PR3

import numpy as np  
  
# Function to perform Union operation on fuzzy sets  
def fuzzy\_union(A, B):  
 return np.maximum(A, B)  
  
# Function to perform Intersection operation on fuzzy sets  
def fuzzy\_intersection(A, B):  
 return np.minimum(A, B)  
  
# Function to perform Complement operation on a fuzzy set  
def fuzzy\_complement(A):  
 return 1 - A  
  
# Function to perform Difference operation on fuzzy sets  
def fuzzy\_difference(A, B):  
 return np.maximum(A, 1 - B)  
  
# Function to create fuzzy relation by Cartesian product of two fuzzy sets  
def cartesian\_product(A, B):  
 return np.outer(A, B)  
  
# Function to perform Max-Min composition on two fuzzy relations  
def max\_min\_composition(R, S):  
 return np.max(np.minimum.outer(R, S), axis=1)  
  
# Example usage  
A = np.array([0.2, 0.4, 0.6, 0.8]) # Fuzzy set A  
B = np.array([0.3, 0.5, 0.7, 0.9]) # Fuzzy set B  
  
# Operations on fuzzy sets  
union\_result = fuzzy\_union(A, B)  
intersection\_result = fuzzy\_intersection(A, B)  
complement\_A = fuzzy\_complement(A)  
difference\_result = fuzzy\_difference(A, B)  
print("Union:", union\_result)  
print("Intersection:", intersection\_result)  
print("Complement of A:", complement\_A)  
print("Difference:", difference\_result)  
  
# Fuzzy relations  
R = np.array([0.2, 0.5, 0.4]) # Fuzzy relation R  
S = np.array([0.6, 0.3, 0.7]) # Fuzzy relation S  
  
# Cartesian product of fuzzy relations  
cartesian\_result = cartesian\_product(R, S)  
# Max-Min composition of fuzzy relations  
composition\_result = max\_min\_composition(R, S)  
print("Cartesian product of R and S:")  
print(cartesian\_result)  
print("Max-Min composition of R and S:")  
print(composition\_result)  
1

Output:

