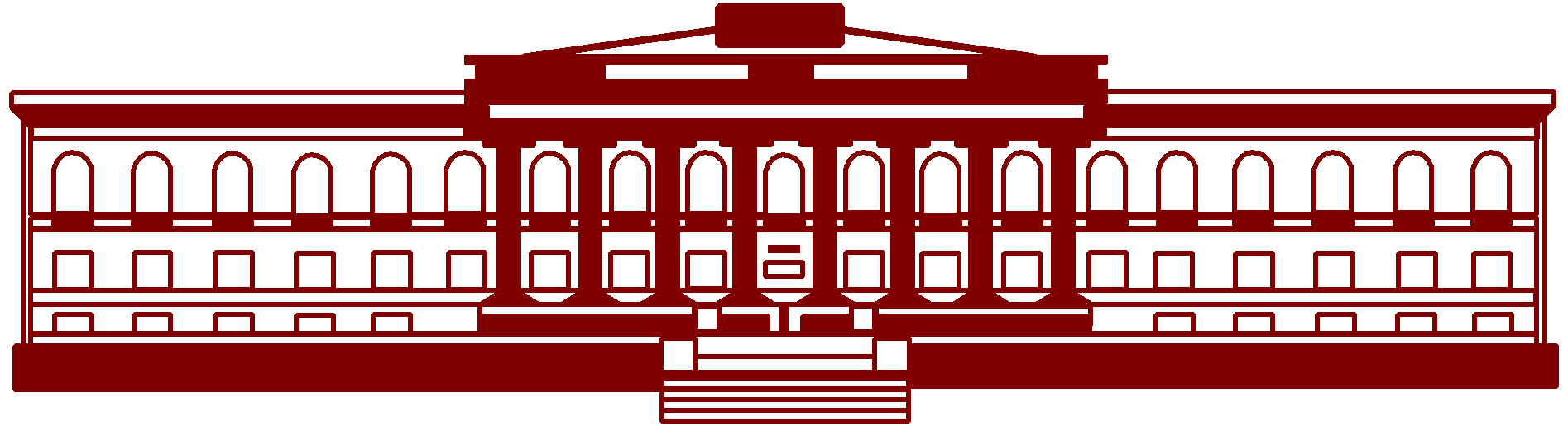
**КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ**

**імені ТАРАСА ШЕВЧЕНКА**

****

**ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ**

**Кафедра прикладних інформаційних систем**

**Звіт до лабораторної роботи №2**

**з курсу**

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

*Студентa 3 курсу*

*групи ПП-32*

*спеціальності 122 «Комп'ютерні науки»*

*ОП «Прикладне програмування»*

*Кондратова Івана Андрійовича*

*Викладач:*

Плескач В.Л.

**Київ – 2023**

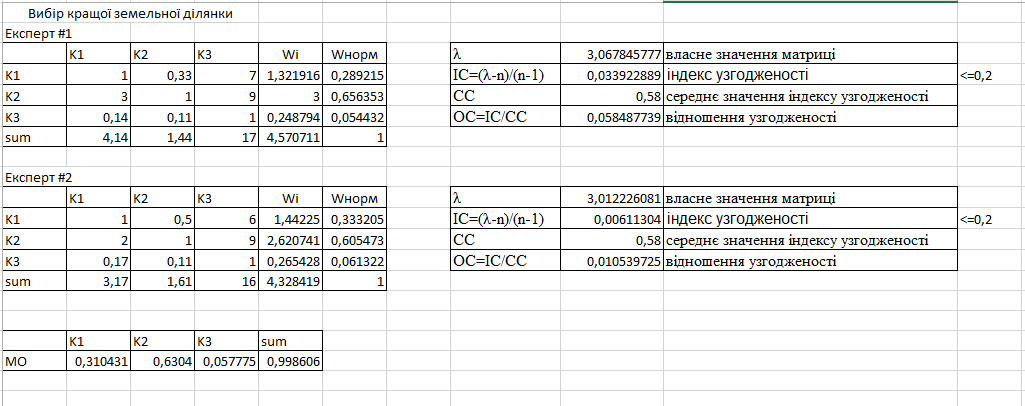
**Тема:** Метод аналізу ієрархій

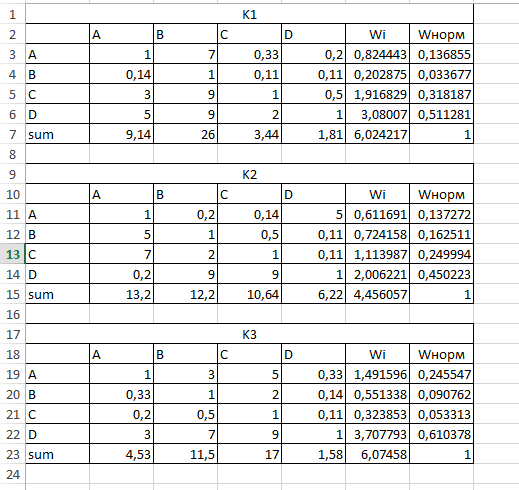
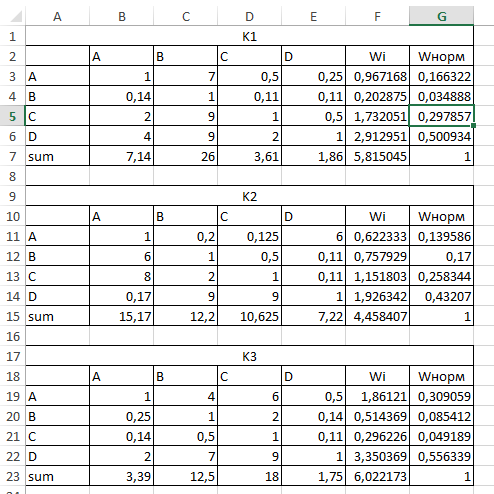
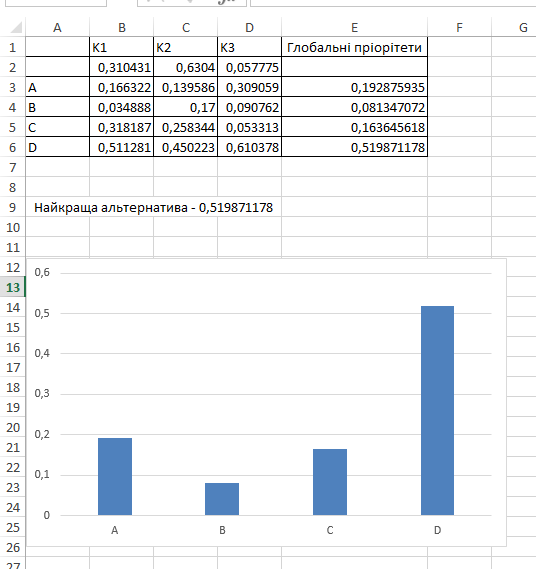
**Мета:** закріпити теоретичні знання та отримати практичні навички здійснення експертної оцінки пріоритетності альтернатив методом аналізу ієрархії Сааті.

**Завдання**

Підприємець, що займається продажем професійного обладнання для перукарських і косметичних салонів вирішив відкрити новий торгівельний пункт і побудувати магазин в одному з районів міста. Міська влада пропонує йому під будівництво чотири земельні ділянки: А, В, С і D. Як криерії при виборі місця будівництва підприємець виділяє три: доступність магазину для клієнтів (місце розташування) – K1; вартість будівництва, доступність комунікацій – K2; можливість подальшого розширення (планується з часом прибудувати приміщення для додаткових відділів) – K3.

**Розв’язок excel**



**Розв’язок Python**

import numpy

import pandas

import seaborn

import matplotlib.pyplot

#сргеом

def geomean(arr):

    return numpy.exp(numpy.log(arr).mean())

#Wi

def priority\_makrs\_avg(matrix):

    result = []

    for array in matrix:

        result.append(geomean(array))

    return result

#Wнорм

def priority\_vector(arr):

    result = []

    sum = numpy.sum(arr)

    for w in arr:

        result.append(w/sum)

    return result

def get\_from\_excel(path,sheet\_name,skiprows,usecols,nrows):

    df = pandas.read\_excel(path, sheet\_name=sheet\_name,header=None, skiprows=skiprows, usecols=usecols, nrows=nrows)

    matrix = df.to\_numpy()

    return matrix

def main():

    first\_matrix = get\_from\_excel(path='D:\SATPR\lab2\SATPR2.xlsx', sheet\_name='1',skiprows=3, usecols='B:D', nrows=3)

    second\_matrix = get\_from\_excel(path='D:\SATPR\lab2\SATPR2.xlsx', sheet\_name='1',skiprows=10, usecols='B:D', nrows=3)

    first\_k1\_matrix = get\_from\_excel(path='D:\SATPR\lab2\SATPR2.xlsx', sheet\_name='2', skiprows=2, usecols='B:E', nrows=4)

    first\_k2\_matrix = get\_from\_excel(path='D:\SATPR\lab2\SATPR2.xlsx', sheet\_name='2', skiprows=10, usecols='B:E', nrows=4)

    first\_k3\_matrix = get\_from\_excel(path='D:\SATPR\lab2\SATPR2.xlsx', sheet\_name='2', skiprows=18, usecols='B:E', nrows=4)

    second\_k1\_matrix = get\_from\_excel(path='D:\SATPR\lab2\SATPR2.xlsx', sheet\_name='3', skiprows=2, usecols='B:E', nrows=4)

    second\_k2\_matrix = get\_from\_excel(path='D:\SATPR\lab2\SATPR2.xlsx', sheet\_name='3', skiprows=10, usecols='B:E', nrows=4)

    second\_k3\_matrix = get\_from\_excel(path='D:\SATPR\lab2\SATPR2.xlsx', sheet\_name='3', skiprows=18, usecols='B:E', nrows=4)

    first\_vector = priority\_vector(priority\_makrs\_avg(first\_matrix))

    second\_vector = priority\_vector(priority\_makrs\_avg(second\_matrix))

    first\_step\_result = []

    for i in range(len(first\_vector)):

        first\_step\_result.append(geomean([first\_vector[i], second\_vector[i]]))

    first\_k1\_vector = priority\_vector(priority\_makrs\_avg(first\_k1\_matrix))

    first\_k2\_vector = priority\_vector(priority\_makrs\_avg(first\_k2\_matrix))

    first\_k3\_vector = priority\_vector(priority\_makrs\_avg(first\_k3\_matrix))

    second\_k1\_vector = priority\_vector(priority\_makrs\_avg(second\_k1\_matrix))

    second\_k2\_vector = priority\_vector(priority\_makrs\_avg(second\_k2\_matrix))

    second\_k3\_vector = priority\_vector(priority\_makrs\_avg(second\_k3\_matrix))

    k1\_result = []

    k2\_result = []

    k3\_result = []

    for i in range(len(first\_k1\_vector)):

        k1\_result.append(numpy.max([first\_k1\_vector[i],second\_k1\_vector[i]]))

        k2\_result.append(numpy.max([first\_k2\_vector[i],second\_k2\_vector[i]]))

        k3\_result.append(numpy.max([first\_k3\_vector[i],second\_k3\_vector[i]]))

    final\_result = []

    for i in range(len(k1\_result)):

        final\_result.append(geomean([k1\_result[i],k2\_result[i], k3\_result[i]]))

    data = pandas.DataFrame([final\_result], columns = ['A','B','C','D'])

    print(data)

    matplotlib.pyplot.figure()

    diagram = seaborn.barplot(data=data)

    diagram.bar\_label(diagram.containers[0], fontsize=12);

    matplotlib.pyplot.show()

if \_\_name\_\_ == '\_\_main\_\_':

    main()

**Висновок:** виконуючи цю лабораторну роботу, я закріпив теоретичні знання та отримав практичні навички здійснення експертної оцінки пріоритетності альтернатив методом аналізу ієрархії Сааті.