ABHINAV RANJAN
RA1911003010003
CSE A1 SECTION
SRMIST , KTR

AI LAB EXP 6 - FUZZY LOGIC

Problem Statement: To implement uncertain methods for an application (to calculate membership of certain values based on user's input and display the same) using Fuzzy logic

Tool : AWS

Algorithm:

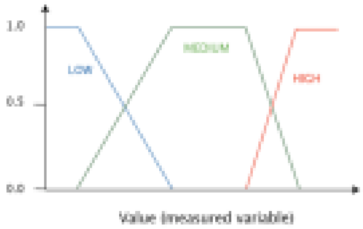
1. Define Non Fuzzy Inputs with Fuzzy Sets. The non-fuzzy inputs are numbers from a certain range, and find how to represent those non-fuzzy values with fuzzy sets.
2. Locate the input, output, and state variables of the plane under consideration.
3. Split the complete universe of discourse spanned by each variable into a number of fuzzy subsets, assigning each with a linguistic label. The subsets include all the elements in the universe.
4. Obtain the membership function for each fuzzy subset.
5. Assign the fuzzy relationships between the inputs or states of fuzzy subsets on one side and output of fuzzy subsets on the other side, thereby forming the rule base.
6. Choose appropriate scaling factors for the input and output variables for normalising the variables between [0, 1]

and [-1, I] intervals.
7. Carry out the fuzzification process.
8. Identify the output contributed from each rule using fuzzy approximate reasoning.
9. Combine the fuzzy outputs obtained from each rule.
10. Finally, apply defuzzification to form a crisp output.

Optimization Technique:

Various numerical optimization techniques can be used such as dynamic programming, Lagrangian relaxation method , mixed integer programming ,and branch-and-bound method . The dynamic programming method is simple but the calculation time required to converge to the optimal solution is quite long. Regarding the branch and-bound method, it adopts a linear function to represent the fuel and start-up costs during a time horizon.

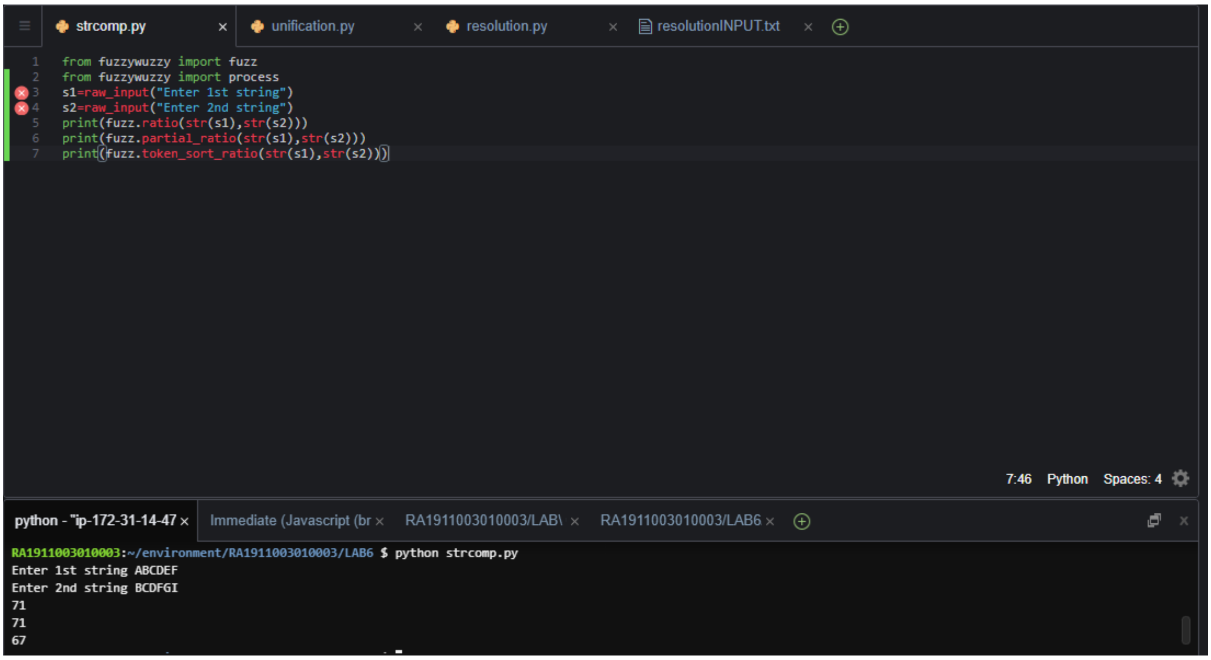
The mixed integer programming uses linear programming to attain optimal solutions. Nevertheless, this method was applied to small problems of unit commitment and they required major assumptions that limit the margin of solutions. For the Lagrangian relaxation method, we note that the convergence time is an advantage, but the obtained solution is not ideal because of the complexity of the problem especially when the optimization problem contains a great number of production units.



SOURCE CODE :

```
from fuzzywuzzy import fuzz
from fuzzywuzzy import process
S1 = raw_input(“Enter 1st string”)
S2 = raw_input(“ENter 2nd string”)
print(fuzz.ratio(str(S1),str(S2)))
print(fuzz.partial_ratio(str(S1),str(S2)))
print(fuzz.token_sort_ratio(str(S1),str(S2)))
```

SCREENSHOTS :



```
from fuzzywuzzy import fuzz
from fuzzywuzzy import process
s1=raw_input("Enter 1st string")
s2=raw_input("Enter 2nd string")
print(fuzz.ratio(str(s1),str(s2)))
print(fuzz.partial_ratio(str(s1),str(s2)))
print(fuzz.token_sort_ratio(str(s1),str(s2)))
```

```
RA1911003010003:~/environment/RA1911003010003/LAB6 $ python strcomp.py
Enter 1st string ABCDEF
Enter 2nd string BCDFGI
71
71
67
```

Result: Implementation of uncertain methods for an application (to calculate membership of certain values) is successfully implemented