LOCATE 1 + j, 17: PRINT CHR$(179): NEXT j
FOR j = 1 TO 3: LOCATE 1 + j, 2: PRINT CHR$(179): NEXT j
LOCATE 1, 17: PRINT CHR$(191)
LOCATE 5, 17: PRINT CHR$(217)
LOCATE 2, 6
PRINT "Solution"
LOCATE 3, 6
PRINT "Refrigerant"
LOCATE 4, 6
PRINT "Throttle"
```

```
LOCATE 1, 50
PRINT "Absorber"
LOCATE 8, 10
PRINT "Evaporator"
LOCATE 14, 25
PRINT "Condenser"
LOCATE 7, 47
PRINT "Desorber"
LOCATE 5, 69
PRINT "Solution"
LOCATE 6, 69:
PRINT "  Heat"
LOCATE 7, 69
PRINT "Exchanger"
LOCATE 18, 24
PRINT "Schematic of Chemical Heat Pump (TA)"

LOCATE 23, 27
PRINT "PRESS ANY KEY TO CONTINUE"
pause$ = INPUT$(1)
CLS : SCREEN 0
END SUB

SUB mainchoicei

' ******************************************
' REQUIRED PARAMETERS NEEDED FOR CHOICE #1
' ******************************************
        COLOR frg, bac: SCREEN 0
        CLS : CALL border: LOCATE 3, 23: PRINT "MASS AND ENERGY BALANCE CALCULATION (TA)";
        FOR jjk = 1 TO 40: LOCATE 4, (22 + jjk): PRINT CHR$(205): NEXT jjk
        LOCATE 6, 8: PRINT "YOU WILL BE ASKED TO SUPPLY";
        LOCATE 8, 8: PRINT "1.  RUN IDENTITY ";
        LOCATE 9, 8: PRINT "2.  WASTE HEAT INLET TEMPERATURE (Saturated Steam is assumed)";
        LOCATE 10, 7: PRINT " 2a.  WASTE HEAT OUTLET IS ASSUMED TO BE WATER AT INLET TEMPERATURE";
        LOCATE 11, 8: PRINT "3.  ABSORBER CAPACITY";
        LOCATE 12, 8: PRINT "4.  CONDENSER WATER INLET TEMPERATURE";
        LOCATE 13, 8: PRINT "5.  APPROACH TEMPERATURE IN CONDENSER";
        LOCATE 14, 8: PRINT "6.  APPROACH TEMPERATURE IN EVAPORATOR";
        LOCATE 15, 8: PRINT "7.  APPROACH TEMPERATURE IN DESORBER";
        LOCATE 16, 8: PRINT "8.  APPROACH TEMPERATURE IN ABSORBER";
        LOCATE 17, 8: PRINT "9.  APPROACH TEMPERATURE IN SOLUTION HEAT EXCHANGER";
        LOCATE 18, 8: PRINT "10.  COOLING WATER TEMPERATURE ENTERING THE ABSORBER";
        LOCATE 19, 8: PRINT "10a. ASSUME STEAM LEAVING ABSORBER AT SAME TEMPERATURE";
        LOCATE 20, 8: PRINT "11.  DESORBER TEMPERATURE";
        LOCATE 23, 10: PRINT "<F1> -INPUT DATA                <ESC> -BACK TO MAIN MENU ";


END SUB

SUB mainchoiceii

' ******************************************
' REQUIRED PARAMETERS NEEDED FOR CHOICE #1
' ******************************************

        COLOR frg, bac: SCREEN 0
        CLS : CALL border: LOCATE 3, 23: PRINT "MASS AND ENERGY BALANCE CALCULATION (HA)";
        FOR jjk = 1 TO 40: LOCATE 4, (22 + jjk): PRINT CHR$(205): NEXT jjk
        LOCATE 6, 8: PRINT "YOU WILL BE ASKED TO SUPPLY";
        LOCATE 8, 8: PRINT "1.  RUN IDENTITY ";
        LOCATE 9, 8: PRINT "2.  WASTE HEAT INLET TEMPERATURE TO THE EVAPORATOR";
        LOCATE 10, 8: PRINT "2a.  EVAPORATOR WASTE HEAT IS ASSUMED TO BE SATURATED STEAM";
        LOCATE 11, 8: PRINT "3.  ABSORBER CAPACITY";
        LOCATE 12, 8: PRINT "4.  COOLING WATER TEMPERATURE ENTERING THE ABSORBER & CONDENSER";
        LOCATE 13, 8: PRINT "4a.  COOLING WATER IS ASSUMED SATURATED AND USED BY BOTH UNITS";
        LOCATE 14, 8: PRINT "5.  APPROACH TEMPERATURE IN CONDENSER";
        LOCATE 15, 8: PRINT "6.  APPROACH TEMPERATURE IN EVAPORATOR";
        LOCATE 16, 8: PRINT "7.  APPROACH TEMPERATURE IN DESORBER";
        LOCATE 17, 8: PRINT "8.  APPROACH TEMPERATURE IN ABSORBER";
        LOCATE 18, 8: PRINT "9.  APPROACH TEMPERATURE IN SOLUTION HEAT EXCHANGER";
        'LOCATE 19, 8: PRINT "10.  WASTE HEAT INLET TEMPERATURE TO THE DESORBER";
        'LOCATE 20, 8: PRINT "10a. DESORBER WASTE HEAT IS ASSUMED TO BE SATURATED STEAM";
        LOCATE 23, 10: PRINT "<F1> -INPUT DATA                <ESC> -BACK TO MAIN MENU ";

END SUB

SUB MAINMENU
```

```
' ************************************MAIN MENU*********************************

    COLOR frg, bac: SCREEN 0: escflag = 0
    CLS : CALL border: LOCATE 6, 27: PRINT " M A I N    M E N U"
    FOR II = 1 TO 20: LOCATE 7, II + 26: PRINT CHR$(205): NEXT II

    LOCATE 12, 12: PRINT " <F1> - MASS AND ENERGY BALANCE CALCULATION (TA)";
    LOCATE 14, 12: PRINT " <F2> - MASS AND ENERGY BALANCE CALCULATION (HA)";
    LOCATE 16, 12: PRINT " <F6> - EXIT THE PROGRAM";
    LOCATE 23, 20: PRINT " K E Y S    <F1>, <F2> & <F6>    A R E    V A L I D ";

END SUB

        SUB mixture

' ********************WORKING FLUID MENU**********************

    escflag = 0

    COLOR frg, bac: SCREEN 0
    CLS : CALL border: LOCATE 6, 23: PRINT " W O R K I N G    F L U I D    M E N U"
    FOR II = 1 TO 40: LOCATE 7, II + 21: PRINT CHR$(205): NEXT II
    LOCATE 10, 12: PRINT " <F1> - LiBr / Water Mixture";
    LOCATE 12, 12: PRINT " <F2> - (Li, K, Na) NO3 / Water Mixture";
    LOCATE 14, 12: PRINT " <F3> - Other *** Exit To Main Menu";
    LOCATE 16, 12: PRINT " <F4> - Exit To Main Menu";
    END SUB

SUB noise STATIC

' *************USED TO GIVE WARNINGS FOR INVALID ENTRIES*************
' SOUND 200, 4
' SOUND 300, 5
'  SOUND 275, 2
'   SOUND 250, 4
'    SOUND 200, 4

        SOUND 300, 2
        SOUND 250, 2

END SUB

        SUB prealk (tf, wt, psat) STATIC

' ***********VAPOR PRESSURE OF (Li, K, Na) NO3 MIXTURES (psat)***********

        T = tf
        a = -1.06427 + .102088 * wt - .0016266 * wt ^ 2 + 8.93054E-06 * wt ^ 3
        B = 291.113 - 13.0305 * wt + .187146 * wt ^ 2 - 7.87688E-04 * wt ^ 3
        tprime = (T - B) / a
        C = 6.21147
        D = -2886.37
        E = -337269.46#
        logpsat = C + D / (tprime + 459.67) + E / (tprime + 459.67) ^ 2
        psat = 10 ^ (logpsat)

' ************************psat IS IN psia*************************

' **************wt IS IN WT% AND T IS IN DEGREES F*************

END SUB

SUB preh2o (tf, P)

' ************************************************************
' Pressure of Saturated Steam (P) in psia
' Desired input is T sat. (Deg. F) to calculate P sat.
' ************************************************************

        T = tf
        C = 6.21147: D = -2886.373: E = -337269.46#
        LogP = C + D / (T + 459.72) + E / (T + 459.72) ^ 2
        P = 10 ^ LogP

END SUB

        SUB preLiBr (tf, wt, psat) STATIC
```

```
' ***********VAPOR PRESSURE OF LiBr SOLUTIONS (psat) IN psia***********

        T = tf
        a = -2.00755 + .16976# wt - 3.133362E-03# wt ^ 2 + 1.97668E-05 * wt ^ 3
        B = 321.128 - 19.322# wt + .374382 * wt ^ 2 - .0020637 * wt ^ 3
        tprime = (T - B) / a
        C = 6.21147
        D = -2886.37
        E = -337269.46#
        logpsat = C + D / (tprime + 459.67) + E / (tprime + 459.67) ^ 2
        psat = 10 ^ (logpsat)

' ******************wt IS IN WT% AND t IS IN DEGREES F*******************

END SUB
```

MODULIII.BAS

```
DECLARE SUB denalk (tf!, wt!, rhosoln!)
DECLARE SUB denlibr (tf!, wt!, rhosoln!)
DECLARE SUB denh2o (tf!, rhoh2o!)
DECLARE SUB condabsin (tcwin!)
DECLARE SUB xhtcalalkHA (hxt!, T!, wtout!, wt!, hvapor!, wtabsout!)
DECLARE SUB xhtcalLiBrHA (hxt!, T!, wtout!, wt!, hvapor!, wtabsout!)
DECLARE SUB mainchoiceii ()
DECLARE SUB tabdata2 (ESTID$, mat$, con$, qout!, tcwin!, twastedesin!, twasteevapin!, tcapp!, teapp!, tdapp!, taapp!, thxapp!)
DECLARE SUB INPTWASTEDES (twastedesin!, twastedesin$, twastedesinUNIT$, trap$, aaaa!)
DECLARE SUB wasteheatevapin (twasteevapin!)
DECLARE SUB wasteheatdesin (twastedesin!)
DECLARE SUB logoTA ()
DECLARE SUB logoHA ()
DECLARE SUB hxtLiBr (tf!, wt, hxt1!)
DECLARE SUB BoostTalk (P!, wt!, tmax!)
DECLARE SUB abswtoutalk (T!, P!, wtout!, wt!)
DECLARE SUB thxoutcalalk (hxt!, wt!, thxout!)
DECLARE SUB xhtcalalk (hxt!, T!, wtout!, wt!, hvapor, wtabsin)
DECLARE SUB hxtalk (T1!, wt!, hxt!)
DECLARE SUB prealk (tf!, wt!, psat!)
DECLARE SUB hgsuper (psat!, Hg!, tsat!, T!, hsuperheat!)
DECLARE SUB TSATH2O (P!, T1!)
DECLARE SUB abswtoutLiBr (T!, P!, wtout!, wt!)
DECLARE SUB thxoutcalLiBr (hxt!, wt!, thxout!)
DECLARE SUB xhtcalLiBr (hxt!, T!, wtout!, wt!, hvapor, wtabsin)
DECLARE SUB preLiBr (tf!, wt!, psat!)
DECLARE SUB BoostTLiBr (P!, wt!, tmax!)
DECLARE SUB hgwater (tf!, Hg!)
DECLARE SUB hfwater (tf!, hf!)
DECLARE SUB preh2o (tf!, P!)
DECLARE SUB TCABSIN (tcain!)
DECLARE SUB Tdesorber (tdeso!)
DECLARE SUB inptcain (tcain!, tcain$, tcainUNIT$, trap$)
DECLARE SUB inptdeso (tdeso!, tdeso$, tdesoUNIT$, trap$)
DECLARE SUB wtabs (wt$, wt!)
DECLARE SUB desTapp (tdapp!)
DECLARE SUB AbsTapp (taapp!)
DECLARE SUB HXTapp (thxapp!)
DECLARE SUB inptcapp (tcapp!, tcapp$, tcappUNIT$, trap$)
DECLARE SUB inpteapp (teapp!, teapp$, teappUNIT$, trap$)
DECLARE SUB inptdapp (tdapp!, tdapp$, tdappUNIT$, trap$)
DECLARE SUB inptaapp (taapp!, taapp$, taappUNIT$, trap$)
DECLARE SUB inpthxapp (thxapp!, thxapp$, thxappUNIT$, trap$)
DECLARE SUB condtapp (tcapp!)
DECLARE SUB evaTapp (teapp!)
DECLARE SUB waterpre (tf!, hf!, Hg!, psat!, T1!)
DECLARE SUB inptcwin (tcwin!, tcwin$, tcwinUNIT$, trap$, ccc!)
DECLARE SUB INPTWASTE (twaste!, twaste$, twasteUNIT$, trap$, aaa!)
DECLARE SUB wasteheatin (twaste!)
DECLARE SUB INPqout (qout!, qout$, qoutUNIT$, trap$, bbb)
DECLARE SUB RUNID (ESTID$, trap$)
DECLARE SUB condh2oin (tcwin)
DECLARE SUB qtout (qout)
```

```
DECLARE SUB tabdata (ESTID$, mat$, con$, qout!, tcwin!, twaste, tcapp, teapp, tdapp, taapp, thxapp, tcain, tdeso)
DECLARE SUB MAINMENU ()
DECLARE SUB TITLESCR ()
DECLARE SUB mainchoicei ()
DECLARE SUB mixture ()
DECLARE SUB noise ()
DECLARE SUB border ()
COMMON SHARED frg, bac, escflag
frg = 14: bac = 1

    SUB qtout (qout) STATIC

' ****************************************************************
' THIS PROGRAM IS USED TO INPUT CAPACITY (qout) BY INDIVIDUAL
' SCREENS.
' ****************************************************************

    escflag = 0
    n = 1: qoutCF = 1: qoutUNIT$ = " Btu/hr    ": qoutunit1$ = qoutUNIT$: qout = 0: qout$ = ""
REDO:
    CLS : CALL border: : LOCATE 6, 10: PRINT "Type in the absorber capacity.Use Function key <F1> to select"
    LOCATE 7, 10: PRINT "the proper units for your entry. Press <F9> when finished"
    LOCATE 23, 8: COLOR 15, bac: PRINT "<F1>-TOGGLE UNITS      <F9>-CONFIRM ENTRY      <ESC>-PREVIOUS MENU"; : COLOR frg, bac
    LOCATE 19, 25: PRINT "Absorber capacity";
110   LOCATE 19, 61: PRINT qoutUNIT$;
trap:
    LOCATE 19, 40, 0: trap$ = INKEY$: IF trap$ = "" GOTO trap:
    IF trap$ = "I" GOTO DISPLAY:
    IF ASC(trap$) = 8 THEN qout$ = "": qout = 0: LOCATE 19, 47: PRINT "          "; : GOTO trap:
    IF trap$ = "A" GOTO UNITS:
    IF ASC(trap$) = 27 THEN
    escflag = 1
    GOTO CONV:
    END IF
    IF ASC(trap$) < 46 OR ASC(trap$) > 57 OR ASC(trap$) = 47 GOTO trap:
    qout$ = qout$ + trap$
    LOCATE 19, 47: PRINT VAL(qout$); : NN = NN + 1: qout = VAL(qout$): GOTO trap:
DISPLAY:
    CLS : CALL border: : LOCATE 12, 16: PRINT "Your Chosen Absorber Capacity is    "; qout; "  "; qoutUNIT$
    COLOR 15, bac: LOCATE 23, 17: PRINT "  <F1>- CORRECT        <F2>- INCORRECT    "; : COLOR frg, bac
CHECK:
    DTCHECK$ = INKEY$: IF DTCHECK$ = "A" GOTO CONV:
    IF DTCHECK$ = "B" THEN qout$ = "": GOTO REDO:
    GOTO CHECK:
UNITS:

' ***************************CHANGING UNITS*****************************

    n = n + 1: IF n = 5 THEN n = 1
    ON n GOTO 150, 200, 300, 400
150   qoutCF = 1: qoutUNIT$ = " Btu/hr    ": GOTO 110
200   qoutCF = 1 / 1055: qoutUNIT$ = " J/hr     ": GOTO 110
300   qoutCF = 1000000!: qoutUNIT$ = " MM Btu/hr ": GOTO 110
400     qoutCF = 3412.32: qoutUNIT$ = "   kW     ": GOTO 110
CONV:

' ********************qout IS IN Btu/hr**********************

    CLS : qout = qout * qoutCF

END SUB

SUB RUNID (ESTID$, trap$) STATIC

' ****************************************************************************
' SUBROUTINE TO ENTER RUN IDENTIFICATION (ESTID$)
' COLOR STATEMENTS ARE USED TO REVERSE THE VIDEO WHENEVER THE DATA IS BEING
' ENTERED AT THE PARTICULAR PLACE ON THE SCREEN
' ****************************************************************************

LOCATE 3, 17: COLOR bac, frg: PRINT SPACE$(8)
LOCATE 3, 17: PRINT ESTID$
111  trap$ = ""
WHILE trap$ = ""
          trap$ = INKEY$
 WEND
IF ASC(trap$) = 9 OR ASC(trap$) = 13 OR ASC(trap$) = 27 THEN GOTO DATAA:
IF ASC(trap$) = 8 THEN ESTID$ = "": LOCATE 3, 17: PRINT SPACE$(8); : GOTO 111
```

```
IF LEN(ESTID$) < 8 THEN   'USER ALLOWED UP TO 8 SPACES FOR RUN ID
ESTID$ = ESTID$ + trap$
ELSE
            CALL noise
            GOTO 111
END IF
LOCATE 3, 17: PRINT (ESTID$); : GOTO 111
DATAA:
LOCATE 3, 17: COLOR frg, bac: PRINT SPACE$(8)
LOCATE 3, 17: PRINT ESTID$

END SUB

SUB tabdata (ESTID$, mat$, con$, qout, tcwin, twaste, tcapp, teapp, tdapp, taapp, thxapp, tcain, tdeso) STATIC
CLS : COLOR frg, bac

' ********************Initializing the parameters*********************

qout$ = "": tcwin$ = "": twaste$ = "": tcapp$ = "": teapp$ = "": tdapp$ = ""
taapp$ = "": thxapp$ = "": tcain$ = "": tdeso$ = ""
qout = 0: tcwin = 0: twaste = 0: tcapp = 0: teapp = 0: tdapp = 0
taapp = 0: thxapp = 0: tcain = 0: tdeso = 0
twasteUNIT$ = " Deg. F": tcwinUNIT$ = " Deg. F"
qoutUNIT$ = " Btu/hr    ": ESTID$ = ""
tcappUNIT$ = " Deg. F": teappUNIT$ = " Deg. F": tdappUNIT$ = " Deg. F"
taappUNIT$ = " Deg. F": thxappUNIT$ = " Deg. F"
tcainUNIT$ = " Deg. F": tdesoUNIT$ = " Deg. F"

' ********************************************************************

            KEY 1, "A"
n = 0

escflag = 0
CLS : CALL border:

' ********************************************************************
' TABULAR FORMAT OF THE SCREEN FOR THE DATA INPUT.
' USER COULD TOGGLE UNITS AND COULD EDIT DATA
' ********************************************************************
LOCATE 3, 3: PRINT "Run Identity:";
LOCATE 3, 36: PRINT "Working Fluid:": ddd = LEN(con$): eee = LEN(" wt%")
LOCATE 5, 3: PRINT "Waste Heat Inlet Temperature:": aaa = LEN("Waste Heat Inlet Temperature:")
LOCATE 6, 3: PRINT "Capacity:": bbb = LEN("Capacity:")
LOCATE 7, 3: PRINT "Condenser Water Inlet Temperature:": ccc = LEN("Condenser Water Inlet Temperature:")
LOCATE 5, (3 + aaa + 12): PRINT " Deg. F"
LOCATE 6, (3 + bbb + 12): PRINT " Btu/hr    "
LOCATE 7, (3 + ccc + 12): PRINT " Deg. F "
LOCATE 8, 3: PRINT "Approach Temperature in Condenser:"
LOCATE 8, 51: PRINT " Deg. F"
LOCATE 9, 3: PRINT "Approach Temperature in Evaporator:"
LOCATE 9, 53: PRINT " Deg. F"
LOCATE 10, 3: PRINT "Approach Temperature in Desorber:"
LOCATE 10, 51: PRINT " Deg. F"
LOCATE 11, 3: PRINT "Approach Temperature in Absorber:"
LOCATE 11, 51: PRINT " Deg. F"
LOCATE 12, 3: PRINT "Approach Temperature in Solution Heat Exchanger:"
LOCATE 12, 66: PRINT " Deg. F"
LOCATE 13, 3: PRINT "Cooling Water Temperature Entering the Absorber:"
LOCATE 13, 66: PRINT " Deg. F"
LOCATE 14, 3: PRINT "Desorber Temperature:"
LOCATE 14, 40: PRINT " Deg. F"
LOCATE 23, 3: PRINT "<F1>-TOGGLE UNITS    <CR>-TOGGLE DATA    <TAB>-END INPUT    <ESC>-ESCAPE";
trap$ = CHR$(13)
REM WAITING FOR THE USER TO ENTER A VALUE OR TO TOGGLE BETWEEN INPUTS,ETC.
DECISION:
WHILE trap$ = ""
            trap$ = INKEY$
WEND
IF ASC(trap$) = 9 OR ASC(trap$) = 27 THEN
GOTO 12
END IF
IF ASC(trap$) = 13 THEN
GOTO TOGDATA:
END IF

' ********************************************************************
' THIS IS THE PART OF THE CODE WHERE THE USER CAN TOGGLE BETWEEN DATA INPUTS
```

```
' ************************************************************************

TOGDATA:
LOCATE 3, 52: PRINT con$
LOCATE 3, (53 + ddd): PRINT "wt%"
LOCATE 3, (53 + ddd + eee): PRINT mat$
n = n + 1: IF n = 12 THEN n = 1
            ON n GOTO 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12
1    CALL RUNID(ESTID$, trap$): GOTO DECISION:
2     CALL INPTWASTE(twaste, twaste$, twasteUNIT$, trap$, aaa): GOTO DECISION:
3      CALL INPqout(qout, qout$, qoutUNIT$, trap$, bbb): GOTO DECISION:
4       CALL inptcwin(tcwin, tcwin$, tcwinUNIT$, trap$, ccc): GOTO DECISION:
5        CALL inptcapp(tcapp, tcapp$, tcappUNIT$, trap$): GOTO DECISION:
6         CALL inpteapp(teapp, teapp$, teappUNIT$, trap$): GOTO DECISION:
7          CALL inptdapp(tdapp, tdapp$, tdappUNIT$, trap$): GOTO DECISION:
8           CALL inptaapp(taapp, taapp$, taappUNIT$, trap$): GOTO DECISION:
9            CALL inpthxapp(thxapp, thxapp$, thxappUNIT$, trap$): GOTO DECISION:
10            CALL inptcain(tcain, tcain$, tcainUNIT$, trap$): GOTO DECISION:
11             CALL inptdeso(tdeso, tdeso$, tdesoUNIT$, trap$): GOTO DECISION:
12 IF ASC(trap$) = 27 THEN escflag = 1



END SUB

SUB tabdata2 (ESTID$, mat$, con$, qout, tcwin, twastedesin, twasteevapin, tcapp, teapp, tdapp, taapp, thxapp)


CLS : COLOR frg, bac

' ********************Initializing the parameters*********************

qout$ = "": tcwin$ = "": tcapp$ = "": teapp$ = "": tdapp$ = ""
taapp$ = "": thxapp$ = "": twasteevapin$ = ""
qout = 0: tcwin = 0: twaste = 0: tcapp = 0: teapp = 0: tdapp = 0
taapp = 0: thxapp = 0: twasteevapin = 0
tcwinUNIT$ = " Deg. F"
qoutUNIT$ = " Btu/hr    ": ESTID$ = ""
tcappUNIT$ = " Deg. F": teappUNIT$ = " Deg. F": tdappUNIT$ = " Deg. F"
taappUNIT$ = " Deg. F": thxappUNIT$ = " Deg. F"
twasteevapinUNIT$ = " Deg. F"

' ************************************************************************
            KEY 1, "A"
n = 0

escflag = 0
CLS : CALL border

' ************************************************************************
' TABULAR FORMAT OF THE SCREEN FOR THE DATA INPUT.
' USER COULD TOGGLE UNITS AND COULD EDIT DATA
' ************************************************************************

LOCATE 3, 3: PRINT "Run Identity:";
LOCATE 3, 36: PRINT "Working Fluid:": ddd = LEN(con$): eee = LEN(" wt%")
LOCATE 5, 3: PRINT "Waste Heat Inlet Temperature (evaporator):": aaa = LEN("Waste Heat Inlet Temperature (evaporator):")
LOCATE 6, 3: PRINT "Absorber Capacity:": bbb = LEN("Absorber Capacity:")
LOCATE 7, 3: PRINT "Cooling Water Temperature (absorber & condenser):": ccc = LEN("Cooling Water Temperature (absorber & condenser):")
LOCATE 5, (3 + aaa + 12): PRINT " Deg. F"
LOCATE 6, (3 + bbb + 12): PRINT " Btu/hr    "
LOCATE 7, (3 + ccc + 12): PRINT " Deg. F"
LOCATE 8, 3: PRINT "Approach Temperature in Condenser:"
LOCATE 8, 51: PRINT " Deg. F"
LOCATE 9, 3: PRINT "Approach Temperature in Evaporator:"
LOCATE 9, 53: PRINT " Deg. F"
LOCATE 10, 3: PRINT "Approach Temperature in Desorber:"
LOCATE 10, 51: PRINT " Deg. F"
LOCATE 11, 3: PRINT "Approach Temperature in Absorber:"
LOCATE 11, 51: PRINT " Deg. F"
LOCATE 12, 3: PRINT "Approach Temperature in Solution Heat Exchanger:"
LOCATE 12, 66: PRINT " Deg. F"
LOCATE 23, 3: PRINT "<F1>-TOGGLE UNITS    <CR>-TOGGLE DATA    <TAB>-END INPUT    <ESC>-ESCAPE";
trap$ = CHR$(13)
REM WAITING FOR THE USER TO ENTER A VALUE OR TO TOGGLE BETWEEN INPUTS,ETC.
DECISION#:
WHILE trap$ = ""
            trap$ = INKEY$
```

```
WEND
IF ASC(trap$) = 9 OR ASC(trap$) = 27 THEN
GOTO 248
END IF
IF ASC(trap$) = 13 THEN
GOTO TOGDATAii:
END IF

' ************************************************************************
' THIS IS THE PART OF THE CODE WHERE THE USER CAN TOGGLE BETWEEN DATA INPUTS
' ************************************************************************

TOGDATAii:
LOCATE 3, 52: PRINT con$
LOCATE 3, (53 + ddd): PRINT "wt%"
LOCATE 3, (53 + ddd + eee): PRINT mat$

n = n + 1: IF n = 10 THEN n = 1
          ON n GOTO 239, 240, 241, 242, 243, 244, 245, 246, 247, 248
239   CALL RUNID(ESTID$, trap$): GOTO DECISIONii:
240    CALL INPTWASTE(twasteevapin, twasteevapin$, twasteevapinUNIT$, trap$, aaa): GOTO DECISIONii:
241      CALL INPqout(qout, qout$, qoutUNIT$, trap$, bbb): GOTO DECISIONii:
242      CALL inptcwin(tcwin, tcwin$, tcwinUNIT$, trap$, ccc): GOTO DECISIONii:
243       CALL inptcapp(tcapp, tcapp$, tcappUNIT$, trap$): GOTO DECISIONii:
244        CALL inpteapp(teapp, teapp$, teappUNIT$, trap$): GOTO DECISIONii:
245         CALL inptdapp(tdapp, tdapp$, tdappUNIT$, trap$): GOTO DECISIONii:
246           CALL inptaapp(taapp, taapp$, taappUNIT$, trap$): GOTO DECISIONii:
247            CALL inpthxapp(thxapp, thxapp$, thxappUNIT$, trap$): GOTO DECISIONii:
248 IF ASC(trap$) = 27 THEN escflag = 1

END SUB

SUB TCABSIN (tcain) STATIC

' ************************************************************************
' THIS PROGRAM IS USED TO INPUT THE COOLING WATER TEMPERATURE ENTERING THE
' ABSORBER (tcain) BY INDIVIDUAL SCREENS
' ************************************************************************

          escflag = 0
CH1INDP8:
ENTPART8:

' ****************************************************************
' USER COULD CHANGE THE UNITS BY PRESSING <F1> KEY OR
' COULD ESCAPE TO PREVIOUS MENU BY PRESSING <ESC> KEY OR
' ACCEPT THE INPUT BY PRESSING <F9>
'****************************************************************

          n = 1: tcainCF = 1: tcainUNIT$ = " Deg. F ": tcainUNIT1$ = tcainUNIT$: tcain = 0: tcain$ = ""
          CLS : CALL border: : LOCATE 6, 6: PRINT "Type in the cooling water temperature entering the absorber. Use Function "
          LOCATE 7, 6: PRINT "key <F1> to select proper units for your entry. Press <F9> when finished."
          COLOR 15, bac: LOCATE 23, 7: PRINT "<F1>-TOGGLE UNITS      <F9>-CONFIRM ENTRY      <ESC>-PREVIOUS MENU"; : COLOR frg, bac
          LOCATE 19, 7: PRINT " Cooling water temperature entering the absorber    ";
PRTUNIT8:
          LOCATE 19, 65: PRINT tcainUNIT$;
20008     LOCATE 19, 40, 0: trap$ = INKEY$: IF trap$ = "" GOTO 20008
          IF trap$ = "I" GOTO 21008
          IF ASC(trap$) = 8 THEN tcain$ = "": tcain = 0: LOCATE 19, 55: PRINT "       ": GOTO 20008
          IF trap$ = "A" GOTO CHANGEUNIT8:
    IF ASC(trap$) = 27 THEN
    escflag = 1
    GOTO 21508
    END IF
          IF ASC(trap$) < 46 OR ASC(trap$) > 57 OR ASC(trap$) = 47 GOTO 20008
          tcain$ = tcain$ + trap$
          LOCATE 19, 55: PRINT VAL(tcain$); : tcain = VAL(tcain$): GOTO 20008

' ************************************************************************
' CONFIRM YOUR TEMPERATURE INPUT PRESS <F1> FOR CORRECT INPUT OR <F2> FOR
' INCORRECT INPUT
' ************************************************************************

21008     CLS : CALL border: : LOCATE 12, 3: PRINT "Your chosen cooling water temperature entering absorber is "; tcain; " "; tcainUNIT$
          LOCATE 23, 12: PRINT "  <F1>- Correct                <F2>-Incorrect    ";
21108     DTCHECK$ = INKEY$: IF DTCHECK$ = "A" GOTO 21508
          IF DTCHECK$ = "B" THEN tcain$ = "": GOTO ENTPART8:
          GOTO 21108
```

```
' **********CHANGING UNITS OF TEMPERATURE**********

CHANGEUNIT8:
        n = n + 1: IF n = 3 THEN n = 1
        ON n GOTO 21208, 21218
21208   tcainCF = 1: tcainUNIT$ = " Deg. F ": GOTO PRTUNIT8:
21218   tcainCF = 2: tcainUNIT$ = " Deg. C ": GOTO PRTUNIT8:

' ****************************************************************

21508   CLS : IF tcainCF = 2 THEN tcain = (tcain * 1.8) + 32

' **********tcain IS IN DEGREES F.**********

END SUB

SUB Tcrysalk (wt, T, tcrys, tcryscode)

' ************************************************************************
' This program determines the crystallization temperature of the (Li, K, Na) NO3
' solution streams and if there is a possibility of crystallization it gives a
' warning when the final result is printed.
' ************************************************************************

 tcryscode = 0

' ********** Crystallization is possible when tcryscode = 1 **********

 IF wt > 90 THEN
 tcrys = 7.2 * wt - 373
 ELSEIF wt <= 90 AND wt > 85 THEN
 tcrys = 15.3 * wt - 1102
 ELSEIF wt <= 85 AND wt > 80 THEN
 tcrys = 5.4 * wt - 260.5
 ELSE
 tcrys = 8.55 * wt - 512.5
 END IF
 IF tcrys > T THEN tcryscode = 1

END SUB

SUB TcrysLiBr (wt, T, tcrys, tcryscode)

' ************************************************************************
' This program determines the crystallization temperature of the LiBr solution
' streams and if there is a possibility of crystallization it gives a warning
' when the final result is printed.
' ************************************************************************
 tcryscode = 0

' ********** Crystallization is possible when tcryscode = 1 **********

 IF wt > 67 THEN
 tcrys = 15 * wt - 830
 ELSEIF wt <= 67 AND wt > 66 THEN
 tcrys = 20 * wt - 1165
 ELSEIF wt <= 66 AND wt > 65 THEN
 tcrys = 35 * wt - 2155
 ELSEIF wt <= 65 AND wt > 63 THEN
 tcrys = 17.5 * wt - 1016.66667#
 ELSE
 tcrys = 10 * wt - 545
 END IF
 IF tcrys > T THEN tcryscode = 1

END SUB

SUB Tdesorber (tdeso) STATIC

' ****************************************************************
' THIS PROGRAM IS USED TO INPUT THE DESORBER TEMPERATURE (tdeso)
' BY INDIVIDUAL SCREENS
' ****************************************************************

        escflag = 0
CH1INDP9:
ENTPART9:
```

```
' ********************************************************************
' USER COULD CHANGE THE UNITS BY PRESSING <F1> KEY OR
' COULD ESCAPE TO PREVIOUS MENU BY PRESSING <ESC> KEY OR
' ACCEPT THE INPUT BY PRESSING <F9>
'*********************************************************************

            n = 1: tdesoCF = 1: tdesoUNIT$ = "Degrees F     ": tdesoUNIT1$ = tdesoUNIT$: tdeso = 0: tdeso$ = ""
            CLS : CALL border: : LOCATE 6, 6: PRINT "Type in the desorber temperature. Use Function key <F1> to select"
            LOCATE 7, 6: PRINT "proper units for your entry. Press <F9> when finished."
            COLOR 15, bac: LOCATE 23, 7: PRINT "<F1>-TOGGLE UNITS        <F9>-CONFIRM ENTRY        <ESC>-PREVIOUS MENU"; : COLOR frg, bac
            LOCATE 19, 20: PRINT " Desorber temperature    ";
PRTUNIT9:
            LOCATE 19, 55: PRINT tdesoUNIT$;
20009    LOCATE 19, 40, 0: trap$ = INKEY$: IF trap$ = "" GOTO 20009
            IF trap$ = "I" GOTO 21009
            IF ASC(trap$) = 8 THEN tdeso$ = "": tdeso = 0: LOCATE 19, 45: PRINT "        ": GOTO 20009
            IF trap$ = "A" GOTO CHANGEUNIT9:
    IF ASC(trap$) = 27 THEN
    escflag = 1
    GOTO 21509
    END IF
            IF ASC(trap$) < 46 OR ASC(trap$) > 57 OR ASC(trap$) = 47 GOTO 20009
            tdeso$ = tdeso$ + trap$
            LOCATE 19, 45: PRINT VAL(tdeso$); : tdeso = VAL(tdeso$): GOTO 20009

' ********************************************************************************
' CONFIRM YOUR TEMPERATURE INPUT PRESS <F1> FOR CORRECT INPUT OR <F2> FOR
' INCORRECT INPUT
' ********************************************************************************

21009    CLS : CALL border: : LOCATE 12, 12: PRINT "Your chosen desorber temperature is "; tdeso; " "; tdesoUNIT$
            LOCATE 23, 12: PRINT "  <F1>- Correct                <F2>-Incorrect    ";
21109    DTCHECK$ = INKEY$: IF DTCHECK$ = "A" GOTO 21509
            IF DTCHECK$ = "B" THEN tdeso$ = "": GOTO ENTPART9:
            GOTO 21109

' ********************CHANGING UNITS OF TEMPERATURE**********************

CHANGEUNIT9:
            n = n + 1: IF n = 3 THEN n = 1
            ON n GOTO 21209, 21219
21209    tdesoCF = 1: tdesoUNIT$ = "Degrees F     ": GOTO PRTUNIT9:
21219    tdesoCF = 2: tdesoUNIT$ = "Degrees C     ": GOTO PRTUNIT9:

' ********************************************************************************

21509    CLS : IF tdesoCF = 2 THEN tdeso = (tdeso * 1.8) + 32

' ********************tdeso IS IN DEGREES F.**************************

END SUB
SUB thxoutcalalk (hxt, wt, thxout) STATIC

' ********************************************************************************
' This program is used to calculate the temperature of (Li, K, Na) NO3
' mixture (thxout in Deg. F) at a given wt% (wt) and specific enthalpy
' (hxt in Btu/lbm).
' ********************************************************************************

            alpha = 394.516 - 8.64996 * wt + .0628787 * wt ^ 2
            beta = .388691 + 3.02719E-03 * wt - 3.80068E-05 * wt ^ 2
            thxout = (hxt - alpha) / beta

END SUB

SUB thxoutcalLiBr (hxt, wt, thxout) STATIC

' ********************************************************************
' This program is used to calculate the temperature of LiBr mixture
' (thxout in Deg. F) at a given wt% (wt) and specific enthalpy
' (hxt in Btu/lbm).
' ********************************************************************

            a0 = -1015.07 + 79.5387 * wt - 2.35802 * wt ^ 2 + .0303158 * wt ^ 3 - 1.4003E-04 * wt ^ 4
            a1 = 4.68108 - .303777 * wt + 8.44845E-03 * wt ^ 2 - 1.04772E-04 * wt ^ 3 + 4.801E-07 * wt ^ 4
            a2 = -.0049107 + 3.83184E-04 * wt - 1.079E-05 * wt ^ 2 + 1.3152E-07 * wt ^ 3 - 5.897E-10 * wt ^ 4
            xxx = (a1 ^ 2 - 4 * (a0 - hxt) * a2)
            IF xxx < 0 THEN PRINT "NO REAL ROOTS"
```

```
            yyy = SQR(xxx)
            x1 = (-a1 - yyy) / (2 * a2)
            x2 = (-a1 + yyy) / (2 * a2)
            IF x1 < 0 THEN
            thxout = x2
            ELSEIF x2 < 0 THEN
            thxout = x1
            ELSEIF x1 < 0 AND x2 < 0 THEN
            PRINT "NO POSITIVE ROOTS"
            END IF

END SUB

SUB TITLESCR

' ********************************TITLE PAGE**********************************

            SCREEN 0: WIDTH 80: COLOR frg, bac: CLS

    CALL border
    LOCATE 12, 21: PRINT " C H E M I C A L    H E A T    P U M P ";

' ************************************************************************
' ALL OF THE CHR$(X) BEING PRINTED IN THE NEXT FEW LINES IS TO PRODUCE A BOX
' AROUND THE ABOVE
' ************************************************************************

            LOCATE 5, 14: PRINT CHR$(201): LOCATE 5, 70: PRINT CHR$(187)
            FOR i = 1 TO 17
            LOCATE 5 + i, 14: PRINT CHR$(186)
            LOCATE 5 + i, 70: PRINT CHR$(186)
            NEXT i
            LOCATE 22, 14: PRINT CHR$(200): LOCATE 22, 70: PRINT CHR$(188)

            FOR i = 1 TO 55
            LOCATE 5, 14 + i: PRINT CHR$(205)
            LOCATE 22, 14 + i: PRINT CHR$(205)
            NEXT i
            LOCATE 24, 25: PRINT "PLEASE PRESS ANY KEY TO CONTINUE";

END SUB

SUB TSATH2O (P, T1)

' ****************************************************
' Temperature of Saturated Steam (T1 in Degrees F)
' Input P sat. (psia) to calculate T sat.
' ****************************************************

            C = 6.21147: D = -2886.373: E = -337269.46#
            LogP = LOG(P) / 2.302585
            T1 = (-D + SQR(D ^ 2 - 4 * E * (C - LogP))) / (2 * (C - LogP)) - 459.72

' ************************************************************************
' x2 = (-D - SQR(D ^ 2 - 4 * E * (C - LogP))) / (2 * (C - LogP)) - 459.72
' PRINT x1, x2
' IF x1 < 0 THEN T1 = x2
' IF x2 < 0 THEN T1 = x1
' ************************************************************************

END SUB

            SUB wasteheatevapin (twasteevapin)

' ************************************************************************
' THIS PROGRAM IS USED TO INPUT EVAPORATOR WASTE INLET TEMPERATURE (twasteevap)
' BY INDIVIDUAL SCREENS
' ************************************************************************

            escflag = 0
CH1INDP2ii:
ENTPART2ii:
            n = 1: twasteevapinCF = 1: twasteevapinUNIT$ = "Degrees F    ": twasteevapinUNIT1$ = twasteevapinUNIT$: twasteevapin = 0
            CLS : CALL border: : LOCATE 6, 6: PRINT "Type in the evaporator waste heat inlet temperature. Use Function key <F1>"
            LOCATE 7, 6: PRINT "to select proper units for your entry. Press <F9> when finished."
            COLOR 15, bac: LOCATE 23, 7: PRINT "<F1>-TOGGLE UNITS        <F9>-CONFIRM ENTRY        <ESC>-PREVIOUS MENU"; : COLOR frg, bac
            LOCATE 19, 7: PRINT "Evaporator waste heat inlet temperature    ";
PRTUNIT2ii:
```

```
            LOCATE 19, 56: PRINT twasteevapinUNIT$;
20002   LOCATE 19, 40, 0: trap$ = INKEY$: IF trap$ = "" GOTO 20002
            IF trap$ = "I" GOTO 21002
            IF ASC(trap$) = 8 THEN twasteevapin$ = "": twasteevapin = 0: LOCATE 19, 47: PRINT "       ": GOTO 20002
            IF trap$ = "A" GOTO CHANGEUNIT2ii:
    IF ASC(trap$) = 27 THEN
    escflag = 1
    GOTO 21502
    END IF
            IF ASC(trap$) < 46 OR ASC(trap$) > 57 OR ASC(trap$) = 47 GOTO 20002
            twasteevapin$ = twasteevapin$ + trap$
            LOCATE 19, 47: PRINT VAL(twasteevapin$); : twasteevapin = VAL(twasteevapin$): GOTO 20002
21002   CLS : CALL border: : LOCATE 12, 3: PRINT "Your chosen evaporator waste heat inlet temperature is "; twasteevapin; " "; twasteevapinUNIT$
            LOCATE 23, 12: PRINT "  <F1>- Correct                <F2>-Incorrect    ";
21102   DTCHECK$ = INKEY$: IF DTCHECK$ = "A" GOTO 21502
            IF DTCHECK$ = "B" THEN twasteevapin$ = "": GOTO ENTPART2ii:
            GOTO 21102
CHANGEUNIT2ii:

' ******************************CHANGING UNITS**************************

            n = n + 1: IF n = 3 THEN n = 1
            ON n GOTO 21202, 21212
21202   twasteevapinCF = 1: twasteevapinUNIT$ = "Degrees F    ": GOTO PRTUNIT2ii:
21212   twasteevapinCF = 2: twasteevapinUNIT$ = "Degrees C    ": GOTO PRTUNIT2ii:

' ********************twasteevapin IS IN DEGREES F*************************

21502   CLS : IF twasteevapinCF = 2 THEN twasteevapin = (twasteevapin * 1.8) + 32

END SUB

SUB wasteheatin (twaste) STATIC

' ************************************************************************
' THIS PROGRAM IS USED TO INPUT WASTE INLET TEMPERATURE (twaste) BY
' INDIVIDUAL SCREENS
' ************************************************************************

            escflag = 0
CH1INDP2:
ENTPART2:
            n = 1: twasteCF = 1: twasteUNIT$ = "Degrees F    ": twasteUNIT1$ = twasteUNIT$: twaste = 0
            CLS : CALL border: : LOCATE 6, 6: PRINT "Type in the waste heat inlet temperature. Use Function key <F1>"
            LOCATE 7, 6: PRINT "to select proper units for your entry. Press <F9> when finished."
            COLOR 15, bac: LOCATE 23, 7: PRINT "<F1>-TOGGLE UNITS       <F9>-CONFIRM ENTRY       <ESC>-PREVIOUS MENU"; : COLOR frg, bac
            LOCATE 19, 7: PRINT "Waste Heat inlet temperature    ";
PRTUNIT2:
            LOCATE 19, 56: PRINT twasteUNIT$;
20000   LOCATE 19, 40, 0: trap$ = INKEY$: IF trap$ = "" GOTO 20000
            IF trap$ = "I" GOTO 21000
            IF ASC(trap$) = 8 THEN twaste$ = "": twaste = 0: LOCATE 19, 45: PRINT "       ": GOTO 20000
            IF trap$ = "A" GOTO CHANGEUNIT2:
    IF ASC(trap$) = 27 THEN
    escflag = 1
    GOTO 21500
    END IF
            IF ASC(trap$) < 46 OR ASC(trap$) > 57 OR ASC(trap$) = 47 GOTO 20000
            twaste$ = twaste$ + trap$
            LOCATE 19, 45: PRINT VAL(twaste$); : twaste = VAL(twaste$): GOTO 20000
21000   CLS : CALL border: : LOCATE 12, 8: PRINT "Your chosen waste heat inlet temperature is  "; twaste; " "; twasteUNIT$
            LOCATE 23, 12: PRINT "  <F1>- Correct                <F2>-Incorrect    ";
21100   DTCHECK$ = INKEY$: IF DTCHECK$ = "A" GOTO 21500
            IF DTCHECK$ = "B" THEN twaste$ = "": GOTO ENTPART2:
            GOTO 21100
CHANGEUNIT2:

' ******************************CHANGING UNITS**************************

            n = n + 1: IF n = 3 THEN n = 1
            ON n GOTO 21200, 21210
21200   twasteCF = 1: twasteUNIT$ = "Degrees F    ": GOTO PRTUNIT2:
21210   twasteCF = 2: twasteUNIT$ = "Degrees C    ": GOTO PRTUNIT2:

' ********************twaste IS IN DEGREES F*************************

21500   CLS : IF twasteCF = 2 THEN twaste = (twaste * 1.8) + 32
```

```
END SUB

SUB wtabs (wt$, wt) STATIC

' ************************************************************************
' THIS PROGRAM IS USED TO INPUT THE CONCENTRATION OF WORKING FLUID LEAVING
' THE DESORBER (WT) BY INDIVIDUAL SCREENS
' ************************************************************************

        escflag = 0
WTINP:
entwt:

' ****************************************************************
' USER COULD ESCAPE TO PREVIOUS MENU BY PRESSING <ESC> KEY OR
' ACCEPT THE INPUT BY PRESSING <F9>
'****************************************************************

        wTUNIT$ = " wt%": wt = 0: wt$ = ""
        CLS : CALL border: : LOCATE 6, 6: PRINT "Type in the working fluid concentration leaving the desorber. "
        LOCATE 7, 6: PRINT "Use Function key <F9> when finished."
        COLOR 15, bac: LOCATE 23, 7: PRINT "<F9>-CONFIRM ENTRY                    <ESC>-PREVIOUS MENU"; : COLOR frg, bac
        LOCATE 19, 7: PRINT "Working fluid concentration leaving the desorber  ";
wTUNIT:
        LOCATE 19, 68: PRINT wTUNIT$;
wtdec:
   LOCATE 19, 40, 0: trap$ = INKEY$: IF trap$ = "" GOTO wtdec:
        IF trap$ = "I" GOTO wtdec1:
        IF ASC(trap$) = 8 THEN wt = 0: wt$ = "": LOCATE 19, 56: PRINT "          ": GOTO wtdec:
    IF ASC(trap$) = 27 THEN
    escflag = 1
    GOTO wtdecend:
    END IF
        IF ASC(trap$) < 46 OR ASC(trap$) > 57 OR ASC(trap$) = 47 GOTO wtdec:
        wt$ = wt$ + trap$
        LOCATE 19, 57: PRINT VAL(wt$); : wt = VAL(wt$): GOTO wtdec:

' ****************************************************************
' CONFIRM YOUR WT% INPUT PRESS <F1> FOR CORRECT INPUT OR <F2> FOR
' INCORRECT INPUT
' ****************************************************************

wtdec1:
        CLS : CALL border: : LOCATE 12, 3: PRINT "Your chosen working fluid concentration leaving the desorber is "; wt; wTUNIT$
        LOCATE 23, 12: PRINT "  <F1>- Correct                <F2>-Incorrect    ";
wtdec3:
        DTCHECK$ = INKEY$: IF DTCHECK$ = "A" GOTO wtdec2:
        IF DTCHECK$ = "B" THEN wt$ = "": GOTO entwt:
        GOTO wtdec3:


' ****************************************************************

wtdec2:
        CLS

' ******************wt IS IN wt%**************************

        IF wt = 0 THEN
        CALL noise
        CLS : CALL border
        LOCATE 15, 3: PRINT "You have entered "; wt; " wt% for the fluid concentration"
        LOCATE 17, 3: PRINT "Please input a number other than zero followed by wt% sign"
    LOCATE 23, 26: PRINT "Press any key to continue"
    DO WHILE INKEY$ = "": LOOP
    GOTO entwt:
    END IF
wtdecend:

END SUB

SUB xhtcalalk (hxt, T, wtout, wt, hvapor, wtabsin) STATIC

' ************************************************************************
' This program is used to calculate the wt% of (Li, K, Na) NO3 mixture (wtout)
' at a given temperature (T in Deg. F) and pressure after throttled.
' wt is wt% leaving the absorber, wtabs is wt% entering the absorber, and
' hvapor is the enthalpy of water vapor in the desorber.
```

```
' ********************************************************************************
            escflag = 0

            alpha1 = 394.516
            alpha2 = -8.64996
            alpha3 = .0628787
            beta1 = .388691
            beta2 = 3.02719E-03
            beta3 = -3.80068E-05
            aaa1 = alpha3 + beta3 * T
            bbb1 = alpha2 + beta2 * T + (hvapor - hxt) / wt
            ccc1 = alpha1 + beta1 * T - hvapor
            sqrterm = bbb1 ^ 2 - 4 * aaa1 * ccc1
            IF sqrterm < 0 THEN PRINT "IMAGINARY ROOTS FOR WT% IN DESORBER": END
            x11 = (-bbb1 + SQR(sqrterm)) / (2 * aaa1)
            x22 = (-bbb1 - SQR(sqrterm)) / (2 * aaa1)
            IF x11 > wt AND x11 < wtabsin THEN
            wtout = x11
            ELSEIF x22 > wt AND x22 < wtabsin THEN
            wtout = x22
            ELSE
            CLS : CALL border
            CALL noise
            LOCATE 10, 3: PRINT "UNREASONABLE NUMBERS FOR WT% IN DESORBER "; x11, x22
            LOCATE 15, 27: PRINT "PLEASE CHECK YOUR INPUTS"
LOCATE 23, 26: PRINT "PRESS ANY KEY TO CONTINUE"
DO WHILE INKEY$ = "": LOOP
            escflag = 1
            END IF

END SUB

SUB xhtcalalkHA (hxt, T, wtout, wt, hvapor, wtabsout)

' ********************************************************************************
' This program is used to calculate the wt% of (Li, K, Na) NO3 mixture (wtout)
' at a given temperature (T in Deg. F) and pressure after throttled.
' wt is wt% leaving the desorber, wtabsout is wt% leaving the absorber, and
' hvapor is the enthalpy of water vapor in the desorber.
' ********************************************************************************
            escflag = 0

            alpha1 = 394.516
            alpha2 = -8.64996
            alpha3 = .0628787
            beta1 = .388691
            beta2 = 3.02719E-03
            beta3 = -3.80068E-05
            aaa1 = alpha3 + beta3 * T
            bbb1 = alpha2 + beta2 * T + (hvapor - hxt) / wt
            ccc1 = alpha1 + beta1 * T - hvapor
            sqrterm = bbb1 ^ 2 - 4 * aaa1 * ccc1
            IF sqrterm < 0 THEN PRINT "IMAGINARY ROOTS FOR WT% IN DESORBER": END
            x11 = (-bbb1 + SQR(sqrterm)) / (2 * aaa1)
            x22 = (-bbb1 - SQR(sqrterm)) / (2 * aaa1)
            IF x11 > wt AND x11 < 1.2 * wt THEN
            wtout = x11
            ELSEIF x22 > wt AND x22 < wt * 1.2 THEN
            wtout = x22
            ELSE
            CLS : CALL border
            CALL noise
            LOCATE 10, 3: PRINT "UNREASONABLE NUMBERS FOR WT% IN ABSORBER "; x11, x22
            LOCATE 15, 27: PRINT "PLEASE CHECK YOUR INPUTS"
LOCATE 23, 26: PRINT "PRESS ANY KEY TO CONTINUE"
DO WHILE INKEY$ = "": LOOP
            escflag = 1
            END IF

END SUB

SUB xhtcalLiBr (hxt, T, wtout, wt, hvapor, wtabsin) STATIC

' ********************************************************************************
' This program is used to calculate the wt% of LiBr solution (wtout) at a
' given temperature (T in Deg. F) and pressure.
' wt is wt% leaving the absorber, wtabsin is wt% entering the absorber, and
' hvapor is the enthalpy of vapor leaving the absorber.
```

```
' ************************************************************************
            escflag = 0

            a01 = -1015.07
            a02 = 79.5387
            a03 = -2.35802
            a04 = .0303158
            a05 = -1.4003E-04
            a11 = 4.68108
            a12 = -.303777
            a13 = 8.44845E-03
            a14 = -1.04772E-04
            a15 = 4.801E-07
            a21 = -.0049107
            a22 = 3.83184E-04
            a23 = -1.079E-05
            a24 = 1.3152E-07
            a25 = -5.897E-10
            i = 0: wtout = wt
3080
            a0 = a01 + a02 * wtout + a03 * wtout ^ 2 + a04 * wtout ^ 3 + a05 * wtout ^ 4
            a1 = a11 + a12 * wtout + a13 * wtout ^ 2 + a14 * wtout ^ 3 + a15 * wtout ^ 4
            a2 = a21 + a22 * wtout + a23 * wtout ^ 2 + a24 * wtout ^ 3 + a25 * wtout ^ 4
            fx = a0 + a1 * T + a2 * T ^ 2 + wtout * (hvapor - hxt) / wt - hvapor
            a0prime = a02 + 2 * a03 * wtout + 3 * a04 * wtout ^ 2 + 4 * a05 * wtout ^ 3
            a1prime = a12 + 2 * a13 * wtout + 3 * a14 * wtout ^ 2 + 4 * a15 * wtout ^ 3
            a2prime = a22 + 2 * a23 * wtout + 3 * a24 * wtout ^ 2 + 4 * a25 * wtout ^ 3
            dfx = a0prime + a1prime * T + a2prime * T ^ 2 + (hvapor - hxt) / wt
            IF i > 200 GOTO 60080
            wtnew = wtout - (fx / dfx)
            IF ABS(fx) > .0001 THEN wtout = wtnew: i = i + 1: GOTO 3080
60080       IF ABS(fx) > .0001 AND i > 200 THEN PRINT "Did not converge for wt% in the desorber": END
            wtout = wtnew
            IF wtout > wtabsin THEN
            CLS : CALL border
            CALL noise
            LOCATE 10, 5: PRINT "UNREASONABLE NUMBERS FOR WT% IN DESORBER "; wtout
            LOCATE 15, 27: PRINT "PLEASE CHECK YOUR INPUTS"
LOCATE 23, 26: PRINT "PRESS ANY KEY TO CONTINUE"
DO WHILE INKEY$ = "": LOOP
            escflag = 1
            END IF
END SUB

SUB xhtcalLiBrHA (hxt, T, wtout, wt, hvapor, wtabsout)

' ************************************************************************
' This program is used to calculate the wt% of LiBr solution (wtout) at a
' given temperature (T in Deg. F) and pressure.
' wt is wt% leaving the desorber, wtabsout is wt% leaving the absorber, and
' hvapor is the enthalpy of vapor leaving the absorber.
' ************************************************************************
            escflag = 0

            a01 = -1015.07
            a02 = 79.5387
            a03 = -2.35802
            a04 = .0303158
            a05 = -1.4003E-04
            a11 = 4.68108
            a12 = -.303777
            a13 = 8.44845E-03
            a14 = -1.04772E-04
            a15 = 4.801E-07
            a21 = -.0049107
            a22 = 3.83184E-04
            a23 = -1.079E-05
            a24 = 1.3152E-07
            a25 = -5.897E-10
            i = 0: wtout = wt
30833
            a0 = a01 + a02 * wtout + a03 * wtout ^ 2 + a04 * wtout ^ 3 + a05 * wtout ^ 4
            a1 = a11 + a12 * wtout + a13 * wtout ^ 2 + a14 * wtout ^ 3 + a15 * wtout ^ 4
            a2 = a21 + a22 * wtout + a23 * wtout ^ 2 + a24 * wtout ^ 3 + a25 * wtout ^ 4
            fx = a0 + a1 * T + a2 * T ^ 2 + wtout * (hvapor - hxt) / wt - hvapor
            a0prime = a02 + 2 * a03 * wtout + 3 * a04 * wtout ^ 2 + 4 * a05 * wtout ^ 3
            a1prime = a12 + 2 * a13 * wtout + 3 * a14 * wtout ^ 2 + 4 * a15 * wtout ^ 3
            a2prime = a22 + 2 * a23 * wtout + 3 * a24 * wtout ^ 2 + 4 * a25 * wtout ^ 3
```

```
        dfx = a0prime + a1prime * T + a2prime * T ^ 2 + (hvapor - hxt) / wt
        IF i > 200 GOTO 60033
        wtnew = wtout - (fx / dfx)
        IF ABS(fx) > .0001 THEN wtout = wtnew: i = i + 1: GOTO 30833
60033   IF ABS(fx) > .0001 AND i > 200 THEN PRINT "Did not converge for wt% in the desorber": END
        wtout = wtnew
        IF wtout > 1.2 * wt OR wtout < wtabsout THEN
        CLS : CALL border
        CALL noise
        LOCATE 10, 5: PRINT "UNREASONABLE NUMBERS FOR WT% IN ABSORBER "; wtout
        LOCATE 15, 27: PRINT "PLEASE CHECK YOUR INPUTS"
LOCATE 23, 26: PRINT "PRESS ANY KEY TO CONTINUE"
DO WHILE INKEY$ = "": LOOP
        escflag = 1
        END IF

END SUB
```

Appendix B

Samples of Input and Output

```
Run Identity: tapp=6              Working Fluid:  73 wt% (Li, K, Na) NO3

Waste Heat Inlet Temperature (evaporator): 66.2        Deg. C
Absorber Capacity: 3.75        MM Btu/hr
Cooling Water Temperature (absorber & condenser): 83          Deg. C
Approach Temperature in Condenser: 5.5           Deg. C
Approach Temperature in Evaporator:  5             Deg. C
Approach Temperature in Desorber:  10            Deg. C
Approach Temperature in Absorber:  6             Deg. C
Approach Temperature in Solution Heat Exchanger:  5.6           Deg. C




<F1>-TOGGLE UNITS      <CR>-TOGGLE DATA      <TAB>-END INPUT      <ESC>-ESCAPE
```

```
05-23-1991              Run Identity tapp=6

Working fluid 73 wt% (Li, K, Na) NO3

Concentration of solution leaving the desorber 73  wt%
Concentration of solution leaving the absorber 69.74132  wt%
Temperature of solution leaving the absorber  192.2  Deg. F
Temperature of solution leaving the desorber  259.2013  Deg. F
Lift temperature  50.04001  Deg. F
Concentration of solution entering the absorber  73.15459  wt%
COP heating and cooling  1.861032  and   .8610326
Steam flow rate  3790.902  lb/hr for a capacity of  3750000  Btu/hr
Refrigerant (water) flow rate  3511.571  lb/hr
Concentrated solution flow rate  75153.56  lb/hr
Dilute solution flow rate  78665.13  lb/hr
Evaporator waste flow rate  3358.494  lb/hr
Desorber waste flow rate  4238.319  lb/hr
Condenser water flow rate  3599.643  lb/hr


                        PRESS ANY KEY TO CONTINUE
```

05-23-1991 Run Identity tapp=6

Working fluid 73 wt% (Li, K, Na) NO3

Total heat transferred in the absorber 3750000 Btu/hr
Total heat transferred in the desorber 3928359 Btu/hr
Total heat transferred in the condenser 3560805 Btu/hr
Total heat transferred in the evaporator 3382446 Btu/hr
Total heat transferred in the solution HX 1587456 Btu/hr
Required UA for the absorber 245255 Btu/hr-Deg. F
Required UA for the desorber 150415.2 Btu/hr-Deg. F
Required UA for the condenser 108113.4 Btu/hr-Deg. F
Required UA for the evaporator 298895.1 Btu/hr-Deg. F
Required UA for the solution HX 95102.81 Btu/hr-Deg. F
Total UA required 897781.5 Btu/hr-Deg. F
Power supplied to pump after absorber 1644.537 Btu/hr
COP corrected for parasitic pumping power 1.857921

PRESS ANY KEY TO CONTINUE

05-23-1991 Run Identity tapp=6

Operating parameters

Working fluid 73 wt% (Li, K, Na) NO3

Stream Description	T (Deg. F)	P (psia)	wt%	Enthalpy (Btu/lb)	Flow (lb/hr)
Steam from Gen.	259.2	9.60	0.0	1173.2	3511.6
Water from Con.	191.3	9.60	0.0	159.2	3511.6
Steam from Evp.	142.2	3.05	0.0	1122.4	3511.6
Sol. from Abs.	192.2	3.05	69.7	176.8	78665.1
Sol. to Gen.	240.8	9.60	69.7	197.0	78665.1
Sol. from Gen.	259.2	9.60	73.0	203.7	75153.6
Sol. to Abs.	202.3	3.05	73.2	180.5	75153.6

PRESS ANY KEY TO CONTINUE

```
Run Identity: tapp=5.5          Working Fluid:  52.8 wt% LiBr

Waste Heat Inlet Temperature (evaporator): 66.2        Deg. C
Absorber Capacity: 3.75        MM Btu/hr
Cooling Water Temperature (absorber & condenser): 83          Deg. C
Approach Temperature in Condenser: 5.5           Deg. C
Approach Temperature in Evaporator:  5             Deg. C
Approach Temperature in Desorber:  10            Deg. C
Approach Temperature in Absorber:  5.5           Deg. C
Approach Temperature in Solution Heat Exchanger:  5.6           Deg. C









<F1>-TOGGLE UNITS     <CR>-TOGGLE DATA     <TAB>-END INPUT     <ESC>-ESCAPE
```

```
05-23-1991            Run Identity tapp=5.5

Working fluid 52.8 wt% LiBr

Concentration of solution leaving the desorber 52.8  wt%
Concentration of solution leaving the absorber 49.80564  wt%
Temperature of solution leaving the absorber  191.3  Deg. F
Temperature of solution leaving the desorber  257.059  Deg. F
Lift temperature  49.14  Deg. F
Concentration of solution entering the absorber  52.92543  wt%
COP heating and cooling  1.781708  and   .7817084
Steam flow rate  3790.902  lb/hr for a capacity of  3750000  Btu/hr
Refrigerant (water) flow rate  3171.527  lb/hr
Concentrated solution flow rate  52752.52  lb/hr
Dilute solution flow rate  55924.05  lb/hr
Evaporator waste flow rate  3033.273  lb/hr
Desorber waste flow rate  4209.442  lb/hr
Condenser water flow rate  3247.935  lb/hr


                      PRESS ANY KEY TO CONTINUE
```

05-23-1991 Run Identity tapp=5.5

Working fluid 52.8 wt% LiBr

Total heat transferred in the absorber 3750000 Btu/hr
Total heat transferred in the desorber 3907986 Btu/hr
Total heat transferred in the condenser 3212892 Btu/hr
Total heat transferred in the evaporator 3054906 Btu/hr
Total heat transferred in the solution HX 1366548 Btu/hr
Required UA for the absorber 261234.1 Btu/hr-Deg. F
Required UA for the desorber 146626.8 Btu/hr-Deg. F
Required UA for the condenser 99363.83 Btu/hr-Deg. F
Required UA for the evaporator 269951.5 Btu/hr-Deg. F
Required UA for the solution HX 78921.27 Btu/hr-Deg. F
Total UA required 856097.4 Btu/hr-Deg. F
Power supplied to pump after absorber 1215.363 Btu/hr
COP corrected for parasitic pumping power 1.779495

PRESS ANY KEY TO CONTINUE

05-23-1991 Run Identity tapp=5.5

Operating parameters

Working fluid 52.8 wt% LiBr

Stream Description	T (Deg. F)	P (psia)	wt%	Enthalpy (Btu/lb)	Flow (lb/hr)
Steam from Gen.	257.1	9.60	0.0	1172.3	3171.5
Water from Con.	191.3	9.60	0.0	159.2	3171.5
Steam from Evp.	142.2	3.05	0.0	1122.4	3171.5
Sol. from Abs.	191.3	3.05	49.8	81.1	55924.0
Sol. to Gen.	237.3	9.60	49.8	105.5	55924.0
Sol. from Gen.	257.1	9.60	52.8	115.5	52752.5
Sol. to Abs.	201.4	3.05	52.9	87.1	52752.5

PRESS ANY KEY TO CONTINUE

```
Run Identity: tapp=10              Working Fluid:  90 wt% (Li, K, Na) NO3

Waste Heat Inlet Temperature: 174.4       Deg. C
Capacity: 2.47        MM Btu/hr
Condenser Water Inlet Temperature: 66          Deg. C
Approach Temperature in Condenser: 10            Deg. C
Approach Temperature in Evaporator:  5             Deg. C
Approach Temperature in Desorber:  10            Deg. C
Approach Temperature in Absorber:  10            Deg. C
Approach Temperature in Solution Heat Exchanger:  10            Deg. C
Cooling Water Temperature Entering the Absorber:  250           Deg. C
Desorber Temperature:   165           Deg. C




<F1>-TOGGLE UNITS     <CR>-TOGGLE DATA     <TAB>-END INPUT     <ESC>-ESCAPE
```

```
05-23-1991              Run Identity tapp=10

Working fluid 90 wt% (Li, K, Na) NO3
Maximum possible boost temperature  538.3148  Deg. F
Concentration of solution entering the absorber 90  wt%
Concentration of solution leaving the absorber 85.84291  wt%
Temperature of solution leaving the absorber  500  Deg. F
Temperature of solution leaving the desorber  327.92  Deg. F
Lift temperature  172.08  Deg. F
Concentration of solution entering the desorber  86.75191  wt%
COP  .4334355
Steam flow rate  3355.923  lb/hr for a capacity of  2470000  Btu/hr
Refrigerant (water) flow rate  3064.083  lb/hr
Concentrated solution flow rate  63272.6  lb/hr
Dilute solution flow rate  66336.68  lb/hr
Evaporator waste flow rate  3630.822  lb/hr
Desorber waste flow rate  2888.493  lb/hr
Condenser water flow rate  218070.2  lb/hr


                        PRESS ANY KEY TO CONTINUE
```

05-23-1991 Run Identity tapp=10

Working fluid 90 wt% (Li, K, Na) NO3
Maximum possible boost temperature 538.3148 Deg. F
Total heat transferred in the absorber 2470000 Btu/hr
Total heat transferred in the desorber 2524886 Btu/hr
Total heat transferred in the condenser 3228657 Btu/hr
Total heat transferred in the evaporator 3173770 Btu/hr
Total heat transferred in the solution HX 3444172 Btu/hr
Required UA for the absorber 123914.1 Btu/hr-Deg. F
Required UA for the desorber 144656 Btu/hr-Deg. F
Required UA for the condenser 39856.75 Btu/hr-Deg. F
Required UA for the evaporator 48367.31 Btu/hr-Deg. F
Required UA for the solution HX 143573.1 Btu/hr-Deg. F
Total UA required 500367.3 Btu/hr-Deg. F
Power supplied to pump after desorber 16517.6 Btu/hr
Power supplied to pump after condenser 1446.526 Btu/hr
Total power supplied to both pumps 17964.13 Btu/hr
COP corrected for parasitic pumping power .4280382

PRESS ANY KEY TO CONTINUE

05-23-1991 Run Identity tapp=10

Operating parameters

Working fluid 90 wt% (Li, K, Na) NO3

Stream Description	T (Deg. F)	P (psia)	wt%	Enthalpy (Btu/lb)	Flow (lb/hr)
Steam from Gen.	327.9	8.13	0.0	1205.2	3064.1
Water from Con.	183.6	8.13	0.0	151.5	3064.1
Steam from Evp.	327.9	100.75	0.0	1187.3	3064.1
Sol. from Abs.	500.0	100.75	85.8	299.6	66336.7
Sol. to Gen.	329.0	8.13	86.8	237.5	66336.7
Sol. from Gen.	327.9	8.13	90.0	241.2	63272.6
Sol. to Abs.	482.0	100.75	90.0	295.6	63272.6

PRESS ANY KEY TO CONTINUE

```
Run Identity: tapp=10              Working Fluid:  70 wt% LiBr

Waste Heat Inlet Temperature: 235          Deg. F
Capacity: 2            MM Btu/hr
Condenser Water Inlet Temperature: 80          Deg. F
Approach Temperature in Condenser: 10            Deg. F
Approach Temperature in Evaporator:  10            Deg. F
Approach Temperature in Desorber:  10            Deg. F
Approach Temperature in Absorber:  10            Deg. F
Approach Temperature in Solution Heat Exchanger:  10              Deg. F
Cooling Water Temperature Entering the Absorber:  338             Deg. F
Desorber Temperature:   230            Deg. F



<F1>-TOGGLE UNITS      <CR>-TOGGLE DATA      <TAB>-END INPUT      <ESC>-ESCAPE
```

```
05-23-1991              Run Identity tapp=10

Working fluid 70 wt% LiBr
Maximum possible boost temperature  375.1813  Deg. F
Concentration of solution entering the absorber 70  wt%
Concentration of solution leaving the absorber 66.00815  wt%
Temperature of solution leaving the absorber  348  Deg. F
Temperature of solution leaving the desorber  225  Deg. F
Lift temperature  123  Deg. F
Concentration of solution entering the desorber  66.40189  wt%
COP  .4828229
Steam flow rate  2271.084  lb/hr for a capacity of  2000000  Btu/hr
Refrigerant (water) flow rate  1960.084  lb/hr
Concentrated solution flow rate  32411.4  lb/hr
Dilute solution flow rate  34371.49  lb/hr
Evaporator waste flow rate  2223.133  lb/hr
Desorber waste flow rate  2111.566  lb/hr
Condenser water flow rate  222966.3  lb/hr


                        PRESS ANY KEY TO CONTINUE
```

05-23-1991 Run Identity tapp=10

Working fluid 70 wt% LiBr
Maximum possible boost temperature 375.1813 Deg. F
Total heat transferred in the absorber 2000000 Btu/hr
Total heat transferred in the desorber 2017845 Btu/hr
Total heat transferred in the condenser 2142306 Btu/hr
Total heat transferred in the evaporator 2124460 Btu/hr
Total heat transferred in the solution HX 1468780 Btu/hr
Required UA for the absorber 131409.6 Btu/hr-Deg. F
Required UA for the desorber 279732.8 Btu/hr-Deg. F
Required UA for the condenser 35745.6 Btu/hr-Deg. F
Required UA for the evaporator 35211.73 Btu/hr-Deg. F
Required UA for the solution HX 97452.59 Btu/hr-Deg. F
Total UA required 579552.3 Btu/hr-Deg. F
Power supplied to pump after desorber 1243.963 Btu/hr
Power supplied to pump after condenser 142.8089 Btu/hr
Total power supplied to both pumps 1386.772 Btu/hr
COP corrected for parasitic pumping power .4821772

PRESS ANY KEY TO CONTINUE

05-23-1991 Run Identity tapp=10

Operating parameters

Working fluid 70 wt% LiBr

Stream Description	T (Deg. F)	P (psia)	wt%	Enthalpy (Btu/lb)	Flow (lb/hr)
Steam from Gen.	225.0	0.94	0.0	1160.6	1960.1
Water from Con.	99.6	0.94	0.0	67.6	1960.1
Steam from Evp.	215.0	15.59	0.0	1151.5	1960.1
Sol. from Abs.	348.0	15.59	66.0	168.3	34371.5
Sol. to Gen.	230.0	0.94	66.4	119.4	34371.5
Sol. from Gen.	225.0	0.94	70.0	125.2	32411.4
Sol. to Abs.	338.0	15.59	70.0	170.6	32411.4

PRESS ANY KEY TO CONTINUE

INTERNAL DISTRIBUTION

1-5.	A. Zaltash	26.	M. R. Patterson
5-10.	M. R. Ally	27.	D. E. Reichle
11.	P. S. Gillis	28.	R. H. Reiner
12.	R. S. Carlsmith	29.	R. B. Shelton
13.	G. E. Courville	30.	J. N. Stone
14.	V. D. Baxter	31.	T. J. Wilbanks
15.	F. A. Creswick	32-36.	Energy Conservation Dist.
16.	R. C. DeVault	37.	ORNL Patent Office
17.	P. D. Fairchild	38-39.	Central Research Library
18.	M. A. Kuliasha	40.	Document Reference Section
19.	V. C. Mei	41-42.	Laboratory Records
20-25.	R. R. Parks	43.	Laboratory Records - RC

EXTERNAL DISTRIBUTION

44. W. Bierman, 45 Foxcroft Drive, Fayetteville, NY 13066
45. B. G. Buchanan, Computer Science Department, University of Pittsburgh, 206 Mineral Industries Bldg, Pittsburgh, PA 15260
46. J. M. Calm, Electric Power Research Institute, 3412 Hillview Avenue, P. O. Box 10412, Palo Alto, CA 94303
47. J. G. Carbonell, Associate Professor of Computer Science, Carnegie-Mellon University, Pittsburgh, PA 15213
48. G. Douglas Carver, TVA, 703 Power Building, Chattanooga, TN 37402
49. R. N. Chappell, Program Manager, Energy Conservation Branch, Energy and Technology Division, DOE Idaho Operations Office, 785 DOE Place, Idaho Falls, ID 83402
50. J. J. Cuttica, Vice President of Research & Development, Gas Research Institute, 8600 W. Bryn Mawr Avenue, Chicago, IL 60631
51. D. C. Erickson, Energy Concepts Company, 627 Ridgely Avenue, Annapolis, MD 21401
52. R. J. Fiskum, Program Manager, Energy Conservation Equipment Branch, CE-422, Room 5H-048, Department of Energy, 1000 Independence Avenue SW, Washington, DC 20585
53. Joel Gilbert, Dames and Moore, Inc., 7101 Wisconsin Avenue, Suite 700, Bethesda, MD 20814
54. S. M. Gillis, Professor, Public Policy and Economics, Duke University, 4875 Duke Station, Durham, NC 27706
55. G. Grossman, Technion Institute of Technology, Faculty of Mechanical Engineering, Haifa Israel
56. W. T. Hanna, Battelle Columbus Laboratories, 505 King Avenue, Columbus, OH 43201

57. A. Hirsch, Vice President, Environmental Sciences and Director, Washington Operations, 5129 Leesburg Pk, Suite 414, Falls Church, VA 22041
58. F. R. Kalhammer, Vice President, Electric Power Research Institute, P.O. Box 10412, Palo Alto, CA 94303
59. Alan Karp, Electric Power Research Institute, Energy Management and Utilization Division, P.O. Box 10412, Palo Alto, CA 94303
60. R. E. Kasperson, Professor of Government and Geography, Graduate School of Geography, Clark University, Worcester, MA 01610
61. M. Lessen, Consulting Engineer, 12 Country Club Drive, Rochester, NY 14618
62. R. Macriss, Institute of Gas Technology, 3424 South State Street, Chicago, IL 60616
63. L. A. McNeely, 7310 Steinmeier Drive, Indianapolis, IN 46250
64. D. K. Miller, Borg-Warner Air Conditioning, Inc., P.O. Box 1592, York PA 17405
65. J. I. Mills, EG&G Idaho, Inc., P. O. Box 1625-WCB, Idaho Falls, ID 83415
66. D. E. Morrison, 333 Oxford Road, East Lansing, MI 48823
67. Ralph Nader, Post Office Box 19367, Washington, D.C. 20036
68. H. Perez-Blanco, 314 Mechanical Engineering Bldg., Penn State, University Park, PA 16802
69. B. A. Phillips, Phillips Engineering Company, 721 Pleasant Street, St. Joseph, MI 49085
70. R. Radermacher, University of Maryland, Mechanical Engineering Department, College Park, MD 20742
71. R. C. Reimann, 5504 Ortloff Road, LaFayette, NY 13084
72. S. L. Richlen, Department of Energy, CE-221, 5F-034/FORSTL, Department of Energy, 1000 Independence Avenue SW, Washington, DC 20585
73. U. Rockenfeller, Rocky Research Company, P.O. Box 1086 Boulder City, NV 89005
74. J. D. Ryan, Building Equipment Division, Department of Energy, CE-422, 5H-048/FORS, 1000 Independence Avenue SW, Washington, DC 20585
75. D. S. Severson, Gas Research Institute, 8800 West Bryn Mawr Avenue, Chicago, IL 60631
76. S. V. Shelton, Georgia Institute of Technology, Department of Mechanical Engineering, Atlanta, GA 30332
77. J. J. Tuzson, Gas Research Institute, 8800 West Bryn Mawr Avenue, Chicago, IL 60631
78. M. Wahlig, Lawrence Berkeley Laboratory, University of California, Berkeley, CA 94720
79. M. Williams, Professor, Department of Economics, Northern Illinois University, Dekalb, IL 60115
89. Office of the Assistant Manager for Energy Research and Development, Department of Energy, Oak Ridge Operations, Oak Ridge, TN 37831
90-99. OSTI, U. S. Department of Energy, P.O. Box 62, Oak Ridge TN 37831