|  |  |  |  |
| --- | --- | --- | --- |
| МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ імені Тараса Шевченка  ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ  **Кафедра програмних систем і технологій**  Дисципліна  **«Ймовірнісні основи програмної інженерії»**  **Лабораторна робота № 2**  **на тему:**  **«** **Лінійне перетворення та Графічне зображення даних»** | | | |
| **Виконала:** | Обруснік Тетяна Віталіївна | **Перевірила**: | Марцафей Анна Сергіївна |
| Група | ІПЗ-24/2 | Дата перевірки |  |
| Форма навчання | денна | Оцінка |  |
| Спеціальність | 121 |
| 2022 | | | |

**Мета –** навчитись використовувати на практиці набуті знання про лінійні перетворення та графічне зображення даних.

**Завдання**

1. Знайдіть Q1 Q3, та P90.

2. Знайдіть середнє та стандартне відхилення цих оцінок.

3. Через незадоволення низькими оцінками викладач вирішив використати

шкалу форми y = ax + b, щоб відредагувати оцінки. Він хотів, щоб середнє

значення масштабних оцінок становило 95, а оцінка 100, щоб залишалася

рівною 100.

4. Показати дані за допомогою діаграми "стовбур – листя".

5. Відобразити дані за допомогою коробкового графіка.

6. Зробити висновок.

**Вимоги до програмного забезпечення**

• Розробляти програму можна на одній з наступних мов

програмування: C/C++ (версія C++11), C# (версія C# 5.0), Java (версія Java SE 8), Python

(версія 2.7).

• Програма повинна розміщуватись в окремому вихідному файлі, без ви

користання додаткових нестандартних зовнішніх модулів.

• Не дозволяється використовувати будь-

які нестандартні бібліотеки та розширення. Програма не повинна залежати від

операційної системи.

• Не реалізуйте жодного інтерфейсу користувача (окрім командного рядку).

Програма не повинна запитувати через пристрій вводу в користувача жодної додаткової

інформації. Вашу програму будуть використовувати виключно у вигляді “чорного

ящику”.

• Розроблена програма повинна зчитувати з командного рядку назву вхідного

файлу та записувати результат у вихідний файл. При запуску першим і єдиним

аргументом командного рядку повинна бути назва вхідного файлу (наприклад,

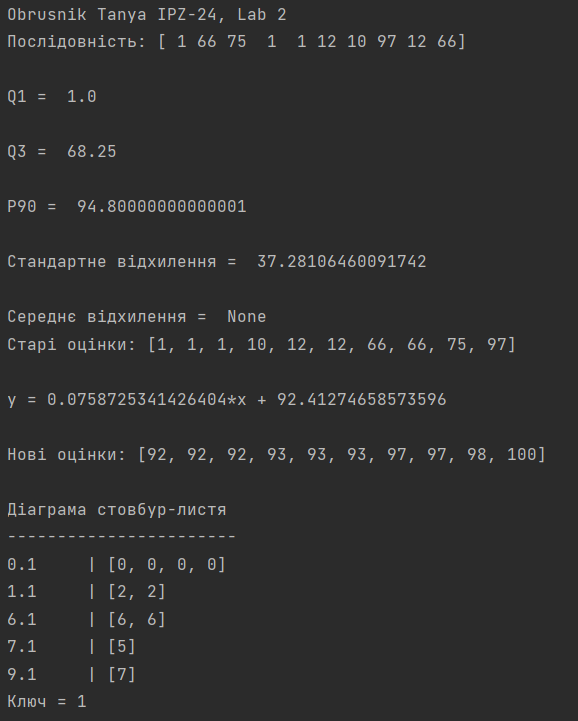
input\_10.txt).

**Псевдокод алгоритму**

import math  
import matplotlib.pyplot as plt  
import numpy as np  
from scipy.linalg import solve  
  
print("Obrusnik Tanya IPZ-24, Lab 2")  
  
def Task1(pindex):  
 index = pindex \* (count + 1) - 1  
 percentile = data[int(index)] + (index % int(index)) \* (data[int(index) + 1] - data[int(index)])  
 return percentile  
  
# ------------------------Task2--------------------------------  
def StandartDeviation():  
 sum = 0  
 totalSum = 0  
 for i in range(len(data)):  
 sum += data[i]  
 midleX = sum / len(data)  
  
 for i in range(len(data)):  
 totalSum += (data[i] - midleX) \*\* 2  
 # квадрат різниці, загальна сума  
 result = np.sqrt(totalSum / (len(data) - 1))  
  
 f.write("\nСтандартне відхилення = " + str(result))  
  
 return result  
  
def MiddleDeviation():  
 sum = 0  
 totalSum = 0  
 for i in range(len(data)):  
 sum += data[i]  
 midleX = sum / len(data)  
 for i in range(len(data)):  
 totalSum += abs(data[i] - midleX)  
 # модуль різниці, загальна сума  
 result = (totalSum / (len(data)))  
  
 f.write("\nСереднє відхилення = " + str(result))  
# -------------------------------------------------------------  
  
def Task3():  
  
 sum = 0  
 result = []  
 for i in data:  
 sum += i  
 a = np.array([[100, 1, ], [(sum / count), 1, ]])  
 # |100 = 100\*a + b  
 # |95 = 74.2\*a + b  
 x = solve(a, np.array([100, 95]))  
 for i in range(count):  
 result.append(round(x[0] \* data[i] + x[1]))  
 print("Старі оцінки: " + str(data))  
 f.write("\nСтарі оцінки: " + str(data))  
  
 print("\ny = " + str(x[0]) + "\*x + " + str(x[1]))  
 f.write("\ny = " + str(x[0]) + "\*x + " + str(x[1]))  
  
 print("\nНові оцінки: " + str(result))  
 f.write("\nНові оцінки: " + str(result))  
  
  
def Task4(i):  
 print("\nДіаграма стовбур-листя")  
 print("-----------------------")  
  
 f.write("\nДіаграма стовбур-листя")  
 f.write("-----------------------")  
  
 i = min(data)  
  
 while i <= max(data):  
 mas = []  
 for j in range(len(data)):  
 if i < data[j] < i + 10:  
 mas.append(data[j] % 10)  
 elif data[j] == i:  
 mas.append(0)  
 if len(mas) != 0:  
 print(str(i / 10) + " \t| " + str(mas))  
 f.write(str(i / 10) + " \t| " + str(mas))  
 i += 10  
 print("Ключ = " + str(data[0]))  
 f.write("Ключ = " + str(data[0]))  
  
# ------------------------Task5---------------------------  
def BoxDiagram():  
 plt.boxplot(data)  
 plt.grid()  
 plt.show()  
plt.title('Task.5 Box Diagram')  
# --------------------------------------------------------  
f = open("answer.txt", "w")  
data = []  
for i in open("input\_10.txt"):  
 data.append(int(i.strip()))  
data = np.delete(data, 0)  
  
print("Послідовність:", data)  
f.write("Послідовність:" + str(data))  
  
data = sorted(data)  
count = len(data)  
  
Q1 = Task1(1 / 4)  
Q3 = Task1(3 / 4)  
P90= Task1(0.9)  
  
print("\nQ1 = ", Q1)  
print("\nQ3 = ", Q3)  
print("\nP90 = ", P90)  
  
f.write("\nQ1 = ")  
f.write(str(Q1))  
f.write("\nQ3 = ")  
f.write(str(Q3))  
f.write("\nP90 = ")  
f.write(str(P90))  
  
print("\nСтандартне відхилення = ", StandartDeviation())  
print("\nСереднє відхилення = ", MiddleDeviation())  
  
Task3()  
Task4(min(data))  
f.close()  
BoxDiagram()

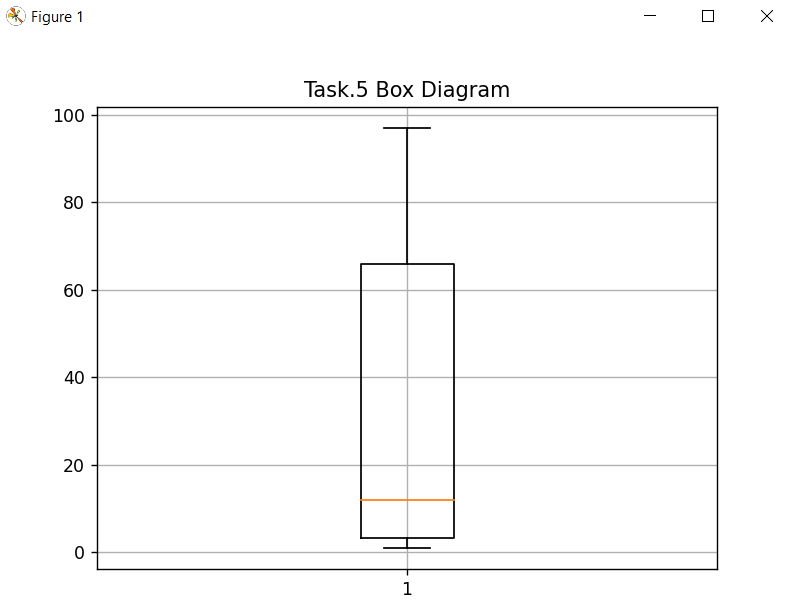
**Результат**

Під час компіляції программа видає наступний результат:



**Коробковий графік**

Коробковий графік або боксплот — засіб візуалізації в описовій статистиці груп числових даних через їх квантилі.



**Висновок:** під час виконання даної лабораторної роботи було використано на практиці набуті знання про лінійні перетворення та графічне зображення даних та побудовано коробковий графік .