МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ “ЛЬВІВСЬКА ПОЛІТЕХНІКА”

ІНСТИТУТ ПРИКЛАДНОЇ МАТЕМАТИКИ ТА ФУНДАМЕНТАЛЬНИХ НАУК

Кафедра прикладної математики

**ЗВІТ**

про виконання лабораторної роботи №2 із дисципліни:

“**Математичні основи штучного інтелекту”**

Виконав:

студент групи ПМ-31

Матолін Марко

Прийняв:

доцент кафедри ПМ

Пабирівський В.В.

Львів ‒ 2023

**Тема:** «Обчислення інтегралу методом Монте-Карло»

# Завдання

Розробити програмну реалізацію методу Монте-Карло для обчислення оцінки значення визначеного інтеграла.

Етапи виконання завдання

1. Вивчити із використанням запропонованих літературних джерел зміст та способи застосування методу МонтеКарло. Для обчислення значення визначеного інтеграла використати метод Монте-Карло для оцінки площі фігури під графіком кривої підінтегральної функції.
2. Обрати для тестового прикладу одну додатно визначену функцію, значення якої можна точно визначити аналітично на заданому інтервалі (для визначеності можна обирати інтервал від 0 до 1, а у якості функції – просту поліноміальну, тригонометричну або показникову).
3. Обрати як основну задачу обчислення значення визначеного інтегралу від деякої додатно визначеної у заданому інтервалі функції, інтеграл від якої не можна порахувати аналітично, проте яка є точно визначеною на цьому інтервалі (наприклад, 𝑓(𝑥) = 𝑒𝑥2 в інтервалі [1, 2]).
4. Реалізувати допоміжну функцію, що обчислюватиме точне значення інтегралу від тестової підінтегральної функції.
5. Реалізувати допоміжну функцію для генерування випадкової точки на

координатній площині – пара рівномірно розподілених випадкових значень (x, y).

1. Реалізувати допоміжну функцію, що повертатиме точне значення

підінтегральної функції в заданій точці, причому передбачити два режими обчислень: 1) значення тестової функції, 2) значення основної функції.

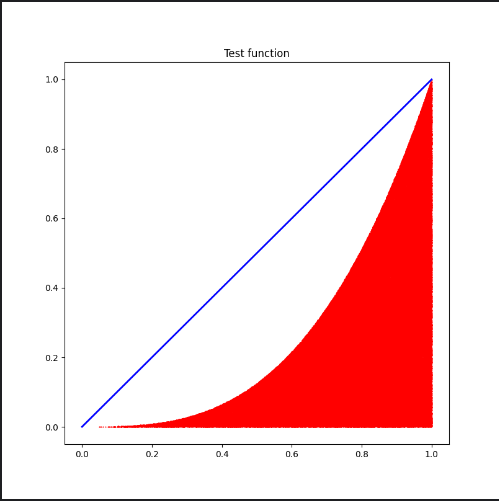
1. Безпосередньо реалізувати алгоритм Монте-Карло для обчислення значення визначеного інтегралу від підінтегральної функції. Передбачити візуалізацію результату роботи алгоритму на графіку.
2. Провести обчислення для тестового прикладу та для основної задачі. Розрахувати похибки (абсолютна, відносна) у тестовому прикладі та провести оцінку похибок основної задачі.

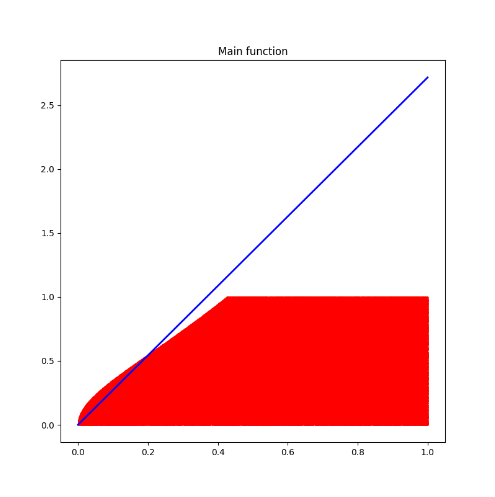
# Варіант-3

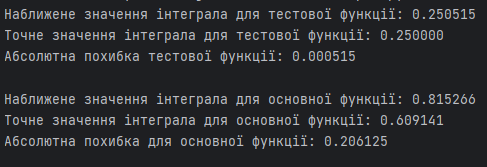
Код програми:

import math  
import random  
import matplotlib.pyplot as plt  
  
# Функція, що обчислює точне значення інтегралу від тестової підінтегральної функції  
def exact\_value\_g():  
 return 1/4  
  
# Функція, що обчислює точне значення інтегралу від основної підінтегральної функції  
def exact\_value\_h():  
 return (2\*math.e - 3)/4  
  
# Функція, що генерує випадкову точку на координатній площині  
def generate\_point():  
 x = random.uniform(0, 1)  
 y = random.uniform(0, 1)  
 return (x, y)  
  
# Функція, що повертає значення підінтегральної функції в заданій точці  
def integrand\_value(x, mode):  
 if mode == "test":  
 return x\*\*3  
 elif mode == "main":  
 return math.sqrt(x) \* math.exp(x)  
  
num\_points = 1000000  
  
# Кількість точок, що потрапили під графік кривої  
num\_hits\_g = 0  
num\_hits\_h = 0  
  
# Списки для зберігання координат точок, які потрапили під графік кривої  
x\_hits\_g = []  
y\_hits\_g = []  
x\_hits\_h = []  
y\_hits\_h = []  
  
# Генерація випадкових точок та підрахунок кількості точок, що потрапили під графік кривої  
for i in range(num\_points):  
 x, y = generate\_point()  
 if y <= integrand\_value(x, "test"):  
 num\_hits\_g += 1  
 x\_hits\_g.append(x)  
 y\_hits\_g.append(y)  
 if y <= integrand\_value(x, "main"):  
 num\_hits\_h += 1  
 x\_hits\_h.append(x)  
 y\_hits\_h.append(y)  
  
# Обчислення оцінки інтегралу за допомогою методу Монте-Карло  
approx\_value\_g = num\_hits\_g / num\_points  
approx\_value\_h = num\_hits\_h / num\_points  
  
error\_g = abs(approx\_value\_g - exact\_value\_g())  
error\_h = abs(approx\_value\_h - exact\_value\_h())  
  
print(f"Наближене значення інтеграла для тестової функції: {approx\_value\_g:.6f}")  
print(f"Точне значення інтеграла для тестової функції: {exact\_value\_g():.6f}")  
print(f"Абсолютна похибка тестової функції: {error\_g:.6f}\n")  
  
print(f"Наближене значення інтеграла для основної функції: {approx\_value\_h:.6f}")  
print(f"Точне значення інтеграла для основної функції: {exact\_value\_h():.6f}")  
print(f"Абсолютна похибка для основної функції: {error\_h:.6f}\n")  
  
plt.figure(figsize=(8, 8))  
plt.plot(x\_hits\_g, y\_hits\_g, "r.", markersize=1)  
plt.plot([0, 1], [0, 1], "b-", linewidth=2)  
plt.title("Test function")  
plt.show()  
  
plt.figure(figsize=(8, 8))  
plt.plot(x\_hits\_h, y\_hits\_h, "r.", markersize=1)  
plt.plot([0, 1], [0, math.e], "b-", linewidth=2)  
plt.title("Main function")  
plt.show()

**Результат виконання:**







**Висновок:**

На лабораторній роботі я навчилась обичислювати інтеграл Монте-Карло та реалізувала алгоритм на мові програмування Python.