EXPERIMENT NO. 6

AIM:

To implement a Fuzzy Controller.
Write a program to design Fuzzy Controller.

THEORY:

Fuzzy Logic is an approach to represent uncertainity

Characteristics of fuzzy logic.

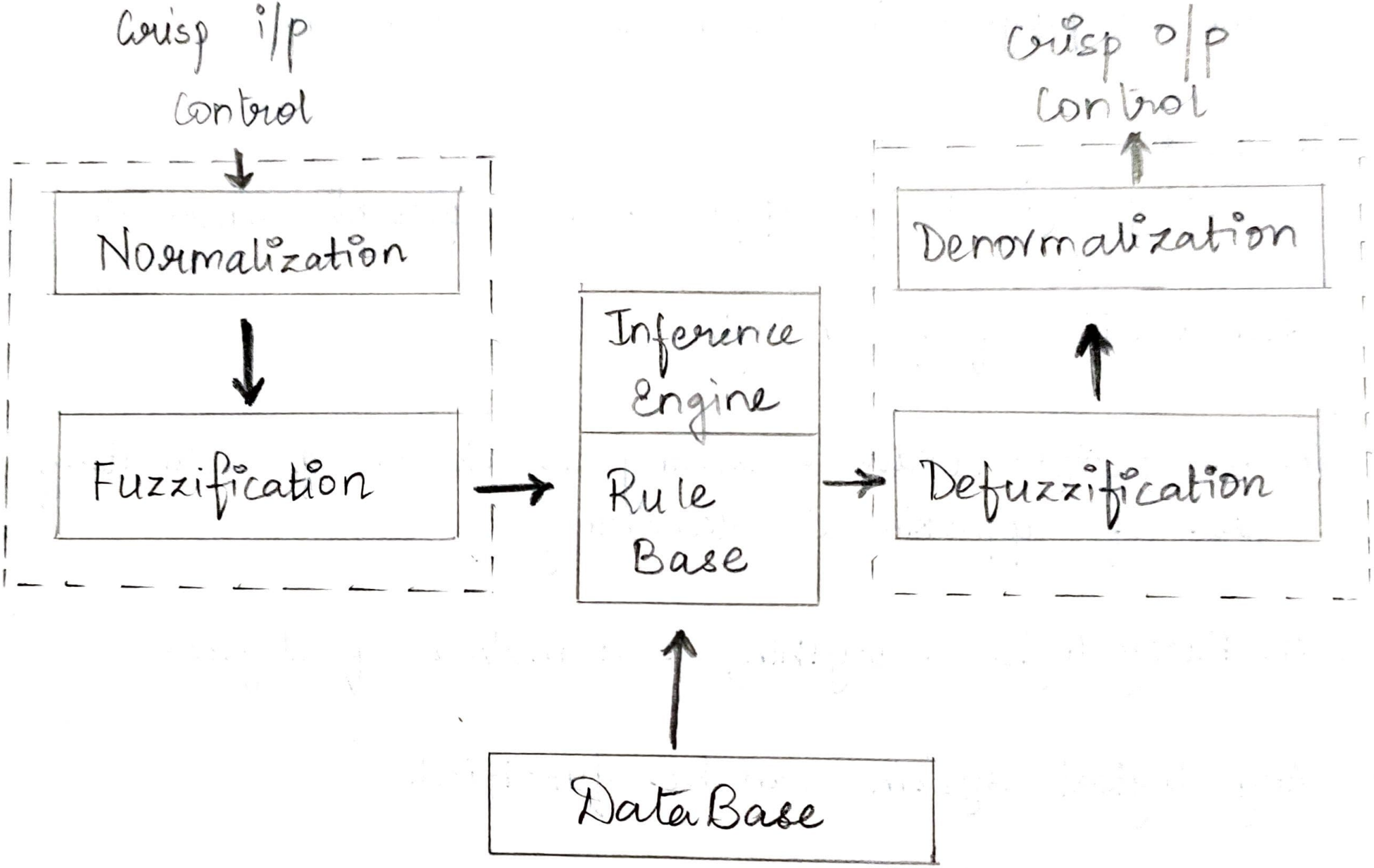
i) In Fuzzy logic, exact reasoning is viewed as a limiting case of approximate reasoning.

ii) In Fuzzy logic everything is a matter of degree.

iii) Any logical system can be fuzzified.

iv) In Fuzzy logic knowledge is interpreted as a collection of elastic or equivalently, fuzzy constraint on a collection of variaties.

v) Inference is viewed as a process of propogation of elastic constraints.


Crisp i/p
Control
Normalization
Fuzzification
Inference
Engine
Rule
Base
Defuzzification
Denormalization
Crisp o/p
Control
DataBase

A FUZZY INFERENCE SYSTEM (FIS)

An FKBC is a highly specialized system designed for performing a specific task during a particular phase of the life cycle of a process control system.

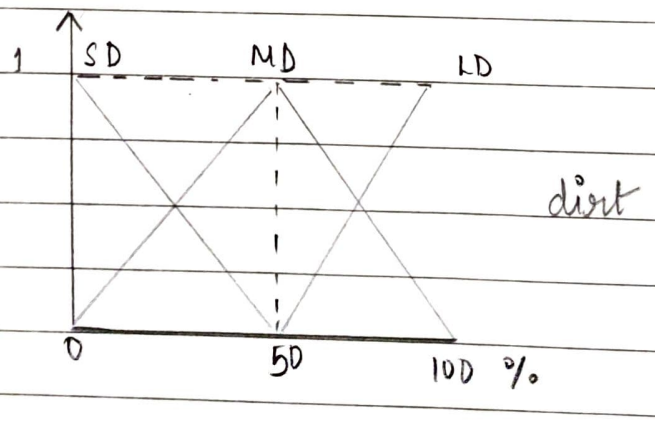
→ Various Modules :

1. NORMALIZATION

Creates a normalized universe of discosure.
Performs a scale transformation.
Eg: Universe of discosure → incy

2. FUZZIFICATION

Converts crisps values ie normalized crisp values to Fuzzy set.
Eg. very tall, medium tall, less tall etc
Generalized way of forming a fuzzy set



3. RULE BASE

if then else

ie combination of all conditions.

if <antecedent> then <consequent>

VARIOUS DESIGN PARAMETERS.

1. Choice of process state and control o/p variable

2. Choice of the content of the rule antecedent and rule consequent.

3. Derivation of set of rules.

4. DATABASE

Provides and stores necessary information

Various design parameters are

i) Choice of Membership function.

ii) Choice of scaling factor

5 INFERENCE ENGINE :

KB = RB + DB infers o/p based on Rulebase and Database.

6. DEFUZZIFICATION :

Converting the set of fuzzy values to single point-wise value.

7. DENORMALIZATION

Maps point wise to its physical domain.

General Steps for Designing a fuzzy controller for real world problems :

Step 1 : Identify i/p and o/p using linguistic variable

Draw Graphs for each

Step 2 : Assigning membership functions.

Step 3 : 3.1 Build a rule base
3.2 Rule Evaluation
3.3 Rule Decision Table

Step 4 : Defuzzification.
4.1 Min - Max Method
4.2 Rule strength Table
4.3 Map RDT with RST get $\bar{z}$ value

→ CONCLUSION :

Thus we have successfully implemented a fuzzy controller using Python Programming Language.

Soft Computing

Title: Program to Solve the Tipping problem

Program:

```
x = int(input('Enter the Rating for Food Quality [0-10] :'))
y = int(input('Enter the Rating for Service [0-10] :'))

'''
Input Descriptors:
1) Food Quality [0-10]:
   i) VP : Very Poor
   ii)P: Poor
   iii)G: Good
   iv)E: Excellent
2) Service[0-10]:
   i) VP : Very Poor
   ii)P: Poor
   iii)G: Good
   iv)E: Excellent
Ouput Descriptor:
1) Tip [0-25%]:
   i) VL: Very Less
   ii) L: Less
   iii) N: Normal
   iv) H: High
   v) VH: Very High
'''

#Food Quality Membership Function

fq_vp = 0
if 0 <= x <= 5:
   fq_vp = (5 - x) / 5

fq_p = 0
```

```
if 0 <= x <= 5:
    fq_p  = x / 50
if 5 <= x <= 8:
    fq_p  = (8 - x) / 3

fq_g = 0
if 5 <= x <= 8:
    fq_g = (x - 5) / 3
if 8 <= x <= 10:
    fq_g = (10-x) /  2

fq_e = 0
if 8 <= x <= 10:
    fq_e = (x - 8) / 2

#Service Membership Function

s_vp = 0
if 0 <= y <= 5:
    s_vp = (5 - y) / 5

s_p = 0
if 0 <= y <= 5:
    s_p  = y / 50
if 5 <= y <= 8:
    s_p  = (8 - y) / 3

s_g = 0
if 5 <= y <= 8:
    s_g = (y - 5) / 3
if 8 <= y <= 10:
    s_g = (10-y) /  2

s_e = 0
if 8 <= y <= 10:
```

```
    s_e = (y - 8) / 2



r1 = min(fq_vp, s_vp)
r2 = min(fq_vp, s_p)
r3 = min(fq_vp, s_g)
r4 = min(fq_vp, s_e)
r5 = min(fq_p, s_vp)
r6 = min(fq_p, s_p)
r7 = min(fq_p, s_g)
r8 = min(fq_p, s_e)
r9 = min(fq_g, s_vp)
r10 = min(fq_g, s_p)
r11 = min(fq_g, s_g)
r12 = min(fq_g, s_e)
r13 = min(fq_e, s_vp)
r14 = min(fq_e, s_p)
r15 = min(fq_e, s_g)
r16 = min(fq_e, s_e)


out = max(r1, r2, r3, r4, r5, r6, r7, r8, r9, r10, r11, r12, r13, r14, r15, r16)
if r1 == out or r2 == out or r5 == out:
    vl = out
    z = 5 - 5 * vl
if r3 == out or r4 == out or r6 == out or r7 == out or r9 == out or r13 == out :
    l = out
    z1 = 5 * l
    z2 = 10 - 5 * l
    z = (z1 + z2) / 2
if r8 == out or r10 == out or r14 == out:
    n = out
    z1 = 5 * n + 5
    z2 = 20 - 10 * n
    z = (z1 + z2) / 2
if r11 == out or r12 == out :
```

```
    h = out
    z1 = 10 * h + 10
    z2 = 25 - 5 * h
    z = (z1 + z2) / 2
if r15 == out or r16 == out:
    vh = out
    z = 20 + 5 *vh
print('The calculated Tip is : {0} %'.format(z))
```

Output:

Enter the Rating for Food Quality [0-10]8

Enter the Rating for Service [0-10]8

The calculated Tip is : 20.0 %

>>>

========== RESTART: C:/Users/asus/Desktop/sc.py ==========

Enter the Rating for Food Quality [0-10] :3

Enter the Rating for Service [0-10] :10

The calculated Tip is : 5.0 %