# **КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ імені**

# **ТАРАСА ШЕВЧЕНКА**



## **ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ**

**Кафедра прикладних інформаційних систем**

**Звіт до лабораторної роботи №6**

# **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.

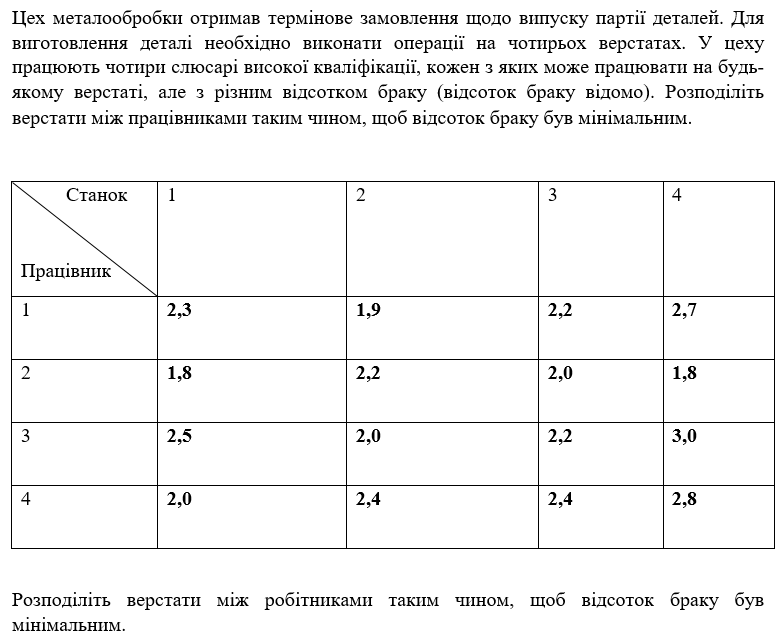
Білий Р.О.

## **Київ – 2023**

**Тема роботи:** Постановка задачі про призначення.

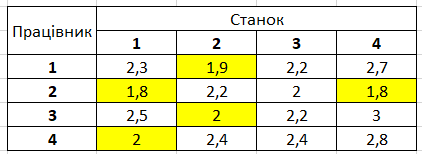
**Мета роботи:** Навчитися складати моделі та вирішувати задачі про призначення.

**Завдання:**

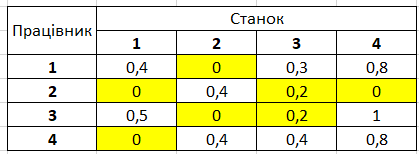


**Хід роботи**

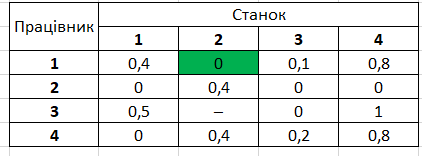
Для матриці витрат визначаємо у кожному рядку найменше значення та віднімаємо від кожного елемента рядка це значення:



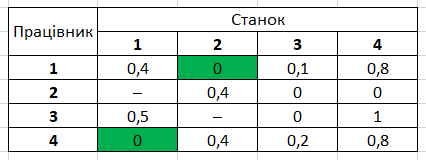
Повторюємо те саме з стовпцями:



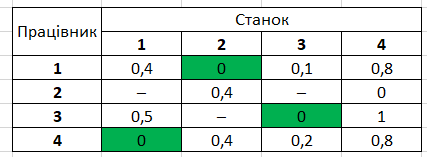
Визначаємо рядок, де один нуль (1 рядок) та всі нулі і цьому стовпці заміняємо прочерком (–)



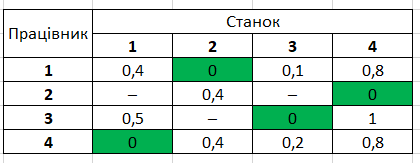
Визначаємо рядок, де один нуль (4 рядок) та всі нулі і цьому стовпці заміняємо прочерком (–)



Визначаємо рядок, де один нуль (3 рядок) та всі нулі і цьому стовпці заміняємо прочерком (–)



Визначаємо рядок, де один нуль (2 рядок) та всі нулі і цьому стовпці заміняємо прочерком (–)



Тоді найменше браку буде, коли:

Працівник 1 працює за станком 2

Працівник 2 працює за станком 4

Працівник 3 працює за станком 3

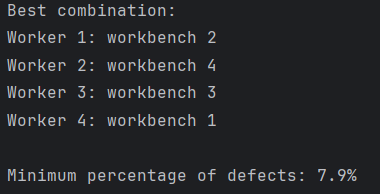
Працівник 4 працює за станком 1

Мінімальний відсоток браку: 1.9+1.8+2.2+2=7.9 (%)

**Лістинг програми:**

import numpy as np  
from scipy.optimize import linear\_sum\_assignment  
import openpyxl  
  
  
def get\_matrix\_from\_sheet(sheet):  
 matrix = [row for row in sheet.iter\_rows(values\_only=True)]  
 return np.array(matrix)  
  
  
def normalize\_rows\_and\_columns(matrix):  
 matrix -= np.min(matrix, axis=0)  
 matrix -= np.min(matrix, axis=1)[:, np.newaxis]  
  
  
def print\_result(row\_ids, col\_ids, min\_defect\_percentage):  
 print('Best combination:')  
 for worker, workbench in zip(row\_ids, col\_ids):  
 print(f'Worker {worker + 1}: workbench {workbench + 1}')  
  
 print(f'\nMinimum percentage of defects: {min\_defect\_percentage:.1f}%')  
  
  
file\_path = 'lab6.xlsx'  
matrix = get\_matrix\_from\_sheet(openpyxl.load\_workbook(file\_path).worksheets[0])  
  
row\_ids, col\_ids = linear\_sum\_assignment(matrix)  
min\_defect\_percentage = matrix[row\_ids, col\_ids].sum()  
print\_result(row\_ids, col\_ids, min\_defect\_percentage)

Результат:



**Висновок:** У ході лабораторної роботи я навчився складати моделі та вирішувати задачі про призначення та реалізувати подібні задачі програмно.