**AIM:** Write a program to show fuzzy relations and operations

**SOURCE CODE:**

import numpy as np

import skfuzzy as fuzz

if \_\_name\_\_ == "\_\_main\_\_":

# Create universe of discourse in Python using linspace ()

X = np.linspace(start=0, stop=75, num=75, endpoint=True, retstep=False)

# Create two fuzzy sets by defining any membership function

# (trapmf(), gbellmf(), gaussmf(), etc).

abc1 = [0, 25, 50]

abc2 = [25, 50, 75]

young = fuzz.membership.trimf(X, abc1)

middle\_aged = fuzz.membership.trimf(X, abc2)

# Compute the different operations using inbuilt functions.

one = np.ones(75)

zero = np.zeros((75,))

# 1. Union = max(µA(x), µB(x))

union = fuzz.fuzzy\_or(X, young, X, middle\_aged)[1]

# 2. Intersection = min(µA(x), µB(x))

intersection = fuzz.fuzzy\_and(X, young, X, middle\_aged)[1]

# 3. Complement (A) = (1- min(µA(x))

complement\_a = fuzz.fuzzy\_not(young)

# 4. Difference (A/B) = min(µA(x),(1- µB(x)))

difference = fuzz.fuzzy\_and(X, young, X, fuzz.fuzzy\_not(middle\_aged)[1])[1]

# 5. Algebraic Sum = [µA(x) + µB(x) – (µA(x) \* µB(x))]

alg\_sum = young + middle\_aged - (young \* middle\_aged)

# 6. Algebraic Product = (µA(x) \* µB(x))

alg\_product = young \* middle\_aged

# 7. Bounded Sum = min[1,(µA(x), µB(x))]

bdd\_sum = fuzz.fuzzy\_and(X, one, X, young + middle\_aged)[1]

# 8. Bounded difference = min[0,(µA(x), µB(x))]

bdd\_difference = fuzz.fuzzy\_or(X, zero, X, young - middle\_aged)[1]

# max-min composition

# max-product composition

# Plot each set A, set B and each operation result using plot() and subplot().

from matplotlib import pyplot as plt

plt.figure()

plt.subplot(4, 3, 1)

plt.plot(X, young)

plt.title("Young")

plt.grid(True)

plt.subplot(4, 3, 2)

plt.plot(X, middle\_aged)

plt.title("Middle aged")

plt.grid(True)

plt.subplot(4, 3, 3)

plt.plot(X, union)

plt.title("union")

plt.grid(True)

plt.subplot(4, 3, 4)

plt.plot(X, intersection)

plt.title("intersection")

plt.grid(True)

plt.subplot(4, 3, 5)

plt.plot(X, complement\_a)

plt.title("complement\_a")

plt.grid(True)

plt.subplot(4, 3, 6)

plt.plot(X, difference)

plt.title("difference a/b")

plt.grid(True)

plt.subplot(4, 3, 7)

plt.plot(X, alg\_sum)

plt.title("alg\_sum")

plt.grid(True)

plt.subplot(4, 3, 8)

plt.plot(X, alg\_product)

plt.title("alg\_product")

plt.grid(True)

plt.subplot(4, 3, 9)

plt.plot(X, bdd\_sum)

plt.title("bdd\_sum")

plt.grid(True)

plt.subplot(4, 3, 10)

plt.plot(X, bdd\_difference)

plt.title("bdd\_difference")

plt.grid(True)

plt.subplots\_adjust(hspace=0.5)

plt.show()

**OUTPUT:**

