| 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 № 1** | | | |
| --- | --- | --- | --- |
| **Виконав:** | Круць Михайло Ярославович | **Перевірила**: | Вечерковська Анастасія  Сергіївна |
| Група | ІПЗ-22 | Дата перевірки |  |
| Форма навчання | денна | Оцінка |  |
| Спеціальність | 121 |
| 2022 | | | |

**Назва роботи:** Центральні тенденції та міра дисперсії

**Мета –** навчитись використовувати на практиці набуті знання про центральні тенденції та міри.

**Постановка задачі**

1. На основі вхідних даних побудувати таблицю частот та сукупних частот
2. Знайти Моду та Медіану заданої вибірки
3. Порахувати Дисперсію та Середнє квадратичне відхилення розподілу
4. Побудувати гістограму частот для даного розподілу
5. Зробити висновок з вигляду гістограми про закон розподілу

**Математична модель**

**Задача:** На основі вибірки даних визначити центральні тенденції (Медіана та Мода), Дисперсію та Середнє квадратичне відхилення розподілу та візуалізувати отримані результати (таблиці частот, сукупних частот та гістограму частот для даного розподілу); зробити висновок з гістограми про закон розподілу

**Дано:** Вибірка переглянутих фільмів (, де n - кількість переглядів)

**Потрібно:**

**Побудувати:** таблицю частот (*,*  …, ),

сукупних частот() та гістограму для даного розподілу

**Знайти:** Моду (Mo), Медіану вибірки(Me); Дисперсію (D) та Середнє квадратичне відхилення розподілу (σ)

**Зробити:** висновок з вигляду гістограми про закон розподілу

**Зв’язки: ;**  ;

*Mo* - the item (*x*) with the highest frequency (max())

- (;

**При:**  N (усі x є натуральними числами)

**Метод:** Використання засобів мови програмування Python для зчитування, обрахунків та запису у файл результатів. Детальні інструкції у псевдокоді.

**Псевдокод**

//Головна частина

file = input(“File name:”)

list\_of\_views = read\_file(file) //Виклик функції (1)

set\_of\_movies = sort(set(list\_of\_movies)) //Відсортована множина переглянутих фільмів (без дублювання))

write\_frequency(set\_of\_movies, list\_of\_views) // Виклик функції (2)

frequencies = empty list

for movie in set\_of\_movies:

add count of movie in list\_of\_view to frequencies

end for

mode = find\_the\_mode(set\_of\_movies, frequencies) //Виклик функції (4)

open output file

//Найчастіше переглянутий фільм (фільми) і є елементом (елементами) моди

if (only one item in mode):

write mode[0]

else:

write f“mode[0], …, mode[n]”

write find\_the\_median(set\_of\_movies) // Виклик функції (3)

//Повторний запис моди у файл

if (only one item in mode):

write mode[0]

else:

write f“mode[0], …, mode[n]”

variance = find\_the\_variance(set\_of\_movies, list\_of\_views, frequencies) //Виклик функції(5)

standard\_deviation = sqrt(variance)

write variance and standard\_deviation

close output file

//Функція зчитування даних з вхідного файлу

**(**1) function read\_file (file\_name):

movies\_file = open input file

count\_of\_views = first line

movie = movie on the current line

movies = empty list

while movie is not empty: //(Поки рядки не скінчились):

add int(movie) to movies

movie = read next line

end while

close movies\_file

return movies

//Функція запису таблиці частот у вихідний файл

(2) function write\_frequency(movies, list\_of\_views):

result = open output file

form table headers

write the top of the header cells

write column names

write bottom of the header cells

cumulative\_frequency = 0

for movie in movies: //перебір фільмів

number = current movie

frequency = count movie in list\_of\_views

cumulative\_frequency += frequency

write current line

write bottom of the row cells

end for

//Form footer

write cumulative frequency

write last bottom

close result

//Функція визначення медіани

(3) function find\_the\_median (movies):

if count of movies is even:

return (movies[n//2] + movies[n//2-1])/2 //середнє арифметичне двох значень у центрі впорядкованої вибірки

else:

return movies[(n-1)//2]

//Функція знаходження моди вибірки

(4) function find\_the\_mode(movies, frequencies):

mode = empty list

for index, frequency in frequencies:

if frequency == max of frequencies:

add movies[index] to mode

end for

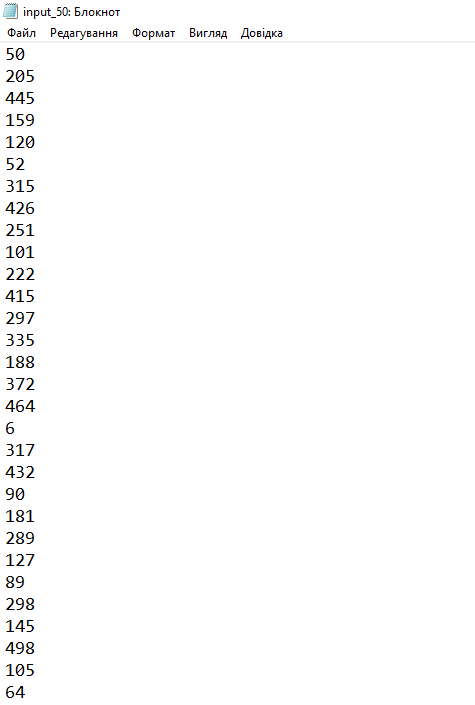
return mode

//Функція обрахунку дисперсії розподілу

(5) function find\_the\_variance(movies, list\_of\_views, frequencies):

return sum of (movies[x]^2 \* frequencies [x] ∀(x) range(count of movies))/sum of frequencies - (sum of list\_of\_views/count of list\_of\_views)^2 //Ф-ла дисперсії вказана у пункті Зв’язки математичної моделі

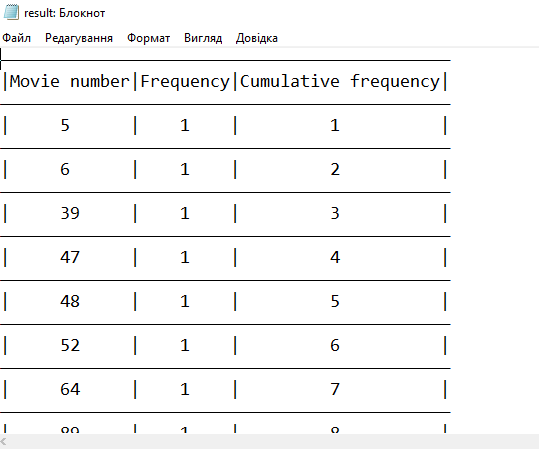
**Випробування алгоритму**

Для перевірки виконання роботи програми використаємо попередньо згенеровану вибірку з 50 елементів. 

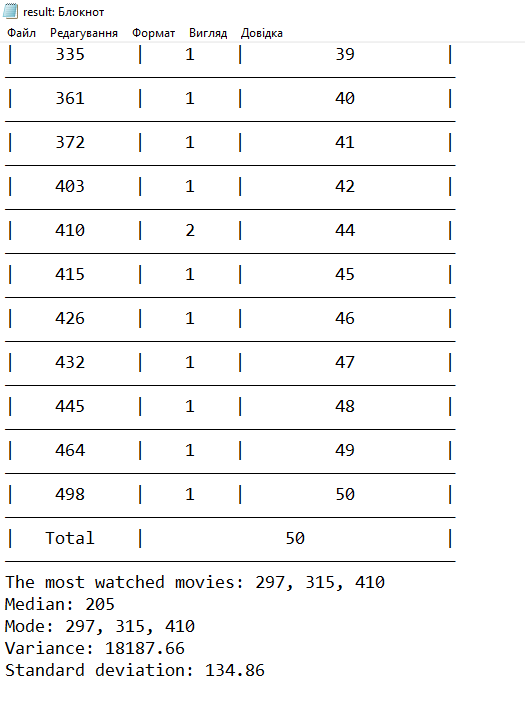
Запускаємо програму і вказуємо необхідний файл з даними:



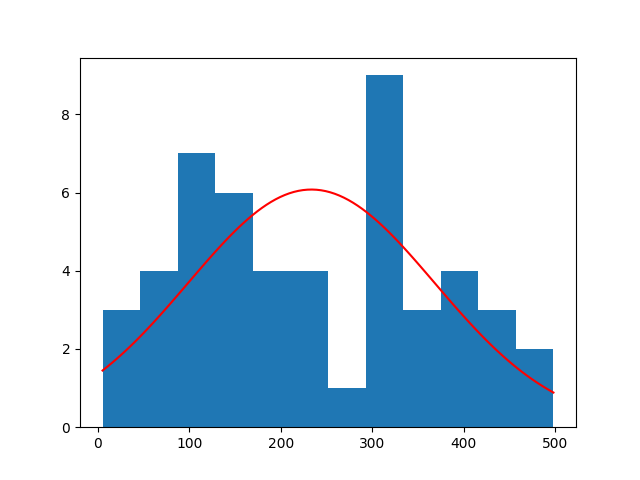
У папці проекту отримуємо новий файл result.txt:



…



На основі отриманих даних будуємо гістограму:



Оскільки для кожної окремої ітерації у вхідних даних обиралося випадкове значення у фіксованому діапазоні, то як і можна було спрогнозувати, отримали mound-shaped distribution (горбоподібний розподіл) з однією вершиною. Це повністю відповідає так званому нормальному закону розподілу випадкової величини. Причому екстремум потрапляє у точку, яка фактично знаходиться на рівні медіани (205). При заданих умовах чим більшою буде вибірка, тим менше дане відхилення.

**Висновки:** впродовж виконання цієї лабораторної роботи мені вдалося спроектувати програмний модуль, який отримуючи на вхід, вибірку даних, формує таблиці частот та сукупних частот, обраховує статистичні характеристики цієї вибірки (Медіана та Мода), знаходить Дисперсію та Середнє квадратичне відхилення розподілу. На основі отриманих даних була створена гістограма, що на практиці підтверджує закон Гаусса про нормалізацію розподілу випадкових величин.