МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

федеральное государственное АВТОНОМНОЕ образовательное учреждение высшего образования

«Национальный исследовательский ядерный университет «МИФИ»

Обнинский институт атомной энергетики

Отделение интеллектуальных кибернетических систем

**Лабораторная работа №1**

Выполнила студентка

Группы ИС-Б17

Отделения ИКС

Петренко В. Ю.

Проверила:

профессор, д.т.н. Гулина О. М.

Обнинск, 2020

1. Алгоритм

Использовала метод середины квадратов.  
Формула заполнения массива псевдослучайными числами:

Формула хи-квадрат распределения (критерий Пирсона):Х2=

(Использовала python 3.6)

1. k=2500 #число элементов

1. r=10 #кол-во интервалов
2. p=0.1 #теоретическая вероятность попадания в каждый интервал
3. array=[] #массив псевдослучайных чисел
4. l\_aper=0 #длина апериодичности
5. l\_per=0 #длина периода
6. p\_i =[] #количество попаданий в каждый интервал
7. X2=0 #хи-вквадрат
8. def fraction(x):
9. # функция для расчета дробной части
10. return x - int(x)
11. def fillArray():
12. # функция для заполнения массива
13. y0=float(input("Введите гамма-нулевое: "))
14. accrs=int(input("Введите количество знаков после запятой: "))
15. for i in range(k):
16. array.append(y0)
17. y0=(10 \*\* -accrs)\*int((10 \*\* accrs)\*fraction(float(((1-y0) \*\* 3)\*(10 \*\* accrs)))) #метод середины квадратов
18. print("Массив заполнен псевдослучайными числами.")
19. def periodLength():
20. global l\_aper, l\_per
21. print("Определение длины периода и апериодичности.")
22. flag=True #пока в последовательности будут одинаковые элементы
23. for i in range(k):
24. for j in range(i+1, k):
25. if(abs(array[i]-array[j])<0.00000001):#сравниваем
26. print("Cовпадение в ", i, "-ом и ", j, "-ом элементах: ", array[i], " и ", array[j])
27. l\_aper = j
28. l\_per = j-i
29. flag=False
30. if not flag:
31. break
32. if flag:
33. #если нет одинаковых элементов, длина апериодичности = длине последовательности
34. l\_aper=k
35. l\_per=0
36. if not flag:
37. break
38. print("Период: ", l\_per)
39. print("Апериодичность: ", l\_aper)
40. def calc\_pi():
41. print("Рассчет количества попаданий в каждый интервал.")
42. for i in range(r):
43. p\_i.append(0)
44. print("[ ", end = ' ')
45. for i in range(r):
46. for j in range(l\_aper):
47. if (array[j]>(i\*p) and array[j]<((i+1)\*p)):
48. p\_i[i]+=1
49. for i in range(r):
50. print(p\_i[i], end = ' ')
51. print(" ] ")
52. def calc\_X2():
53. print("Рассчет Х2.")
54. X2 = 0
55. for i in range(r):
56. X2+=((p\_i[i]-(l\_aper\*p)) \*\* 2)/(l\_aper\*p)
57. print("X2 = ", X2)
58. def show():
59. n=int(input("Вывести последовательность до: "))
60. if n>k:
61. n=k
62. for i in range(n):
63. print(array[i], end = ', ')
64. print("Программа генерирует псевдослучайную последовательнсоть из 2500 элементов на интервале (0;1)\n")
65. print("Затем расчитывает длины периода и апериодичности и вероятность попадания в каждый интервал (10 интервалов)\n")
66. print("После чего считает Х-квадрат для последовательности длиной апериодичности.")
67. fillArray()
68. periodLength()
69. calc\_pi()
70. calc\_X2()
71. show()

2. Результаты

k=2

|  |  |  |  |
| --- | --- | --- | --- |
| № эксперимента | γ0 | P | L |
| 1 | 0,12 | 1 | 22 |
| 2 | 0,34 | 1 | 15 |
| 3 | 0,56 | 1 | 12 |
| 4 | 0,63 | 1 | 6 |
| 5 | 0,88 | 1 | 2 |

k=4

|  |  |  |  |
| --- | --- | --- | --- |
| № эксперимента | γ0 | P | L |
| 1 | 0,1234 | 102 | 124 |
| 2 | 0,1111 | 102 | 118 |
| 3 | 0,1222 | 102 | 159 |
| 4 | 0,8765 | 102 | 127 |
| 5 | 0,5678 | 102 | 130 |

k=6

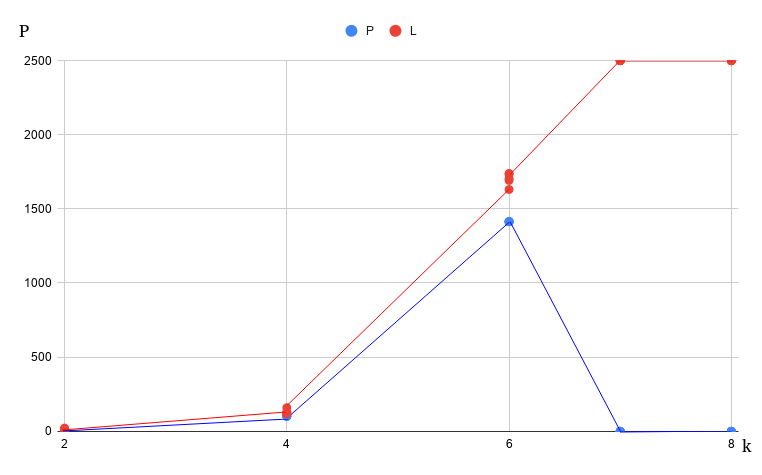
|  |  |  |  |
| --- | --- | --- | --- |
| № эксперимента | γ0 | P | L |
| 1 | 0.234567 | 1414 | 1739 |
| 2 | 0.342365 | 1414 | 1631 |
| 3 | 0.236790 | 1414 | 1704 |
| 4 | 0.456456 | 1414 | 1734 |
| 5 | 0.512345 | 1414 | 1692 |

k=7

|  |  |  |  |
| --- | --- | --- | --- |
| № эксперимента | γ0 | P | L |
| 1 | 0.1234567 | 0 | 2500 |
| 2 | 0.2312453 | 0 | 2500 |
| 3 | 0.2457654 | 0 | 2500 |
| 4 | 0.4562853 | 0 | 2500 |
| 5 | 0.8673945 | 0 | 2500 |

k=8

|  |  |  |  |
| --- | --- | --- | --- |
| № эксперимента | γ0 | P | L |
| 1 | 0.13243546 | 0 | 2500 |
| 2 | 0,45347234 | 0 | 2500 |
| 3 | 0,54657629 | 0 | 2500 |
| 4 | 0,56428546 | 0 | 2500 |
| 5 | 0,96542378 | 0 | 2500 |

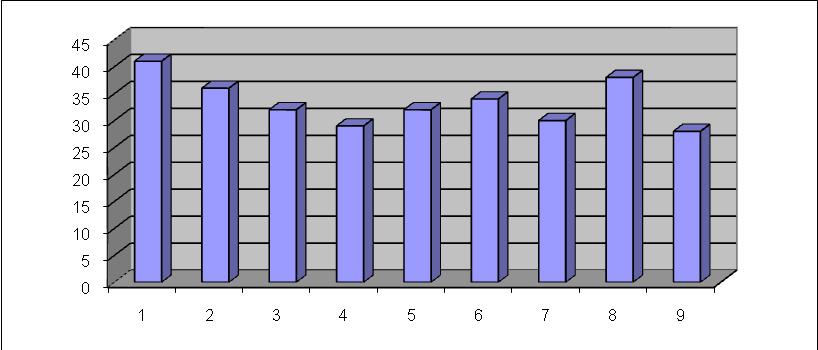


Проверка по критерию Х-квадрат.

|  |  |  |
| --- | --- | --- |
| γ0i | χ2 | Вывод |
| **0,00012531** | 4,5 | данная последовательность распределена по равномерному закону с точностью примерно 50% |
| **0,23540014** | 8,52 | данная последовательность распределена по равномерному закону с точностью примерно 10% |
| **0,39554201** | 6,06 | данная последовательность распределена по равномерному закону с точностью примерно 40% |
| **0,87126303** | 13,5 | данная последовательность распределена по равномерному закону с точностью примерно 95% |
| **0,91001369** | 5,16 | данная последовательность распределена по равномерному закону с точностью примерно 20% |

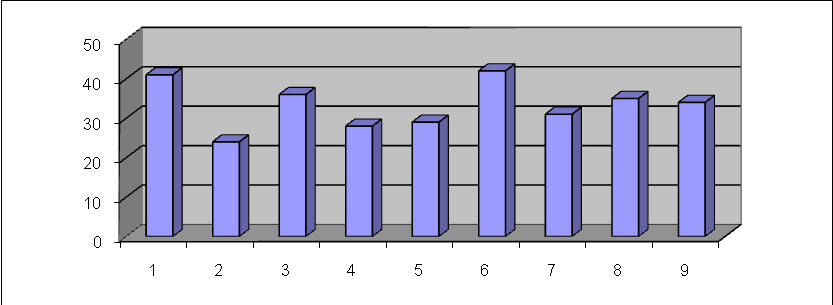
Случай 1. При γ0 =**0,00012531**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 41 | 36 | 32 | 29 | 32 | 34 | 30 | 38 | 28 |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |



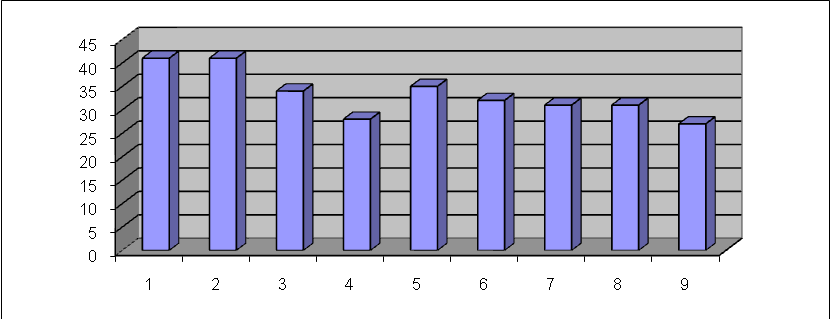
Случай 2. При γ0 =**0,23540014**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 41 | 24 | 36 | 28 | 29 | 42 | 31 | 35 | 34 |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |



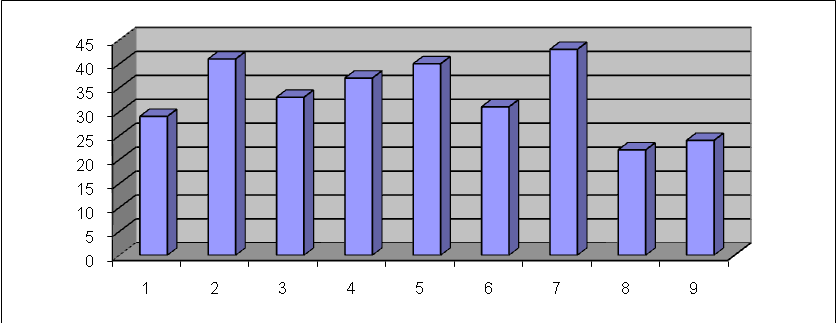
Случай 3. При γ0 =**0,39554201**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 41 | 41 | 34 | 28 | 35 | 32 | 31 | 31 | 27 |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |



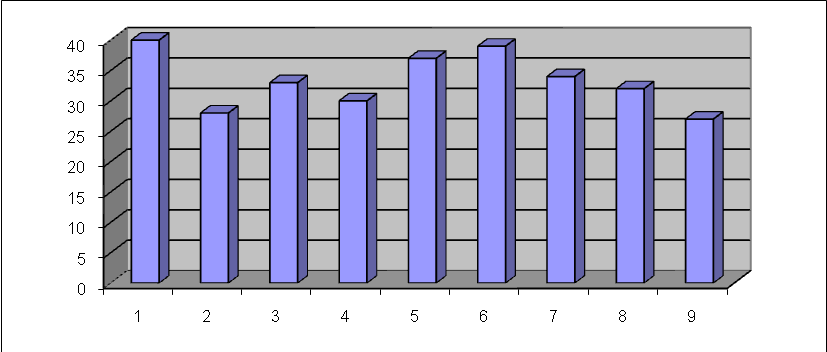
Случай 4. При γ0 =**0,87126303**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 29 | 41 | 33 | 37 | 40 | 31 | 43 | 22 | 24 |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |



Случай 5. При γ0 =**0,91001369**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 40 | 28 | 33 | 30 | 37 | 39 | 34 | 32 | 27 |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |



3. Выводы: При увеличении количества знаков после запятой растут длины периода и апериодичности. Гипотеза о соответствии равномерному закону распределения по значениям Х2 подтвердилась лишь в некоторых случаях.