DESIGN & Innovation Centre, RGIPT

Supervisor : **Dr. Milan Kumar**

Project ID : DIC/2020/LD/017

Project Title : **GUI based incoming multi-component feed quality calculator**

Executives: Raunak Garg (Eche18-054)
Divanshu Jha (Eche18-047)

Contents

Project Details

- The project was aimed at obtaining the quality of incoming feed in the distillation.
- Python language was used and data(Antoine's equation constants, components) were taken as input from excel file according to component information feed, using Ideal Raoult's law followed by estimating feed conditions, calculating bubble point and dew point using iterative method and focusing on least error, as a result getting accurate feed quality.

Theory

- Antoine Equation

A relatively simple empirical equation that correlates vapor pressure–temperature data extremely well is the **Antoine equation.**

Here, pressure is in mmHg & Temperature is in Celsius.

$$\log_{10} p^* = A - \frac{B}{T + C}$$

- Raoult's Law

Suppose A is a substance contained in a gas–liquid system in equilibrium at temperature T and pressure P.

Raoult's law is an approximation that is generally valid when x_A is close to 1,that is, when the liquid is almost pure A. It is also sometimes valid over the entire range of compositions for mixtures of similar substances, such as paraffinic hydrocarbons of similar molecular weights.

$$p_A \equiv \boxed{y_A P = x_A p_A^*(T)}$$

Theory

- Finding feed condition, Liquid & Vapour Composition (L + V system)

for given T, P, z,
Assuming solution to be ideal liquid, Raoult's law will be applicable

$$y_i P = x_i P_i^{sat}$$

& $$K_i = \frac{y_i}{x_i} = \frac{P_i^{sat}}{P} \qquad \{K_i = f(T,P)\}$$

Using, Antoine eqn,

$$\ln P^{sat} = A - \frac{B}{T+C}$$

Now, calculating,

$$\sum z_i K_i = z_A K_A + z_B K_B + \dots$$

$$\sum \frac{z_i}{K_i} = \frac{z_A}{K_A} + \frac{z_B}{K_B} + \dots$$

if,

$\sum K_i z_i$	$\sum \frac{z_i}{K_i}$	Feed Condition
>1	>1	$L+V$
$=1$	-	Saturated liquid
-	$=1$	Saturated Vapour
<1	-	Subcooled liquid
	<1	Superheated Vapour.

for $L+V$ system,

$$F = L + V$$

$$F z_i = L x_i + V y_i$$

$\{$ We know that $\sum x_i = 1$, $\sum y_i = 1$ $\}$

Theory

Bubble point & Dew point

When a liquid is heated slowly at constant pressure, the temperature at which the first vapor bubble forms is the **bubble-point temperature** of the liquid at the given pressure.

When a gas (vapor) is cooled slowly at constant pressure, the temperature at which the first liquid droplet forms is the **dew-point temperature** at the given pressure

Bubble Point Calculation

$$p_i = x_i p_i^*(T_{\text{bp}}), \quad i = \text{A}, \text{B}, \ldots$$

$$y_A + y_B + y_C + \ldots = 1$$

$$P = x_{\text{A}} p_{\text{A}}^*(T_{\text{bp}}) + x_{\text{B}} p_{\text{B}}^*(T_{\text{bp}}) + \cdots$$

Dew Point Calculation

$$x_i = \frac{y_i P}{p_i^*(T_{\text{dp}})}, \qquad i = \text{A}, \text{B}, \text{C}, \ldots$$

$$x_{\text{A}} + x_{\text{B}} + x_{\text{C}} + \cdots = 1$$

$$\frac{y_{\text{A}} P}{p_{\text{A}}^*(T_{\text{dp}})} + \frac{y_{\text{B}} P}{p_{\text{B}}^*(T_{\text{dp}})} + \cdots = 1$$

How did we proceed ?

- In 1st meeting, our supervisor Dr. Milan Sir gives us the project details and some suggestions about using the appropriate computer language.
- We then studied thoroughly about the concepts that were going to be used in our project and then break the whole project in small fragments so that we can make our strategy efficiently.
- In 2nd meeting, we clarified our doubts and after clarifying we started developing the code in python.
- We then extracted the data(Antoine's constants) from the excel sheet and various inputs were taken to calculate the summations of K_by_Z and K_Z to know about the condition of our input feed in the distillation column and in 3rd meeting we showed our progress.
- In 3rd meeting , Milan sir told us to crosscheck the result we have obtained and include it in the code and he also told us the method of cross checking the results, i.e to calculate the bubble and dew point of feed mixture.

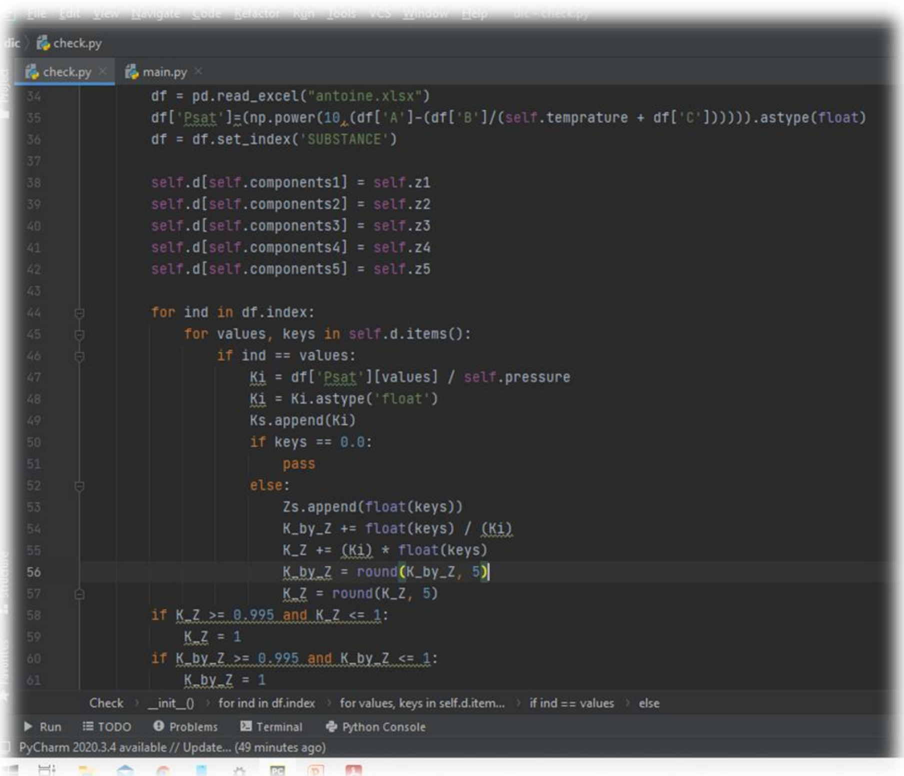
How did we proceed ?

- In 4^{th} & 5^{th} meeting, we showed the calculation of bubble and dew point. The next task assigned to us was to calculate the vapor, liquid compositions and flow rates for the liquid-vapor systems.
- In 6^{th} meeting, we showed our work to Milan sir and then sir gives us some suggestions to improve the efficiency of our code.
- In further meetings, sir told us to finalize the code in the form of interactive GUI(Graphical User Interface) which asks the user to input values such as Temperature, pressure, feed rate, no. of components, and there respective compositions and it showed the desired output.
- In series of other meetings, we worked on creating GUI and showed our weekly progress to sir and finally we had our program ready in the form of GUI.

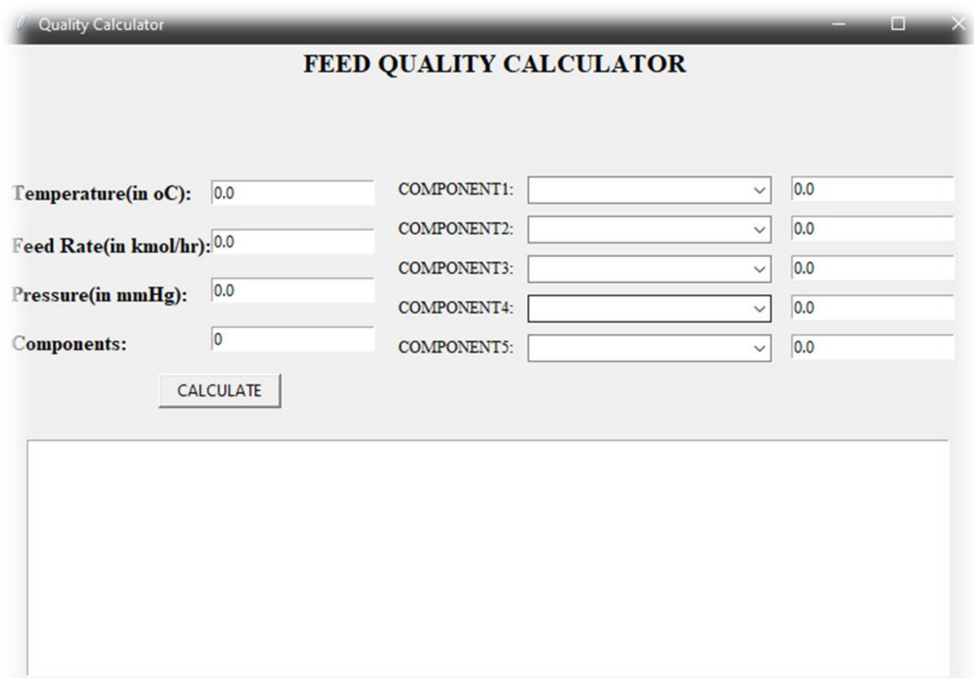
Problems Faced

- In bubble and dew point calculations, we faced difficulty in applying the theoritical(hit &trial) concepts to practical application(code) .
- It was easy to do the calculation for binary mixture but for multi-component mixtures it was bit time-consuming and tricky to derive the general relation from mass balances and also to imply that relation in the code as well.
- The biggest hurdle before us was to be finalize the code in the form of interactive GUI as we both were not familiar in making GUIs in python.

Glimpse



GUI



Code

Illustration 1 (L + V system)

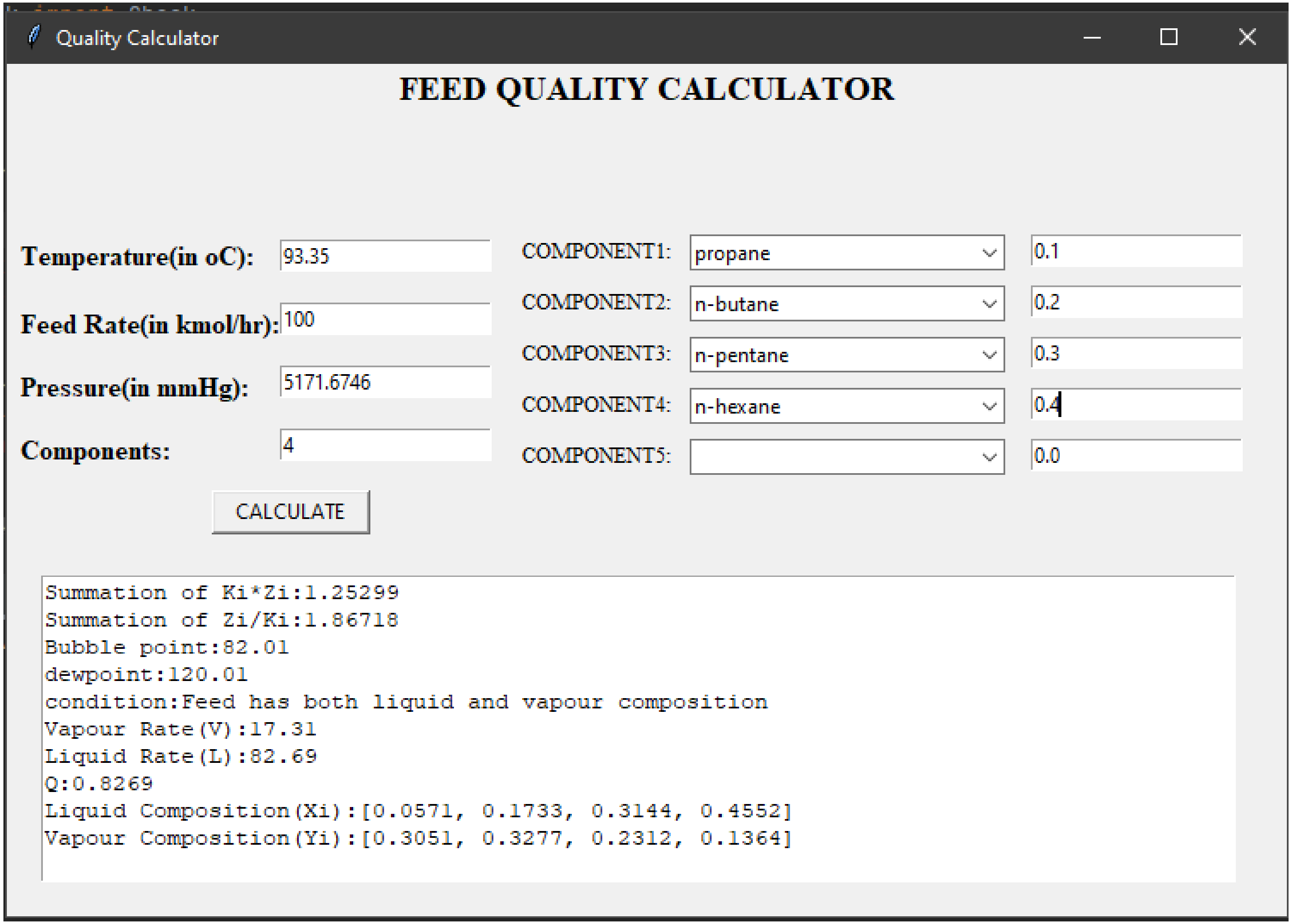


Illustration 2 (Saturated Vapor)

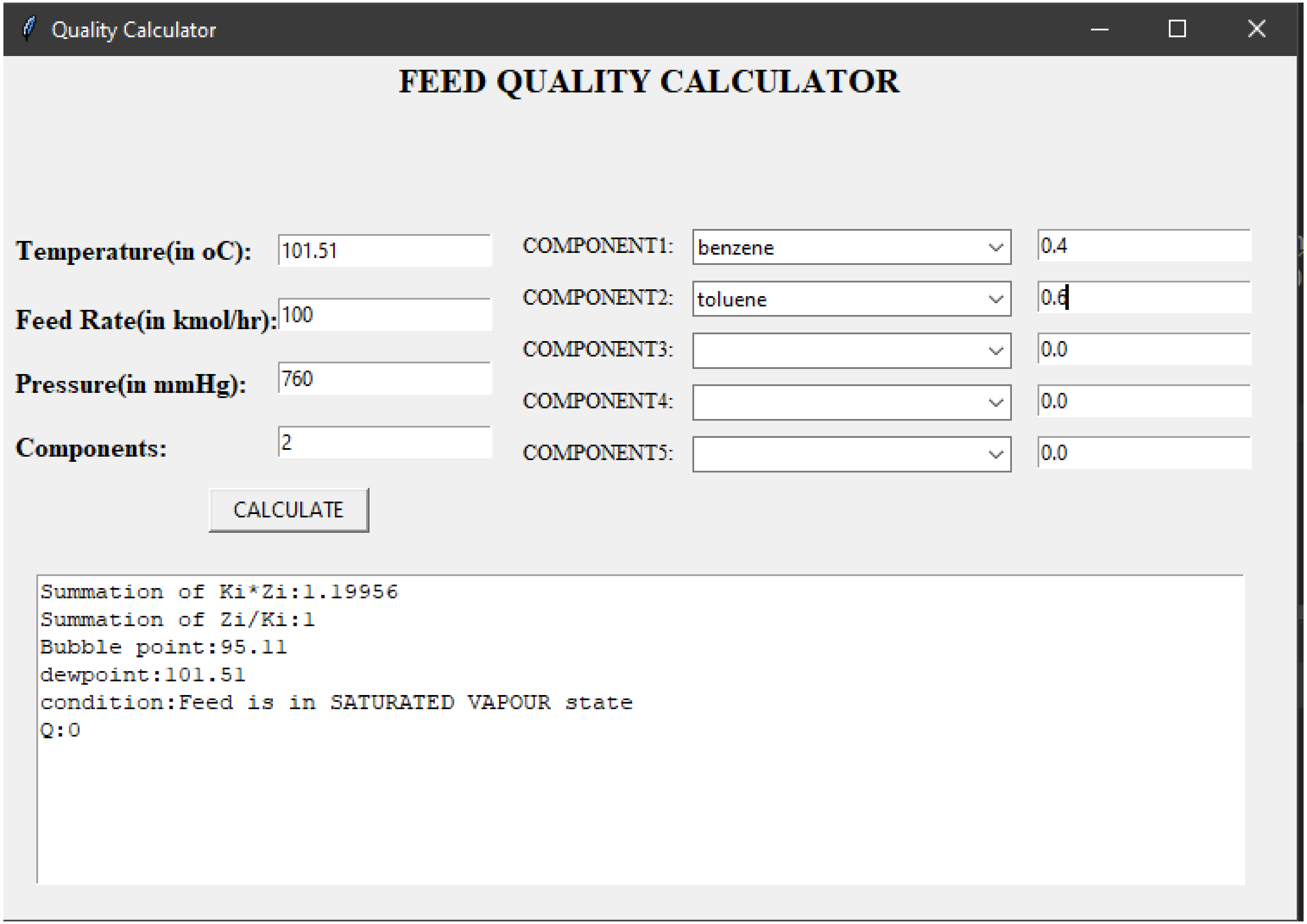
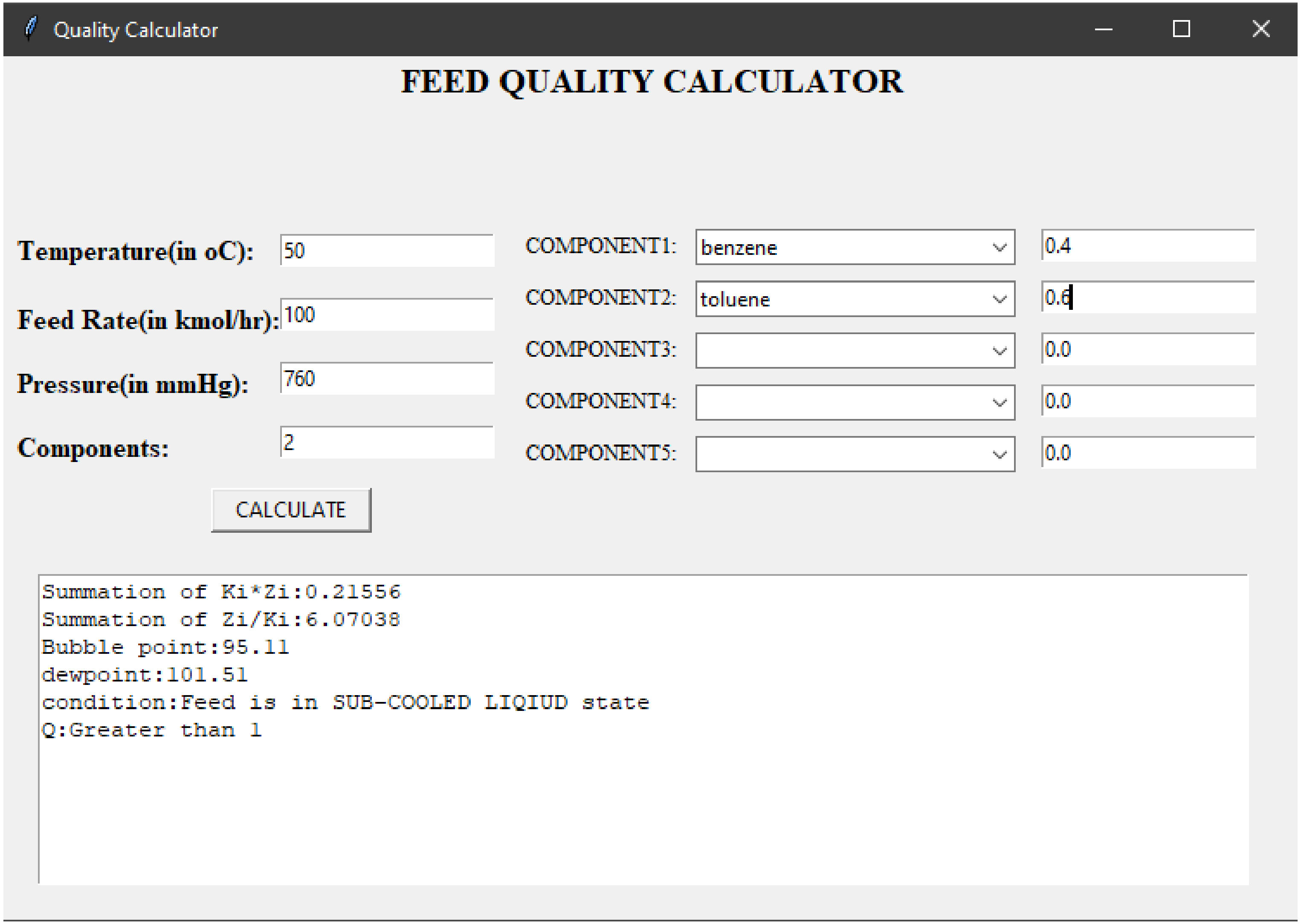


Illustration 3 (Sub-cooled liquid)



Libraries used in Python

- ‘Pandas’ library for extracting data from excel sheet and using that data.
- ‘Numpy’ library for calculating mathematical relations.
- ‘Tkinter’ library for making GUI.

References

- Mass & Energy balance lectures by Rakesh sir.
- Lecture and study material given by Milan sir.

THANK YOU