# **КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ імені**

# **ТАРАСА ШЕВЧЕНКА**



## **ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ**

**Кафедра прикладних інформаційних систем**

**Звіт до лабораторної роботи №1**

# **з курсу**

**«Системний аналіз та теорія прийняття рішень»**

*Студента 3 курсу*

*групи ПП-31 спеціальності 122 «Комп'ютерні науки» ОП «Прикладне програмування»*

Момотюка Михайла Тарасовича

*Викладач:*

Плескач В.Л.

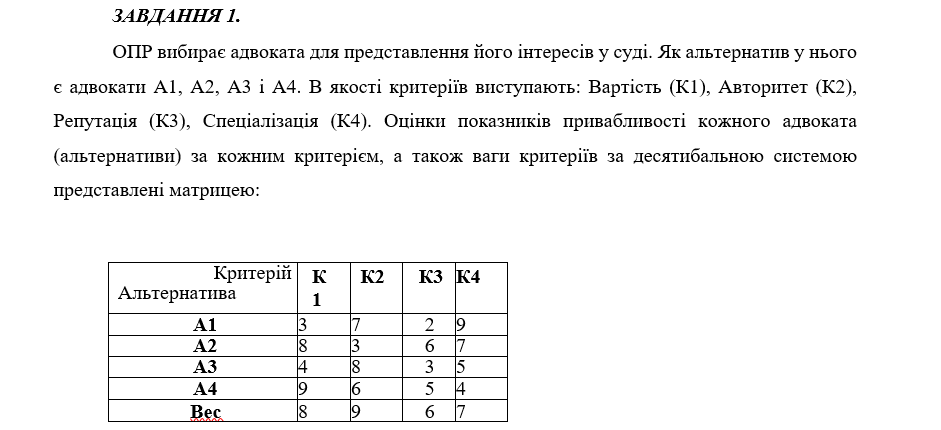
Білий Р.О

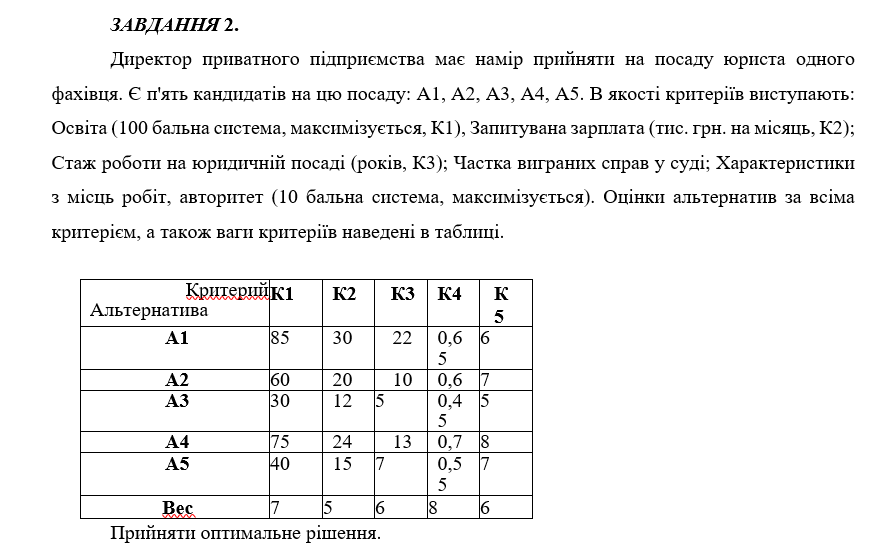
## **Київ – 2023**

**Тема роботи:** Прийняття рішення в умовах повної визначеності.

**Мета роботи:** Дослідити методи прийняття рішення в умовах повної визначеності.

**Завдання:**

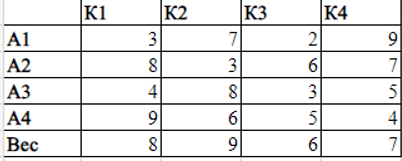




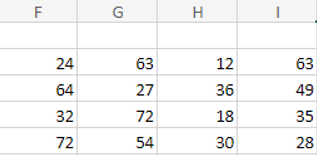
**Хід роботи**

Завдання 1:

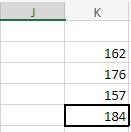
Вводимо таблицю в Excel



Розраховуємо оцінки за допомогою формул

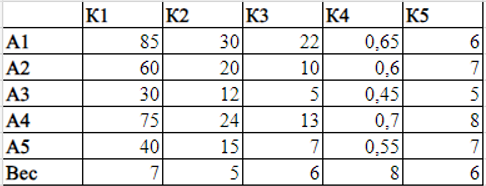


Розраховуємо функцію корисності, максимальне значення 184 у альтернативи A4, отже найкраще буде обрати саме його як адвоката.

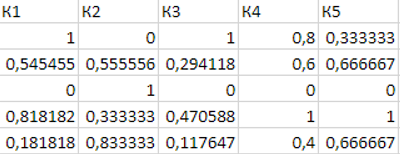


Завдання 2:

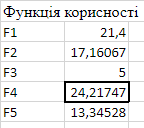
Вводимо таблицю в Excel



Розраховуємо оцінки за допомогою формул



Розраховуємо функцію корисності, максимальне значення 24,21 у альтернативи A4, отже обираємо саме цього кандидата

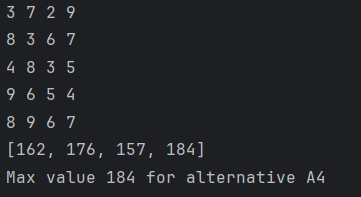


**Лістинг програми:**

Завдання 1:

file = open('task1.txt', 'r')  
  
file\_data = []  
  
for row in file:  
 file\_data.append([int(num) for num in row.split() if num.isdigit()])  
  
formatted\_file\_data = '\n'.join([' '.join(map(str, row)) for row in file\_data])  
print(formatted\_file\_data)  
  
results = []  
  
for i in range(len(file\_data) - 1):  
 sum\_val = sum(file\_data[i][j] \* file\_data[len(file\_data) - 1][j] for j in range(len(file\_data[0])))  
 results.append(sum\_val)  
  
max\_result = max(results)  
max\_result\_position = results.index(max\_result) + 1  
  
print(results)  
print(f'Max value {max\_result} for alternative A{max\_result\_position}')

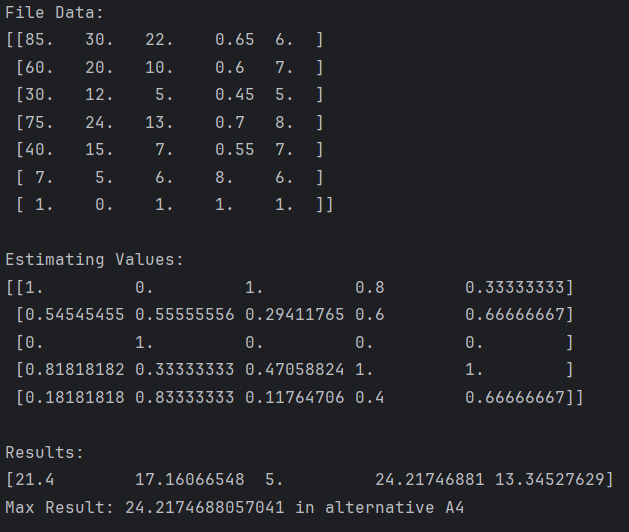
Результат:



Завдання 2:

import numpy as np  
  
file\_data = np.loadtxt('task2.txt')  
  
print("File Data:")  
print(file\_data)  
  
estimating\_values = np.where(file\_data[-1, None, :],  
 (file\_data[:-2, :] - np.min(file\_data[:-2, :], axis=0)) /  
 (np.max(file\_data[:-2, :], axis=0) - np.min(file\_data[:-2, :], axis=0)),  
 (np.max(file\_data[:-2, :], axis=0) - file\_data[:-2, :]) /  
 (np.max(file\_data[:-2, :], axis=0) - np.min(file\_data[:-2, :], axis=0)))  
  
print("\nEstimating Values:")  
print(estimating\_values)  
  
results = np.dot(estimating\_values, file\_data[-2, :])  
print("\nResults:")  
print(results)  
  
max\_result\_position = np.argmax(results) + 1  
max\_result = results[max\_result\_position - 1]  
print(f'Max Result: {max\_result} in alternative A{max\_result\_position}')

Результат:



**Висновок:** У ході лабораторної роботи я розглянув прийняття рішення в умовах повної визначеності з використанням Ехсеl та реалізував прийняття рішення програмно на мові Python.