**Сибирский государственный университет**

**телекоммуникаций и информатики**

Институт ИВТ

Кафедра вычислительных систем

**Отчёт по лабораторной работе № 1**

Дисциплина: Моделирование

На тему: Датчики случайных чисел.

Группа МГ-211

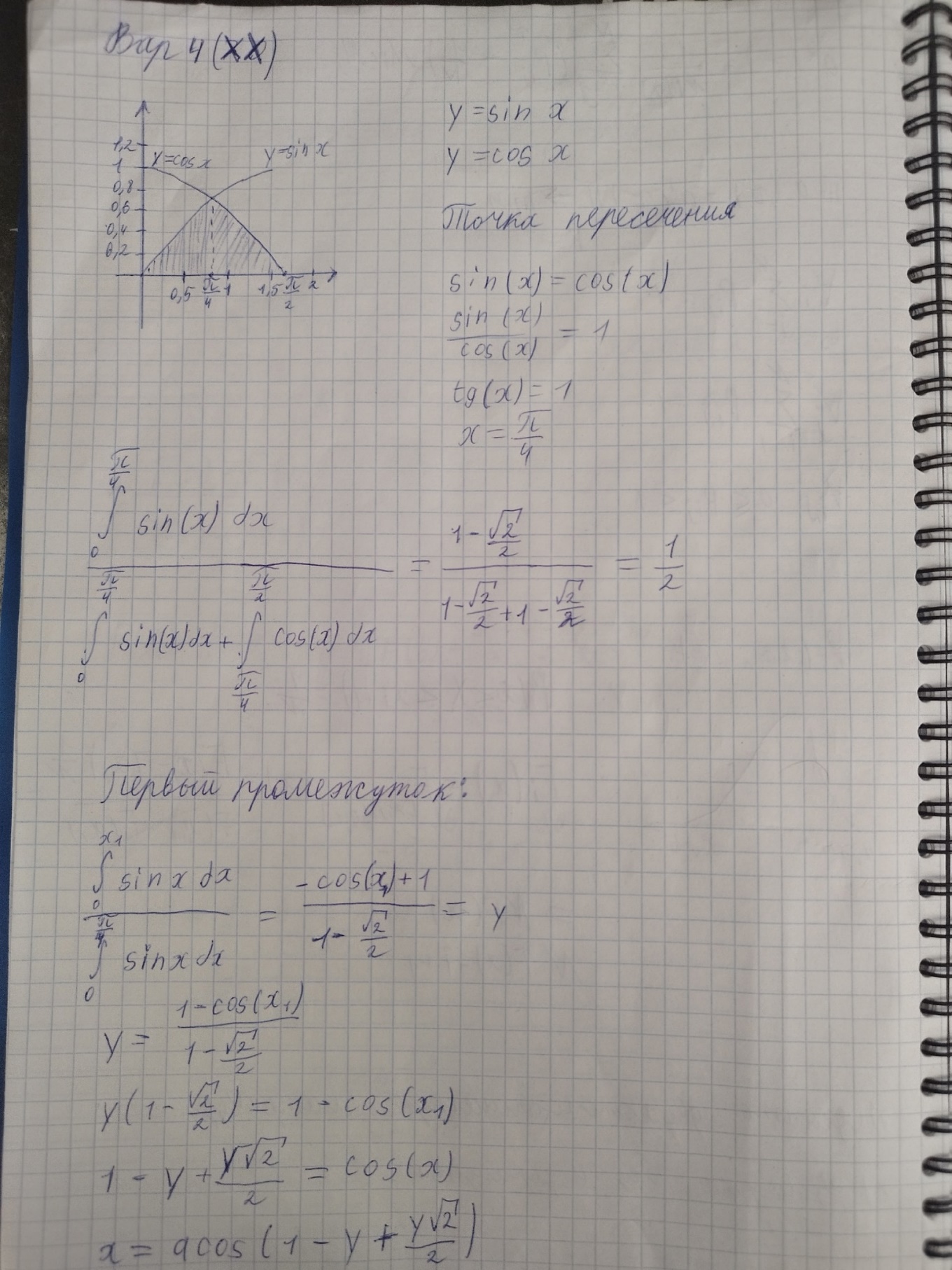
Выполнил: Бурдуковский И.А.

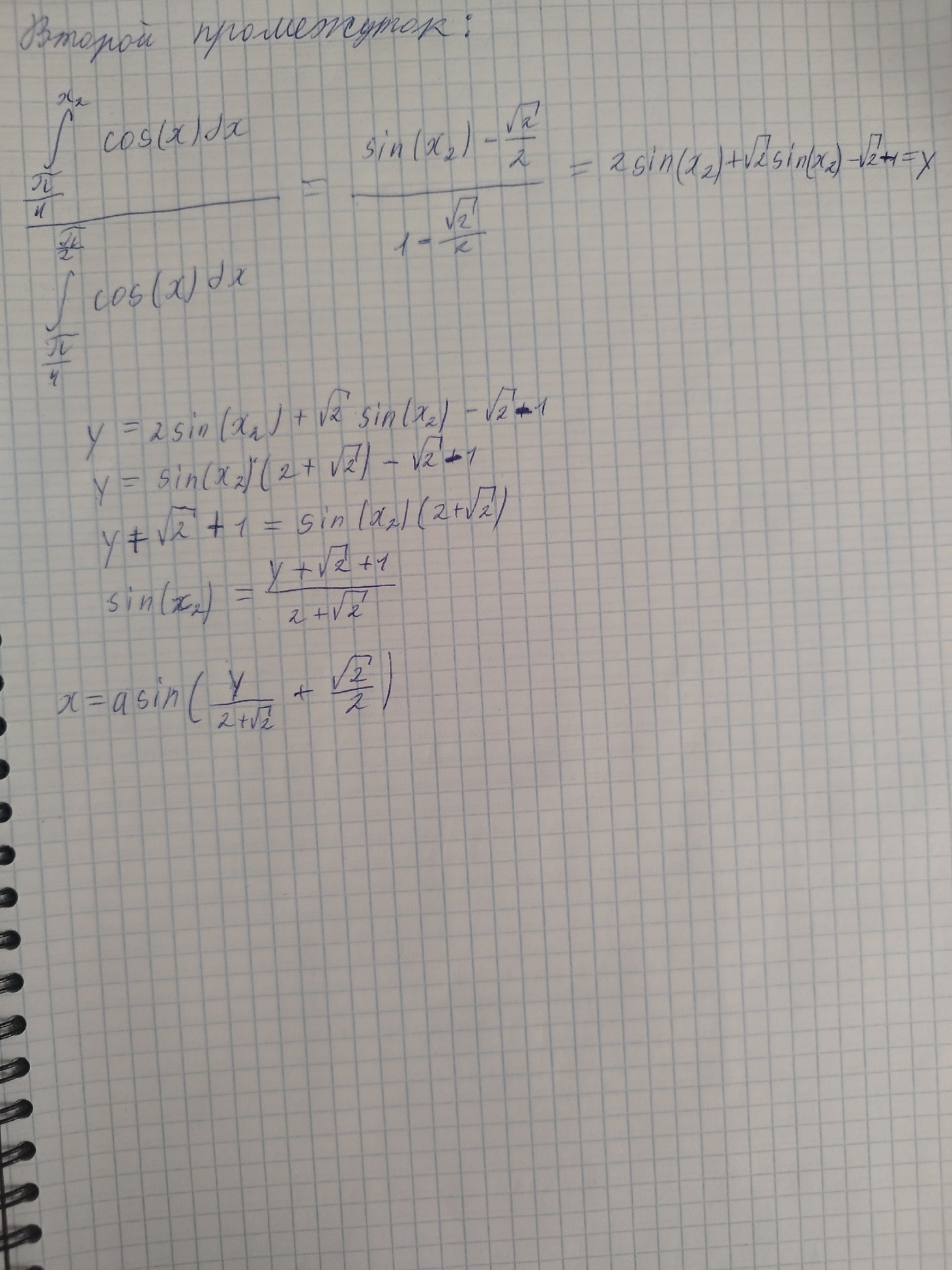
Проверил: д.т.н, доцент кафедры ВС Родионов А.С.

Новосибирск

2023 г.

1. Датчик случайных чисел для области



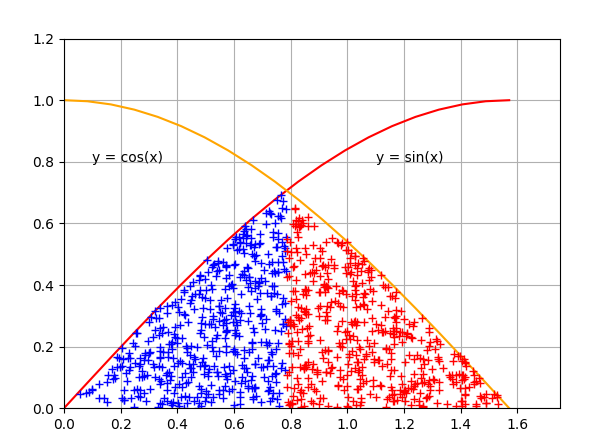


Код программы:

import matplotlib.pyplot as plt  
import math  
import random  
import numpy as np

n = 500  
  
plt.grid()  
  
x\_sin = np.linspace(0, math.pi / 2, 20)  
y\_sin = np.sin(x\_sin)  
plt.plot(x\_sin, y\_sin, color="red")  
  
x\_cos = np.linspace(0, math.pi / 2, 20)  
y\_cos = np.cos(x\_cos)  
plt.plot(x\_cos, y\_cos, color="orange")  
  
x1 = []  
x2 = []  
f1 = []  
f2 = []  
  
# Функция y = sin(x)  
for i in range(n):  
  
 rand = random.random()  
 x1.append(math.acos(1 - rand + (rand \* math.sqrt(2)) / 2))  
  
 f1.append(math.sin(x1[i]) \* random.uniform(0, 1))  
  
# Функция y = сos(x)  
for i in range(n):  
  
 rand = random.random()  
 x2.append(math.asin(rand / (2 + math.sqrt(2)) + (math.sqrt(2) / 2)))  
  
 f2.append(math.cos(x2[i]) \* random.uniform(0, 1))  
  
plt.plot(x1, f1, 'b+')  
  
plt.plot(x2, f2, 'r+')  
  
plt.xlim(0, 1.75)  
plt.ylim(0, 1.2)  
plt.text(0.1, 0.8, 'y = сos(x)')  
plt.text(1.1, 0.8, 'y = sin(x)')  
  
plt.show()

Результат:



1. Датчик случайных чисел для отрезков
2. Для построения датчика случайных чисел для отрезков, заданных функциями: y=sin(x) и y=cos(x), необходимо найти частное интегралов длин кривых.

Длина кривой находится по формуле

а) Для функции : . - данный интеграл не берётся.

Воспользуемся методом линейной аппроксимации.

Получим уравнение:



Вместо x подставим и получим координату x искомой точки.

Найдём y как . Построим график.

б) Для функции : . - данный интеграл не берётся.

Воспользуемся методом линейной аппроксимации.

Получим уравнение:



Вместо x подставим и получим координату x искомой точки.

Найдём y как . Построим график.

Код программы:

import matplotlib.pyplot as plt  
import math  
import random  
import numpy as np

n = 50  
middle = np.pi / 4  
parts = 5  
plt.grid()  
  
x\_sin = np.linspace(0, middle, 20)   
y\_sin = np.sin(x\_sin)  
t\_sin = np.polyfit(x\_sin, y\_sin, parts)  
f\_sin = np.poly1d(t\_sin)  
x\_sin\_full = np.linspace(0, np.pi / 2, 20)  
y\_sin\_full = np.sin(x\_sin\_full)  
print(f\_sin)  
plt.plot(x\_sin\_full, y\_sin\_full, label='y = sin(x)', color="green")  
plt.plot(x\_sin, y\_sin, x\_sin, f\_sin(x\_sin), label='y = sin(x) full', color="red")  
  
x\_cos = np.linspace(middle, np.pi / 2, 20)   
y\_cos = np.cos(x\_cos)  
t\_cos = np.polyfit(x\_cos, y\_cos, parts)  
f\_cos = np.poly1d(t\_cos)  
x\_cos\_full = np.linspace(0, np.pi / 2, 20)  
y\_cos\_full = np.cos(x\_cos\_full)  
print('\n', f\_cos)  
plt.plot(x\_cos\_full, y\_cos\_full, label='y = cos(x)', color="magenta")  
plt.plot(x\_cos, y\_cos, x\_cos, f\_cos(x\_cos), label='y = cos(x) full', color="orange")  
  
x1 = []  
x2 = []  
f1 = []  
f2 = []  
  
# Функция y = sin x  
for i in range(n):  
 buf = 0  
 for j in range(parts):  
 buf += f\_sin.c[j] \* random.uniform(0, middle)  
 buf += f\_sin.c[parts]  
 x1.append(buf)  
 f1.append(np.sin(buf))   
  
plt.plot(x1, f1, 'b+')  
  
# Функция y = cos x  
for i in range(n):  
 buf = 0  
 rand = random.uniform(middle, np.pi/2)   
 for j in range(parts):  
 buf += f\_sin.c[j] \* random.uniform(middle, np.pi/2)   
 buf += f\_sin.c[parts]  
 x2.append(buf)   
 f2.append(np.cos(buf))   
  
plt.plot(x2, f2, 'r+')  
  
plt.xlim(0, 1.75)  
plt.ylim(0, 1.2)  
plt.text(0.1, 0.8, 'y = сos(x)')  
plt.text(1.1, 0.8, 'y = sin(x)')  
  
plt.show()

Результат:

