

**Міністерство освіти і науки України
Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»
Факультет інформатики та обчислювальної техніки
Кафедра обчислювальної техніки**

Лабораторна робота №1

з дисципліни

«Методи та технології штучного інтелекту»

Виконав:

студент групи ІМ-13

Тавлуй Дмитро Олександрович

Перевірив:

Шимкович В. М.

Київ 2023

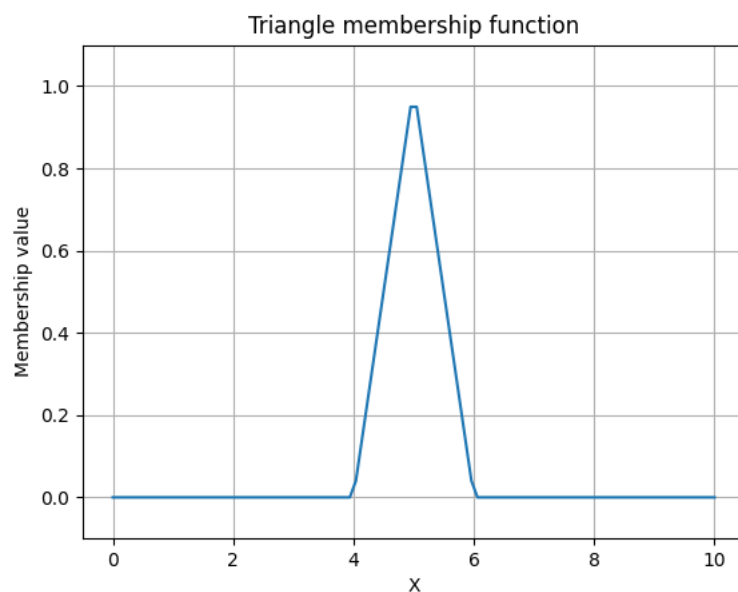
Тема: Дослідження способів формування нечітких множин і операцій над ними

Мета роботи: Побудувати нечіткі множин з використанням різних типів функцій приналежності. Виконати найбільш поширені логічні операції над нечіткими множинами.

Індивідуальні завдання

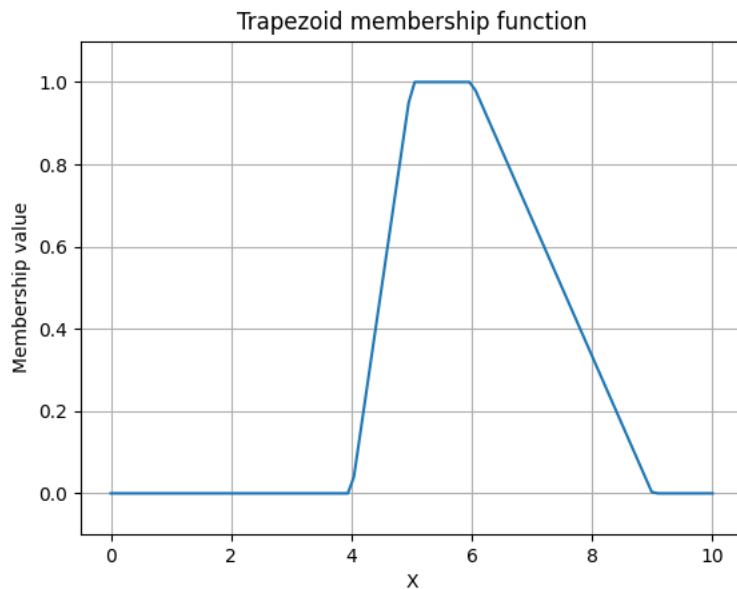
За допомогою мови програмування python та додаткових бібліотек було виконано наступні завдання:

1. Побудувати трикутну і трапецієподібну функцію приналежності.



```
x = np.linspace(0, 10, 100)
func = fuzz.trimf(x, [4, 5, 6])

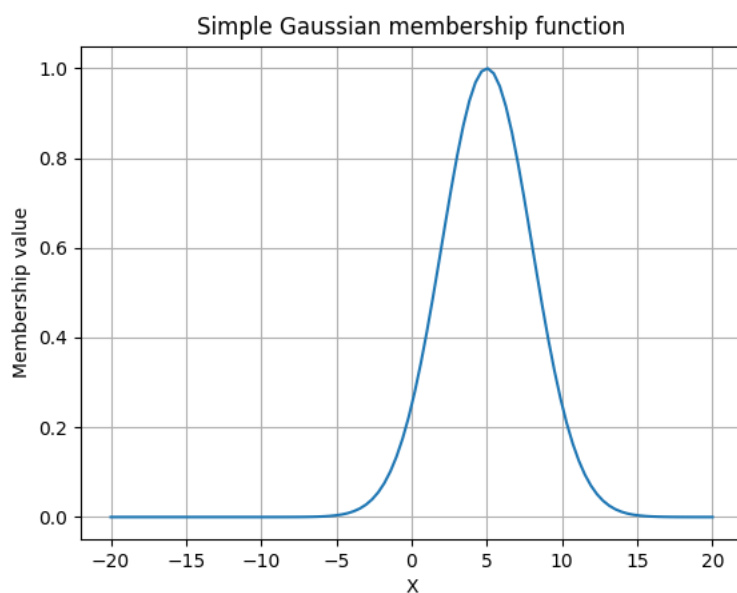
plt.plot(x, func)
plt.title('Triangle membership function')
plt.xlabel('X')
plt.ylabel('Membership value')
plt.ylim(-0.1, 1.1)
plt.grid(True)
plt.show()
```



```
x = np.linspace(0, 10, 100)
func = fuzz.trapmf(x, [4, 5, 6, 9])

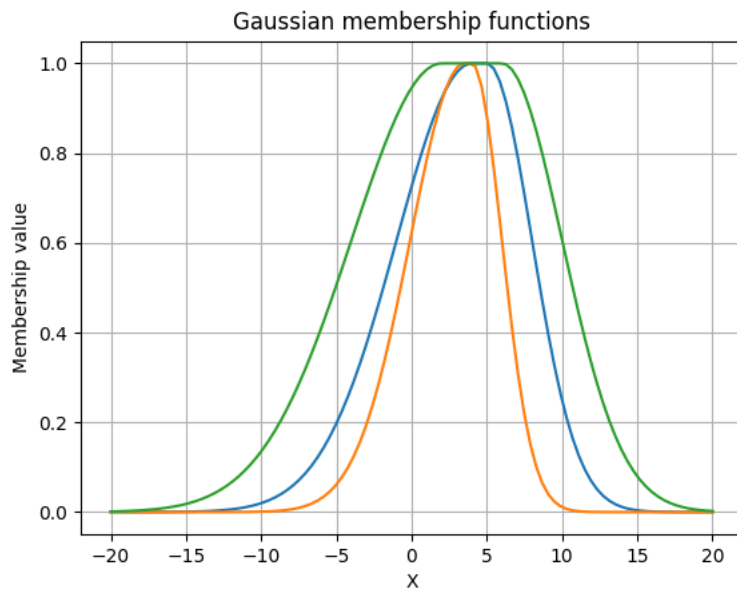
plt.plot(x, func)
plt.title('Trapezoid membership function')
plt.xlabel('X')
plt.ylabel('Membership value')
plt.ylim(-0.1, 1.1)
plt.grid(True)
plt.show()
```

2. Побудувати просту і двосторонню функцію приналежності Гаусса, утворену за допомогою різних функцій розподілу.



```
x = np.linspace(-20, 20, 100)
func = fuzz.gaussmf(x, 5, 3)

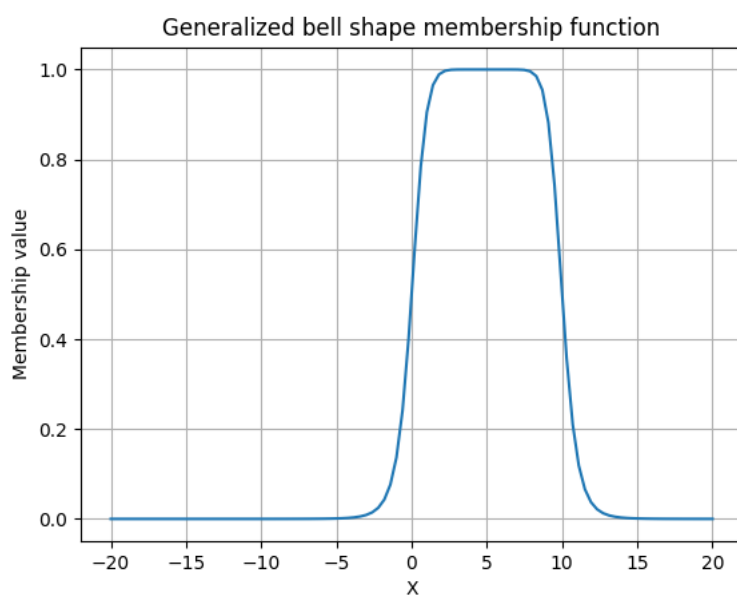
plt.plot(x, func)
plt.title('Simple Gaussian membership function')
plt.xlabel('X')
plt.ylabel('Membership value')
plt.grid(True)
plt.show()
```



```
x = np.linspace(-20, 20, 100)
func1 = fuzz.gauss2mf(x, 4, 5, 5, 3)
func2 = fuzz.gauss2mf(x, 3.5, 3.6, 4, 2)
func3 = fuzz.gauss2mf(x, 2, 6, 6, 4)

plt.plot(x, func1)
plt.plot(x, func2)
plt.plot(x, func3)
plt.title('Gaussian membership functions')
plt.xlabel('X')
plt.ylabel('Membership value')
plt.grid(True)
plt.show()
```

3. Побудувати функцію приналежності "узагальнений дзвін", яка дозволяє представляти нечіткі суб'єктивні переваги.

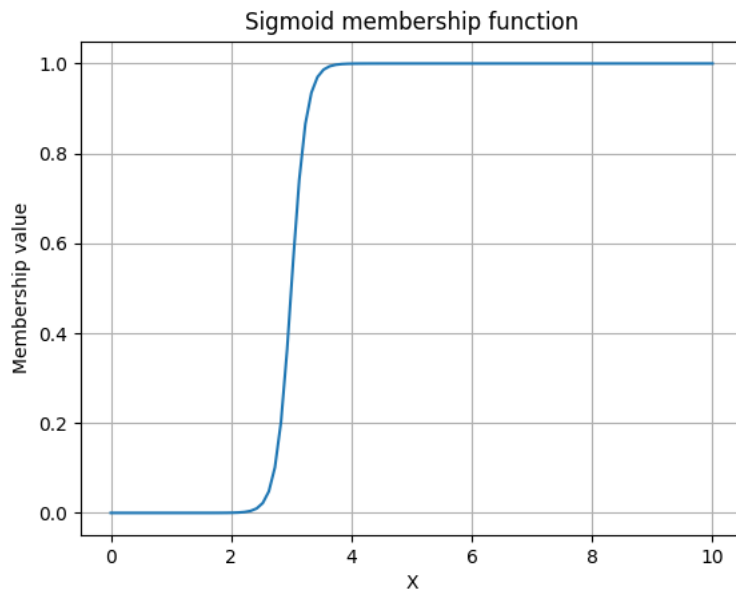


```
x = np.linspace(-20, 20, 100)
func = fuzz.gbellmf(x, 5, 5, 5)

plt.plot(x, func)
plt.title('Generalized bell shape membership function')
plt.xlabel('X')
plt.ylabel('Membership value')
```

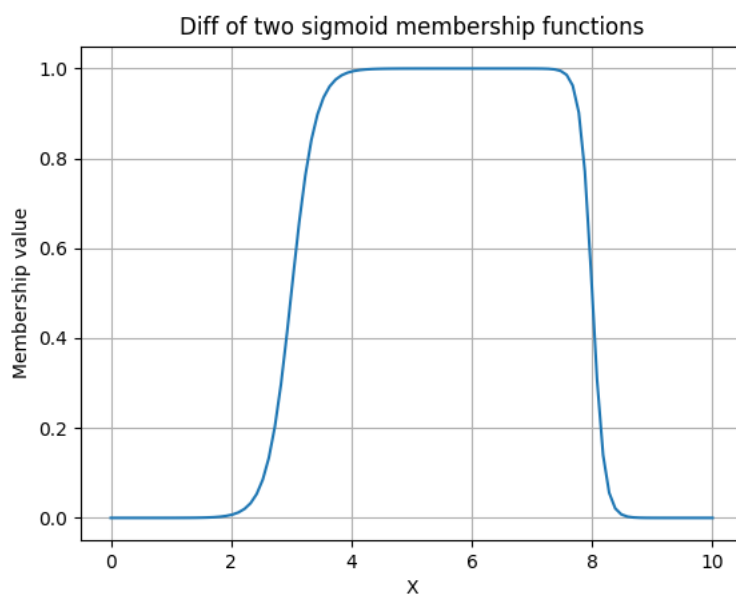
```
plt.grid(True)
plt.show()
```

4. Побудувати набір сігмоїдних функцій: основну односторонню, яка відкрита зліва чи справа; додаткову двосторонню; додаткову несиметричну.



```
x = np.linspace(0, 10, 100)
func = fuzz.sigmf(x, 3, 8)

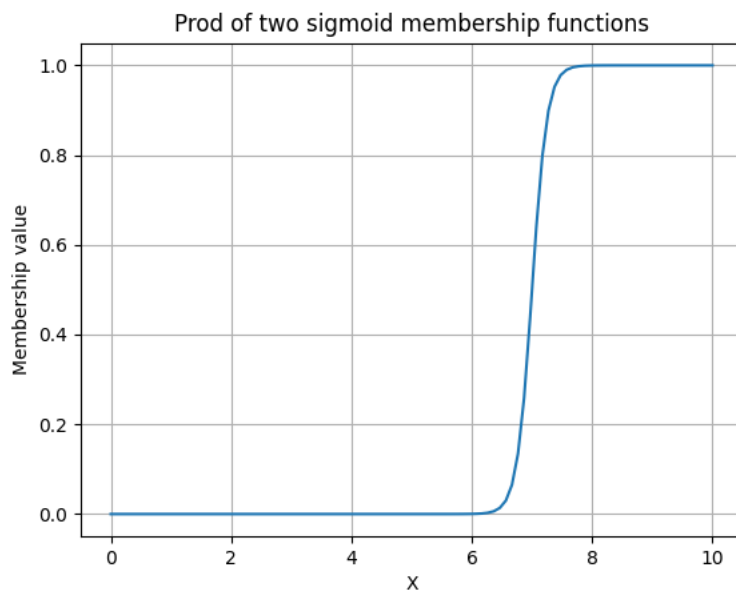
plt.plot(x, func)
plt.title('Sigmoid membership function')
plt.xlabel('X')
plt.ylabel('Membership value')
plt.grid(True)
plt.show()
```



```
x = np.linspace(0, 10, 100)
func = fuzz.dsigmf(x, 3, 5, 8, 10)

plt.plot(x, func)
plt.title('Diff of two sigmoid membership functions')
plt.xlabel('X')
plt.ylabel('Membership value')
```

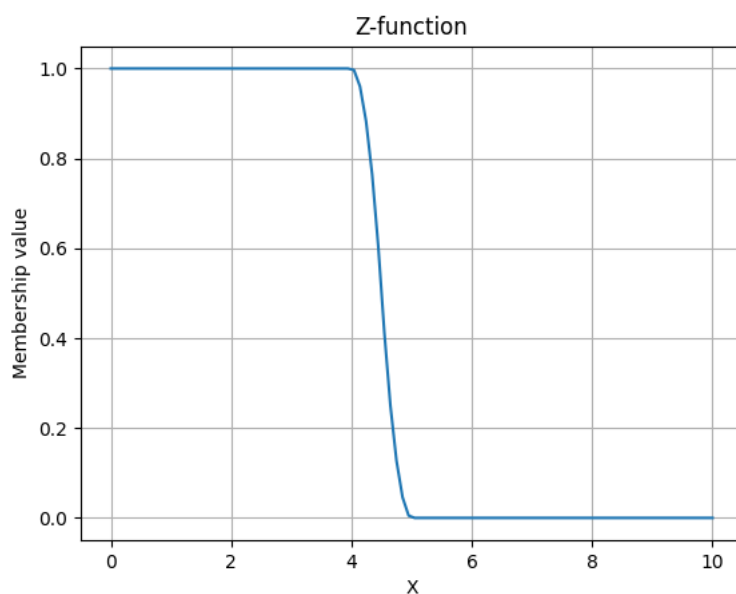
```
plt.grid(True)
plt.show()
```



```
x = np.linspace(0, 10, 100)
func = fuzz.psigmf(x, 3, 5, 7, 8)

plt.plot(x, func)
plt.title('Prod of two sigmoid membership functions')
plt.xlabel('X')
plt.ylabel('Membership value')
plt.grid(True)
plt.show()
```

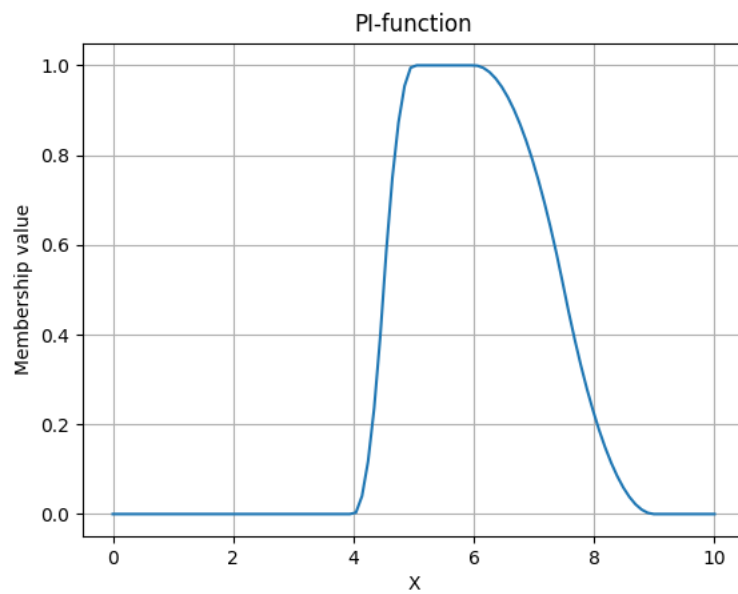
5. Побудувати набір поліноміальних функцій приналежності (Z-, PI- і S-функцій).



```
x = np.linspace(0, 10, 100)
func = fuzz.zmf(x, 4, 5)

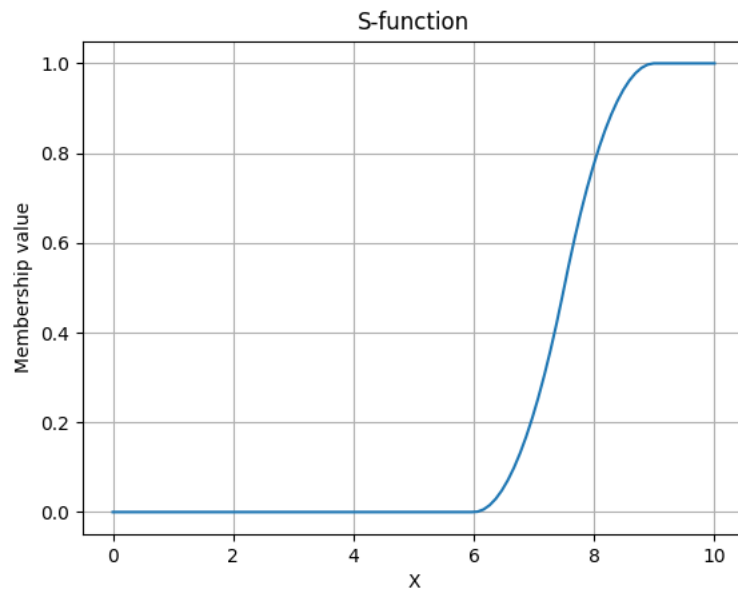
plt.plot(x, func)
plt.title('Z-function')
plt.xlabel('X')
plt.ylabel('Membership value')
```

```
plt.grid(True)
plt.show()
```



```
x = np.linspace(0, 10, 100)
func = fuzz.pimf(x, 4, 5, 6, 9)

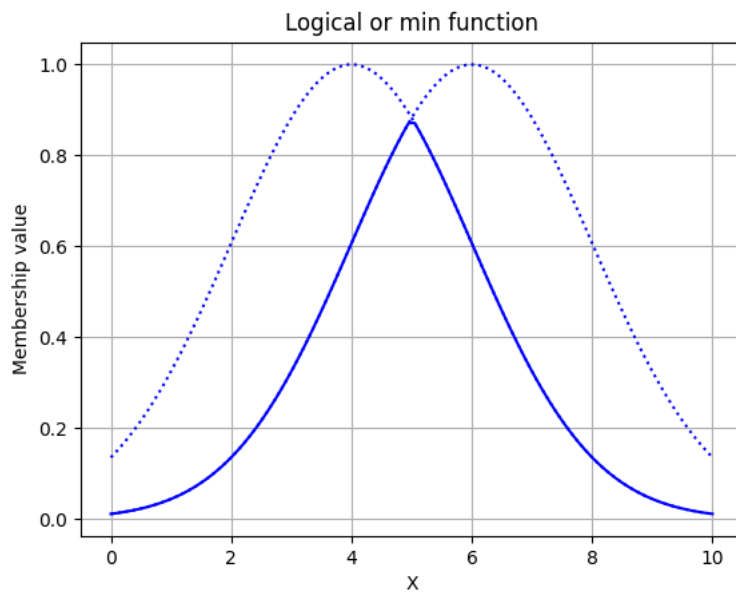
plt.plot(x, func)
plt.title('PI-function')
plt.xlabel('X')
plt.ylabel('Membership value')
plt.grid(True)
plt.show()
```



```
x = np.linspace(0, 10, 100)
func = fuzz.smf(x, 6, 9)

plt.plot(x, func)
plt.title('S-function')
plt.xlabel('X')
plt.ylabel('Membership value')
plt.grid(True)
plt.show()
```

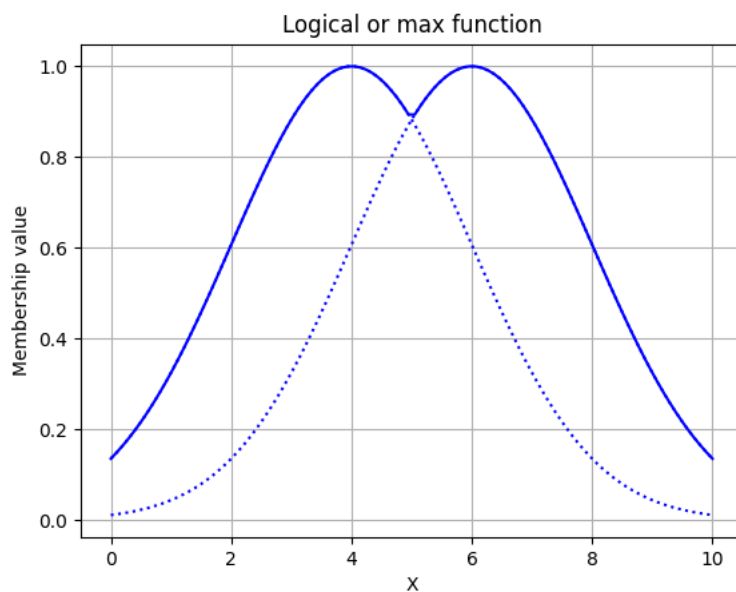
6. Побудувати мінімаксну інтерпретацію логічних операторів з використанням операцій пошуку мінімуму і максимуму.



```
x = np.linspace(0, 10, 100)
func1 = fuzz.gaussmf(x, 4, 2)
func2 = fuzz.gaussmf(x, 6, 2)

minFunc = np.fmin(func1, func2)

plt.plot(x, func1, 'b', linestyle='dotted')
plt.plot(x, func2, 'b', linestyle='dotted')
plt.plot(x, minFunc, 'b')
plt.title('Logical or min function')
plt.xlabel('X')
plt.ylabel('Membership value')
plt.grid(True)
plt.show()
```



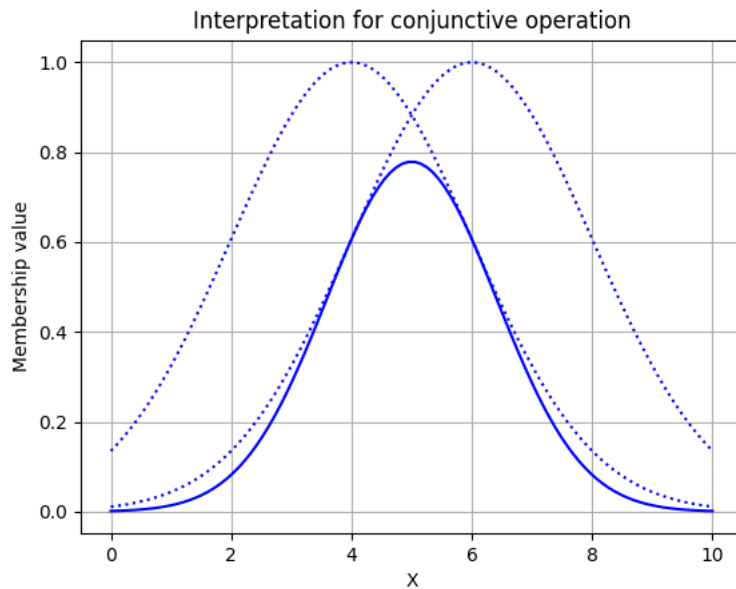
```
x = np.linspace(0, 10, 100)
func1 = fuzz.gaussmf(x, 4, 2)
func2 = fuzz.gaussmf(x, 6, 2)
maxFunc = np.fmax(func1, func2)

plt.plot(x, func1, 'b', linestyle='dotted')
plt.plot(x, func2, 'b', linestyle='dotted')
```



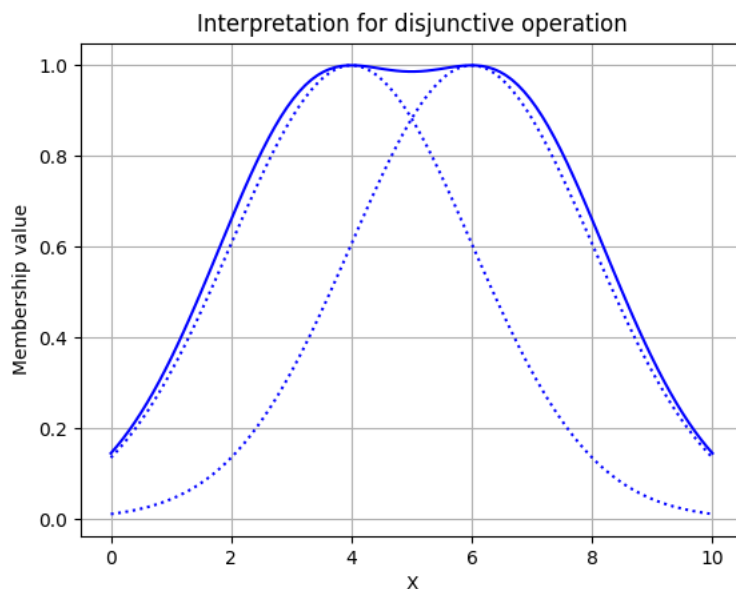
```
plt.plot(x, maxFunc, 'b')
plt.title('Logical or max function')
plt.xlabel('X')
plt.ylabel('Membership value')
plt.grid(True)
plt.show()
```

7. Побудувати вірогідну інтерпретацію кон'юнктивну і диз'юнктивних операторів.



```
x = np.linspace(0, 10, 100)
func1 = fuzz.gaussmf(x, 4, 2)
func2 = fuzz.gaussmf(x, 6, 2)
production = func1 * func2

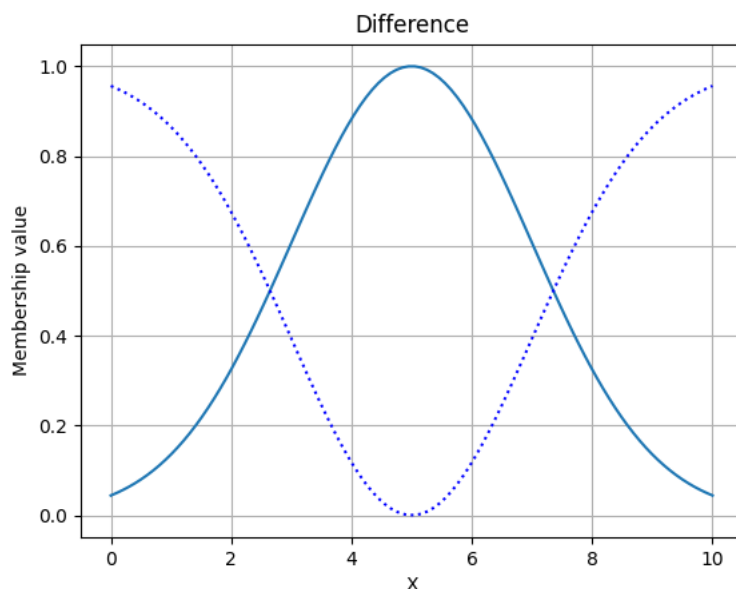
plt.plot(x, func1, 'b', linestyle='dotted')
plt.plot(x, func2, 'b', linestyle='dotted')
plt.plot(x, production, 'b')
plt.title('Interpretation for conjunctive operation')
plt.xlabel('X')
plt.ylabel('Membership value')
plt.grid(True)
plt.show()
```



```
x = np.linspace(0, 10, 100)
func1 = fuzz.gaussmf(x, 4, 2)
func2 = fuzz.gaussmf(x, 6, 2)
algSum = func1 + func2 - func1 * func2

plt.plot(x, func1, 'b', linestyle='dotted')
plt.plot(x, func2, 'b', linestyle='dotted')
plt.plot(x, algSum, 'b')
plt.title('Interpretation for disjunctive operation')
plt.xlabel('X')
plt.ylabel('Membership value')
plt.grid(True)
plt.show()
```

8. Побудувати доповнення нечіткої множини, яке описує деяке розмите судження і представляє собою математичний опис вербального вираження, який заперечує це нечітка множина.



```
x = np.linspace(0, 10, 100)
func = fuzz.gaussmf(x, 5, 2)

plt.plot(x, func)
plt.plot(x, 1 - func, 'b', linestyle='dotted')
plt.title('Difference')
plt.xlabel('X')
plt.ylabel('Membership value')
plt.grid(True)
plt.show()
```

Висновки:

Під час виконання лабораторної роботи було набуто нові знання про різні типи функцій приналежності, такі як трикутні, трапецієподібні, функції Гаусса, сігмоїдні та інші. Також було виконано їх побудову за допомогою мови програмування python, так бібліотек numpy, matplotlib.pyplot та skfuzzy. Крім того, вивчив основні логічні операції, які можна застосовувати до нечітких множин, такі як об'єднання, перетин та доповнення.