Федеральное государственное бюджетное образовательное учреждение высшего образования

«Сибирский государственный университет телекоммуникаций и информатики» (СибГУТИ)

Кафедра ВС

Лабораторная работа 2

по дисциплине «Моделирование»

Выполнил: ст. гр. ЗМП-41 Лёвкин И. А.

Проверил: Родионов А.С.

Новосибирск 2025

**Задание**

Построить датчик точек, равномерно распределённых в заданной области и по её периметру.

**Ход работы:**

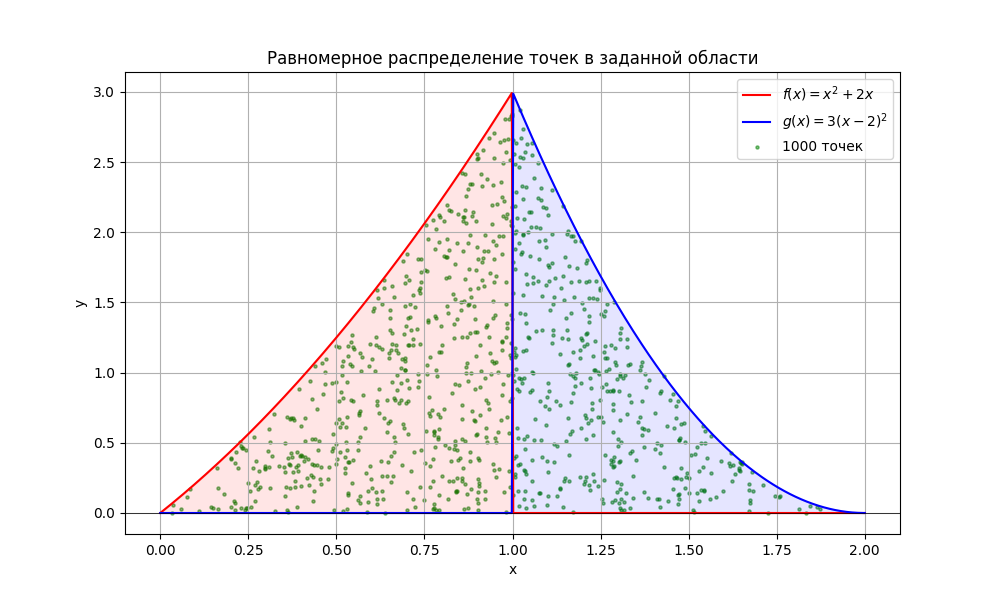
1. Даны 2 функции:
2. Точка пересечения фнукций
3. Нахождение функции распределения:
4. Рассчитаем длину дуг

Введём дополнительную величину a:

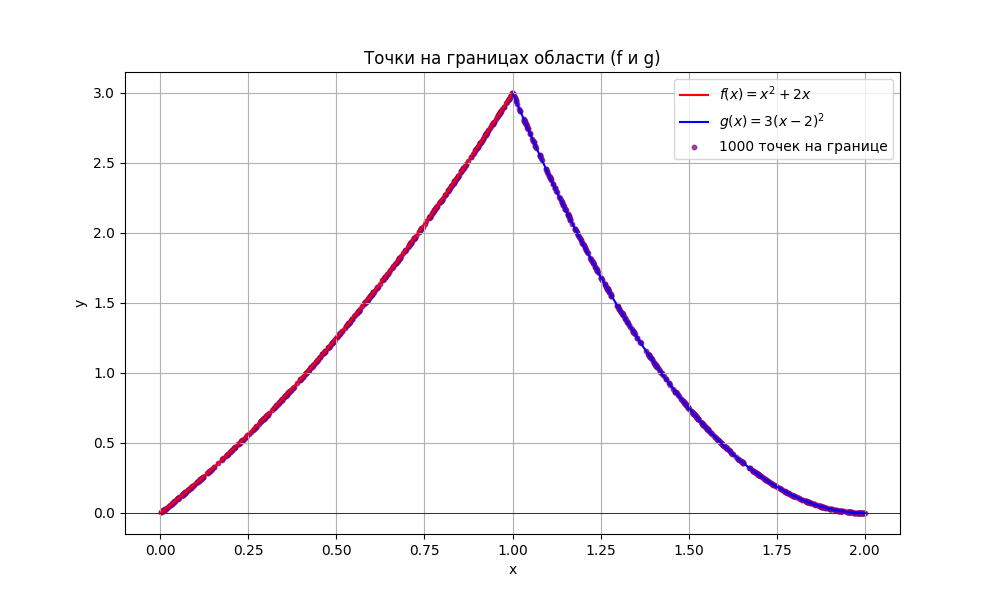
– равномерное распределение от 0 до ,

**Результаты:**

Распределение по области:



Распределение по периметру:

****

**Листинг**

**Часть 1**

import numpy as np

import matplotlib.pyplot as plt

def f(x):

return x\*\*2 + 2 \* x

def g(x):

return 3 \* (x - 2) \*\* 2

def generate\_points(n):

points = []

while len(points) < n:

x = np.random.uniform(0, 2)

y = np.random.uniform(0, 3)

if (x <= 1 and y <= f(x)) or (x > 1 and y <= g(x)):

points.append((x, y))

return np.array(points)

def plot\_points(n):

points = generate\_points(n)

x = np.linspace(0, 2, 500)

y\_f = [f(xi) if xi <= 1 else 0 for xi in x]

y\_g = [g(xi) if xi >= 1 else 0 for xi in x]

plt.figure(figsize=(10, 6))

plt.plot(x, y\_f, "r-", label="$f(x) = x^2 + 2x$")

plt.plot(x, y\_g, "b-", label="$g(x) = 3(x-2)^2$")

plt.scatter(

points[:, 0], points[:, 1], s=5, c="green", alpha=0.5, label=f"{n} точек"

)

plt.fill\_between(x, y\_f, color="red", alpha=0.1)

plt.fill\_between(x, y\_g, color="blue", alpha=0.1)

plt.axhline(0, color="black", linewidth=0.5)

plt.xlabel("x")

plt.ylabel("y")

plt.legend()

plt.title("Равномерное распределение точек в заданной области")

plt.grid(True)

plt.show()

plot\_points(n=1000)

**Часть 2**

import numpy as np

import matplotlib.pyplot as plt

def f(x):

return x\*\*2 + 2 \* x

def g(x):

return 3 \* (x - 2) \*\* 2

def generate\_points\_on\_boundary(n):

points = []

for \_ in range(n):

if np.random.rand() < 0.5:

# Точка на f(x): x ∈ [0, 1]

x = np.random.uniform(0, 1)

y = f(x)

else:

# Точка на g(x): x ∈ [1, 2]

x = np.random.uniform(1, 2)

y = g(x)

points.append((x, y))

return np.array(points)

def plot\_boundary\_points(n):

points = generate\_points\_on\_boundary(n)

x = np.linspace(0, 2, 500)

y\_f = [f(xi) if xi <= 1 else np.nan for xi in x]

y\_g = [g(xi) if xi >= 1 else np.nan for xi in x]

plt.figure(figsize=(10, 6))

plt.plot(x, y\_f, "r-", label="$f(x) = x^2 + 2x$")

plt.plot(x, y\_g, "b-", label="$g(x) = 3(x - 2)^2$")

plt.scatter(

points[:, 0],

points[:, 1],

s=10,

c="purple",

alpha=0.7,

label=f"{n} точек на границе",

)

plt.axhline(0, color="black", linewidth=0.5)

plt.xlabel("x")

plt.ylabel("y")

plt.legend()

plt.title("Точки на границах области (f и g)")

plt.grid(True)

plt.show()

plot\_boundary\_points(n=1000)