USER MANUAL:

1. Enter the number of components in the system.
    a. Example: Binary - 2 , Tertiary - 3 , Quaternary - 4
    b. Note: This code is only for systems with 2 or more components. It will not work with only 1 species.
2. Enter the temperature of the system in KELVIN.
3. Enter the pressure of the system in BAR.
4. Enter the constants of each species. Matlab will ask the user to enter each of the constants, starting with species 1, species 2 and so on. Matlab will ask for the following for each species:
    a. Gas/Vapor composition (y)
    b. $\boldsymbol{\omega}$
    c. Critical temperature in KELVIN (T_c)
    d. Critical pressure in BAR (P_c)
    e. V_c in cm^3/mol
5. After all of these are entered, Matlab will automatically show the following:
    a. All possible second virial coefficients (B_{ij})
    b. Fugacity coefficient of each species ($\widehat{\phi_i^v}$)
    c. Fugacity of each species ($\hat{f}_i$)

NOTE: Please input NUMBERS only. The code will error if there are non-numerical symbols entered.

WRITE-UP FOR THE CODES

```
clear;
clc;
```

This part of the code prompts the user to input the required constants for the calculations. This includes the temperature and pressure of the mixture, the mole fractions of the components, and the critical properties of each individual species (T_{ci}, P_{ci}, ω_i, Z_{ci}, and V_{ci}).

```
%input
components=input('Input number of components: ');
y=zeros(1, components);
temperature=input('\nInput mixture temperature (in Kelvin): ');
pressure=input('Input mixture pressure (in bar): ');
crit_temp=zeros(1, components);
crit_pressure=zeros(1, components);
z_crit=zeros(1, components);
v_crit=zeros(1, components);ω
w=zeros(1, components);


for i=1:components
    y(1,i)=input(['\nInput gas/vapor composition y' num2str(i) ': ']);
    w(1,i)=input(['Input w' num2str(i) ': ']);
    crit_temp(1,i)=input(['Input critical temperature (in Kelvin) of 
component ' num2str(i) ': ']);
    crit_pressure(1,i)=input(['Input critical pressure (in bar) of 
component ' num2str(i) ': ']);
    z_crit(1,i)=input(['Input Zc of component ' num2str(i) ': ']);
    v_crit(1,i)=input(['Input Vc (in cm3/mol) of component ' num2str(i) ': 
']);
end
```

This part creates an array for each critical property where the input values can be stored.

The code is looped to ask for each critical property n number of times, where n is the number of components in the mixture.

SAMPLE RESULT FOR THE INPUT

```
Input number of components: 3

Input mixture temperature (in Kelvin): 373.15
Input mixture pressure (in bar): 35

Input gas/vapor composition y1: .21
Input w1: .012
Input critical temperature (in Kelvin) of component 1: 190.6
Input critical pressure (in bar) of component 1: 45.99
Input Zc of component 1: .286
Input Vc (in cm3/mol) of component 1: 98.6

Input gas/vapor composition y2: .43
Input w2: .1
Input critical temperature (in Kelvin) of component 2: 305.3
Input critical pressure (in bar) of component 2: 48.72
Input Zc of component 2: .279
Input Vc (in cm3/mol) of component 2: 145.5

Input gas/vapor composition y3: .36
Input w3: .152
Input critical temperature (in Kelvin) of component 3: 369.8
Input critical pressure (in bar) of component 3: 42.48
Input Zc of component 3: .276
Input Vc (in cm3/mol) of component 3: 200
```

This part of the code solves for the second virial coefficient B_{ij}.

$$\omega_{ij} = \frac{\omega_i + \omega_j}{2}$$

$$T_{c,ij} = \sqrt{T_{c,i}T_{c,j}}(1 - k_{ij})$$

$$Z_{c,ij} = \frac{Z_{c,i} + Z_{c,j}}{2}$$

$$V_{c,ij} = \left(\frac{V_{c,i}^{1/3} + V_{c,j}^{1/3}}{2}\right)^3$$

$$P_{c,ij} = \frac{Z_{c,ij}RT_{c,ij}}{V_{c,ij}}$$

$$T_{r,ij} = {}^{T}/_{T_{c,ij}}$$

$$B_{ij} = \left[0.083 - \frac{0.422}{T_{r,ij}^{1.6}} + \omega_{ij}\left(0.139 - \frac{0.172}{T_{r,ij}^{4.2}}\right)\right] R\frac{T_{C,ij}}{P_{C,ij}}$$

```
%solving for bij
crit_temp_final=zeros(components, components);
crit_pressure_final=zeros(components, components);
z_crit_final=zeros(components, components);
v_crit_final=zeros(components, components);
w_final=zeros(components, components);
b=zeros(components, components);
reduced_temp=zeros(components, components);
```

In order to solve for B_{ij}, there are several variables that are needed (T_{cij}, T_{rij}, P_{cij}, ω_{ij}, Z_{cij}, and V_{cij}). This part of the code creates a matrix for each variable where the values can be stored.

```
for i=1:components
    for j=1:components
        crit_temp_final(i,j)=sqrt(crit_temp(1,i)*crit_temp(1,j));
        w_final(i,j)=(w(1,i)+w(1,j))/2;
        z_crit_final(i,j)=(z_crit(1,i)+z_crit(1,j))/2;
        v_crit_final(i,j)=((v_crit(1,i)^(1/3)+v_crit(1,j)^(1/3))/2)^3;
         crit_pressure_final(i,j)=(83.14*z_crit_final(i,j)*crit_temp_final(
i,j))/v_crit_final(i,j);
        reduced_temp(i,j)=temperature/crit_temp_final(i,j);
        b(i,j)=(0.083-(0.422/(reduced_temp(i,j)^1.6))+w_final(i,j)*(0.139-
(0.172/(reduced_temp(i,j)^4.2))))*83.14*crit_temp_final(i,j)/crit_pressure_
final(i,j);
    end
end
```

Since values for each combination of i and j are required, it would be more efficient to calculate for the values of each variable if it is performed inside a nested loop that runs until n, where n is the number of components. The formulas for the calculation of these variables are indicated above. The value for R that is used is 83.14 to correspond with the units of the other variables.

```
fprintf('\nThe possible Bij values are: \n');
for i=1:components-1
    for j=i+1:components
        fprintf('B%d%d: %g\n', i, j, b(i,j));
    end
end
```

This part of the code provides the B_{ij} values that are required. The limit of the for loop is adjusted in consideration of the property B_{ij}= B_{ji} and so that the B_{ii} and B_{jj} values will not be provided.

SAMPLE RESULT FOR THE CALCULATION OF B_{IJ}

```
The possible Bij values are:
B12: -51.515
B13: -78.4526
B23: -166.84
```

This part of the code solves for the fugacity coefficients $\hat{\phi}_k$ and the fugacity of each species in the solution $\hat{f}_k$.

$$\delta_{ik} \equiv 2B_{ik} - B_{ii} - B_{kk}$$
$$\delta_{ii} = \delta_{jj} = \delta_{kk} = 0$$
$$\delta_{ij} \equiv 2B_{ij} - B_{ii} - B_{jj}$$
$$\ln \hat{\phi}_k = \frac{P}{RT}\left[B_{kk} + \frac{1}{2}\sum_i^N \sum_j^N y_i y_j (2\delta_{ik} - \delta_{ij})\right]$$
$$\hat{f}_1 = \hat{\phi}_1 y_1 P, \qquad \hat{f}_2 = \hat{\phi}_2 y_2 P$$

```
%solving for fugacity
delta_matrix=zeros(components, components);
for i=1:components
    for j=1:components
        if i==j
           delta_matrix=delta_matrix;
        else
            delta_matrix(i,j)=2*b(i,j)-b(i,i)-b(j,j);
        end
    end
end

for k=1:components
    sum=0;
    for i=1:components
        for j=1:components
            sum=sum+(y(1,i)*y(1,j)*(2*delta_matrix(i,k)-
delta_matrix(i,j)));
        end
    end
    ln_phi=pressure/83.14/temperature*(b(k,k)+0.5*sum);
    phi=exp(ln_phi);
    fprintf('\nphi_hat_%d = %g\n', k, phi);
    fugacity=phi*y(1,k)*pressure;
    fprintf('f_hat_%d = %g bar\n', k, fugacity);
end
```

This line creates a matrix where the values for δ can be stored.

This part of the code solves for the values of δ, while considering the restrictions as indicated above.

This part of the code solves for the values of fugacity coefficients $\hat{\phi}_k$ and the fugacity of each species in the solution $\hat{f}_k$. The formulas for each variable are indicated above. It was more efficient to solve for the summation part for $\hat{\phi}_k$ first, so it was isolated and solved inside a separate nested for loop.

SAMPLE RESULT FOR THE CALCULATION OF FUGACITY

```
phi_hat_1 = 1.0192
f_hat_1 = 7.4911 bar

phi_hat_2 = 0.880658
f_hat_2 = 13.2539 bar

phi_hat_3 = 0.774943
f_hat_3 = 9.76429 bar
```

OUTPUT FOR SYSTEM A: METHANE(1), ETHANE(2), AND PROPANE(3) AT 373.15K AND 35 BAR

y_1=0.21, y_2= 0.43, y_3=0.36

```
Input number of components: 3

Input mixture temperature (in Kelvin): 373.15
Input mixture pressure (in bar): 35

Input gas/vapor composition y1: .21
Input w1: .012
Input critical temperature (in Kelvin) of component 1: 190.6
Input critical pressure (in bar) of component 1: 45.99
Input Zc of component 1: .286
Input Vc (in cm3/mol) of component 1: 98.6

Input gas/vapor composition y2: .43
Input w2: .1
Input critical temperature (in Kelvin) of component 2: 305.3
Input critical pressure (in bar) of component 2: 48.72
Input Zc of component 2: .279
Input Vc (in cm3/mol) of component 2: 145.5

Input gas/vapor composition y3: .36
Input w3: .152
Input critical temperature (in Kelvin) of component 3: 369.8
Input critical pressure (in bar) of component 3: 42.48
Input Zc of component 3: .276
Input Vc (in cm3/mol) of component 3: 200
```

```
The possible Bij values are:
B12: -51.515
B13: -78.4526
B23: -166.84

phi_hat_1 = 1.0192
f_hat_1 = 7.4911 bar

phi_hat_2 = 0.880658
f_hat_2 = 13.2539 bar

phi_hat_3 = 0.774943
f_hat_3 = 9.76429 bar
```

OUTPUT FOR SYSTEM B: N-PENTANE(1), N-HEXANE(2), CYCLOHEXANE (3), CYCLOPENTANE (4) AT 500K AND 7 BAR

y_1=0.35, y_2= 0.35, y_3=0.15, y_4=0.15

```
Input number of components: 4

Input mixture temperature (in Kelvin): 500
Input mixture pressure (in bar): 7

Input gas/vapor composition y1: .35
Input w1: .252
Input critical temperature (in Kelvin) of component 1: 469.7
Input critical pressure (in bar) of component 1: 33.7
Input Zc of component 1: .27
Input Vc (in cm3/mol) of component 1: 313

Input gas/vapor composition y2: .35
Input w2: .301
Input critical temperature (in Kelvin) of component 2: 507.6
Input critical pressure (in bar) of component 2: 30.25
Input Zc of component 2: .266
Input Vc (in cm3/mol) of component 2: 371

Input gas/vapor composition y3: .15
Input w3: .21
Input critical temperature (in Kelvin) of component 3: 553.6
Input critical pressure (in bar) of component 3: 40.73
Input Zc of component 3: .273
Input Vc (in cm3/mol) of component 3: 308

Input gas/vapor composition y4: .15
Input w4: .196
Input critical temperature (in Kelvin) of component 4: 511.8
Input critical pressure (in bar) of component 4: 45.02
Input Zc of component 4: .273
Input Vc (in cm3/mol) of component 4: 258
```

```
The possible Bij values are:
B12: -417.44
B13: -415.739
B14: -346.28
B23: -503.752
B24: -420.125
B34: -414.23

phi_hat_1 = 0.944208
f_hat_1 = 2.31331 bar

phi_hat_2 = 0.918881
f_hat_2 = 2.25126 bar

phi_hat_3 = 0.919905
f_hat_3 = 0.9659 bar

phi_hat_4 = 0.943844
f_hat_4 = 0.991036 bar
```