

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
«Центральні тенденції та міра дисперсії»

Виконав:	Сирота Ангеліна Олександрівна	Перевірила:	Вечерковська Анастасія Сергіївна
Група	ІПЗ-21	Дата перевірки	
Форма навчання	денна	Оцінка	
Спеціальність	121		
2022			

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

Хід роботи

Постановка задачі:

Написати програму, що зчитує дані з файла і виконує наступні функції:

- Побудувати таблицю частот та сукупних частот для переглянутих фільмів. Визначити фільм, який був переглянутий частіше за інші.
- Знайти Моду та Медіану заданої вибірки.
- Порахувати Дисперсію та Середнє квадратичне відхилення розподілу.
- Побудувати гістограму частот для даного розподілу.

Усі результати записувати в окремий текстовий файл

Побудова математичної моделі:

Частота: кількість переглядів фільму, міститься у вхідному файлі

Сукупна частота: сума попередніх значень частот

Мода: значення, що має найбільшу частоту

- Якщо частоти усіх елементів дорівнюють 1, то моди немає.
- Якщо декілька елементів мають найбільшу частоту, то модами будуть ці елементи

Медіана: центральне значення. Знаходиться за формулою:

- Якщо парна кількість елементів у вибірці: $\frac{x_{\frac{n}{2}} + x_{(\frac{n}{2}+1)}}{2}$
- Якщо непарна кількість елементів у вибірці: $x_{\frac{n+1}{2}}$

де n – кількість елементів у вибірці

Середнє \bar{x} обчислюється за формулою: $\bar{x} = \frac{x_1 f_1 + x_2 f_2 + \dots + x_n f_n}{f_1 + f_2 + \dots + f_n}$

Дисперсія рахується за формулою: $Var(x) = \frac{\sum_{x \in X} f_x (x - \bar{x})^2}{\sum_{x \in X} f_x}$

Середнє квадратичне відхилення обчислюється за формулою: $\sigma = \sqrt{Var(x)}$

Гістограма складається з прямокутників, де ширина є фільмом / інтервалом, у якому розташовані фільми, а висота є кількістю переглядів. У випадку з інтервалами кількість переглядів є сумарною для кожного фільма з цього інтервалу.

Псевдокод алгоритму:

Заповнення масиву значеннями частоти і сукупної частоти:

```
frequency(f, arrfF):  
    data = [for item in f]    // масив частот, зчитаних з файлу  
    for el in data:  
        number += 1          // фільм  
        cf += el              // сукупна частота  
        arrfF.extend([[number, el, cf]])  
    return arrfF
```

Визначення фільму, який був переглянутий частіше за інших:

```
MaxFrequency(arrfF):  
    for i in range(arrfF):  
        if arrfF[i][1] > max:    // max - максимальна частота  
            max = arrfF[i][1]  
            index = i           // індекс фільму  
    print(index + 1)
```

Визначення моди:

```
moda(arrfF):  
    for i in range(arrfF):  
        if arrfF[i][1] > frmax:    // frmax - максимальна частота  
            frmax = arrfF[i][1]  
    for i in range(arrfF):  
        if arrfF[i][1] == frmax:  
            fmax.append(i + 1) 0    // fmax – масив елементів з максимальними  
частотами  
    if frmax == 1:  
        print("Моди немає")  
        fileOutput.write("Моди немає")  
    else:  
        for i in range(fmax):  
            print(fmax[i])
```

Визначення медіани:

```
median(arrfF):  
    n = arrfF[len(arrfF) - 1][2]    // в якості кількості елементів взято сукупну частоту  
                                     // останнього елемента  
    if n % 2 == 0:                  // якщо парна кількість  
        for i in range(arrfF):  
            if n / 2 > arrfF[i][2] and n / 2 <= arrfF[i + 1][2]:  
                index = i + 1      // індекс елемента n/2  
  
        median = (arrfF[index][0] + arrfF[index + 1][0]) / 2 // медіана  
    else:                            // якщо непарна кількість  
        for i in range(arrfF):  
            if (n + 1) / 2 > arrfF[i][2] and (n + 1) / 2 <= arrfF[i + 1][2]:  
                index = i + 1  
        median = arrfF[index][0]  
    print(median)
```

Визначення середнього значення:

```
average(arrfF):
    for i in range(len(arrfF)):
        numerator += arrfF[i][0] * arrfF[i][1] // чисельник
        denominator += arrfF[i][1] // знаменник
    Xave = numerator / denominator // середнє значення
    return Xave
```

Визначення дисперсії та середнього квадратичного відхилення:

```
def dispersion(arrfF):
    Xave = average(arrfF)
    for i in range(arrfF):
        numerator += arrfF[i][1] * math.pow(arrfF[i][0] - Xave, 2) // чисельник
        denominator += arrfF[i][1] // знаменник
    dis = numerator / denominator // дисперсія
    print(dis)

    msd = sqrt(dis) // середнє квадратичне відхилення
    print(msd)
```

Побудова гістограми:

```
histogram(arrfF):
    if arrfF == arrfF10:
        interval = len(arrfF) / 10
    elif arrfF == arrfF100:
        interval = len(arrfF) / 20
    elif arrfF == arrfF1000:
        interval = len(arrfF) / 50

    # створити масив з усіх значень з урахуванням частот
    for i in range(len(arrfF)):
        for j in range(arrfF[i][1]):
            arr.append(arrfF[i][0])

    // заповнення масиву інтервалів
    while (len(arrInt) <= len(arrfF) / interval):
        arrInt.append(p)
        p += interval

    pyplot.hist(arr, arrInt)
```

Випробування алгоритму:

Набір з 10 фільмів:

```
Введіть значення кількості елементів у вхідному файлі (10/100/1000):
10
-----

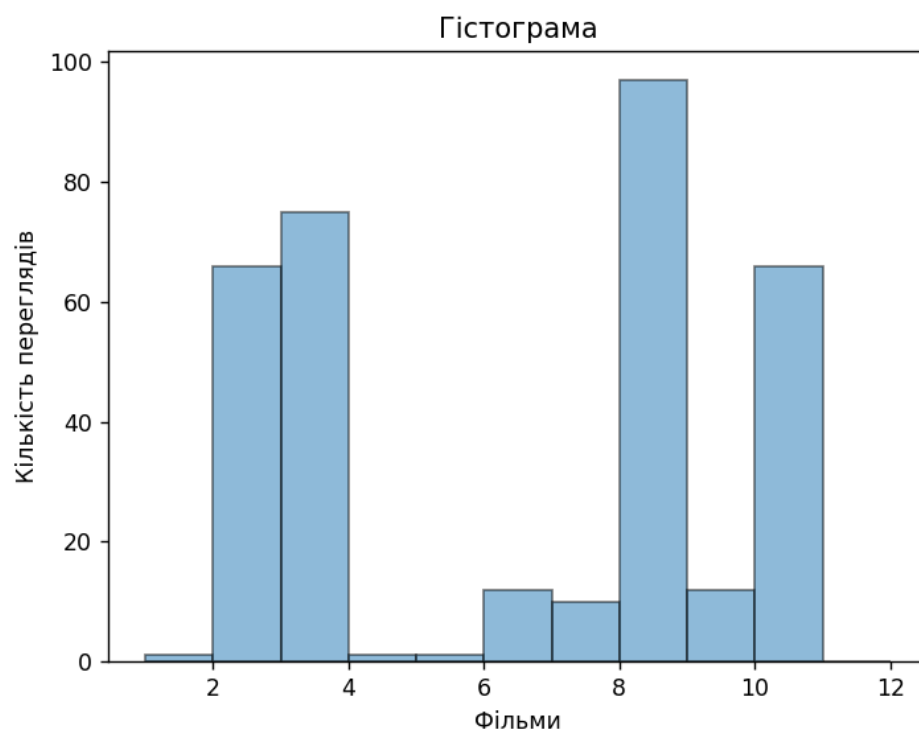
Фільм з максимальною кількістю переглядів ( 97 ): 8

----- Таблиця частот -----
Елемент | Частота | Сукупна частота
-----+-----+-----
1       | 1       | 1
2       | 66      | 67
3       | 75      | 142
4       | 1       | 143
5       | 1       | 144
6       | 12      | 156
7       | 10      | 166
8       | 97      | 263
9       | 12      | 275
10      | 66      | 341

Мода: 8
Медіана: 8

Дисперсія: 10
Середнє квадратичне відхилення: 3
```

Figure 1



Набір зі 100 фільмів:

Введіть значення кількості елементів у вхідному файлі (10/100/1000):
100

Фільм з максимальною кількістю переглядів (999): 6

----- Таблиця частот -----

Елемент	Частота	Сукупна частота
1	642	642
2	51	693
3	97	790
4	529	1319
5	46	1365
6	999	2364
7	317	2681
8	99	2780
9	880	3660
10	46	3706
11	79	3785
12	548	4333
13	361	4694
14	821	5515
15	71	5586
16	288	5874
17	51	5925
18	255	6180
19	429	6609
20	80	6689
21	657	7346
22	22	7368
23	817	8185
24	168	8353
25	688	9041
26	858	9899
27	162	10061
28	587	10648
29	775	11423
30	51	11474
31	566	12040
32	738	12778
33	763	13541
34	832	14373

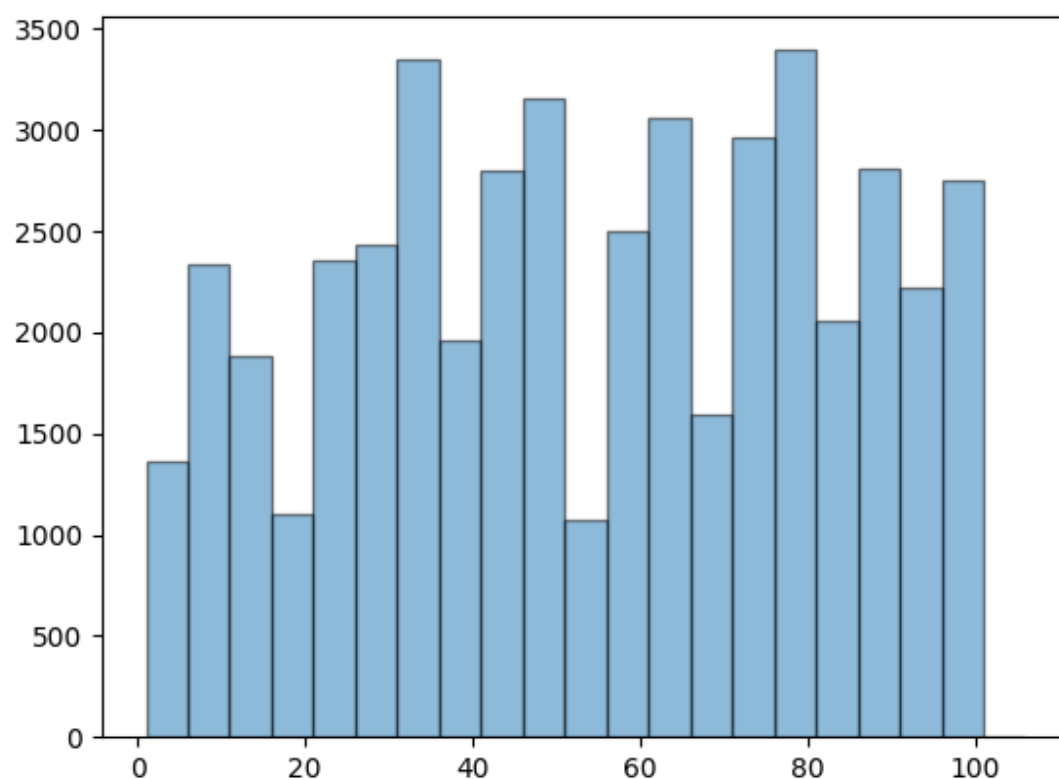
35	447	14820
36	414	15234
37	784	16018
38	355	16373
39	154	16527
40	251	16778
41	660	17438
42	250	17688
43	813	18501
44	382	18883
45	694	19577
46	613	20190
47	923	21113
48	362	21475
49	687	22162
50	571	22733
51	103	22836
52	79	22915
53	535	23450
54	162	23612
55	193	23805
56	198	24003
57	607	24610
58	91	24701
59	928	25629
60	676	26305
61	569	26874
62	503	27377
63	945	28322
64	777	29099
65	269	29368
66	47	29415
67	615	30030
68	685	30715
69	225	30940
70	22	30962
71	824	31786
72	553	32339
73	589	32928
74	22	32950
75	976	33926
76	384	34310
77	702	35012
78	612	35624

79	878	36502
80	820	37322
81	77	37399
82	834	38233
83	147	38380
84	879	39259
85	119	39378
86	736	40114
87	768	40882
88	146	41028
89	707	41735
90	450	42185
91	498	42683
92	119	42802
93	636	43438
94	612	44050
95	359	44409
96	984	45393
97	782	46175
98	22	46197
99	354	46551
100	607	47158

Мода: 6
 Медіана: 54.5
 Дисперсія: 784
 Середнє квадратичне відхилення: 28



Figure 1



Набір із 1000 фільмів:

Введіть значення кількості елементів у вхідному файлі (10/100/1000):
1000

Фільм з максимальною кількістю переглядів (99970): 925

Таблиця частот

Елемент	Частота	Сукупна частота
1	54831	54831
2	76418	131249
3	57391	188640
4	58530	247170
5	48009	295179
6	53988	349167
7	60817	409984
8	36476	446460
9	18316	464776
10	84352	549128
11	20598	569726
12	51505	621231
13	32702	653933
14	79484	733417
15	46431	779848
16	55168	835016
17	5646	840662
18	32180	872842
19	94377	967219
20	51172	1018391
21	98419	1116810
22	51768	1168578
23	80007	1248585
24	98249	1346834
25	54703	1401537
26	9894	1411431
27	92429	1503860
28	8254	1512114
29	85688	1597802
30	92880	1690682
31	57339	1748021
32	56039	1804060
33	13823	1817883
34	79415	1897298
35	76577	1973875
36	23784	1997659
37	35177	2032836
38	19310	2052146
39	69144	2121290
40	35909	2157199
41	30270	2187469
42	68698	2256167
43	78130	2334297
44	90713	2425010
45	39709	2464719
46	38514	2503233
47	37071	2540304
48	37463	2577767
49	88333	2666100
50	6606	2672706
51	88363	2761069
52	23082	2784151
53	38404	2822555
54	74854	2897409
55	83088	2980497
56	22361	3002858
57	33210	3036068
58	95159	3131227
59	20092	3151319
60	5818	3157137
61	60028	3217165
62	15732	3232897
63	61013	3293910
64	19128	3313038
65	2723	3315761
66	21903	3337664
67	44994	3382658
68	9426	3392084
69	38499	3430583

70	38984	3469567
71	78150	3547717
72	82757	3630474
73	11759	3642233
74	84329	3726562
75	90323	3816885
76	19032	3835917
77	66110	3902027
78	82039	3984066
79	30405	4014471
80	18771	4033242
81	896	4034138
82	77743	4111881
83	69168	4181049
84	71016	4252065
85	83925	4335990
86	91400	4427390
87	30578	4457968
88	77711	4535679
89	80667	4616346
90	44919	4661265
91	5199	4666464
92	64950	4731414
93	17250	4748664
94	7568	4756232
95	62069	4818301
96	27304	4845605
97	39725	4885330
98	88453	4973783
99	91505	5065288
100	77781	5143069
101	56176	5199245
102	87976	5287221
103	14833	5302054
104	49362	5351416
105	94550	5445966
106	99024	5544990
107	7874	5552864
108	8757	5561621

110	50448	5707185
111	74720	5781905
112	51057	5832962
113	80885	5913847
114	98046	6011893
115	47564	6059457
116	79727	6139184
117	90950	6230134
118	33571	6263705
119	76181	6339886
120	1975	6341861
121	61406	6403267
122	81772	6485039
123	42043	6527082
124	58550	6585632
125	99172	6684804
126	35361	6720165
127	68148	6788313
128	53120	6841433
129	80606	6922039
130	22132	6944171
131	9402	6953573
132	9585	6963158
133	65598	7028756
134	93515	7122271
135	69599	7191870
136	34177	7226047
137	13436	7239483
138	93616	7333099
139	74182	7407281
140	99968	7507249
141	46006	7553255
142	66437	7619692
143	14929	7634621
144	12947	7647568
145	18048	7665616
146	820	7666436
147	34679	7701115
148	83815	7784930
149	61492	7846422
150	23168	7869590
151	22741	7892331
152	48149	7940480
153	38931	7979411
154	45915	8025326

157	90429	8220990
158	7437	8228427
159	76180	8304607
160	22740	8327347
161	67318	8394665
162	45759	8440424
163	85106	8525530
164	24946	8550476
165	62067	8612543
166	94324	8706867
167	80	8706947
168	42076	8749023
169	70222	8828245
170	26844	8855089
171	18605	8873694
172	50131	8923825
173	79964	9003789
174	9683	9013472
175	12026	9025498
176	12204	9037702
177	14775	9052477
178	69352	9121829
179	70421	9192250
180	72531	9264781
181	98228	9363009
182	44920	9407929
183	21097	9429026
184	61451	9490477
185	45751	9536228
186	59715	9595943
187	56940	9652883
188	44255	9697138
189	63161	9760299
190	44594	9804893
191	52609	9857502
192	31199	9888701
193	9696	9898397
194	22294	9920691
195	91742	10012433
196	83094	10095527
197	51426	10146953
198	21592	10168545
199	74176	10242721
200	83096	10325817
201	47854	10373671

204	51666	10576902
205	21121	10598023
206	45704	10643727
207	14023	10657750
208	51504	10709254
209	36689	10745943
210	45870	10791813
211	10193	10802006
212	38109	10840115
213	13481	10853596
214	82390	10935986
215	20002	10955988
216	66434	11022422
217	44234	11066656
218	11293	11077949
219	31691	11109640
220	41976	11151616
221	49636	11201252
222	60544	11261796
223	58938	11320734
224	75657	11396391
225	22739	11419130
226	39549	11458679
227	70394	11529073
228	41402	11570475
229	10077	11580552
230	97399	11677951
231	10280	11688231
232	97990	11786221
233	73090	11859311
234	87541	11946852
235	27059	11973911
236	79181	12053092
237	69586	12122678
238	85766	12208444
239	11449	12219893
240	21533	12241426
241	26242	12267668
242	35563	12303231
243	8636	12311867
244	5437	12317304
245	77255	12394559
246	73153	12467712
247	98790	12566502
248	4364	12570866

297	24465	14803094
298	50732	14853826
299	86326	14940152
300	22790	14962942
301	62522	15025464
302	47189	15072653
303	43306	15115959
304	14920	15130879
305	57523	15188402
306	4602	15193004
307	51910	15244914
308	82357	15327271
309	28664	15355935
310	79874	15435809
311	36223	15472032
312	15514	15487546
313	68999	15556545
314	11403	15567948
315	51460	15619408
316	27434	15646842
317	52957	15699799
318	90433	15790232
319	50304	15840536
320	58020	15898556
321	47997	15946553
322	72024	16018577
323	17385	16035962
324	46646	16082608
325	2626	16085234
326	14018	16099252
327	13924	16113176
328	45791	16158967
329	55844	16214811
330	70949	16285760
331	61511	16347271
332	68880	16416151
333	78517	16494668
334	95129	16589797
335	12703	16602500
336	55346	16657846
337	10904	16668750
338	41293	16710043
339	11065	16721108
340	17150	16738258
341	35961	16774219

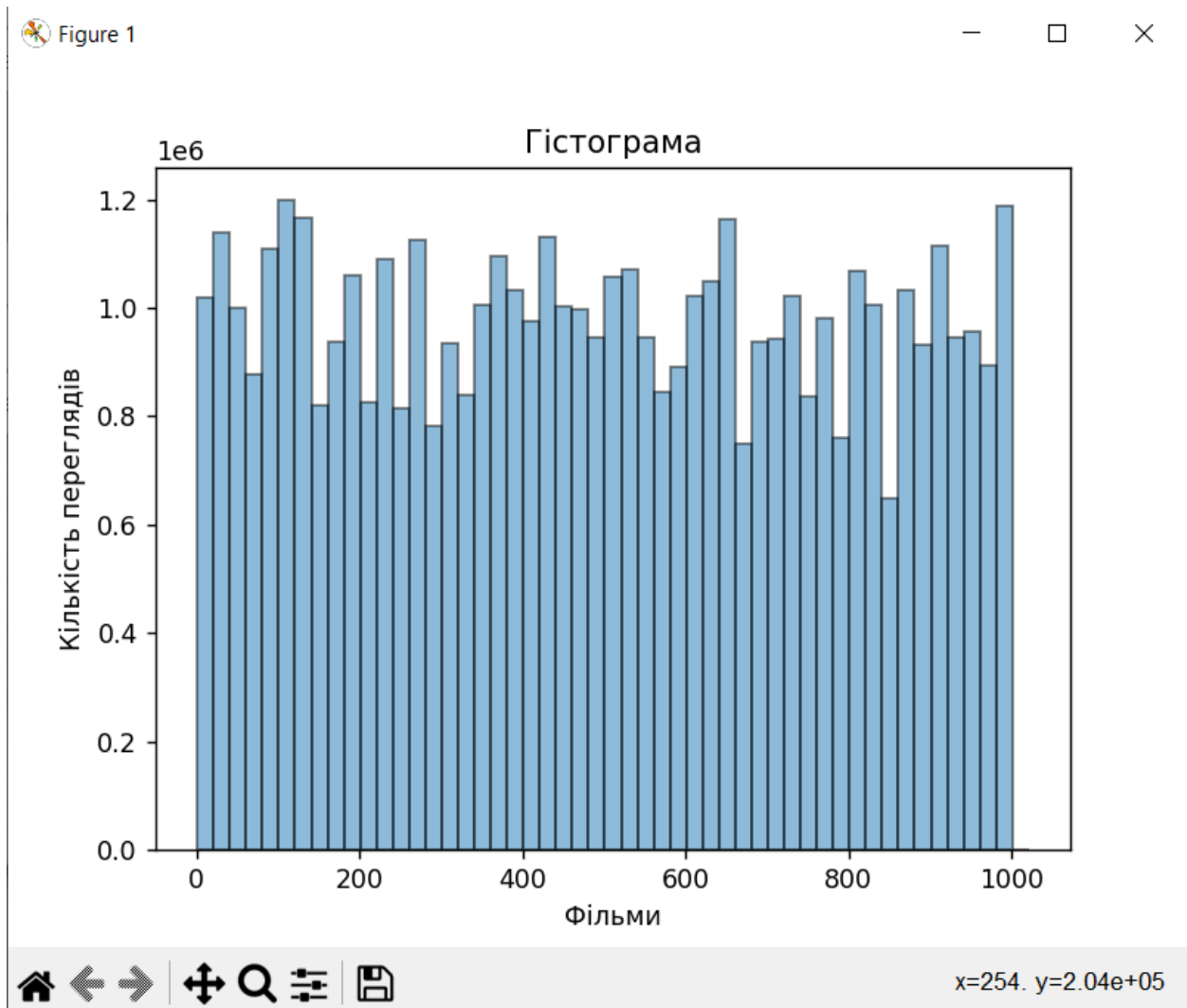
250	24535	12640572
251	39054	12679626
252	129	12679755
253	77205	12756960
254	5322	12762282
255	26755	12789037
256	37212	12826249
257	300	12826549
258	66469	12893018
259	99403	12992421
260	62835	13055256
261	2374	13057630
262	25454	13083084
263	80390	13163474
264	96170	13259644
265	6965	13266609
266	2352	13268961
267	73289	13342250
268	1210	13343460
269	91492	13434952
270	72400	13507352
271	59021	13566373
272	80466	13646839
273	50122	13696961
274	79841	13776802
275	79666	13856468
276	40655	13897123
277	88853	13985976
278	85516	14071492
279	72186	14143678
280	36955	14180633
281	7903	14188536
282	14785	14203321
283	65777	14269098
284	70947	14340045
285	11915	14351960
286	56517	14408477
287	7341	14415818
288	40519	14456337
289	37445	14493782
290	45170	14538952
291	79362	14618314
292	14922	14633236
293	89503	14722739
294	113	14722852

344	78075	16938375
345	40218	16978593
346	81232	17059825
347	5308	17065133
348	2317	17067450
349	930	17068380
350	59007	17127387
351	98668	17226055
352	13406	17239461
353	93669	17333130
354	54271	17387401
355	33030	17420431
356	47835	17468266
357	86709	17554975
358	61897	17616872
359	41770	17658642
360	84950	17743592
361	68300	17811892
362	99820	17911712
363	26997	17938709
364	70836	18009545
365	53362	18062907
366	99256	18162163
367	38925	18201088
368	91342	18292430
369	40617	18333047
370	84738	18417785
371	6567	18424352
372	27838	18452190
373	76786	18528976
374	83946	18612922
375	9114	18622036
376	42778	18664814
377	12474	18677288
378	53586	18730874
379	53581	18784455
380	55686	18840141
381	60476	18900617
382	80522	18981139
383	96701	19077840
384	37223	19115063
385	36978	19152041
386	19966	19172007
387	729	19172736
388	25702	19198438

...

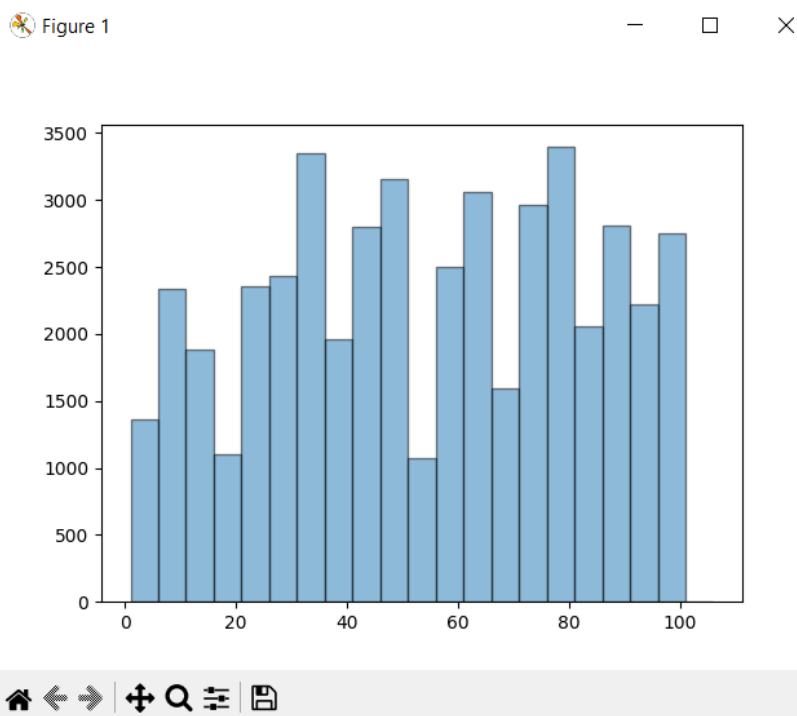
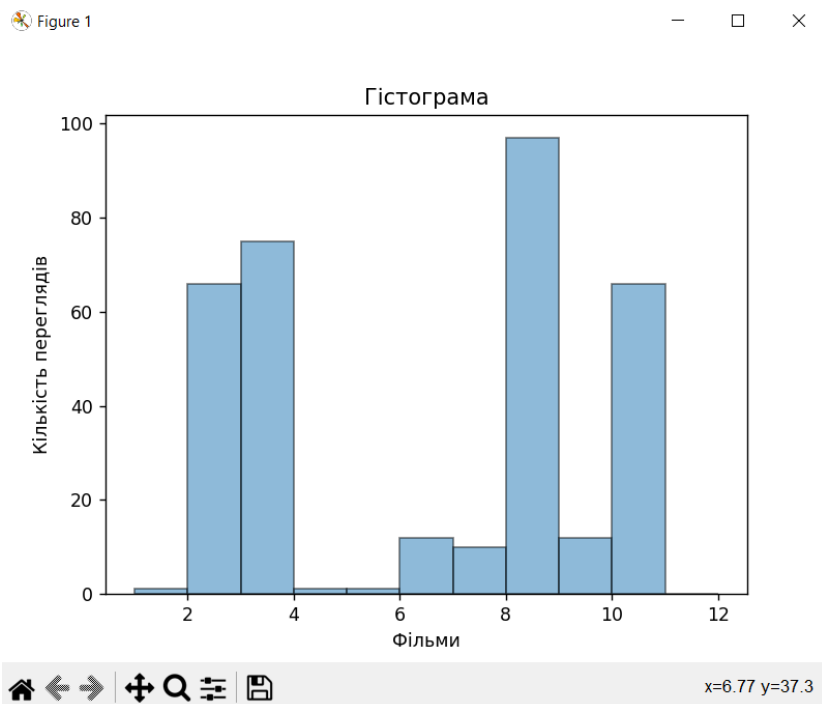
982	82040	47941935
983	7382	47949317
984	81739	48031056
985	59959	48091015
986	86731	48177746
987	60294	48238040
988	7783	48245823
989	86957	48332780
990	91975	48424755
991	85263	48510018
992	57128	48567146
993	39651	48606797
994	84147	48690944
995	24393	48715337
996	48960	48764297
997	83063	48847360
998	3008	48850368
999	64327	48914695
1000	69601	48984296

Мода: 925
Медіана: 492.5
Дисперсія: 84399
Середнє квадратичне відхилення: 291

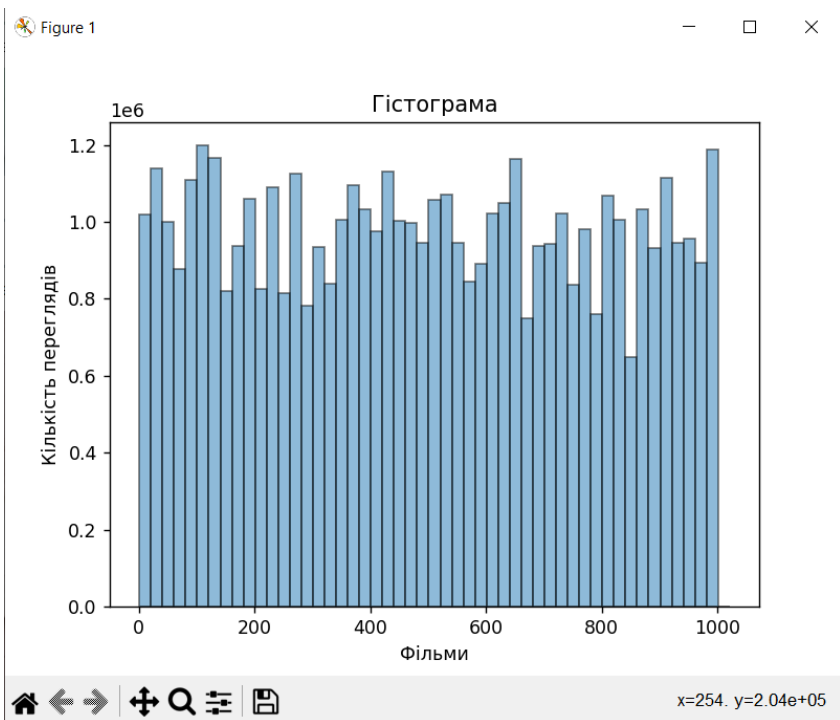


Висновки з вигляду гістограми, про закон розподілу:

Дана гістограма не є симетричною. Складається з одного кластеру.



Дана гістограма не є симетричною. Складається з одного кластеру.



Дана гістограма не є симетричною. Складається з одного кластеру. Через розбиття на інтервали отримана гістограма тяжіє до рівномірного розподілу.

Висновок: в ході цієї лабораторної роботи було побудовано таблицю частот та сукупних частот для переглянутих фільмів, визначено фільм, який був переглянутий частіше за інші, знайдено моду та медіану заданої вибірки, пораховано дисперсію та середнє квадратичне відхилення, побудовано гістограму частот за допомогою бібліотеки matplotlib і проаналізовано отримані діаграми. Вони не є симетричними і складається з одного кластеру. Отримані результати внесено у новий текстовий файл.