

Міністерство освіти та науки України
Львівський національний університет імені Івана Франка

Факультет електроніки та
Комп'ютерних технологій

Звіт

Про виконання лабораторної роботи №4
“Композиція нечітких бінарних відношень”

Виконав:
Студент групи ФеІ-44
Сапанюк М.І.
Перевірила:
Притула М.

Львів 2022

Мета:

ознайомитися з основними операціями над нечіткими відношеннями, застосувати операцію композиції бінарних нечітких відношень у практичних задачах.

Хід роботи:

1. Застосувати операцію композиції бінарних нечітких відношень у задачі, пов'язаній з консалтингом в області вибору професії, щодо навчання та отримання відповідної спеціальностей. Для цього на основі психо-фізіологічного профілювання спеціальностей, яке визначається нечітким бінарним відношенням $S = \{\mu_S()\}$, заданим на декартовому добутку $X \times Y$, а також психо-фізіологічних характеристик претендентів на ці спеціальності, яке визначаються нечітким бінарним відношенням $Q = \{\mu_Q()\}$, заданим на декартовому добутку $Y \times Z$, визначити оптимальні рекомендації для вибору професій. X – множина спеціальностей: x_1 – менеджер; x_2 – програміст; x_3 – водій; x_4 – секретар. Y – множина характеристик: y_1 – гнучкість мислення; y_2 – уміння швидко приймати рішення; y_3 – концентрація уваги; y_4 – зорова пам'ять; y_5 – витривалість; y_6 – швидкість реакції рухів; y_7 – відповідальність. Z – множина претендентів: z_1 – Андрієнко; z_2 – Василенко; z_3 – Іваненко; z_4 – Дмитренко; z_5 – Петренко; z_6 – Романенко.
2. На основі інформації експертів задати нечіткі відношення $S = \{\mu_S()\}$ і $Q = \{\mu_Q()\}$ у вигляді матриці профілювання спеціальностей MS , розміром 4×7 , і матриці профілювання претендентів MQ , розміром 7×6 .

S							
	'y1'	'y2'	'y3'	'y4'	'y5'	'y6'	'y7'
x1	0.55	1.0	0.7	0.4	0.25	0.1	0.85
x2	1.0	0.55	0.85	0.4	0.25	0.1	0.7
x3	0.5	0.85	0.5	0.55	0.7	1.0	0.4
x4	0.4	0.7	0.85	0.55	0.6	0.1	1.0
Q							
	'z1'	'z2'	'z3'	'z4'	'z5'	'z6'	
y1	0.8	0.23	0.55	0.76	0.2	0.11	
y2	0.06	0.97	0.2	0.64	0.27	0.28	
y3	0.1	0.59	0.54	0.61	0.24	0.96	
y4	0.43	0.55	0.59	0.22	0.29	0.44	
y5	0.85	0.73	0.94	0.96	0.44	0.86	
y6	0.52	0.61	0.53	0.78	0.81	0.48	
y7	0.34	0.73	0.98	0.79	0.44	0.99	

3. Запрограмувати алгоритм, який за допомогою операції композиції нечітких відношень $S \circ Q$ знаходить переваги і недоліки претендентів на кожну зі спеціальностей

```

Composition
['z1', 'z2', 'z3', 'z4', 'z5', 'z6']
y1 [0.55, 0.97, 0.85, 0.79, 0.44, 0.85]
y2 [0.8, 0.7, 0.7, 0.76, 0.44, 0.85]
y3 [0.7, 0.85, 0.7, 0.78, 0.81, 0.7]
y4 [0.6, 0.73, 0.98, 0.79, 0.44, 0.99]

```

4. Визначити переваги і недоліки претендентів на кожну зі спеціальностей за допомогою альтернативної операції композиції бінарних нечітких відношень. Порівняти результати.

```

Alternative Composition
['z1', 'z2', 'z3', 'z4', 'z5', 'z6']
y1 [0.44, 0.44, 0.44, 0.44, 0.44, 0.44, 0.13, 0.97, 0.97, 0.97, 0.97, 0.97, 0.97, 0.3, 0.3, 0.38, 0.38, 0.38, 0.38, 0.83, 0.42, 0.64, 0.64, 0.64, 0.64, 0.64, 0.67, 0.11, 0.27, 0.27, 0.27, 0.27, 0.37, 0.06, 0.28, 0.67, 0.67, 0.67, 0.67, 0.84]
y2 [0.8, 0.8, 0.8, 0.8, 0.8, 0.8, 0.23, 0.53, 0.53, 0.53, 0.53, 0.53, 0.55, 0.55, 0.55, 0.55, 0.55, 0.55, 0.69, 0.76, 0.76, 0.76, 0.76, 0.76, 0.76, 0.2, 0.2, 0.2, 0.2, 0.31, 0.11, 0.15, 0.82, 0.82, 0.82, 0.82, 0.82]
y3 [0.4, 0.4, 0.4, 0.4, 0.59, 0.59, 0.59, 0.12, 0.82, 0.82, 0.82, 0.82, 0.82, 0.28, 0.28, 0.28, 0.32, 0.66, 0.66, 0.66, 0.38, 0.54, 0.54, 0.54, 0.67, 0.78, 0.78, 0.1, 0.23, 0.23, 0.23, 0.31, 0.81, 0.81, 0.06, 0.24, 0.48, 0.48, 0.6, 0.6, 0.6]
y4 [0.32, 0.32, 0.32, 0.32, 0.51, 0.51, 0.51, 0.09, 0.68, 0.68, 0.68, 0.68, 0.68, 0.73, 0.22, 0.22, 0.46, 0.46, 0.56, 0.56, 0.98, 0.3, 0.45, 0.52, 0.52, 0.58, 0.58, 0.79, 0.08, 0.19, 0.2, 0.2, 0.26, 0.26, 0.44, 0.04, 0.2, 0.82, 0.82, 0.82, 0.82, 0.99]

```

Висновок:

Виконавши лабораторну роботу, ознайомився з основними операціями над нечіткими відношеннями, застосував операцію композиції бінарних нечітких відношень у практичних задачах.

Додаток:

```

from random import random as rand
# X
# x1 – менеджер;
# x2 – програміст;
# x3 – водій;
# x4 – секретар.
# Y
# y1 – гнучкість мислення;
# y2 – уміння швидко приймати рішення;
# y3 – концентрація уваги;
# y4 – зорова пам'ять;
# y5 – витривалість;
# y6 – швидкість реакції рухів;
# y7 – відповідальність. -
# Z
# z1 – Андрієнко;
# z2 – Василенко;
# z3 – Іваненко;
# z4 – Дмитренко;
# z5 – Петренко;
# z6 – Романенко.
from typing import List

def printMarrix(matrix: List[List[float]], column_label: List[str], row_label:
List[str]) -> None:
    print(column_label)
    for i in range(0, len(matrix)):
        print(row_label[i], matrix[i])

```

```

def alternative_composition(matrix1: List[List[float]], matrix2:
List[List[float]]):
    answer: List[List[float]] = []
    for i in range(0, len(matrix1)):
        answer.append([])
        for j in range(0, len(matrix2[0])):
            temp: List[float] = []
            for k in range(0, len(matrix1[0])):
                temp.append(matrix1[i][k] * matrix2[k][j])
            # answer[i].append(max(temp))
            answer[i].append(float(format(max(temp), '.2f')))
    return answer

```

```

def composition(matrix1: List[List[float]], matrix2: List[List[float]]):
    answer: List[List[float]] = []
    for i in range(0, len(matrix1)):
        answer.append([])
        for j in range(0, len(matrix2[0])):
            temp: List[float] = []
            for k in range(0, len(matrix1[0])):
                temp.append(min(matrix1[i][k], matrix2[k][j]))
            # answer[i].append(max(temp))
            answer[i].append(float(format(max(temp), '.2f')))
    return answer

```

```

def main() -> int:
    x_labels: List[str] = ['x1', 'x2', 'x3', 'x4']
    y_labels: List[str] = ['y1', 'y2', 'y3', 'y4', 'y5', 'y6', 'y7']
    z_labels: List[str] = ['z1', 'z2', 'z3', 'z4', 'z5', 'z6']
    # y1 y2 y3 y4 y5 y6 y7
    S: List[List[float]] = [[0.55, 1.00, 0.70, 0.40, 0.25, 0.10, 0.85], # x1
                            [1.00, 0.55, 0.85, 0.40, 0.25, 0.10, 0.70], # x2
                            [0.50, 0.85, 0.50, 0.55, 0.70, 1.00, 0.40], # x3
                            [0.40, 0.70, 0.85, 0.55, 0.60, 0.10, 1.00]] # x4
    ## z1 z2 z3 z4 z5 z6
    # Q: list[list[float]] = [[0.00, 0.00, 0.00, 0.00, 0.00, 0.00], #y1
    # [0.00, 0.00, 0.00, 0.00, 0.00, 0.00], #y2
    # [0.00, 0.00, 0.00, 0.00, 0.00, 0.00], #y3
    # [0.00, 0.00, 0.00, 0.00, 0.00, 0.00], #y4
    # [0.00, 0.00, 0.00, 0.00, 0.00, 0.00], #y5
    # [0.00, 0.00, 0.00, 0.00, 0.00, 0.00], #y6
    # [0.00, 0.00, 0.00, 0.00, 0.00, 0.00]] #y7
    Q: List[List[float]] = [[float(format(rand(), '.2f')) for i in range(0, 6)]
                             for j in range(0, 7)]
    print("S")
    printMarrix(S, y_labels, x_labels)
    print("Q")
    printMarrix(Q, z_labels, y_labels)
    print("Composition")
    printMarrix(composition(S, Q), z_labels, y_labels)
    print("Alternative Composition")
    printMarrix(alternative_composition(S, Q), z_labels, y_labels)
    composition(S, Q)
    # printMarrix(S, z_labels, x_labels)
    return 0

```

```

if __name__ == '__main__':
    main()

```