Министерство образования и науки Российской Федерации

Федеральное государственное автономное образовательное учреждение высшего образования «Национальный исследовательский университет   
«Московский институт электронной техники»

Кафедра высшей математики №1

Стадник Александр Михайлович

Лабораторная работа № 3

по теме «Этапы построения математической модели»

Направленность (профиль) «Применение математических методов к решению инженерных и естественнонаучных задач»

Вычисление числа методом Монте-Карло

Студент Стадник А.М

Москва 2022

Объект исследования задачи

Задача

# Содержательная постановка задачи

Разработать математическую модель, позволяющую вычислить число через площадь круга методом Монте Карло. Модель должна:

* Вычислять число при четверти окружности
* Вычислять число при полной окружности

Исходные данные:

* – количество точек (бросаний)
* – оригинальное число для сравнения результатов и уточнения точности.
* – количество проведенных опытов – для повышения точности.

# Концептуальная постановка задачи

Все броски проводятся случайным образом и должны попасть в окружность или четверть окружности радиусом 1. Применим следующие гипотезы:

* Координаты броска являются случайными.
* Учитываем только попадания в окружность или четверть.

Сокращенная формулировка задачи концептуальной постановки задачи:

* Сравнить точность числа при бросании на окружность и четверть.

# Математическая постановка задачи

Для решения задачи используется формула для площади окружности

и формулой круга , также учтем отношение , отсюда

# Качественный анализ и проверка конкретности модели

Контроль размерности:

* Все расчеты приведены в условных единицах

# Выбор и обоснование методов решения

# Аналитический (численный) метод

import math

import random

dots\_quantity = 1\_000\_000

PI = math.pi

def main() -> None:

def quater\_circle() -> float:

result\_pi = 0.0

for dot in range(dots\_quantity):

x = random.random() # [0.0, 1.0)

y = random.random() # [0.0, 1.0)

if math.sqrt(x\*\*2 + y\*\*2) <= 1.0:

result\_pi += 1.0

result\_pi = 4 \* (result\_pi / dots\_quantity)

return result\_pi

def full\_circle() -> float:

result\_pi = 0.0

for dot in range(dots\_quantity):

x = random.uniform(-1.0, 1.0)

y = random.uniform(-1.0, 1.0)

if math.sqrt(x\*\*2 + y\*\*2) <= 1.0:

result\_pi += 1.0

result\_pi = 4 \* (result\_pi / dots\_quantity)

return result\_pi

tests\_quantity = 5

quater\_cicle\_average\_PI, full\_cicle\_average\_PI = 0.0, 0.0

for i in range(tests\_quantity):

quater\_cicle\_average\_PI += quater\_circle()

full\_cicle\_average\_PI += full\_circle()

quater\_cicle\_average\_PI /= tests\_quantity

full\_cicle\_average\_PI /= tests\_quantity

print(

f"Точность числа PI для четверти круга на {tests\_quantity} при {dots\_quantity} бросаниях:",

PI / quater\_cicle\_average\_PI,

)

print(

f"Точность числа PI для целого круга на {tests\_quantity} при {dots\_quantity} бросаниях:",

PI / full\_cicle\_average\_PI,

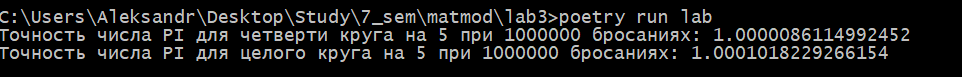
)

# Проверка адекватности модели

Данная математическая модель для решения поставленной задачи годится. По полученным данным, при 5 тестах и количестве бросков в каждый тест равном 1.000.000 можно получить число с точностью до 5 знака после запятой

# Практическое использование построенной модели

Пример работы программы:



По результатам исследования можно сделать следующие вывод:

* Для расчета числа методом Монте Карло лучше использовать четверть окружности. Можно сделать вывод, что из-за более концентрированного количества бросков, площадь круга выходит точнее из-за этого и точность выше