МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ   
РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

**«БЕЛГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ**

**ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ им. В. Г. ШУХОВА»**

**(БГТУ им. В.Г. Шухова)**

Кафедра программного обеспечения вычислительной техники и автоматизированных систем

Лабораторная работа №2

по дисциплине «Системный анализ и обработка информации»

тема: «Разработка датчика случайных чисел распределённых по произвольному закону»

Выполнил: ст. группы ВТ-32

Воскобойников И. С.

Проверил: Полунин А. И.

Белгород 2021 г.

**Цель работы:** Получение теоретических знаний о принципах моделирования физических систем с помощью метода сил и моментов.

**Задания к лабораторной работе**

Разработать датчик случайных чисел, распределенных по заданному закону. Вид закона и интервал распределения задает сам учащийся. При задании функции плотности вероятности необходимо учесть, что в области ее существования она положительна, интеграл от нее на интервале [a,b] равен единице. Для удовлетворения первому условию студент должен проанализировать выбранную функцию и выбрать интервал [a,b] там, где она положительна. Для удовлетворения второму условию поступаем следующим образом.

Вычисляем интеграл от функции на интервале [a,b]. Получаем

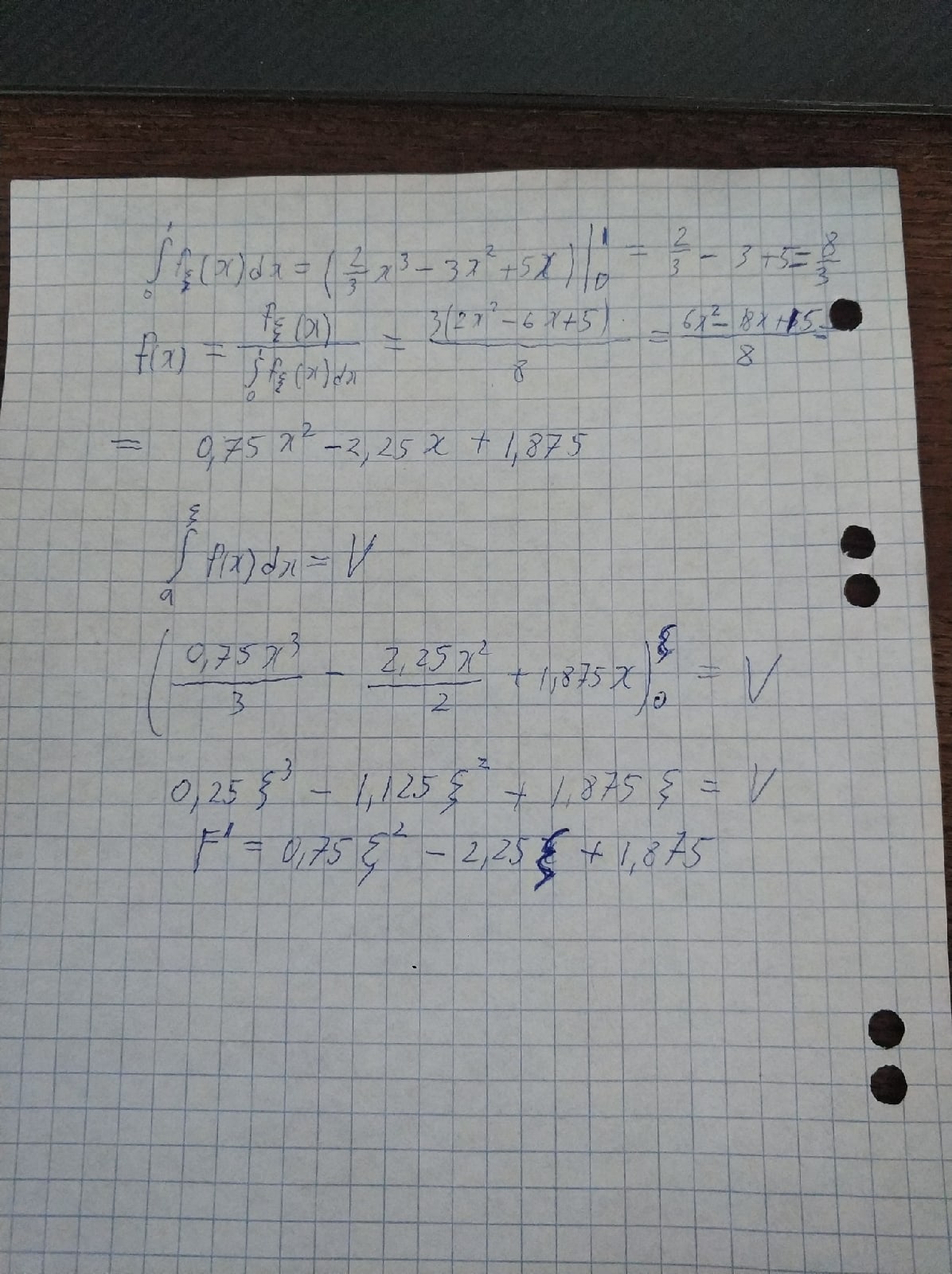
S=.

Задаем новую функцию плотности вероятности

Интеграл от нее на интервале [a,b] равен единице. Для этой функции разрабатываем датчик случайных чисел.

При задании использовать нелинейную зависимость.

**Вариант 3.**



**Код программы**

import random

import matplotlib.pyplot as plt

import math

#Функция возвращающая значение СВ

def get\_f():

h = 2 \* 1e-6

i = 0

res = list()

x = list()

b = math.e \*\* 2 - 1

while i < 2:

x.append(i)

res.append(math.e \*\* i / b)

i += h

return x, res

def get\_value(v):

return math.log((math.e \*\* 2 - 1) \* v + 1)

#Функция генерирующая выборку из N чисел

def get\_samp(N):

samp = list() #Выборка

for i in range(N):

samp.append(get\_value(random.random())) #Добавляем в выборку число полученое по выведенной формуле

return samp

samp = get\_samp(1000000)

plt.hist(samp, density=True, bins=30)

x, samp2 = get\_f()

plt.plot(x, samp2)

plt.show()

