

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ**

**ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ**

**«ДОНСКОЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

**(ДГТУ)**

Факультет Информатика и вычислительная техника

Кафедра Кибербезопасность информационных систем

**ЛАБОРАТОРНАЯ РАБОТА № 4**

на тему «Вероятностное программирование»

Выполнил обучающийся гр. ВКБ31

Анисимов Даниил Георгиевич

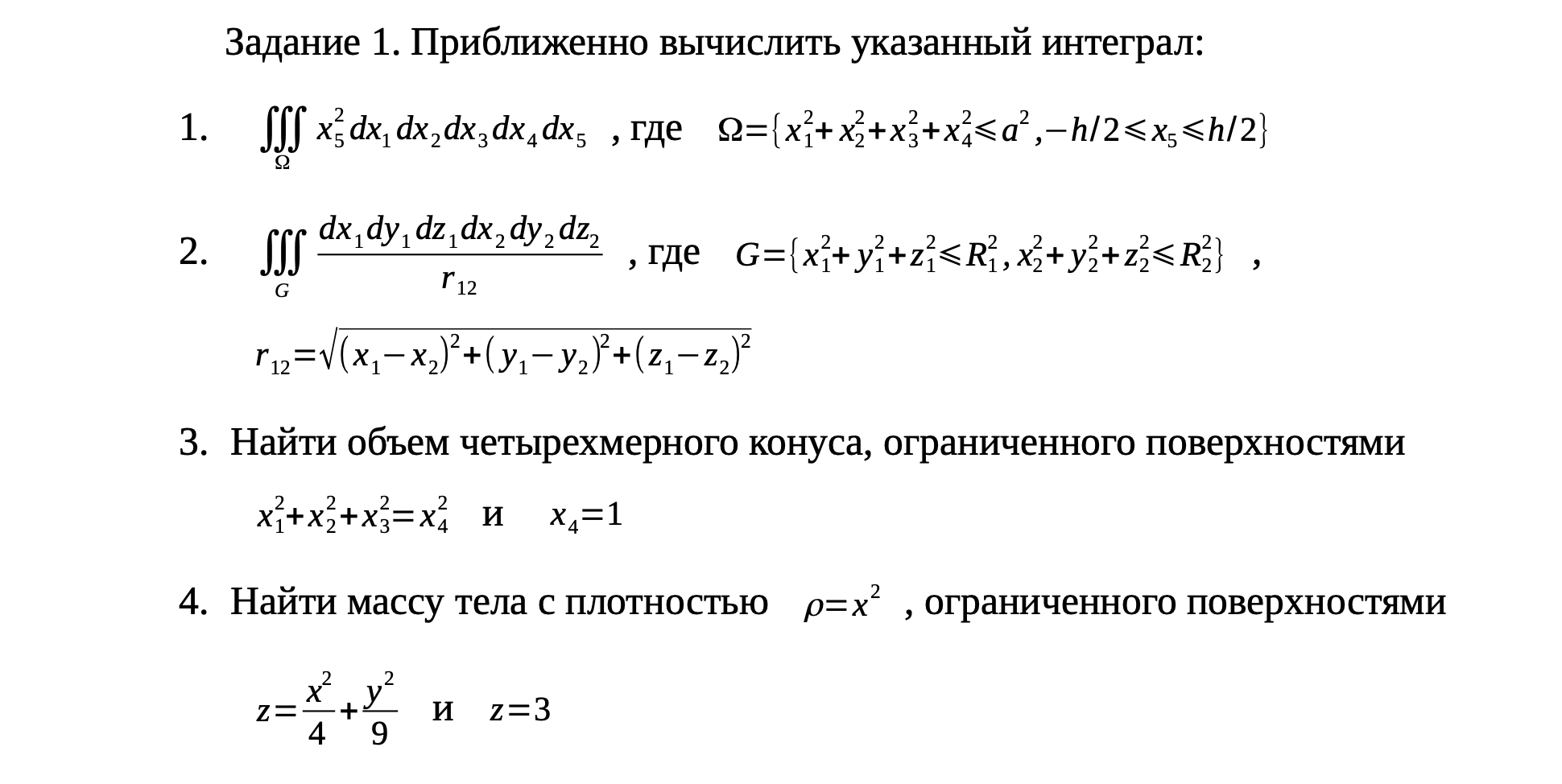
Проверил

Доцент, Савельев Василий Александрович

Ростов-на-Дону

2022

Вероятностное программирование включает в себя широкий выбор задач, моделей и методов, опирающихся на статистические закономерности. При решении задач может возникать необходимость использовать случайные (псевдослучайные) числа. Практически все современные системы программирования содержат генераторы последовательности псевдослучайных чисел. Однако качество таких генераторов может очень сильно отличаться. Поэтому при использовании генераторы следует тестировать на качество последовательности с помощью наборов тестов dieharder или тестов NIST SP800-22. Если встроенный генератор не проходит тесты – следует найти или реализовать качественный PRNG (pseudorandom numbers generator). Использование библиотечных или опубликованных PRNG не возбраняется.



**from** scipy **import** integrate

**import** numpy **as** np

f = **lambda** x, y, z : x

g = **lambda** x : 0

h = **lambda** x : (1 - x) / 2

q = **lambda** x, y : 0

r = **lambda** x, y : 1 - x - 2 \* y

v, err = integrate.tplquad(f, 0, 1, g, h, q, r)

**print**(v)

**import** math

pi = math.pi

**def** volume(r, h):

**return** ((1 / 3) \* pi \* r \* r \* h)

**def** surfacearea(r, s):

**return** (pi \* r \* s + pi \* r \* r)

radius = float(5)

height = float(12)

slat\_height = float(13)

**print**("Volume Of Cone : ", surfacearea(radius, slat\_height))

**from** scipy **import** integrate

**import** numpy **as** np

**def** f(x, y):

**return** x \* y

**def** h(x):

**return** x

v, err = integrate.dblquad(f, 1, 2, **lambda** x: 1, h)

**print**(v)