МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Національний аерокосмічний університет ім. М.Є. Жуковського

“Харківський авіаційний інститут”

Кафедра комп’ютерних систем, мереж і кібербезпеки

Титульный лист заполняется на украинском языке

Остальная часть отчета может быть как на украинском, так и на русском

Лабораторна робота № 4

з дисципліни “Обробка даних засобами Python”

Дослідження статистичних методів в Python

Виконав студент гр. 555іМБ Литвинов.О.А Фамилия И.О.

(підпис, дата)

Перевірив к.т.н зав.кафедри каф. 301

(науковий ступінь, вчене звання, посада)

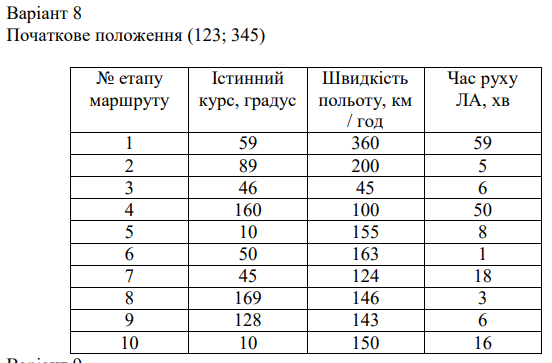
Дергачов.К.Ю

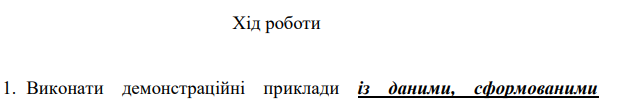
(підпис, дата) (П.І.Б.)

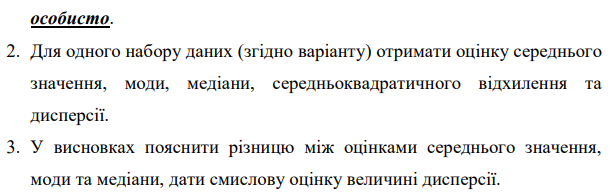
2023

Мета роботи: набути практичний досвід використання функцій, які надає вбудована бібліотека statistics. Також, відпрацювати вміння працювати з цими інструментами, використовуючи довільний набір даних згідно варіанту завдання.

Варіант 8:







Код завдання 1:

import statistics

import random

from fractions import Fraction as F

from decimal import Decimal as D

print("TASK 1")

print('statistics.mean([34, 12, 65, 15]) = '

      f'{statistics.mean([34, 12, 65, 15])}')

print('statistics.mean([F(10, 15), F(47, 93), F(5, 1)]) = '

      f'{statistics.mean([F(10, 15), F(47, 93), F(5, 1)])}')

print('statistics.mean([D("3.4"), D("1.2"), D("6.5"), D("1.5")]) = '

      f'{statistics.mean([D("3.4"), D("1.2"), D("6.5"), D("1.5")])}')

print(f'{"-" \* 100}')

print('statistics.mean([random.randint(1, 100) for x in range(1, 1001)]) = '

      f'{statistics.mean([random.randint(1, 100) for x in range(1, 1001)])}')

print(

    'statistics.mean([random.triangular(1, 100, 70) for x in range(1, 1001)]) = '

    f'{statistics.mean([random.triangular(1, 100, 70) for x in range(1, 1001)])}'

)

print(f'{"-" \* 100}')

print('statistics.mode([random.randint(1, 100) for x in range(1, 1001)]) = '

      f'{statistics.mode([random.randint(1, 100) for x in range(1, 1001)])}')

print('statistics.mode([random.randint(1, 100) for x in range(1, 1001)]) = '

      f'{statistics.mode([random.randint(1, 100) for x in range(1, 1001)])}')

print('statistics.mode([random.randint(1, 100) for x in range(1, 1001)]) = '

      f'{statistics.mode([random.randint(1, 100) for x in range(1, 1001)])}')

print('statistics.mode(["F-35", "F-16", "SU-27", "F-35", "Cessna 172"]) = '

      f'{statistics.mode(["F-35", "F-16", "SU-27", "F-35", "Cessna 172"])}')

print(f'{"-" \* 100}')

print('statistics.median([random.randint(1, 100) for x in range(1, 50)]) = '

      f'{statistics.median([random.randint(1, 100) for x in range(1, 51)])}')

print(

    'statistics.median\_grouped([random.randint(1, 100) for x in range(1, 50)]) = '

    f'{statistics.median\_grouped([random.randint(1, 100) for x in range(1, 51)])}'

)

print(

    'statistics.median\_high([random.randint(1, 100) for x in range(1, 50)]) = '

    f'{statistics.median\_high([random.randint(1, 100) for x in range(1, 51)])}'

)

print(

    'statistics.median\_low([random.randint(1, 100) for x in range(1, 50)]) = '

    f'{statistics.median\_low([random.randint(1, 100) for x in range(1, 51)])}')

print(f'{"-" \* 100}')

data = [1, 2, 3, 4, 5, 6, 7, 8, 9]

print(f'data = {data}')

print('statistics.pvariance(data) = '

      f'{statistics.pvariance(data)}')

print('statistics.pstdev(data) = '

      f'{statistics.pstdev(data)}')

print('statistics.variance(data) = '

      f'{statistics.variance(data)}')

more\_data = [3, 4, 5, 5, 5, 5, 5, 6, 6]

print(f'more\_data = {more\_data}')

print('statistics.pvariance(more\_data) = '

      f'{statistics.pvariance(more\_data)}')

print('statistics.pstdev(more\_data) = '

      f'{statistics.pstdev(more\_data)}')

print('statistics.variance(more\_data) = '

      f'{statistics.variance(more\_data)}')

some\_fractions = [F(5, 6), F(2, 7), F(11, 76)]

print(f'some\_fractions = {some\_fractions}')

print('statistics.pvariance(some\_fractions) = '

      f'{statistics.pvariance(some\_fractions)}')

print('statistics.pstdev(some\_fractions) = '

      f'{statistics.pstdev(some\_fractions)}')

print('statistics.variance(some\_fractions) = '

      f'{statistics.variance(some\_fractions)}')

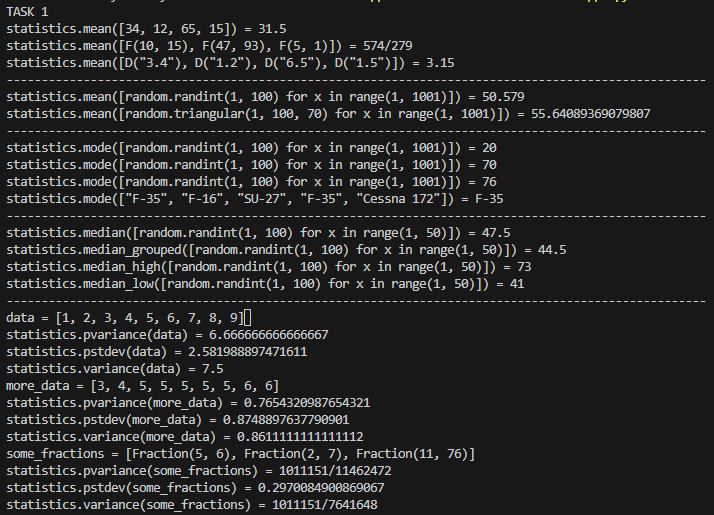


Рисунок 1 – Завдання 1

Код завдання 2:

import statistics

class Colors:

    DEFAULT = '\033[0m'

    CHANGED = '\033[95m'

def statistics\_calc(data):

    print(f'{Colors.DEFAULT}Mean = {statistics.mean(data)}')

    print(f'Mode = {statistics.mode(data)}')

    print(f'Median = {statistics.median(data)}')

    print(f'St\_deviation = {statistics.stdev(data)}')

    print(f'Variance = {statistics.variance(data)}')

print(f'{Colors.CHANGED}Degree:')

statistics\_calc([59, 89, 46, 160, 10, 50, 45, 169, 128, 10])

print(f'{Colors.CHANGED}Velocity:')

statistics\_calc([360, 25, 158, 200, 160, 63, 240, 58, 65, 50])

print(f'{Colors.CHANGED}Time:')

statistics\_calc([5, 4, 18, 3, 6, 16, 18, 12, 6, 16])

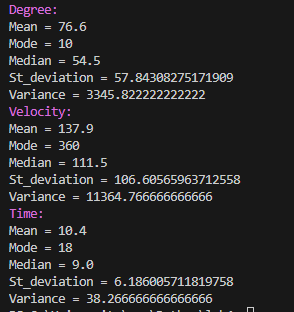


Рисунок 2 – Завдання 2

Висновок:

При виконанні цієї роботи використовувалися стандартні методи бібліотеки statistics, а точніше:

1. **Середнє значення(mean):** Це статистична міра, що представляє собою середнє арифметичне всіх значень в наборі. Воно чутливе до викидів, тобто значень, які суттєво відрізняються від інших. Якщо у нас, наприклад, є набір даних про доходи, великі викиди можуть значно вплинути на середнє значення, роблячи його не представницьким для всього набору.
2. **Мода(mode):** Це значення чи значення, які зустрічаються найчастіше в наборі. Мода особливо корисна, коли вам цікаво саме те значення, яке найбільше повторюється. Мода може бути корисною для категоріальних даних або в тих випадках, коли ви хочете знати конкретне значення, яке найчастіше зустрічається в наборі.
3. **Медіана(median):** Це значення, яке знаходиться в середині впорядкованого набору даних. Медіана не чутлива до великих викидів, оскільки вона не залежить від конкретних значень, а лише від їх порядку. Вона особливо корисна в тих випадках, коли у вас є великі викиди, і ви хочете отримати репрезентативне значення "середнього" для вашого набору даних.
4. **Дисперсія(variance):** Дисперсія вимірює розсіювання даних відносно їхнього середнього значення. Велика дисперсія свідчить про те, що дані мають тенденцію розташовуватися далеко від середнього значення, що може вказувати на великі варіації в наборі. Маленька дисперсія свідчить про те, що дані мають тенденцію групуватися близько до середнього значення. Оцінка дисперсії допомагає нам розуміти ступінь розподілу даних і визначити, наскільки набір даних є однорідним або різноманітним.

Інакше кажучи ці параметри вказують на різні аспекти вибірки: типове, найчастіше, середнє значення та розкид даних