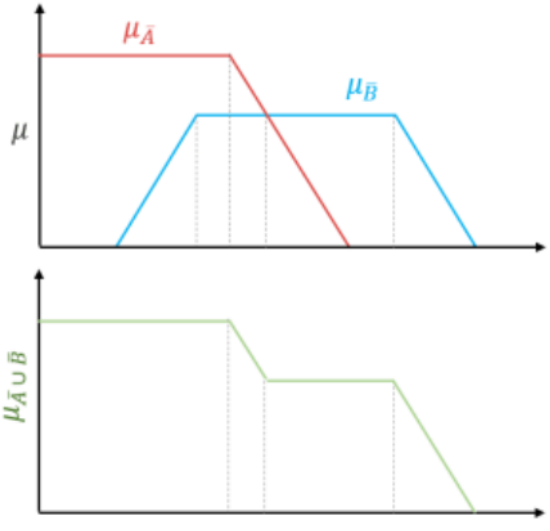
Date: **Ex No:** 6.1	**Title of the Lab** Implementation of Fuzzy Logic	**Name:** Yuvraj Singh Chauhan **Registration Number:** RA1911027010058 **Section:** N1 **Lab Batch:** 1 **Day Order:** 3

AIM:

To implement Fuzzy Logic.

Description of the Concept or Problem given:

In case of union of crisp sets, we simply have to select repeated elements only once. In case of fuzzy sets, when there are common elements in both the fuzzy sets, we should select the element with **maximum membership value**.
The **union** of two fuzzy sets A and B is a fuzzy set C , written as C = A ∪ B
Graphically we can represent union operation as follow. Red and Blue membership functions represents the fuzzy value for elements in set A and B, respectively. Wherever these fuzzy functions overlaps, we have to consider the point with maximum membership value.



Fuzzy union

Manual Solution:

1. Import matplotlib from the python library.
2. Initialize an array of numbers in two 2D arrays a and b which would act as sets of numbers.
3. Define two functions checkset() and set_and_mf_of_set() which will check whether the elements in the array contains more than one membership value or not, if not than return the set.
4. Initialize another set of numbers in two 2D array x and y with slightly different values and store them in a different variable and check whether they contain they contain membership value or not.

5. Find the union of arrays x and y using the fuzzy logic code of union for finding the union between the arrays and print them.
6. Plot the corresponding graphs of set_and_mf_of_set(x) and set_and_mf_of_set(y).
7. The middle graph represents the union between the two sets of arrays.

Program Implementation [Coding]:

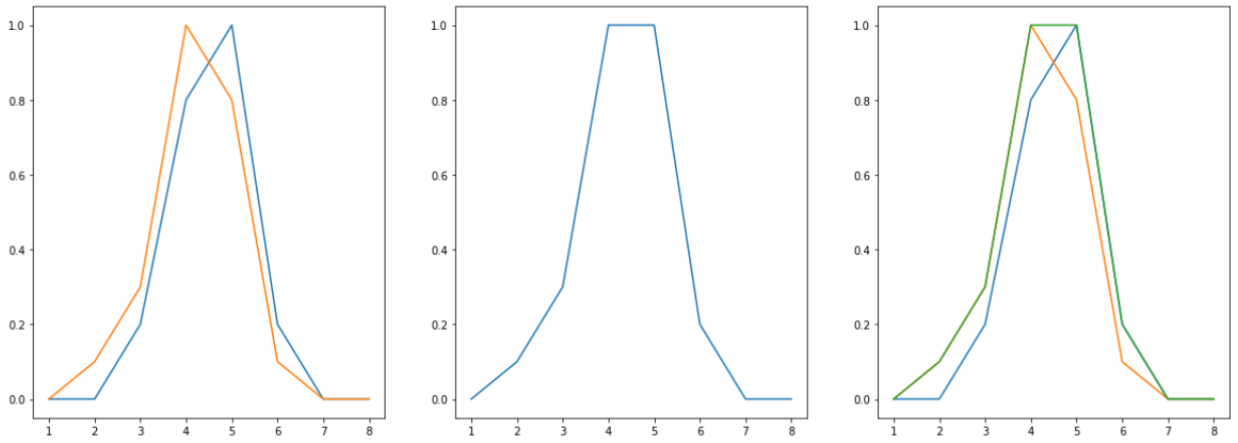
```
import matplotlib.pyplot as plt
plt.rcParams["figure.figsize"] = (20,7)
a =[[1,0.0],[2,0.0],[3,0.2],[4,0.8],[5,1],[6,0.2],[7,0.0],[8,0.0]]
b =[[1,0.0],[2,0.1],[3,0.3],[4,1],[5,0.5],[6,0.1],[7,0.0],[8,0.0]]

print()
def checkset(a):
    r=0
    for i in range(len(a)):
        for j in range(i+1,len(a)):
            if a[i][0]==a[j][0]:
                r+=1
                break
    return r

def set_and_mf_of_set(a):
    p = checkset(a)
    if p == 0:
        set1 =[]
        mfset=[]
        for i in range(len(a)):
            set1.append(a[i][0])
            mfset.append(a[i][1])
        return set1,mfset
    else:
        print("In Set at one element more than one MemberShip Value")

a = set_and_mf_of_set(a)
x =[[1,0.0],[2,0.0],[3,0.2],[4,0.8],[5,1],[6,0.2],[7,0.0],[8,0.0]]
y =[[1,0.0],[2,0.1],[3,0.3],[4,1],[5,0.8],[6,0.1],[7,0.0],[8,0]]
p,mfp = set_and_mf_of_set(x)
q,mfq = set_and_mf_of_set(y)

def Union(a,b):
    p = checkset(a)
    q = checkset(b)
    if p == 0 & q==0:
        union=[]
        if len(a) < len(b) :
            temp = a
            a = b
            b = temp
        for i in range (len(a)):
            for j in range(0,len(b)):
                if a[i][0] == b[j][0]:
                    if a[i][1] > b[j][1]:
                        union.append(a[i])
```

```
                else:
                    union.append(b[j])
            else:
                if len(union)==0:
                    union.append(a[i])
                else:
                    p=0
                    for k in range(0,len(union)):
                        if union[k][0]==a[i][0]:
                            p+=1
                            if p==0:
                                union.append(a[i])



        return union
    else:
        print("In Set at one element more than one MemberShip Value")


z=Union(y,x)
r,mfr = set_and_mf_of_set(z)

print(x)
print(y)
print("Union is",z)

plt.subplot(131)
plt.plot(p,mfp)
plt.plot(q,mfq)

plt.subplot(132)
plt.plot(r,mfr)

plt.subplot(133)

plt.plot(p,mfp)
plt.plot(q,mfq)
plt.plot(r,mfr)
```

Screenshots of the Outputs:

```
[[1, 0.0], [2, 0.0], [3, 0.2], [4, 0.8], [5, 1], [6, 0.2], [7, 0.0], [8, 0.0]]
[[1, 0.0], [2, 0.1], [3, 0.3], [4, 1], [5, 0.8], [6, 0.1], [7, 0.0], [8, 0]]
Union is [[1, 0.0], [2, 0.1], [3, 0.3], [4, 1], [5, 1], [6, 0.2], [7, 0.0], [8, 0.0]]
```

Out[1]: [<matplotlib.lines.Line2D at 0x187c7729910>]



Signature of the Student

[YUVRAJ SINGH CHAUHAN]