МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ
БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ФАКУЛЬТЕТ ПРИКЛАДНОЙ МАТЕМАТИКИ И ИНФОРМАТИКИ
Кафедра технологий программирования

Толкун Кирилл Юрьевич

Отчёт по лабораторным работам по курсу “Имитационное и статистическое моделирование”

студента 4 курса 8 группы

Вариант 10

Преподаватель:
Лобач Сергей Викторович
ассистент кафедры ММАД

Работа сдана ________ 2020 г.

Зачтена ________ 2020 г.

____
(подпись преподавателя)

Минск
2020

1 Лабораторная 1

1.1 Условие

Согласно варианту: $\mathbf{X_0} = \alpha = \mathbf{16807}, \mathbf{K} = \mathbf{64}$.

Используя метод Маклерена-Марсальи построить датчик БСВ (1 датчик должен быть мультипликативно конгруэнтный, второй – на выбор). Исследовать точность построенной БСВ.

1. Осуществить моделирование $n = 1000$ реализаций БСВ с помощью мультипликативного конгруэнтного метода (МКМ) с параметрами $X0, \alpha, m = 231$;

2. Осуществить моделирование $n = 1000$ реализаций БСВ с помощью метода Макларена-Марсальи (один датчик должен быть мультипликативно конгруэнтный (п. 1), второй – на выбор). K – объем вспомогательной таблицы;

3. Проверить точность моделирования обоих датчиков (п. 1 и п. 2) с помощью критерия согласия Колмогорова и χ^2 - критерия Пирсона с уровнем значимости $\varepsilon = 0.05$.

1.2 Теория

1.2.1 Датчики БСВ

Для моделирования на ЭВМ реализаций ***Базовой случайной величины*** используются специальные программы, называемые программными датчиками БСВ.
В основе программных датчиков БСВ лежат рекуррентные формулы вида:

$$x_n = \varphi(x_{n-1}, \dots, x_{n-p}), n = 1, 2, \dots, \tag{1.1}$$

где $x_{1-p}, x_{2-p}, \dots, x_0$ $(p \geqslant 1)$ - заданные стартовые значения. Описанное соотношение (1.1) описывает детерминированный алгоритм, однако при соответствующем подборе преобразования $\varphi(\cdot)$ получаемые на его основе псевдослучайные числа x_n по своим функциональным и числовым характеристикам близки к БСВ.

Алгоритмы моделирования вида (1.1) обладают общим недостатком: начиная с некоторого момента $\mathbf{t_0}$, последовательность псевдослучайных чисел образует цикл, который повторяется бесконечное число раз. Длина $\mathbf{T}$ циклически повторяющейся последовательности называется ***периодом датчика*** БСВ ($T \leq m - 1$).

Период $\mathbf{T}$ и ***коэффициент использования*** БСВ $\mathbf{k}$ являются основными показателями качества программных датчиков БСВ. Лучшим датчикам соответствуют большие значения $\mathbf{T}$ и $\mathbf{k}$.

1.2.2 Линейный конгруэнтный метод

Линейный конгруэнтный метод - один из методов генерации псевдослучайных чисел. Применяется в простых случаях и не обладает криптографической стойкостью. Входит в библиотеки различных компиляторов.

Суть метода заключается в вычислении последовательности случайных чисел X_n следующим образом:

$$X_{n+1} = \frac{\alpha X_n + c) \bmod m}{m}, \tag{1.2}$$

где:

$$\left.\begin{array}{l} 1.\ X_0 \text{ - начальное значение } (0 \leqslant X_0 < 1) \\ 2.\ \alpha \text{ - множитель } (0 \leqslant \alpha < m) \\ 3.\ c \text{ - приращение } (0 \leqslant c < m) \\ 4.\ m \geq 2 \text{ - модуль} \end{array}\right\} \text{ - } \textbf{\textit{параметры датчика}}.$$

Типовые значения параметров: $m = 2^{31}, x_0 = \alpha = 65539$.

1.2.3 Мультипликативный конгруэнтный метод

Метод генерации *линейной конгруэнтной последовательности* (раздел 1.2.2) при $\mathbf{c = 0}$ называют ***мультипликативным конгруэнтным методом***.

1.2.4 Метод Макларена-Марсальи

Генератор Макларена-Марсальи - криптографически стойкий генератор псевдослучайных чисел, который основан на комбинации двух конгруэнтных генераторов и вспомогательной матрице, с помощью которой происходит перемешивание двух последовательностей, полученных от двух генераторов.

Данный генератор псевдослучайных чисел оперирует с тремя объектами: двумя конгруэнтными генераторами, которые порождают последовательности $\langle \mathbf{X_n} \rangle, \langle \mathbf{Y_n} \rangle$, и массива $\mathbf{V}$, состоящей из $\mathbf{k}$ элементов, обычно $k \in \{64, 28, 256\}$. На выходе последовательность $\langle \mathbf{Z_n} \rangle$.

Генератор состоит из четырёх основных стадий:

1. Инициализация $\mathbf{V}$ и $\mathbf{Z}$ первыми $\mathbf{k}$ элементами последовательности $\langle \mathbf{X_n} \rangle$ - выполняется один раз;

2. Выборка $\mathbf{X}, \mathbf{Y}$ из $\langle \mathbf{X_n} \rangle, \langle \mathbf{Y_n} \rangle$, то есть $\mathbf{X}, \mathbf{Y}$ - очередные члены последовательностей $\langle \mathbf{X_n} \rangle, \langle \mathbf{Y_n} \rangle$;

3. Вычисление $\mathbf{j} = \lfloor \mathbf{k} \cdot \mathbf{Y} \rfloor$, где $\mathbf{j} \in [\mathbf{0}, \mathbf{k})$ - случайное число, определяемое Y;

4. Присвоение $\mathbf{Z_i} = \mathbf{V_i}$ и замена $\mathbf{V_j} = \mathbf{X}$.

Последние три стадии могут повторяться необходимое число раз.

Данный метод позволяет ослабить зависимость между членами последовательности $\mathbf{Z_n}$ и получить сколь угодно большие значения её периода T при условии, что периоды T_1, T_2 исходных датчиков являются взаимно простыми числами. Коэффициент использования БСВ для данного датчика $\mathbf{k} = \frac{\mathbf{1}}{\mathbf{2}}$ (за исключением первой реализации, для моделирования которой используется $K + 1$ реализация).

1.2.5 χ^2 критерий согласия Пирсона

Критерий согласия Пирсона - это непараметрический метод, который позволяет оценить значимость различий между фактическим (выявленным в результате исследования) количеством исходов или качественных характеристик выборки, попадающих в каждую категорию, и теоретическим количеством, которое можно ожидать в изучаемых группах при справедливости нулевой гипотезы. Метод позволяет оценить статистическую значимость различий двух или нескольких относительных показателей (частот, долей).

Данный критерий применяют для проверки гипотезы о соответствии эмпирического распределения предполагаемому теоретическому распределению $F(X)$ при большом объёме выборки ($n \geqslant 100$). *Критерий применим для любых видов функции $F(x)$, даже при неизвестных значениях их параметров, что обычно имеет место при анализе результатов механических испытаний.*

Статистика критерия проверки гипотез имеет вид:

$$\chi^2 = \sum_{i=1}^{p} \frac{(n_i - n \cdot p_i)^2}{n \cdot p_i}, \tag{1.3}$$

где n_i - наблюдаемые частоты, $n \cdot p_i$ - ожидаемые частоты.

Чем больше χ^2, тем сильнее выборка X не согласуется с гипотезой H_0 (***нулевая гипотеза***: наблюдаемые частоты соответствуют ожидаемым).

Чтобы проверить гипотезу по *критерию Пирсона* необходимо сравнить *статистику критерия* с *критическим значения*, которой находится в таблице для соответствующего *уровня значимости* и количеству *степеней свободы*.

Пример: при уровне значимости $\alpha = 0.05$ и количестве степеней свободы $\nu = 9$ *критерий Пирсона* согласуется с *нулевой гипотезой* при $\chi^2 < 16.919$.

ν \ α	0,99	0,98	0,95	0,90	0,80	0,70	0,50	0,30	0,20	0,10	0,05	0,02	0,01	α / ν
1	0,00016	0,00628	0,00393	0,0158	0,0642	0,148	0,455	1,074	1,642	2,706	3,841	5,412	6,635	1
2	0,0201	0,0404	0,103	0,211	0,446	0,713	1,386	2,408	3,219	4,605	5,991	7,824	9,210	2
3	0,115	0,185	0,352	0,584	1,005	1,424	2,366	3,605	4,642	6,251	7,815	9,837	11,345	3
4	0,297	0,429	0,711	1,064	1,649	2,195	3,357	4,878	5,989	7,779	9,488	11,668	13,277	4
5	0,554	0,752	1,145	1,610	2,343	3,000	4,351	6,064	7,289	9,236	11,070	13,388	15,086	5
6	0,872	1,134	1,635	2,204	3,070	3,828	5,348	7,231	8,558	10,645	12,592	15,033	16,812	6
7	1,239	1,564	2,167	2,833	3,822	4,671	6,346	8,383	9,803	12,017	14,067	16,622	18,475	7
8	1,646	2,032	2,733	3,490	4,594	5,527	7,344	9,524	11,030	13,362	15,507	18,168	20,090	8
9	2,088	2,532	3,325	4,168	5,380	6,393	8,343	10,656	12,242	14,684	16,919	19,679	21,666	9
10	2,558	3,059	3,940	4,865	6,179	7,267	9,342	11,781	13,442	15,987	18,307	21,161	23,209	10
11	3,053	3,609	4,575	5,578	6,989	8,148	10,341	12,899	14,631	17,275	19,675	22,618	24,725	11
12	3,571	4,178	5,226	6,304	7,807	9,034	11,340	14,011	15,812	18,549	21,026	24,054	26,217	12
13	4,107	4,765	5,892	7,042	8,634	9,926	12,340	15,119	16,985	19,812	22,362	25,472	27,688	13
14	4,660	5,368	6,571	7,790	9,467	10,821	13,339	16,222	18,151	21,064	23,685	26,873	29,141	14
15	5,229	5,985	7,261	8,547	10,307	11,721	14,339	17,322	19,311	22,307	24,996	28,259	30,578	15
16	5,812	6,614	7,962	9,312	11,152	12,624	15,338	18,418	20,465	23,542	26,296	29,633	32,000	16
17	6,408	7,255	8,672	10,085	12,002	13,531	16,338	19,511	21,615	24,769	27,587	30,995	33,409	17
18	7,015	7,906	9,390	10,865	12,857	14,440	17,338	20,601	22,760	25,989	28,869	32,346	34,805	18
19	7,633	8,567	10,117	11,651	13,716	15,352	18,338	21,689	23,900	27,204	30,144	33,687	36,191	19
20	8,260	9,237	10,851	12,443	14,578	16,266	19,337	22,775	25,038	28,412	31,410	35,020	37,566	20

Рис. 1: Значения χ^2 при различных P_{χ^2} в зависимости от числа степеней свобод ν.

1.2.6 Критерий согласия Колмогорова

Критерий согласия Колмогорова предназначен для проверки гипотезы о принадлежности выборки некоторому закону распределения, то есть проверки того, что эмпирическое распределение соответствует предполагаемой модели.

Эмпирическая функция распределения $\mathbf{F_n}$, построенная по выборке $X = (X_1, \ldots, X_n)$, имеет вид:

$$F_n(x) = \frac{1}{n}\sum_{i=1}^{n} I_{X_i \leqslant x} \tag{1.4}$$

где $I_{X_i \leqslant x}$ указывает, попало ли наблюдение X_i в область $(-\infty, x]$:

$$I_{X_i \leqslant x} = \begin{cases} 1, X_i \leqslant x \\ 0, X_i > x \end{cases} \tag{1.5}$$

Статистика критерия для эмпирической функции распределения $F_n(x)$ определяется следующим образом:

$$D_n = \sup_{x \in R} |F_n(x) - F(x)| \tag{1.6}$$

Принятие решения по критерию Колмогорова: В случае справедливости *нулевой гипотезы* (H_0) при $n \to +\infty$ статистика D_n имеет распределение Колмогорова:

$$\lim_{n \to \infty} P(\sqrt{n} D_n < x) = K(x) \tag{1.7}$$

здесь

$$K(x) = \sum_{k=-\infty}^{+\infty} (-1)^k e^{-2k^2x^2} \approx 1 - 2e^{-2x^2}, x \geqslant 0 \tag{1.8}$$

- функция Колмогорова.

При ***уровне значимости*** α пороговое значение C_α, находится из соотношения:

$$K(C_\alpha) = 1 - \alpha \tag{1.9}$$

Таким образом, для проверки гипотезы о виде распределения получаем:

$$\rho(X) = \begin{cases} H_0, \sqrt{n} D_n \leqslant \alpha, \\ H_1, \sqrt{n} D_n > \alpha. \end{cases} \tag{1.10}$$

C_α	α	C_α	α	C_α	α	C_α	α	C_α	α	C_α	α	C_α	α
$\boldsymbol{\alpha \le 0,29}$	1,00000	**0,62**	0,8368	**0,95**	0,3275	**1,28**	0,0755	**1,61**	0,0112	**1,94**	0,0011	**2,27**	0,0001
0,30	0,99999	**0,63**	0,8222	**0,96**	0,3154	**1,29**	0,0717	**1,62**	0,0105	**1,95**	0,0010	**2,28**	0,0001
0,31	0,99998	**0,64**	0,8073	**0,97**	0,3036	**1,30**	0,0681	**1,63**	0,0098	**1,96**	0,0009	**2,29**	0,0001
0,32	0,99995	**0,65**	0,7920	**0,98**	0,2921	**1,31**	0,0646	**1,64**	0,0092	**1,97**	0,0009	**2,30**	0,0001
0,33	0,99991	**0,66**	0,7764	**0,99**	0,2809	**1,32**	0,0613	**1,65**	0,0086	**1,98**	0,0008	**2,31**	0,000046
0,34	0,99993	**0,67**	0,7604	**1,00**	0,2700	**1,33**	0,0582	**1,66**	0,0081	**1,99**	0,0007	**2,32**	0,000042
0,35	0,9997	**0,68**	0,7442	**1,01**	0,2594	**1,34**	0,0551	**1,67**	0,0076	**2,00**	0,0007	**2,33**	0,000038
0,36	0,9995	**0,69**	0,7278	**1,02**	0,2492	**1,35**	0,0522	**1,68**	0,0071	**2,01**	0,0006	**2,34**	0,000035
0,37	0,9992	**0,70**	0,7112	**1,03**	0,2392	**1,36**	0,0495	**1,69**	0,0066	**2,02**	0,0006	**2,35**	0,000032
0,38	0,9987	**0,71**	0,6945	**1,04**	0,2296	**1,37**	0,0469	**1,70**	0,0062	**2,03**	0,0005	**2,36**	0,000030
0,39	0,9981	**0,72**	0,6777	**1,05**	0,2202	**1,38**	0,0444	**1,71**	0,0058	**2,04**	0,0005	**2,37**	0,000027
0,40	0,9972	**0,73**	0,6609	**1,06**	0,2111	**1,39**	0,0420	**1,72**	0,0054	**2,05**	0,0004	**2,38**	0,000024
0,41	0,9960	**0,74**	0,6440	**1,07**	0,2024	**1,40**	0,0397	**1,73**	0,0050	**2,06**	0,0004	**2,39**	0,000022
0,42	0,9945	**0,75**	0,6272	**1,08**	0,1939	**1,41**	0,0375	**1,74**	0,0047	**2,07**	0,0004	**2,40**	0,000020
0,43	0,9926	**0,76**	0,6104	**1,09**	0,1857	**1,42**	0,0354	**1,75**	0,0044	**2,08**	0,0004	**2,41**	0,000018
0,44	0,9903	**0,77**	0,5936	**1,10**	0,1777	**1,43**	0,0335	**1,76**	0,0041	**2,09**	0,0003	**2,42**	0,000016
0,45	0,9874	**0,78**	0,5770	**1,11**	0,1700	**1,44**	0,0316	**1,77**	0,0038	**2,10**	0,0003	**2,43**	0,000014
0,46	0,9840	**0,79**	0,5605	**1,12**	0,1626	**1,45**	0,0298	**1,78**	0,0035	**2,11**	0,0003	**2,44**	0,000013
0,47	0,9800	**0,80**	0,5441	**1,13**	0,1555	**1,46**	0,0282	**1,79**	0,0033	**2,12**	0,0002	**2,45**	0,000012
0,48	0,9753	**0,81**	0,5280	**1,14**	0,1486	**1,47**	0,0266	**1,80**	0,0031	**2,13**	0,0002	**2,46**	0,000011
0,49	0,9700	**0,82**	0,5120	**1,15**	0,1420	**1,48**	0,0250	**1,81**	0,0029	**2,14**	0,0002	**2,47**	0,000010
0,50	0,9639	**0,83**	0,4962	**1,16**	0,1356	**1,49**	0,0236	**1,82**	0,0027	**2,15**	0,0002	**2,48**	0,000009
0,51	0,9572	**0,84**	0,4806	**1,17**	0,1294	**1,50**	0,0222	**1,83**	0,0025	**2,16**	0,0002	**2,49**	0,000008
0,52	0,9497	**0,85**	0,4653	**1,18**	0,1235	**1,51**	0,0209	**1,84**	0,0023	**2,17**	0,0002	**2,50**	0,0000075
0,53	0,9415	**0,86**	0,4503	**1,19**	0,1177	**1,52**	0,0197	**1,85**	0,0021	**2,18**	0,0001	**2,55**	0,0000044
0,54	0,9325	**0,87**	0,4355	**1,20**	0,1122	**1,53**	0,0185	**1,86**	0,0020	**2,19**	0,0001	**2,60**	0,0000026
0,55	0,9228	**0,88**	0,4209	**1,21**	0,1070	**1,54**	0,0174	**1,87**	0,0019	**2,20**	0,0001	**2,65**	0,0000016
0,56	0,9124	**0,89**	0,4067	**1,22**	0,1019	**1,55**	0,0164	**1,88**	0,0017	**2,21**	0,0001	**2,70**	0,0000010
0,57	0,9013	**0,90**	0,3927	**1,23**	0,0970	**1,56**	0,0154	**1,89**	0,0016	**2,22**	0,0001	**2,75**	0,0000006
0,58	0,8896	**0,91**	0,3791	**1,24**	0,0924	**1,57**	0,0145	**1,90**	0,0015	**2,23**	0,0001	**2,80**	0,0000003
0,59	0,8772	**0,92**	0,3657	**1,25**	0,0879	**1,58**	0,0136	**1,91**	0,0014	**2,24**	0,0001	**2,85**	0,00000018
0,60	0,8643	**0,93**	0,3527	**1,26**	0,0836	**1,59**	0,0127	**1,92**	0,0013	**2,25**	0,0001	**2,90**	0,00000010
0,61	0,8508	**0,94**	0,3399	**1,27**	0,0794	**1,60**	0,0120	**1,93**	0,0012	**2,26**	0,0001	**2,95**	0,00000006

Рис. 2: Значения C_α при различных α.

Пример: при уровне значимости $\alpha = 0.05$ и пороговое значение из соотношений (1.8), (1.9) $\mathbf{C_\alpha \approx 1.359}$.

Критерий согласия Колмогорова для непрерывного равномерное распределения

Непрерывное равномерное распределение - распределение случайной вещественной величины, принимающей значения, принадлежащие некоторому промежутку конечно длины, характеризующая тем, что плотность вероятности на этом промежутке почти всюду постоянна.

Функция распределения:

$$F_X(x) \equiv P(X \leqslant x) = \begin{cases} 0, x < a \\ \frac{x-a}{b-a}, a \leqslant x < b \\ 1, x \geqslant b \end{cases} \tag{1.11}$$

Значения теоретическая функция распределения для интервала $[0, 1)$:

$$F(x) = \frac{x - 0}{1 - 0} = x \tag{1.12}$$

Значения эмпирической функции распределения.

Для x_i из выборки X, значение эмпирической функции распределения:

$$F_n(x) = \frac{n_i}{n} \tag{1.13}$$

где n - количество элементов в выборке, n_i - количество элементов в выборке меньших x_i.

1.3 Код программы

```
import math

import matplotlib.pyplot as plt


def linear_congruential_generator(x, alpha, c, m):
    while (True):
        x = (alpha * x + c) % m
        yield x / m


def multiplexial_congruential_generator(x, alpha, m):
    generator = linear_congruential_generator(x, alpha, 0, m)
    while (True):
        yield next(generator)


def mclaren_marsaglia_generator(x_generator, y_generator, k):
    V = [next(x_generator) for _ in range(k)]
    while (True):
        X = next(x_generator)
        Y = next(y_generator)
        j = math.floor(k * Y)
        yield V[j]
        V[j] = X


def hi_squared_test(values, k, critical_value):
    nu = [0] * k
    for value in values:
        nu[math.floor(value * k)] += 1
    p_k = len(values) / k
    hi_squared = 0
    for value in nu:
        hi_squared += ((value - p_k) ** 2) / p_k
    # Дляуровнязначимости      0.05 прити 9- степеняхсвободы .
    return hi_squared < critical_value, hi_squared
```

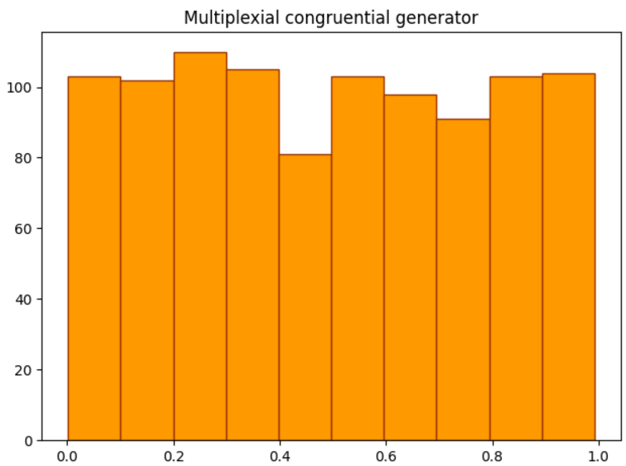
```


def kolmogorov_test(values, critical_value):
    values.sort()
    Dn = 0
    i = 0
    n = len(values)
    for value in values:
        i += 1
        # F(X) = (x-a)/(b-a) = для[ a = 0 иb = 1] = x.
        theoretical_func_res = value
        # колво- значениеввыборкеменьшихтекущегозначенияизвыборки      .
        empirical_function_result = i / n
        Dn = max(Dn, theoretical_func_res - empirical_function_result)
    Dn *= math.sqrt(n)
    return Dn < critical_value, Dn


x0 = 16807
alpha0 = 16807
K = 64

x1 = 8195
alpha1 = 8195
c = 46
k = 64

m = 2 ** 31

hi_squared_critical_value = 16.919
kolmogorov_critical_value = 1.359

mult_congr_gen = multiplexial_congruential_generator(x0, alpha0, m)
x = [next(mult_congr_gen) for _ in range(1000)]
# print('\n'.join(map(str, x)))
hi_squa_test1 = hi_squared_test(x, 10, hi_squared_critical_value)
kolm_test1 = kolmogorov_test(x, kolmogorov_critical_value)

print('Multiplexial congruential generator:')
print('Hi Squared Pirson criteria: ' + str(hi_squa_test1[1]) + ' <= '
      + str(hi_squared_critical_value) if hi_squa_test1[0] else
      'Zero hypothesis fails by Hi Squared Pirson criteria.')
print('Kolmogorov criteria: ' + str(kolm_test1[1]) + ' <= '
      + str(kolmogorov_critical_value) if kolm_test1[0]
      else 'Zero hypothesis fails by Kolmogorov criteria.')

plt.hist(x, 10, ec='#993300', facecolor='#ff9900')
plt.title('Multiplexial congruential generator')
plt.show()

x = linear_congruential_generator(x0, alpha0, 0, m)
y = linear_congruential_generator(x1, alpha1, c, m)
mclar_mars_gen = mclaren_marsaglia_generator(x, y, k)
z = [next(mclar_mars_gen) for _ in range(1000)]
# print('\n'.join(map(str, z)))
```

```python
hi_squa_test2 = hi_squared_test(z, 10, hi_squared_critical_value)
kolm_test2 = kolmogorov_test(z, kolmogorov_critical_value)

print('\nMcLaren marsaglia generator:')
print('Hi Squared Pirson criteria: ' + str(hi_squa_test2[1]) + ' <= '
      + str(hi_squared_critical_value) if hi_squa_test2[0]
      else 'Zero hypothesis fails by Hi Squared Pirson criteria.')
print('Kolmogorov criteria: ' + str(kolm_test2[1]) + ' <= '
      + str(kolmogorov_critical_value) if kolm_test2[0]
      else 'Zero hypothesis fails by Kolmogorov criteria.')

plt.hist(z, 10, ec='#666633', facecolor="#99ff33")
plt.title('McLaren marsaglia generator')
plt.show()
```

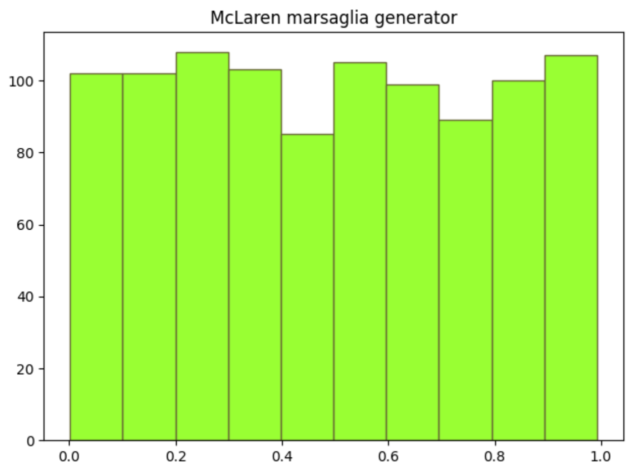
1.4 Результат выполнения

```
Multiplexial congruential generator:
Hi Squared Pirson criteria: 7.300000000000001 <= 16.919
Kolmogorov criteria: 0.12043782997824995 <= 1.359

McLaren marsaglia generator:
Hi Squared Pirson criteria: 4.840000000000001 <= 16.919
Kolmogorov criteria: 0.1608008491673147 <= 1.359
```

Рис. 3: Результат выполнения программы: проверка критерием согласия Пирсона и критерием согласия Колмогорова.



(a) Диаграмма выборки, полученной мультипликативным конгруэнтным методом.



(b) Диаграмма выборки, полученной методом Макларена-Марсальи.