REFERENCES

Bird, R.B., Stewart, W.E., and Lightfoot, E.N. Transport Phenomena, Wiley, New York, (1970).

Christoffel, E.G., "Kinetics of Simultaneous Isomerization and Cracking of the Five Hexane Isomers by Use of the Wei-Prater Method", Ind. Eng. Chem. Prod. Res. Dev., Vol. 18, No.2, (1979), pp. 143-148.

Ciapetta, F.G., Ind. Eng. Chem., 45 , (1953), pp. 159.

Ciapetta, F.G., and D. N. Wallace, Catal. Rev., 5, (1971), pp. 67.

Clark, A., Ind. Eng. Chem., 45, (1953), pp. 1476.

Constantinides Alkis Applied Numerical Methods with Personal Computers, McGraw-Hill, Singapore, 1987.

Emmett, P.H., ed., Catalysis, Volume VI. Alkylation, Isomerization, Polymerization, Cracking and Hydroreforming, Reinhold, New York, 1958.

Froment, G.F., Bischoff, K.B. Chemical Reactor Analysis and Design, John Wiley & Sons, 1979.

Froment, G.F., "Model Discrimination and Parameter Estimation in Heterogeneous Catalysis", AIChE J., Vol.21, No. 6, (1975), pp. 1041-1057.

Gates, B.C., Katzer, J.R., Schuit, G.A. Chemistry of Catalystic Processes, McGraw-Hill, 1979.

Greensfield, B.S., et al. "Catalytic and thermal hydrocracking of pure hydrocarbons," Ind. Eng. Chem., 41(1949), pp. 2573-84.

Heinemann, H., Ind.Eng.Chem., 43, (1951), pp. 2098.

Henningsen J. and Bundgaard-Nielson M., "Catalytic Reforming", British Chemical Engineering, 15, (1970), pp. 1433.

Hettinger, W.P., Keith, C.D., Gring, J.L., and Teter, J. W., Ind. Eng. Chem., 47, (1955), 719.

Himmelblau, D.M. and Bischoff, K.B. Process Analysis and Simulation, Wiley, New York (1968).

Hougen, O.A., and Watson, K.M. Chemical Process Principles, Vol. III, Wiley, New York, 1947.

Krane, H.G., Groh, A.B., Schulman, B.L. and Sinfelt, J.H. Proc. 5th World Pet. Cong., New York, 1959, sec. III. p.39.

Kugelman, A.M., "What Affects Cat Reformer Yield?", Hydrocarbon Processing, 55, 1 (1976), pp. 95.

Levenspiel, O. Chemical Reaction Engineering, 2nd ed., Wiley, New York, 1972.

Little, D.M., Catalyst Reforming, PennWell Books, Oklahoma, 1985.

Marin, G.B., G.F. Froment, "Reforming of C_6 hydrocarbons on a Pt-Al_2O_3 catalyst", Chem. Eng. Science, Vol. 37, No. 5, , (1982), pp. 759-773.

Meerbott, W.K., Cherry, A.H., Chernoff, B., Crocoll, J.D., and Kaemmerlen, C.J., Ind. Eng. Chem., 46, (1954), pp. 2026.

Mills, G.A., Heinemann, H., Milliken, T.H. , and Oblad ,A.G., Ind. Eng. Chem., 45, (1953), pp. 136.

Reid, C.R., Prausnitz, J.M., Poling, B.E. The Properties of Gases & Liquids, Fourth Edition, McGraw-Hill, 1987.

Selman, D.M., Voorhies, A., Jr. "Methylcyclopentane Reforming with Platinum-Rhenium Catalyst. Mechanism Studies", Ind. Eng. Chem., Prod. Res. Dev., Vol. 14, No.2, (1975), pp. 118-123.

Smith, R.B., "Kinetic Analysis of Naphtha Reforming with Platinum Catalyst", Chem. Eng. Prog., 55, (1959), pp. 76.

Sterba, M.J. and Haensel V., "Catalytic Reforming", Ind. Eng. Chem., Prod. Res. Dev., Vol. 15, No. 1, 1976.

Van Trimpont, P.A., Marin, G.B., Froment, G.F., "Kinetics of Methylcyclohexane Dehydrogenation on Sulfided Commercial Platinum/Alumina and Platinum-Rhenium /Alumina Catalyst", Ind. Eng. Chem. Fundam., 25, (1986), pp. 544-553.

Van Trimpont, P.A., Marin, G.B., Froment, G.F., "Reforming of C_7 hydrocarbons on a sulfided commercial $Pt-Al_2O_3$ catalyst", Ind. Eng. Chem. Res.,27, (1988), pp.51-57.

Yang, K.H. and Hougen, O.A., Chem. Eng. Prog., 46, (1950), pp. 146.

Yaws, C. L., Chiang, P. Y., Hydrocarbon Processing, Nov 1988, pp. 81-84.

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

APPENDICES

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

APPENDIX A

PROPERTY OF DATA BANK

The simbols and equation are shown below. The enthalpy and Gibbs energy of formation at 298.2 K are for the ideal-gas state. The reference states chosen for the elements are as follows:

MW = Molecular weight, g/mol

T_c = critical temperature, K

P_c = critical pressure, bar

V_c = critical volume, cm^3/mole

Z_c = critical compressibility factor, P_cV_c/RT_c

Omega = Pitzer's acentric factor

Dipm = dipole moment, debyes

A, B, C, D = constants to calculate the isobaric heat capacity of the ideal gas with Cp in J/(mol.K) and T in kelvins:

$$Cp = A + BT + CT^2 + DT^3$$

ΔH_f = heat of formation, kcal/g-mol.

ΔG_f = Gibbs free energy, kJ/g-mol.

Table A-1 Physical properties constant (Reid et al., (1987))

COMPONENT	MW.	Tc (K)	Pc (bar)	Vc cm^3/mole	Zc	Omega	Dipm debyes	ΔH_f° kcal/g-mol
HYDROGEN	2.016	33.0	12.9	64.3	0.303	-0.216	0	0
METHANE	16.043	190.4	46.0	99.2	0.288	0.011	0	-17.88
ETHANE	30.07	305.4	48.3	148.3	0.285	0.099	0	-20.23
PROPANE	44.094	369.8	42.5	203.0	0.281	0.153	0	-24.82
BUTANE	58.124	425.2	38.0	255.0	0.274	0.199	0	-30.15
n-PENTANE	72.151	469.7	33.7	304.0	0.263	0.251	0	-35.00
n-HEXANE	86.178	507.5	30.1	370.0	0.264	0.299	0	-39.96
CYCLOHEXANE	84.162	553.5	40.7	308.0	0.273	0.212	0.3	-29.43
METHYLCYCLOPENTANE	84.162	532.7	37.8	319.0	0.272	0.231	0	-25.50
2-METHYLPENTANE	86.178	497.5	30.1	367.0	0.267	0.278	0	-41.66
3-METHYLPENTANE	86.178	504.5	31.2	367.0	0.273	0.272	0	-41.02
2,3 DIMETHYLBUTANE	86.178	500.0	31.3	358.0	0.269	0.247	0	-42.49
2,2 DIMETHYLBUTANE	86.178	488.8	30.8	359.0	0.272	0.232	0	-44.35
BENZENE	78.114	562.2	48.9	259.0	0.271	0.212	0	19.82
TOLUENE	92.141	591.8	41.0	316.0	0.263	0.263	0.4	11.95
n-HEPTANE	100.205	540.3	27.4	432.0	0.263	0.349	0	-44.89
METHYLCYCLOHEXANE	98.189	572.2	34.7	368.0	0.268	0.236	0	-36.99
ETHYLCYCLOPENTANE	98.189	569.5	34.0	375.0	0.269	0.271	0	-30.37
3-METHYLHEXANE	100.205	535.3	28.1	404.0	0.255	0.323	0	-45.96
2-METHYLHEXANE	100.205	530.4	27.3	421.0	0.261	0.329	0	-46.60
O-XYLENE	106.168	630.3	37.3	369.0	0.262	0.31	0.5	4.54
M-XYLENE	106.168	617.1	35.4	376.0	0.259	0.325	0.3	4.12
P-XYLENE	106.168	616.2	35.1	379.0	0.260	0.32	0.1	4.29

Note: ΔH_f° from Perry's Chemical Engineers' Handbook (1984)

Table A-2 Heat capacity, $Cp = A + BT + CT^2 + DT^3$ (Reid et al., (1987)) , kJ/g-mol

COMPONENT	A	B	C	D
HYDROGEN	2.7140E+01	9.2740E-03	-1.3810E-05	7.6450E-09
METHANE	1.9250E+01	5.2130E-02	1.1970E-05	-1.1320E-08
ETHANE	5.4090E+00	1.7810E-01	-6.9380E-05	8.7130E-09
PROPANE	-4.2240E+00	3.0630E-01	-1.5860E-04	3.2100E-08
BUTANE	9.4870E+00	3.3130E-01	-1.1080E-04	-2.8220E-09
n-PENTANE	-3.6260E+00	4.8730E-01	-2.5800E-04	5.3000E-08
n-HEXANE	-4.4130E+00	5.8200E-01	-3.1190E-04	6.4900E-08
CYCLOHEXANE	-5.4540E+01	6.1130E-01	-2.5230E-04	1.3210E-08
METHYLCYCLOPENTANE	-5.0110E+01	6.3810E-01	-3.6420E-04	8.0410E-08
2-METHYLPENTANE	-1.0570E+01	6.1800E-01	-3.5730E-04	8.0580E-08
3-METHYLPENTANE	-2.3860E+00	5.6900E-01	-2.8700E-04	5.0330E-08
2,3 DIMETHYLBUTANE	-1.4610E+01	6.1500E-01	-3.3760E-04	6.8200E-08
2,2 DIMETHYLBUTANE	-1.6630E+01	6.2930E-01	-3.4810E-04	6.5800E-08
BENZENE	-3.3920E+01	4.7390E-01	-3.0170E-04	7.1300E-08
TOLUENE	-2.4350E+01	5.1250E-01	-2.7650E-04	4.9110E-08
n-HEPTANE	-5.1460E+00	6.7620E-01	-3.6510E-04	7.6580E-08
METHYLCYCLOHEXANE	-6.1920E+01	7.8420E-01	-4.4380E-04	9.3660E-08
ETHYLCYCLOPENTANE	-5.5310E+01	7.5110E-01	-4.3960E-04	1.0040E-07
3-METHYLHEXANE	-7.0460E+00	6.8370E-01	-3.7340E-04	7.8340E-08
2-METHYLHEXANE	-3.9390E+01	8.6420E-01	-6.2890E-04	1.8360E-07
O-XYLENE	-1.5850E+01	5.9620E-01	-3.4430E-04	7.5280E-08
M-XYLENE	-2.9170E+01	6.2970E-01	-3.7470E-04	8.4780E-08
P-XYLENE	-2.5090E+01	6.0420E-01	-3.3740E-04	6.8200E-08

Table A-3 Gibbs free energy of formation of gas (Yaws and Chiang, 1988), $\Delta G_f = A + BT + CT^2$, (kJ/g-mol), by T in kelvins.

Component	A	B	C	ΔG_f at 298K
n-Hexane	-170.447	5.5417E-01	5.0303E-05	-0.84
2-methylpentane	-177.675	5.6303E-01	4.8313E-05	-5.60
3-methylpentane	-174.861	5.6271E-01	5.0351E-05	-2.70
2,2-dimethylbutane	-189.225	5.8649E-01	4.7623E-05	-10.22
2,3-dimethylbutane	-181.310	5.7783E-01	4.9722E-05	-4.70
Methylcyclopentane	-110.437	4.7401E-01	4.9123E-05	35.18
Benzene	81.512	1.5282E-01	2.6522E-05	129.41
n-Hepatane	-191.520	6.5052E-01	5.6444E-05	7.35
Ethylcyclopentane	-131.223	5.7136E-01	5.4772E-05	43.91
Methylcyclohexane	-160.038	6.1255E-01	4.6303E-05	26.61
Toluene	47.813	2.3831E-01	3.1916E-05	121.66
2-methylhexane	-198.645	6.5837E-01	5.6475E-05	2.56
3-methylhexane	-196.032	6.5427E-01	5.6454E-05	3.95
3-ethylpentane	-193.374	6.6687E-01	5.6450E-05	10.37
2,2-dimethylpetane	-209.894	6.8563E-01	5.6352E-05	-0.57
2,3-dimethylpetane	-203.028	6.6456E-01	5.6286E-05	0.01
2,4-dimethylpetane	-205.762	6.8189E-01	5.6332E-05	2.44
3,3-dimethylpetane	-205.327	6.7887E-01	5.6327E-05	1.98
2,2,3-trimethylbutane	-209.101	6.9811E-01	5.2621E-05	3.61
O-XYLENE	17.048	3.3940E-01	3.9428E-05	122.09
M-XYLENE	15.063	3.3452E-01	4.1387E-05	118.87
P-XYLENE	15.763	3.3952E-01	4.2301E-04	121.13

APPENDIX B

GROUPS IN KINETIC EQUATIONS

Table B-1 Groups in kinetic equation for reactions on solid catalyst (Yang and Hougen, (1950))

Driving-Force Groups

Reaction	$A \rightleftharpoons R$	$A \rightleftharpoons R + S$	$A + B \rightleftharpoons R$	$A + B \rightleftharpoons R + S$
Adsorption of A controlling	$p_A - \frac{p_R}{K}$	$p_A - \frac{p_R p_S}{K}$	$p_A - \frac{p_R}{K p_B}$	$p_A - \frac{p_R p_S}{K p_B}$
Adsorption of B controlling	0	0	$p_B - \frac{p_R}{K p_A}$	$p_B - \frac{p_R p_S}{K p_A}$
Desorption of R controlling	$p_A - \frac{p_R}{K}$	$\frac{p_A}{p_S} - \frac{p_R}{K}$	$p_A p_B - \frac{p_R}{K}$	$\frac{p_A p_B}{p_S} - \frac{p_R}{K}$
Surface reaction controlling	$p_A - \frac{p_R}{K}$	$p_A - \frac{p_R p_S}{K}$	$p_A p_B - \frac{p_R}{K}$	$p_A p_B - \frac{p_R p_S}{K}$
Impact of A controlling (A not adsorbed)	0	0	$p_A p_B - \frac{p_R}{K}$	$p_A p_B - \frac{p_R p_S}{K}$
Homogeneous reaction controlling	$p_A - \frac{p_R}{K}$	$p_A - \frac{p_R p_S}{K}$	$p_A p_B - \frac{p_R}{K}$	$p_A p_B - \frac{p_R p_S}{K}$

Replacements in the General Adsorption Groups
$(1 + K_A p_A + K_B p_B + K_R p_R + K_S p_S + K_I p_I)^n$

Reaction	$A \rightleftharpoons R$	$A \rightleftharpoons R + S$	$A + B \rightleftharpoons R$	$A + B \rightleftharpoons R + S$
Where adsorption of A is rate controlling, replace $K_A p_A$ by	$\frac{K_A p_R}{K}$	$\frac{K_A p_R p_S}{K}$	$\frac{K_A p_R}{K p_B}$	$\frac{K_A p_R p_S}{K p_B}$
Where adsorption of B is rate controlling, replace $K_B p_B$ by	0	0	$\frac{K_B p_R}{K p_A}$	$\frac{K_B p_R p_S}{K p_A}$
Where desorption of R is rate controlling, replace $K_R p_R$ by	$K K_R p_A$	$K K_R \frac{p_A}{p_S}$	$K K_R p_S p_B$	$K K_R \frac{p_A p_B}{p_S}$
Where adsorption of A is rate controlling with dissociation of A, replace $K_A p_A$ by	$\sqrt{\frac{K_A p_R}{K}}$	$\sqrt{\frac{K_A p_R p_S}{K}}$	$\sqrt{\frac{K_A p_R}{K p_B}}$	$\sqrt{\frac{K_A p_R p_S}{K p_B}}$
Where equilibrium adsorption of A takes place with dissociation of A, replace $K_A p_A$ by and similarly for other components adsorbed with dissociation	$\sqrt{K_A p_A}$	$\sqrt{K_A p_A}$	$\sqrt{K_A p_A}$	$\sqrt{K_A p_A}$

Table B-1 (continued)

Where A is not adsorbed, replace $K_A p_A$ by	0	0	0	0
and similarly for other components that are not adsorbed				

Kinetic Groups

Adsorption of A controlling	k_A
Adsorption of B controlling	k_B
Desorption of R controlling	$k_R K$
Adsorption of A controlling with dissociation	k_A
Impact of A controlling	$k_A K_B$
Homogeneous reaction controlling	k

Surface Reaction Controlling

	$A \rightleftharpoons R$	$A \rightleftharpoons R + S$	$A + B \rightleftharpoons R$	$A + B \rightleftharpoons R + S$
Without dissociation	$k_{sr} K_A$	$k_{sr} K_A$	$k_{sr} K_A K_B$	$k_{sr} K_A K_B$
With dissociation of A	$k_{sr} K_A$	$k_{sr} K_A$	$k_{sr} K_A K_B$	$k_{sr} K_A K_B$
B not adsorbed	$k_{sr} K_A$	$k_{sr} K_A$	$k_{sr} K_A$	$k_{sr} K_A$
B not adsorbed, A dissociated	$k_{sr} K_A$	$k_{sr} K_A$	$k_{sr} K_A$	$k_{sr} K_A$

Exponents of Adsorption Groups

Adsorption of A controlling without dissociation	$n = 1$
Desorption of R controlling	$n = 1$
Adsorption of A controlling with dissociation	$n = 2$
Impact of A without dissociation $A + B \rightleftharpoons R$	$n = 1$
Impact of A without dissociation $A + B \rightleftharpoons R + S$	$n = 2$
Homogeneous reaction	$n = 0$

Surface Reaction Controlling

	$A \rightleftharpoons R$	$A \rightleftharpoons R + S$	$A + B \rightleftharpoons R$	$A + B \rightleftharpoons R + S$
No dissociation of A	1	2	2	2
Dissociation of A	2	2	3	3
Dissociation of A (B not adsorbed)	2	2	2	2
No dissociation of A (B not adsorbed)	1	2	1	2

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

APPENDIX C

EQUILIBRIUM CONSTANT

The equilibrium constant can be calculated from

$$\Delta G^{\circ}_{T,P} = -RT \ln K \qquad \text{(C-1)}$$

For any given reaction

$$\Delta G^{\circ} = \Sigma n_p \Delta G^{\circ}_f - \Sigma n_r \Delta G^{\circ}_f \qquad \text{(C-2)}$$

product reactant

where

ΔG°_f is the standard free energy of formation of compound i (Appendix A-3).

n_p is stoichiometric coefficients of moles for product.

n_r is stoichiometric coefficients of moles for reactant.

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

APPENDIX D

VISCOSITY OF GAS MIXTURE

The viscosity of gas mixture in reforming system is predicted by method of Wilke (Reid et al., (1987)).

$$\eta_m = \frac{\Sigma\ y_i\ \eta_i}{\Sigma y_i\ \phi_{ij}} \qquad \text{(D-1)}$$

where $$\phi_{ij} = \frac{[1 + (\eta_i/\eta_j)^{1/2}\ (M_j/M_i)^{1/4})\ ]^2}{[8\ (1 + M_i/M_j)\ ]^{1/2}} \qquad \text{(D-2)}$$

ϕ_{ji} is found by interchanging subscripts or by

$$\phi_{ji} = \frac{\eta_j\ M_i\ \phi_{ij}}{\eta_i\ M_j} \qquad \text{(D-3)}$$

where η_m is the viscosity of the mixture.

η_i is pure component viscosity of i in the mixture.

y_i is the mole fraction of i in the mixture.

M_i is the molecular weight of i in the mixture.

The pure component viscosity, η_i is predicted by method of Chung et al. (Reid et al., (1987)).

$$n = \frac{40.785\ Fc\ (M/T)^{1/2}}{V_c^{2/3}\ \Omega_V} \qquad \text{(D-4)}$$

where n is the viscosity, μP.

M is the molecular weight.

T is temperature, K.

V_c is critical volume, cm^3/mole.

Ω_V is viscosity collision.

$$\Omega_V = A\ (T^*)^{-B} + C\ [\exp(-DT^*)] + E\ [\exp(-FT^*)] \qquad \text{(D-5)}$$

where $T^* = 1.2593$, $A = 1.16145$, $B = 0.14874$, $C = 0.52487$,

$D = 0.77320$, $E = 2.16178$, $F = 2.43787$.

$$Fc = 1 - 0.2756 \cdot \omega + 0.05905\ \mu^4 + k \qquad \text{(D-6)}$$

ω is the acentric factor and k is a special correction for highly polar substances is shown in APPENDIX A.

$$\mu_r = \frac{1.313\ \mu}{(V_C\ T_C)^{1/2}} \qquad \text{(D-7)}$$

μ is dipole moment, debyes.

μ_r is a dimensionless dipole moment.

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

APPENDIX E

PROGRAMMING

```
/*********************************************************************
*   Program to simulate reforming processes for C6 to C7 hydrocarbons *
*   Source file name is SIM.C                                         *
*   File data bank for specific heat, Cp is  cpmix.dat                *
*   File data bank for viscosity is vismix.dat                        *
*********************************************************************/

/* include standard file */
#include <stdio.h>
#include <math.h>
#include <conio.h>

/* define global variable */
#define  MAX  100
#define  NAME 20
#define  COMPONENT  30
#define  REACTION   30

struct data_bank {        /* structure for physical properties */
        char  word[COMPONENT];   /* name of component */
        float mw;         /* Molecular weight, MW. */
        float tc;         /* Critical temperature, Tc K  */
        float vc;         /* Critical volume cm^3/mole, Vc */
        float acentric;   /* Acentric factor w  */
        float dipolem;    /* Dipole moment ,unit debyes */
        };

struct cpm {          /* structure for heat capacity */
        char word[COMPONENT];    /* name of component */
        double a,b,c,d;   /* cp= a +bT +cT^2 +dt^3  */
        };


/* function  */
double  viscosity_pure(); /* function for viscosity of pure component */
double  vismixture();   /* function for viscosity of mixture */
double  cpmixture();    /* function for heat capacity of mixture */
void    heat();         /* function for heat of reactions */
void    equilibrium();  /* function for equilibrium constant */
void    ratec6c7();     /* function for rates of reactions   */
double  getvalue();     /* function for input value from keybroad */
void    checkfile();    /* function for open file  */
double  atoi();         /* function for change string to numeric */
void    textbox ();     /* function for draw box */
```

```
/* array variable */
double  heatreaction[REACTION];  /* for heat of reaction */
double  kequilibrium[REACTION];  /* for equilibrium constant */
double  rate[REACTION];          /* for rates of reactions */
double  partialp[COMPONENT];     /* for partial pressure of C6 to C7 */


FILE    *printer;


main()  /* main program */
{
        FILE    *cpdatabank;
        FILE    *physical_property;
        struct  cpm     cpmix[MAX];
        struct  data_bank   physical[MAX];
        double  a, b, c, d, e;
        double  temperature;
        double  temp1;
        double  pressure;
        double  vismix_ture;
        double  feedhc=0;


/*************************************************************
*               Array of mole feed is                        *
*               molefeed[0]  = Hydrogen                      *
*               molefeed[1]  = methane                       *
*               molefeed[2]  = ethane                        *
*               molefeed[3]  = propane                       *
*               molefeed[4]  = n-butane                      *
*               molefeed[5]  = n-pentane                     *
*               molefeed[6]  = nHexane                       *
*               molefeed[7]  = 2MP                           *
*               molefeed[8]  = 3MP                           *
*               molefeed[9]  = 22DMB                         *
*               molefeed[10] = 23DMB                         *
*               molefeed[11] = MCP                           *
*               molefeed[12] = BZ                            *
*               molefeed[13] = Toluene                       *
*               molefeed[14] = MCH                           *
*               molefeed[15] = ECP                           *
*               molefeed[16] = n-Heptane                     *
*               molefeed[17] = SBP7                          *
*               molefeed[18] = MBP7                          *
*************************************************************/


        double  molefeed[COMPONENT]={
                        0,   /* hydrogen  */
                        0,0,0,0,0,  /* C1-C5  */
                        0,   /* nhexane */
                        0,0,0,0,0,0,    /* 2MP,3MP,22DMB,23DMB,MCP,BZ */
                        0,0,0,0,0,0     /*Tol,MCH,ECP,n-Heptane,SBP7,MBP7*/
                        };
```

```
        double  h2_c6=0,h2_c7=0;
        double  c1_c6=0,c1_c7=0;
        double  c2_c6=0,c2_c7=0;
        double  c3_c6=0,c3_c7=0;
        double  c4_c6=0,c4_c7=0;
        double  c5_c6=0,c5_c7=0;
        double  c6_c6=0,c6_c7=0;
        double  mole_feed[COMPONENT];
        double  moleinitial[COMPONENT];
        double  slope[REACTION];
        double  totalmole=0;
        double  h2_hydrocarbon;
        double  kequilibrium[REACTION];
        double  k1[20];  /* slope k1 for Range-Kutta method */
        double  k2[20];  /* slope k2 for Range-Kutta method */
        double  k3[20];  /* slope k3 for Range-Kutta method */
        double  k4[20];  /* slope k4 for Range-Kutta method */
        double  cpmix_ture;
        double  slopetemp=0;
        double  slope_temp=0;
        double  viscosity;
        char    namecp[]="cpmix.dat";  /* File name of databank for cp */
        char    namephysical[]="vismix.dat";  /* File name of physical properties */

        int     i=0,j;
        int     iii=0,jjj=0;
        int     row=1; /* row of the monitor */
        int     col=1;
        int     count=0;
        float   initial, final, h, printin;
        int     nnp, nnc;
        char    nameoutput[40];
/*************************************************************
* read data for viscosity from file                          *
*************************************************************/

        physical_property = fopen(namephysical,"r");
        while (fgets(physical[count].word,20,physical_property) !=NULL) {
        fscanf(physical_property,"%lE %lE %lE %lE %lE\n",&a,&b,&c,&d,&e);
        physical[count].mw=a;
        physical[count].tc=b;
        physical[count].vc=c;
        physical[count].acentric=d;
        physical[count].dipolem=e;
        count=count+1;
        }
/*************************************************************
* read data for specific heat from file                      *
*************************************************************/
        count=0;
        cpdatabank = fopen(namecp,"r");
        while (fgets(cpmix[count].word,20,cpdatabank) !=NULL) {
        fscanf(cpdatabank,"%lE %lE %lE %lE\n",&a,&b,&c,&d);
        cpmix[count].a=a;
        cpmix[count].b=b;
```

```
        cpmix[count].c=c;
        cpmix[count].d=d;
        count=count+1;
        }


/***********************************************
*  Input Data for Simulation                   *
*  variable:          meaning                  *
*  temperature        degree C                 *
*  pressure           bar                      *
*  h2_hydrocarbon     H2 to Hydrocarbon ratio  *
*  initial            W/Hydrocarbon=0          *
*  final              W/Hydrocarbon at W/F     *
*  printin            Print interval           *
***********************************************/
        clrscr();
        textbox (1,1,79,24,1);
        row=1;
        gotoxy(3,row=row+1);
        cprintf("INPUT FEED");

/***********************************************
*   Input value from keybroad                  *
***********************************************/
        for( i=1; i<count; ++i) {
        gotoxy(3,row);
        cprintf("\n%s :", cpmix[i].word);
        molefeed[i]=getvalue(26,row+1);
        moleinitial[i]=molefeed[i];
        row=row+1;
        }

/***********************************************
*   Input file name to save data               *
***********************************************/
        clrscr();
        row=1;
        checkfile(nameoutput,col,row);

/* recieve temperature, pressure etc. */
        row=9;
        gotoxy(col,row);
        cprintf("Feed Temperature (C)       : ");
        temperature=(double)getvalue(col+30,row);

        gotoxy(col,row+1);
        cprintf("Pressure, bar   is         : ");
        pressure=(double)getvalue(col+30,row+1);

        gotoxy(col,row+2);
        cprintf("H2 to Hydrocarbon ratio    : ");
        h2_hydrocarbon=(double)getvalue(col+30,row+2);
```

```
        gotoxy(col,row+3);
        cprintf("Input initial condition <0> :");
        initial=(float)getvalue(col+30,row+3);

        gotoxy(col,row+4);
        cprintf("Input Final condition >0    :");
        final=(float)getvalue(col+30,row+4);

        gotoxy(col,row+5);
        cprintf("Step size                   : ");
        h=(float)getvalue(col+30,row+5);

        gotoxy(col,row+6);
        cprintf("Print interval              : ");
        printin=(float)getvalue(col+30,row+6);
        nnp=((final-initial)/printin)+1;
        nnc=(printin/h)+1;

        temperature=temperature+273.16;

/*  Mole of feed into the reactor No.1  */
        for( i=0; i<count; ++i) {
         feedhc =feedhc+ molefeed[i];
         }

/* use hydrogen/hydrocarbon ratio to calculate h2 in put */
        molefeed[0]=h2_hydrocarbon*feedhc;
        moleinitial[0]=molefeed[0];

/***********************************************
*       Working program                        *
***********************************************/
              gotoxy(1,18);
              highvideo();
              textbackground(GREEN);
              textcolor(BLINK);
              cprintf("Working");
              gotoxy(10,18);
              normvideo();
              textbackground(CYAN);
              textcolor(BLUE);


              cprintf("Finish at W/Hydrocarbon feed = %d ",nnp);

              gotoxy(1,21);
              cprintf("Internal Loop      = ");

/* First, print text file to output */
          cpmix_ture=cpmixture(temperature-273.16,cpmix,count, molefeed);
          vismix_ture=vismixture(molefeed,physical,temperature-273.16,count);
          fprintf(printer,"%f  ",0);
          for (i=0;i<count; ++i) {
                fprintf(printer,"%lf ",molefeed[i]);
                }
```

```
        fprintf(printer," %lf ", temperature-273.16);
        fprintf(printer," %lf ", cpmix_ture);
        fprintf(printer," %lf ", vismix_ture);
        fprintf(printer,"\n");
/*************************************************************
*              Loop for Fourth-order Range-Kutta             *
*************************************************************/
        for (iii=1; iii<=nnp; ++iii) {        /* print interval */
            gotoxy(1,20);
            cprintf("W/Hydrocarbon feed = ");
            gotoxy(23,20);
            cprintf("%d",iii);
        for (jjj=1;jjj<nnc; ++jjj) {    /* internal loop  */
            gotoxy(23,21);
            cprintf("%d",jjj);
            /* search for partial pressure */
            totalmole=0;
            for( i=0; i<count; ++i) {
                    totalmole =totalmole+ molefeed[i]; }
            /* partial pressure */
            for(i=0; i<count; ++i) {
                    partialp[i]=molefeed[i]*pressure/totalmole;
            }
/*********************************************************
*                      Step 1  search for k1             *
*********************************************************/
/* Call ratec6c7 function and
   Call heat of reaction function at one temperature */

            ratec6c7(partialp, rate, temperature-273.16);
            heat(temperature-273.16,heatreaction);

/********** Meaning of array k[1] of Range-Kutta ******
*             k1[0]= H2                                 *
*             k1[1]= C1                                 *
*             k1[2]= C2                                 *
*             k1[3]= C3                                 *
*             k1[4]= C4                                 *
*             k1[5]= C5                                 *
*             k1[6]= nhexane                            *
*             k1[7]= 2MP                                *
*             k1[8]= 3MP                                *
*             k1[9]= 22DMB                              *
*             k1[10]= 23DMB                             *
*             k1[11]= MCP                               *
*             k1[12]= Bz                                *
*             k1[13]= Toluene                           *
*             k1[14] = MCH                              *
*             k1[15] = ECP                              *
*             k1[16] = n-Heptane                        *
*             k1[17] = SBP7                             *
*             k1[18] = MBP7                             *
*             k1[19] = Temperature, K                   *
*********************************************************/
```

```
/* c6   */
                h2_c6= (rate[5]+3*rate[6]-rate[7]-rate[8]-rate[9])*feedhc;
                c1_c6= (rate[7]+rate[8]+rate[9])*feedhc*0.07*2;
                c2_c6= (rate[7]+rate[8]+rate[9])*feedhc*0.31*2;
                c3_c6= (rate[7]+rate[8]+rate[9])*feedhc*0.38*2;
                c4_c6= (rate[7]+rate[8]+rate[9])*feedhc*0.21*2;
                c5_c6= (rate[7]+rate[8]+rate[9])*feedhc*0.03*2;
                c6_c6= (-rate[0]-rate[1]-rate[5])*feedhc;
/*  c7  */
                h2_c7= (rate[12]+3*rate[14]-rate[15]-rate[16])*feedhc;
                c1_c7= (rate[15]+rate[16])*feedhc*0*2;
                c2_c7= (rate[15]+rate[16])*feedhc*0.187*2;
                c3_c7= (rate[15]+rate[16])*feedhc*0.375*2;
                c4_c7= (rate[15]+rate[16])*feedhc*0.313*2;
                c5_c7= (rate[15]+rate[16])*feedhc*0.125*2;
                c6_c7= (rate[15]+rate[16])*feedhc*0*2;

/*  for mixed c6 to c7 */
                k1[0]= h2_c6 + h2_c7;
                k1[1]= c1_c6 + c1_c7;
                k1[2]= c2_c6 + c2_c7;
                k1[3]= c3_c6 + c3_c7;
                k1[4]= c4_c6 + c4_c7;
                k1[5]= c5_c6 + c5_c7;
                k1[6]= c6_c6 + c6_c7;
                k1[7]= (rate[0]-rate[2]-rate[3]-rate[7])*feedhc;
                k1[8]= (rate[1]+rate[2]-rate[8])*feedhc;
                k1[9]= (rate[4]-rate[9])*feedhc;
                k1[10]= (rate[3]-rate[4])*feedhc;
                k1[11]= (rate[5]-rate[6])*feedhc;
                k1[12]= rate[6]*feedhc;
                k1[13]= rate[14]*feedhc;
                k1[14]= (rate[13]-rate[14])*feedhc;
                k1[15]= (rate[12]-rate[13])*feedhc;
                k1[16]= (-rate[10]-rate[12])*feedhc;
                k1[17]= (rate[10]-rate[11]-rate[15])*feedhc;
                k1[18]= (rate[11]-rate[16])*feedhc;

                   /* Call cp miture function  */
                cpmix_ture=cpmixture(temperature-273.16,cpmix,count, molefeed);
                slopetemp=0;
                   for (i=0; i<17;++i) {  /*  reaction is 17 */
                slopetemp= slopetemp-heatreaction[i]*rate[i]; }
                    k1[19]=slopetemp/cpmix_ture;

                /* search for molefeed at slope k1 */
               for (i=0; i<count;++i) {
                mole_feed[i]= molefeed[i] + h*k1[i]/2; }
                    templ=temperature+h*k1[19]/2;

/*****************************************************
*                      Step 2  search for k2          *
*****************************************************/
                totalmole=0;
```

```
            for( i=0; i<count; ++i) {
            totalmole =totalmole+ mole_feed[i]; }

            for(i=0; i<count; ++i) {
            partialp[i]=molefeed[i]*pressure/totalmole; }

/* Call ratec6c7 function and
   Call heat of reaction function at one temperature */

            ratec6c7(partialp, rate, temp1-273.16);
            heat(temperature-273.16,heatreaction);

/* c6   */
            h2_c6= (rate[5]+3*rate[6]-rate[7]-rate[8]-rate[9])*feedhc;
            c1_c6= (rate[7]+rate[8]+rate[9])*feedhc*0.07*2;
            c2_c6= (rate[7]+rate[8]+rate[9])*feedhc*0.31*2;
            c3_c6= (rate[7]+rate[8]+rate[9])*feedhc*0.38*2;
            c4_c6= (rate[7]+rate[8]+rate[9])*feedhc*0.21*2;
            c5_c6= (rate[7]+rate[8]+rate[9])*feedhc*0.03*2;
            c6_c6= (-rate[0]-rate[1]-rate[5])*feedhc;
/*  c7  */
            h2_c7= (rate[12]+3*rate[14]-rate[15]-rate[16])*feedhc;
            c1_c7= (rate[15]+rate[16])*feedhc*0*2;
            c2_c7= (rate[15]+rate[16])*feedhc*0.187*2;
            c3_c7= (rate[15]+rate[16])*feedhc*0.375*2;
            c4_c7= (rate[15]+rate[16])*feedhc*0.313*2;
            c5_c7= (rate[15]+rate[16])*feedhc*0.125*2;
            c6_c7= (rate[15]+rate[16])*feedhc*0*2;

/*  for mixed c6 to c7 */
            k2[0]= h2_c6 + h2_c7;
            k2[1]= c1_c6 + c1_c7;
            k2[2]= c2_c6 + c2_c7;
            k2[3]= c3_c6 + c3_c7;
            k2[4]= c4_c6 + c4_c7;
            k2[5]= c5_c6 + c5_c7;
            k2[6]= c6_c6 + c6_c7;
            k2[7]= (rate[0]-rate[2]-rate[3]-rate[7])*feedhc;
            k2[8]= (rate[1]+rate[2]-rate[8])*feedhc;
            k2[9]= (rate[4]-rate[9])*feedhc;
            k2[10]= (rate[3]-rate[4])*feedhc;
            k2[11]= (rate[5]-rate[6])*feedhc;
            k2[12]= rate[6]*feedhc;
            k2[13]= rate[14]*feedhc;
            k2[14]= (rate[13]-rate[14])*feedhc;
            k2[15]= (rate[12]-rate[13])*feedhc;
            k2[16]= (-rate[10]-rate[12])*feedhc;
            k2[17]= (rate[10]-rate[11]-rate[15])*feedhc;
            k2[18]= (rate[11]-rate[16])*feedhc;

            /* Call cp miture function  */
            cpmix_ture=cpmixture(temp1-273.16,cpmix,count, mole_feed);
            slopetemp=0;
            for (i=0; i<17; ++i) {  /*  reaction is 17  */
            slopetemp= slopetemp-heatreaction[i]*rate[i]; }
                k2[19]=slopetemp/cpmix_ture;
```

```
            /* search for molefeed at slope k2 */
            for (i=0; i<count; ++i) {
            mole_feed[i]= molefeed[i] + h*k2[i]/2; }
            temp1=temperature+h*k2[19]/2;



/*******************************************************
*                      Step 3  search for k3          *
*******************************************************/

            totalmole=0;
            for( i=0; i<count; ++i) {
                        totalmole =totalmole+ mole_feed[i]; }

            for(i=0; i<count; ++i) {
                        partialp[i]=mole_feed[i]*pressure/totalmole; }

/* Call ratec6c7 function and
   Call heat of reaction function at one temperature */

            ratec6c7(partialp, rate, temp1-273.16);
            heat(temperature-273.16,heatreaction);

/* c6   */
            h2_c6= (rate[5]+3*rate[6]-rate[7]-rate[8]-rate[9])*feedhc;
            c1_c6= (rate[7]+rate[8]+rate[9])*feedhc*0.07*2;
            c2_c6= (rate[7]+rate[8]+rate[9])*feedhc*0.31*2;
            c3_c6= (rate[7]+rate[8]+rate[9])*feedhc*0.38*2;
            c4_c6= (rate[7]+rate[8]+rate[9])*feedhc*0.21*2;
            c5_c6= (rate[7]+rate[8]+rate[9])*feedhc*0.03*2;
            c6_c6= (-rate[0]-rate[1]-rate[5])*feedhc;
/*  c7  */
            h2_c7= (rate[12]+3*rate[14]-rate[15]-rate[16])*feedhc;
            c1_c7= (rate[15]+rate[16])*feedhc*0*2;
            c2_c7= (rate[15]+rate[16])*feedhc*0.187*2;
            c3_c7= (rate[15]+rate[16])*feedhc*0.375*2;
            c4_c7= (rate[15]+rate[16])*feedhc*0.313*2;
            c5_c7= (rate[15]+rate[16])*feedhc*0.125*2;
            c6_c7= (rate[15]+rate[16])*feedhc*0*2;

/*  for mixed c6 to c7 */
            k3[0]= h2_c6 + h2_c7;
            k3[1]= c1_c6 + c1_c7;
            k3[2]= c2_c6 + c2_c7;
            k3[3]= c3_c6 + c3_c7;
            k3[4]= c4_c6 + c4_c7;
            k3[5]= c5_c6 + c5_c7;
            k3[6]= c6_c6 + c6_c7;
            k3[7]= (rate[0]-rate[2]-rate[3]-rate[7])*feedhc;
            k3[8]= (rate[1]+rate[2]-rate[8])*feedhc;
            k3[9]= (rate[4]-rate[9])*feedhc;
            k3[10]= (rate[3]-rate[4])*feedhc;
            k3[11]= (rate[5]-rate[6])*feedhc;
            k3[12]= rate[6]*feedhc;
            k3[13]= rate[14]*feedhc;
```

```
k3[14]= (rate[13]-rate[14])*feedhc;
k3[15]= (rate[12]-rate[13])*feedhc;
k3[16]= (-rate[10]-rate[12])*feedhc;
k3[17]= (rate[10]-rate[11]-rate[15])*feedhc;
k3[18]= (rate[11]-rate[16])*feedhc;


    /* Call cp miture function  */
cpmix_ture=cpmixture(temp1-273.16,cpmix,count, mole_feed);
slopetemp=0;
for (i=0; i<17; ++i) {  /*  reaction is 17  */
slopetemp= slopetemp-heatreaction[i]*rate[i]; }
k3[19] =slopetemp/cpmix_ture;


/* search for molefeed at slope k3 */
for (i=0; i<count; ++i) {
  mole_feed[i]= molefeed[i] + h*k3[i]; }
  temp1=temperature+h*k3[13];


/*******************************************************
*                    Step 4  search for k4            *
*******************************************************/

totalmole=0;
for( i=0; i<count; ++i) {
totalmole =totalmole+ mole_feed[i]; }

for(i=0; i<count; ++i) {
  partialp[i]=mole_feed[i]*pressure/totalmole; }


/* Call ratec6c7 function and
   Call heat of reaction function at one temperature */

      ratec6c7(partialp, rate, temp1-273.16);
      heat(temperature-273.16,heatreaction);

/* c6   */
      h2_c6= (rate[5]+3*rate[6]-rate[7]-rate[8]-rate[9])*feedhc;
      c1_c6= (rate[7]+rate[8]+rate[9])*feedhc*0.07*2;
      c2_c6= (rate[7]+rate[8]+rate[9])*feedhc*0.31*2;
      c3_c6= (rate[7]+rate[8]+rate[9])*feedhc*0.38*2;
      c4_c6= (rate[7]+rate[8]+rate[9])*feedhc*0.21*2;
      c5_c6= (rate[7]+rate[8]+rate[9])*feedhc*0.03*2;
      c6_c6= (-rate[0]-rate[1]-rate[5])*feedhc;
/*  c7  */
      h2_c7= (rate[12]+3*rate[14]-rate[15]-rate[16])*feedhc;
      c1_c7= (rate[15]+rate[16])*feedhc*0*2;
      c2_c7= (rate[15]+rate[16])*feedhc*0.187*2;
      c3_c7= (rate[15]+rate[16])*feedhc*0.375*2;
      c4_c7= (rate[15]+rate[16])*feedhc*0.313*2;
      c5_c7= (rate[15]+rate[16])*feedhc*0.125*2;
      c6_c7= (rate[15]+rate[16])*feedhc*0*2;
```

```
/*  for mixed c6 to c7 */
                k4[0]= h2_c6 + h2_c7;
                k4[1]= c1_c6 + c1_c7;
                k4[2]= c2_c6 + c2_c7;
                k4[3]= c3_c6 + c3_c7;
                k4[4]= c4_c6 + c4_c7;
                k4[5]= c5_c6 + c5_c7;
                k4[6]= c6_c6 + c6_c7;
                k4[7]= (rate[0]-rate[2]-rate[3]-rate[7])*feedhc;
                k4[8]= (rate[1]+rate[2]-rate[8])*feedhc;
                k4[9]= (rate[4]-rate[9])*feedhc;
                k4[10]= (rate[3]-rate[4])*feedhc;
                k4[11]= (rate[5]-rate[6])*feedhc;
                k4[12]= rate[6]*feedhc;
                k4[13]= rate[14]*feedhc;
                k4[14]= (rate[13]-rate[14])*feedhc;
                k4[15]= (rate[12]-rate[13])*feedhc;
                k4[16]= (-rate[10]-rate[12])*feedhc;
                k4[17]= (rate[10]-rate[11]-rate[15])*feedhc;
                k4[18]= (rate[11]-rate[16])*feedhc;

                /* Call cp miture function  */
                cpmix_ture=cpmixture(temp1-273.16,cpmix,count, mole_feed);
                slopetemp=0;
                for (i=0; i<17; ++i) {  /*  reaction is 17  */
                slopetemp= slopetemp-heatreaction[i]*rate[i]; }
                k4[19]=slopetemp/cpmix_ture;

                /* search for slope  */
                for (i=0; i<count; ++i)  {
                    slope[i]= (k1[i]+2*k2[i]+2*k3[i]+k4[i])/6;
                    }


                /* search for slope temperature */
                slope_temp=(k1[19]+2*k2[19]+2*k3[19]+k4[19])/6;

                for (i=0; i<count; ++i) {
                    molefeed[i]= molefeed[i] + h*slope[i]; }
                    temperature=temperature+h*slope_temp;
                if((temperature < 720.2))
                    temperature=720.2;

            }  /* end loop internal of RK */
                gotoxy(23,20);
                cprintf("         ");
                gotoxy(23,21);
                cprintf("         ");

/* Call: heat capacity function and
   Call: viscosity of mixture */
                cpmix_ture=cpmixture(temperature-273.16,cpmix,count, molefeed);
                vismix_ture=vismixture(molefeed,physical,temperature-273.16,count
);

/* Print text file to output */
```

```
        fprintf(printer,"%f  ",(iii*printin));
        for (i=0;i<count; ++i) {
          fprintf(printer,"%lf ",molefeed[i]);
        }
        fprintf(printer," %lf ", temperature-273.16);
        fprintf(printer," %lf ", cpmix_ture);
        fprintf(printer," %lf ", vismix_ture);
        fprintf(printer,"\n");

        }  /* end loop external of RK */
        fclose(printer);
        sound(200);
        delay(1000);
        nosound();
}   /*  End main program */


/*************************************************************
* Function vismixture()  is to calculate viscosity of gas     *
* mixture                                                      *
* Call: vismixture(molefeed, temperature, number of component)*
*************************************************************/

double vismixture(c, physical_vis, temp_mix, nc)
double temp_mix;
int    nc;
double c[];
struct data_bank physical_vis[];
{

        double  nviscos[40];  /* viscosity of pure component */
        int     i,j;
        double  total_mole =0;  /* total mole */
        double  y[40];   /* mole fraction */
        double  phe[40][40];
        double  phe1,phe2;
        double  visc_mix=0; /* viscosity of gas mixture */
        double  sumphe;  /* summation of the mole_fraction
                            multiply with phe  */

/*****************************************************
*        Search all viscosity of pure component       *
*        and save in aray of nviscos                  *
*****************************************************/

        for (i=0; i<nc; ++i) {
          nviscos[i]=viscosity_pure(physical_vis[i],temp_mix);
        }

/*************************************************************
*        Search all mole fraction and save in aray of y      *
*************************************************************/
        for (i=0; i<nc; ++i) {
          total_mole =total_mole + c[i];
          }
```

```
        for (i=0; i<nc; ++i) {
          y[i]= c[i]/total_mole;
          }

/*****************************
*         Search all phe     *
*****************************/

        for(i=0; i<nc; ++i) {           /* first loop */
          for (j=0; j<nc; ++j) {        /*second loop */
            if( i!=j) {
              if (j<i) {
                 phe[i][j] = nviscos[i]*physical_vis[j].mw*phe[j][i]
                             /nviscos[j]/physical_vis[i].mw;
                }
              else {
                phe1 =pow((double)(1+ (sqrt(nviscos[i]/nviscos[j]))*pow((double)
                             physical_vis[j].mw/physical_vis[i].mw,0.25)),2);
                phe2 = sqrt((double)8*(1+(physical_vis[i].mw/physical_vis[j].mw)));

                phe[i][j]=(double) phe1/phe2;
                }
            }   /* end if i=j */
          }     /* end second loop */
        }       /* end first  loop */

/*********************************************************
*        Search for viscosity of gas mixture             *
*********************************************************/
        for(i=0; i<nc; ++i) {
           sumphe = y[i];
           for(j=0; j<nc; ++j) {
             if (j!=i) {
                sumphe=sumphe +y[j]*phe[i][j];
                }
           }
           visc_mix=visc_mix + y[i]*nviscos[i]/sumphe;
        }
        return visc_mix;  /* return viscosity of gas mixture */
}


/*********************************************************************
* Function viscosity_pure()to calculate viscosity of pure component*
* viscosity_pure (physical properties, temperature Kelvin)          *
*                                                                   *
*       parameter use are                                           *
*       1. Tc           *Critical temperature, Tc K                 *
*       2. Vc       * Critical volume cm^3/mole, Vc                 *
*       3. w            * Acentric factor w                         *
*       4. Dipolem  * Dipole moment ,unit debyes                    *
*       5. Mw;          * Molecular weight, MW.                     *
*                                                                   *
*********************************************************************/
```

```
double viscosity_pure(physic_constant,temp)
double temp;                        /* temperature variable */
struct data_bank physic_constant;
{
        int i;
        double fc;          /* fc=1-0.2756w + 0.059035ur^4  */
        double tstar;       /* dimension less temperature */
                            /* T* =1.2593 Tr  */
        double omegav;      /* viscosity collision  */
        double ur;          /* ur=131.3*dipolemoment/(VcTc)^0.5 */
        double npure;       /* viscosity pure component */
        temp=temp+273.16;     /* change temperature C to K */

            ur=  131.3*physic_constant.dipolem
                /(pow((double)physic_constant.tc
                *physic_constant.vc,0.5));

            fc= 1-0.2756*physic_constant.acentric
               + 0.059035*pow(ur,4);

            tstar = 1.2593*(temp)/physic_constant.tc;

            omegav= 1.16145*pow(tstar,-0.14876)
                   +0.52487*exp((double)-0.77320*tstar)
                   +2.16178*exp((double)-2.43787*tstar);

            npure= 40.785*fc*pow((double)physic_constant.mw
                      *temp,0.5)/pow((double)physic_constant.vc,
                      (double)2/3)/omegav;
        return npure;  /* return viscosity of pure component */
}

/*****************************************************************
*  CPMIXTURE FUNCTION                                            *
*  K = temperature (C)                                           *
*  cpmixc = A,B,C,D                                              *
*  nc = total number of component (int)                          *
*  conc= concentration of mixture                                *
*****************************************************************/

double cpmixture(K,cpmixc,nc,conc)
double K;
struct cpm cpmixc[];
int    nc;
double conc[];
{
        double  cptemp;
        int     i;
        double total=0; /* for total mole*/
        double cpmixture=0;
        for (i=0;i<nc;i++)
          total=total+conc[i];

        /* chang temperature from degree C to degree K  */
        K=K+273.16;
```

```
        /* loop to search cp of mixture at given T  */
        for (i=0; i<nc;i=i+1) {
        cptemp=cpmixc[i].a+(cpmixc[i].b)*K+(cpmixc[i].c)*K*K+(cpmixc[i].d)*K*K*K;
        cpmixture= cpmixture+ cptemp*conc[i]/total;
        }
        return cpmixture;  /* return value of the cp mixture */
}

/*********************************************************
*  Function equilibrium( temperature, keq)               *
*  Equilibrium constant is depend on temperature         *
*  Call by : temperature C, array of kequilibrium        *
*  function is not return value but use pointer.         *
*********************************************************/
void    equilibrium(temp,keq)
double  temp;
double  keq[];

{
        float  R=1.987;   /* Gas constant 1.987 cal/mol K */
        temp=temp+273.16;  /* change temperature C to K */

/*********************************************************
* Reaction   n-Hexane =====  2 MP                        *
*********************************************************/
keq[0]= exp(-(-7.228+(8.8600E-03)*temp -(1.99E-06)*pow(temp,2.0))*1000/R/temp/4.18)
;
/*********************************************************
* Reaction   n-Hexane =====  3 MP                        *
*********************************************************/
keq[1]=exp(-( -4.414+(8.54E-03)*temp -(4.80E-08)*pow(temp,2.0))*1000/R/temp/4.18);
/*********************************************************
* Reaction   2 MP     =====  3 MP                        *
*********************************************************/
keq[2]= exp(-(2.814-(3.2E-04)*temp -(2.038E-06)*pow(temp,2.0))*1000/R/temp/4.18);
/*********************************************************
* Reaction   2 MP     =====  2,3 DMB                     *
*********************************************************/
keq[3]= exp(-(-3.635+(1.48E-02)*temp +(1.409E-06)*pow(temp,2.0))*1000/R/temp/4.18);
/*********************************************************
* Reaction   2,3 DMB     =====  2,2 DMB                  *
*********************************************************/
keq[4]= exp(-(-7.915+(8.66E-03)*temp -(2.099E-06)*pow(temp,2.0))*1000/R/temp/4.18);
/*********************************************************
* Reaction   n-Hexane     =====  MCP +  H2               *
*********************************************************/
keq[5]= exp(-(60.01-(8.016E-02)*temp -(1.18E-06)*pow(temp,2.0))*1000/R/temp/4.18);
/*********************************************************
* Reaction   MCP     =====  Bz +  3H2                    *
*********************************************************/
keq[6]= exp(-(191.949-(3.2119E-01)*temp -(2.2601E-05)*pow(temp,2.0))*1000/R/temp/4.
18);
```

```
/*********************************************************
* Reaction . n-P7 =====  SBP7                             *
*********************************************************/
keq[7]= exp(-(-7.125+(7.85E-03)*temp +(3.10E-08)*pow(temp,2.0))*1000./R/temp/4.18);
/*********************************************************
* Reaction   SBP7  =====  MBP7                             *
*********************************************************/
keq[8]= exp(-(-4.383+(6.19E-03)*temp -(1.89E-07)*pow(temp,2.0))*1000./R/temp/4.18);
/*********************************************************
* Reaction    n P7   =====  5N7  + H2                      *
*********************************************************/
keq[9]= exp(-(60.297-(7.916E-02)*temp -(1.672E-06)*pow(temp,2.0))*1000/R/temp/4.18)
;
/*********************************************************
* Reaction    5 N7   =====  MCH                           *
*********************************************************/
keq[10]= exp(-(-28.815+(4.119E-02)*temp -(8.469E-06)*pow(temp,2.0))*1000/R/temp/4.1
8);
/*********************************************************
* Reaction    MCH    =====  TOL +  3H2                    *
*********************************************************/
keq[11]= exp(-(207.851-(3.7424E-01)*temp -(1.4387E-05)*pow(temp,2.0))*1000/R/temp/4
.18);
}


/*********************************************************
*  Function heat( temperature, heat of reaction)        *
*  Because heat of reaction is depend on temperature    *
*  Call by : temperature C, array of heat of reaction   *
*  function is not return value but use pointer.        *
*********************************************************/

void    heat(temp,heat_reaction)
double  temp;
double  heat_reaction[];
{
        int     i;
        temp=temp+273.16;
/*********************************************************
* Reaction   n-Hexane =====  2 MP                         *
*********************************************************/
heat_reaction[0]=-1.70*1000 +((-6.15*(temp-298))
                +(3.60E-2)*(pow(temp,2)-pow(298,2))/2
                -(4.54E-5)*(pow(temp,3)-pow(298,3))/3
                +(1.568E-8)*(pow(temp,4)-pow(298,4))/4)
                /4.184;

/*********************************************************
* Reaction   n-Hexane =====  3 MP                         *
*********************************************************/
heat_reaction[1]=-1.06*1000 +((2.027*(temp-298))
                +(-1.3E-2)*(pow(temp,2)-pow(298,2))/2
                +(2.49E-5)*(pow(temp,3)-pow(298,3))/3
                +(-1.457E-8)*(pow(temp,4)-pow(298,4))/4)
                /4.184;
```

```
/*********************************************************
* Reaction   2 MP     =====  3 MP                        *
*********************************************************/
heat_reaction[2]= 0.64*1000 +((8.184*(temp-298))
                  +(-4.9E-2)*(pow(temp,2)-pow(298,2))/2
                  +(7.03E-5)*(pow(temp,3)-pow(298,3))/3
                  +(-3.025E-8)*(pow(temp,4)-pow(298,4))/4)
                  /4.184;

/*********************************************************
* Reaction   2 MP     =====  2,3 DMB                     *
*********************************************************/
heat_reaction[3]=-0.83*1000 +((-4.04*(temp-298))
                  +(-3.0E-3)*(pow(temp,2)-pow(298,2))/2
                  +(1.97E-5)*(pow(temp,3)-pow(298,3))/3
                  +(-1.236E-8)*(pow(temp,4)-pow(298,4))/4)
                  /4.184;

/*********************************************************
* Reaction   2,3 DMB     =====  2,2 DMB                  *
*********************************************************/
heat_reaction[4]=-0.86*1000 +((-2.02*(temp-298))
                  +(1.430E-2)*(pow(temp,2)-pow(298,2))/2
                  +(-1.05E-5)*(pow(temp,3)-pow(298,3))/3
                  +(3.0E-10)*(pow(temp,4)-pow(298,4))/4)
                  /4.184;

/*********************************************************
* Reaction   n-Hexane     =====  MCP +  H2               *
*********************************************************/
heat_reaction[5]= 14.46*1000 +(((-1.8557E+1)*(temp-298))
                  +(6.53740E-2)*(pow(temp,2)-pow(298,2))/2
                  +(-6.611E-5)*(pow(temp,3)-pow(298,3))/3
                  +(2.3155E-8)*(pow(temp,4)-pow(298,4))/4)
                  /4.184;

/*********************************************************
* Reaction   MCP      =====  Bz +  3H2                   *
*********************************************************/
heat_reaction[6]= 45.32*1000 +(((9.761E+1)*(temp-298))
                  +(-1.3638E-1)*(pow(temp,2)-pow(298,2))/2
                  +(2.1070E-5)*(pow(temp,3)-pow(298,3))/3
                  +(1.3825E-8)*(pow(temp,4)-pow(298,4))/4)
                  /4.184;

/*********************************************************
* Reaction   2MP + H2 =====  C5 + C4 + C3 + C2 + C1      *
*********************************************************/
heat_reaction[7]= -7.0170*1000 +((-9.9647*(temp-298))
                  +(-1.0838E-1)*(pow(temp,2)-pow(298,2))/2
                  +(1.4722E-4)*(pow(temp,3)-pow(298,3))/3
                  +(-5.8017E-8)*(pow(temp,4)-pow(298,4))/4)
                  /4.184;
```

```
/*********************************************************
* Reaction   3MP + H2 =====  C5 + C4 + C3 + C2 + C1      *
*********************************************************/
heat_reaction[8]= -7.6570*1000 +(((-1.8149E+1)*(temp-298))
                  +(-5.9382E-2)*(pow(temp,2)-pow(298,2))/2
                  +(7.6918E-5)*(pow(temp,3)-pow(298,3))/3
                  +(-2.7767E-8)*(pow(temp,4)-pow(298,4))/4)
                  /4.184;
/*********************************************************
* Reaction   2,2 DMB + H2 =====  C5 + C4 + C3 + C2 + C1  *
*********************************************************/
heat_reaction[9]= -4.3270*1000 +(((-3.9047)*(temp-298))
                  +(-1.1968E-1)*(pow(temp,2)-pow(298,2))/2
                  +(1.3802E-4)*(pow(temp,3)-pow(298,3))/3
                  +(-4.5937E-8)*(pow(temp,4)-pow(298,4))/4)
                  /4.184;
/*********************************************************
* Reaction   n-P7 =====  SBP7                            *
*********************************************************/
heat_reaction[10]=-1.39*1000 +((-34.244*(temp-298))
                  +(1.88E-1)*(pow(temp,2)-pow(298,2))/2
                  +(-2.638E-4)*(pow(temp,3)-pow(298,3))/3
                  +(1.0702E-7)*(pow(temp,4)-pow(298,4))/4)
                  /4.184;
/*********************************************************
* 2. Reaction   SBP7  =====  MBP7                        *
*********************************************************/
heat_reaction[11]=-1.34*1000 +((32.344*(temp-298))
                  +(-1.805E-2)*(pow(temp,2)-pow(298,2))/2
                  +(2.555E-4)*(pow(temp,3)-pow(298,3))/3
                  +(-1.0526E-7)*(pow(temp,4)-pow(298,4))/4)
                  /4.184;
/*********************************************************
* 3. Reaction    n P7   =====  5N7                       *
*********************************************************/
heat_reaction[12]= 14.51*1000 +((-23.024*(temp-298))
                  +(8.4174E-2)*(pow(temp,2)-pow(298,2))/2
                  +(-8.831E-5)*(pow(temp,3)-pow(298,3))/3
                  +(3.1465E-8)*(pow(temp,4)-pow(298,4))/4)
                  /4.184;
/*********************************************************
* 4. Reaction    5 N7   =====  MCH                       *
*********************************************************/
heat_reaction[13]=-6.61*1000 +((-6.61*(temp-298))
                  +(3.31E-2)*(pow(temp,2)-pow(298,2))/2
                  +(-4.2E-6)*(pow(temp,3)-pow(298,3))/3
                  +(-6.74E-9)*(pow(temp,4)-pow(298,4))/4)
                  /4.184;
/*********************************************************
* 5. Reaction    MCH    =====  TOL +  3H2                *
*********************************************************/
heat_reaction[14]=48.94*1000 +((118.99*(temp-298))
```

```
                +(-2.4388E-1)*(pow(temp,2)-pow(298,2))/2
                +(1.2587E-4)*(pow(temp,3)-pow(298,3))/3
                +(-2.1615E-8)*(pow(temp,4)-pow(298,4))/4)
                /4.184;
/*********************************************************
* 6.Reaction   SBP7 + H2 ===== C6+ C5 + C4 + C3 + C2 + C1 *
*********************************************************/
heat_reaction[15]= -7.5272*1000 +((16.137*(temp-298))
                +(-2.4792E-1)*(pow(temp,2)-pow(298,2))/2
                +(3.6395E-4)*(pow(temp,3)-pow(298,3))/3
                +(-1.5243E-7)*(pow(temp,4)-pow(298,4))/4)
                /4.184;
/*********************************************************
* 7.Reaction   MBP7 + H2 =====  C6+ C5 + C4 + C3 + C2 + C1 *
*********************************************************/
heat_reaction[16]= -6.19*1000 +(((-1.6207E+1)*(temp-298))
                +(-6.7421E-1)*(pow(temp,2)-pow(298,2))/2
                +(1.0845E-4)*(pow(temp,3)-pow(298,3))/3
                +(-4.7168E-8)*(pow(temp,4)-pow(298,4))/4)
                /4.184;
}

/*********************************************************************
* Function to calculate rate of reaction of C6 to C7 hydrocarbons *
* Because heat of reaction is depend on temperature                *
* Call by :ratec6c7(partial pressure, rate, temperature)           *
*********************************************************************/
void ratec6c7(partial, rate, temp)
double partial[];
double rate[];
double temp;  /*   temperature  c  */
{
        double  phe;
        double  acid_ad, metal_ad;
        int     i;
        float   R=1.987*4.183;
                /* R= gas constant = 1.987 cal/mole */
                /* 1 cal = 4.183 j                  */

/*************************************************************
*   Reaction of C6 hydrocarbons                              *
*                                                            *
*************************************************************/
/*****************************
*       n-Hexane <=====> 2MP *
*****************************/
        float  A1=1.715E+10;    /* kmol/kg cat. hr  */
        float  E1=147.3E+3;     /* J/mole  */

/*****************************
*       n-Hexane <=====> 3MP *
*****************************/
        float  A2=1.510E+10;
        float  E2=147.3E+3;

/*****************************
*       2MP <=====> 3MP      *
*****************************/
```

```
        float  A3=8.766E+8;
        float  E3=125.2E+3;

/*****************************
*       2MP <=====> 2,3DMB   *
*****************************/
        float  A4=8.587E+9;
        float  E4=147.3E+3;

/*****************************
*       2,3DMB <====> 2,2DMB *
*****************************/
        float  A5=1.029E+9;
        float  E5=125.2E+3;

/*****************************
*   n-Hexane <====> MCP + H2 *
*****************************/
        float  A6=4.004E+17;
        float  E6=264.6E+3;

/*****************************
*       MCP <====> Bz + 3H2  *
*****************************/
        float  A7=8.496E+10;
        float  E7=147.3E+3;

/*****************************
*       2MP + H2 ----> 2C5-  *
*****************************/
        float  A8=6.759E+8;
        float  E8=147.3E+3;

/*****************************
*       3MP + H2 ---->2C5-   *
*****************************/
        float  A9=9.494E+8;
        float  E9=147.3E+3;

/*****************************
*    2,2DMB + H2 ---->2C5-   *
*****************************/
        float  A10=1.076E+9;
        float  E10=147.3E+3;

/* common adsorption term  */
        float  khex=7.601;
        float  kmcp=2.016E+2;

/*****************************************************************
*   Reaction of C7 hydrocarbons                                  *
*****************************************************************/

/*****************************
*       n-Heptane <====> SBP *
*****************************/
        float  A11=1.83E+6;
        float  E11=87.75E+3;
```

```
/*****************************
*       SBP7 <====> MBP7     *
*****************************/
        float  A12=1.83E+6;
        float  E12=87.75E+3;

/*******************************
*   n-Heptane <====> 5N7  + H2 *
*******************************/
        float  A13=2.48E+17;
        float  E13=256.4E+3;

/*****************************
*       5N7 <====> MCH       *
*****************************/
        float  A14=9.08E+17;
        float  E14=256.4E+3;

/*****************************
*       MCH <====> Tol + 3H2 *
*****************************/
        float  A15=3.45E+11;
        float  E15=121.7E+3;

/*****************************
*      SBP7 + H2 -----> 2C6-  *
*****************************/
        float  A16=1.43E+17;
        float  E16=256.4E+3;

/*****************************
*      MBP7 + H2 ----->2C6-   *
*****************************/
        float  A17=1.43E+17;
        float  E17=256.4E+3;

/* common adsorption term  */
        float  kc6_ = 107;
        float  kp7  = 21.9;
        float  kn7  = 659;
        float  ktol = 70.3;
        float  kmch = 0.34;
        float  preexp = 1.47E+10;
        float  delh   = 99.77E+3;

/* for C7 hydrocarbons */
        acid_ad  = (partial[0]+kc6_*(partial[6]+partial[5]
                   +partial[4]+partial[3]+partial[2]+partial[1])
                   +kp7*partial[10]+ kn7*(partial[8]+partial[9])
                   +ktol*partial[7]*partial[0])/partial[0];
        metal_ad = 1+kmch*partial[8]
                          +preexp*(exp(-delh/R/(temp+273)))*(partial[8]/partial[0]
/partial[0]);
/* for C6 hydrocarbons */
        phe=pow((1+(khex*(partial[6]+partial[7]+partial[8]+partial[9]
                +partial[10])/partial[0])
                +(kmcp*partial[11]/partial[0])),2);
```

```
/* Call function equilibrium()  */
        equilibrium(temp,kequilibrium);

/* Isomerization  */
        temp=temp+273.16;

/*****************************************************************
* Rate equation for C6 hydrocarbons                              *
*****************************************************************/

rate[0]= A1*(exp(-E1/R/temp))*( partial[6]-(partial[7]/kequilibrium[0]))/(partial[0
]*phe);
rate[1]= A2*(exp(-E2/R/temp))*( partial[6]-(partial[8]/kequilibrium[1]))/(partial[0
]*phe);
rate[2]= A3*(exp(-E3/R/temp))*( partial[7]-(partial[8]/kequilibrium[2]))/(partial[0
]*phe);
rate[3]= A4*(exp(-E4/R/temp))*( partial[7]-(partial[10]/kequilibrium[3]))/(partial[
0]*phe);
rate[4]= A5*(exp(-E5/R/temp))*( partial[10]-(partial[9]/kequilibrium[4]))/(partial[
0]*phe);
rate[5]= A6*(exp(-E6/R/temp))*( partial[6]-(partial[11]*partial[0]/kequilibrium[5])
)/(partial[0]*phe);
rate[6]= A7*(exp(-E7/R/temp))*( partial[11]-(partial[12]*pow(partial[0],3)/kequilib
rium[6]))/partial[0]/phe;
rate[7]= A8*(exp(-E8/R/temp))*partial[7]/phe;
rate[8]= A9*(exp(-E9/R/temp))*partial[8]/phe;
rate[9]= A10*(exp(-E10/R/temp))*partial[9]/phe;

/*****************************************************************
* Rate equation for C7 hydrocarbons                              *
*****************************************************************/
rate[10]= A11*(exp(-E11/R/temp))*( partial[16]-(partial[17]/kequilibrium[7]))/(part
ial[0]*acid_ad);
rate[11]= A12*(exp(-E12/R/temp))*( partial[17]-(partial[18]/kequilibrium[8]))/(part
ial[0]*acid_ad);
rate[12]= A13*(exp(-E13/R/temp))*( partial[16]-(partial[15]*partial[0]/kequilibrium
[9]))/(partial[0]*acid_ad);
rate[13]= A14*(exp(-E14/R/temp))*( partial[15]-(partial[14]/kequilibrium[10]))/(par
tial[0]*acid_ad);
rate[14]= A15*(exp(-E15/R/temp))*( partial[14]-(partial[13]*pow(partial[0],3)/kequi
librium[11]))/(pow(partial[0]*metal_ad,2));
rate[15]= A16*(exp(-E16/R/temp))*partial[17]/acid_ad;
rate[16]= A17*(exp(-E17/R/temp))*partial[18]/acid_ad;
}  /* end function ratec6c7() */


/*******************************************************
* Function check file to output                        *
* Call by: checkfiel(name, coordinate x, coordinate y) *
*******************************************************/
void checkfile(s,x,y)
char s[];
{
        int  c;
        int  i;
        int  check;
        int  get_ch;
```

```
clrscr();
/* Draw box */
 textbox (1,1,30,7,1);
 gotoxy(x+2,y+1);
 cprintf("Meaning of output name");
 gotoxy(x+2,y+3);
 cprintf(" con    = output to screen");
 gotoxy(x+2,y+4);
 cprintf(" prn    = output to printer");
 gotoxy(x+2,y+5);
 cprintf(" other  = output to file");
 y=y+7;
 do {
 check=1;
 gotoxy(x,y);
 cprintf("                                                    ");
 gotoxy(x,y+1);
 cprintf("                                                    ");
 gotoxy(x,y);
 cprintf("Enter output name  :");
 i=0;
  while ((c=getch())!='\r'|| check==1) {
    if(c==0x1B) {
      gotoxy(1,y+2);
      cprintf("Exit program   <Y/N> :");
      if((get_ch=getch())=='y' || get_ch=='Y') {
         gotoxy(1,y+2);
         cprintf("                                          ");
         exit(1);}
      else if (get_ch==0) {
         check=1;
         i=0;
         getch();
         gotoxy(1,y+2);
         cprintf("                                    ");
         gotoxy(x+20,y);
         cprintf("                                    ");
         gotoxy(x+20,y);
         }
      else {
         check=1;
         i=0;
         gotoxy(1,y+2);
         cprintf("                                    ");
         gotoxy(x+20,y);
         cprintf("                                    ");
         gotoxy(x+20,y);
         }
      }
    else if (c==0x8)
      {i=0;
       check=1;
       gotoxy(x+20,y);
       cprintf("                                                    ");
       gotoxy(x,y+1);
```

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

```
                cprintf("                                                    ");
                gotoxy(x+20,y);
                }
              else if (c==0)
                {getch();
                 check=1;
                 i=0;
                 gotoxy(x+20, y);
                 cprintf("                                                   ");
                 gotoxy(x, y+1);
                 cprintf("                                                   ");
                 gotoxy(x+20,y);
                 }
               else if (c=='\r')
                 {check=1;
                  i=0;
                 }
                    .
               else
                 {check=2;
                  cprintf("%c",c);
                  s[i]=c;
                  i=i+1;
                  }
              }
            s[i]='\0';
            if((searchpath(s))!=NULL)
            {
            gotoxy(x,y+1);
            cprintf("Replace old file name <Y/N> :");
            if ((get_ch=getch())=='y' || get_ch=='Y') {
                gotoxy(x,y+1);
                cprintf("                                          ");
                break;}
            else {
                 gotoxy(x,y+1);
                 cprintf("                                          ");
                 }
            }

          } /* end loop do  */

          while ((searchpath(s))!=NULL);
          printer=fopen(s,"w");
          if (printer==NULL)
           {
            clrscr();
            gotoxy(x,y);
            cprintf("Can't open file");
            exit(3);
            }
}
```

```
/* Function getvalue()  */
double getvalue(x,y)
int x,y;
{
        char    s[30];
        int     c;
        int     i=0;
        int     check;
        int     get_ch;
        double  num;
        gotoxy(x,y);
        check=1;
        while ((c=getch())!='\r'|| check==1) {
             if(c==0x1B) {
                   gotoxy(3,y+1);
                   cprintf("Exit program   <Y/N> :");
                  if((get_ch=getch())=='y' || get_ch=='Y') {
                    gotoxy(3,y+1);
                    cprintf("                                        ");
                    exit(1);}
                  else if (get_ch==0) {
                            getch();
                            gotoxy(3,y+1);
                            cprintf("                                                ");
                            gotoxy(x+i,y);
                            }
                       else {
                             gotoxy(3,y+1);
                             cprintf("                                                ");
                             gotoxy(x+i,y);
                             }
                        }
                  else if (c==0x8)
                      { i=0;
                        check=1;
                        gotoxy(x,y);
                        cprintf("                                        ");
                        gotoxy(x,y);
                      }
                  else if (c==0)
                      { getch();
                        i=0;
                        check=1;
                        gotoxy(x,y);
                        cprintf("                                        ");
                        gotoxy(x,y);
                      }
                      else if (c=='\r')
                               { check=1;
                                    i=0;
                               }

                  else
                      {
                        if (c>='0'&& c<='9'|| c=='.') {
                              check=2;
                              cprintf("%c",c);
```

```
                        s[i]=c;
                        i=i+1;
                       }
                else {   /* remove character  */
                       i=0;
                       check=1;
                       sound(200);
                       delay(100);
                       nosound();
                       gotoxy(x,y);
                       cprintf("                                        ");
                       gotoxy(x,y);
                }
             }
        } /* end while  */

        s[i]='\0';
        if (s[0]==NULL)
           return 12;
        num=atoi(s);
        return (num);
}

double atoi(s)
char s[];
{
        int i;
        int j;
        double n,n1=0;
        n=0;
        for(i=0;s[i]>='0'&& s[i]<='9'&& s[i]!='.'; i=i+1)
           n=10*n+s[i]-'0';
        if(s[i]=='.') {
        for(i=i+1,j=1;s[i]>='0'&& s[i]<='9'; i=i+1,j=j+1)
           n1=n1+(s[i]-'0')/pow(10,j);
        n=n+n1;
        }
        return n;
}

/*-------------------------------------------------------*
*  Routine to draw boxes in text mode                    *
*  The style argument determines the type of box drawn:  *
*         0 = no box                                     *
*           1 = single-scored                            *
*           2 = double-scored                            *
*-------------------------------------------------------*/
void textbox (int left, int top, int right,
              int bottom, int style)
{
       register  r,c;
       static bord [][6] = {
              {196, 179, 218, 191, 217, 192},
              {205, 186, 201, 187, 188, 200}
       };
```

```
        if(style==0) return;
        --style;
    /* draw horizontals  */
        for (c=left+1; c<right; c++) {
          gotoxy(c,top);
          cprintf("%c",bord[style][0]);
          gotoxy(c,bottom);
          cprintf("%c",bord[style][0]);
        }

    /* draw verticals */
        for (r=top+1; r<bottom; r++) {
        gotoxy(left,r);
        cprintf("%c",bord[style][1]);
        gotoxy(right,r);
        cprintf("%c",bord[style][1]);
        }
    /* set corners  */
        gotoxy(left,top);     cprintf("%c",bord[style][2]);
        gotoxy(right,top);    cprintf("%c",bord[style][3]);
        gotoxy(right,bottom); cprintf("%c",bord[style][4]);
        gotoxy(left,bottom);  cprintf("%c",bord[style][5]);
} /*-------------------------------------------------*/
```

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

Bibliography

Mr. Prathompong Panyavivanond Graduated with a bachelor degree in Engineering Chemical from the Department of Chemical Technology, Faculty of Science, Chulalongkorn University in 1986.

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย