МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ   
РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

**«БЕЛГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ**

**ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ им. В. Г. ШУХОВА»**

**(БГТУ им. В.Г. Шухова)**

Кафедра программного обеспечения вычислительной техники и автоматизированных систем

Лабораторная работа №3

по дисциплине «Теория надежности»

тема: «Муха в треугольнике»

Выполнил: ст. группы ВТ-32

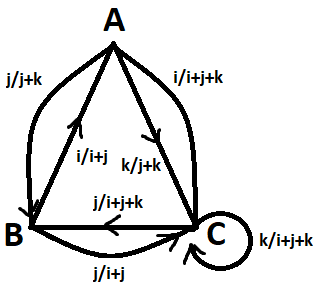
Егоров В. Ю.

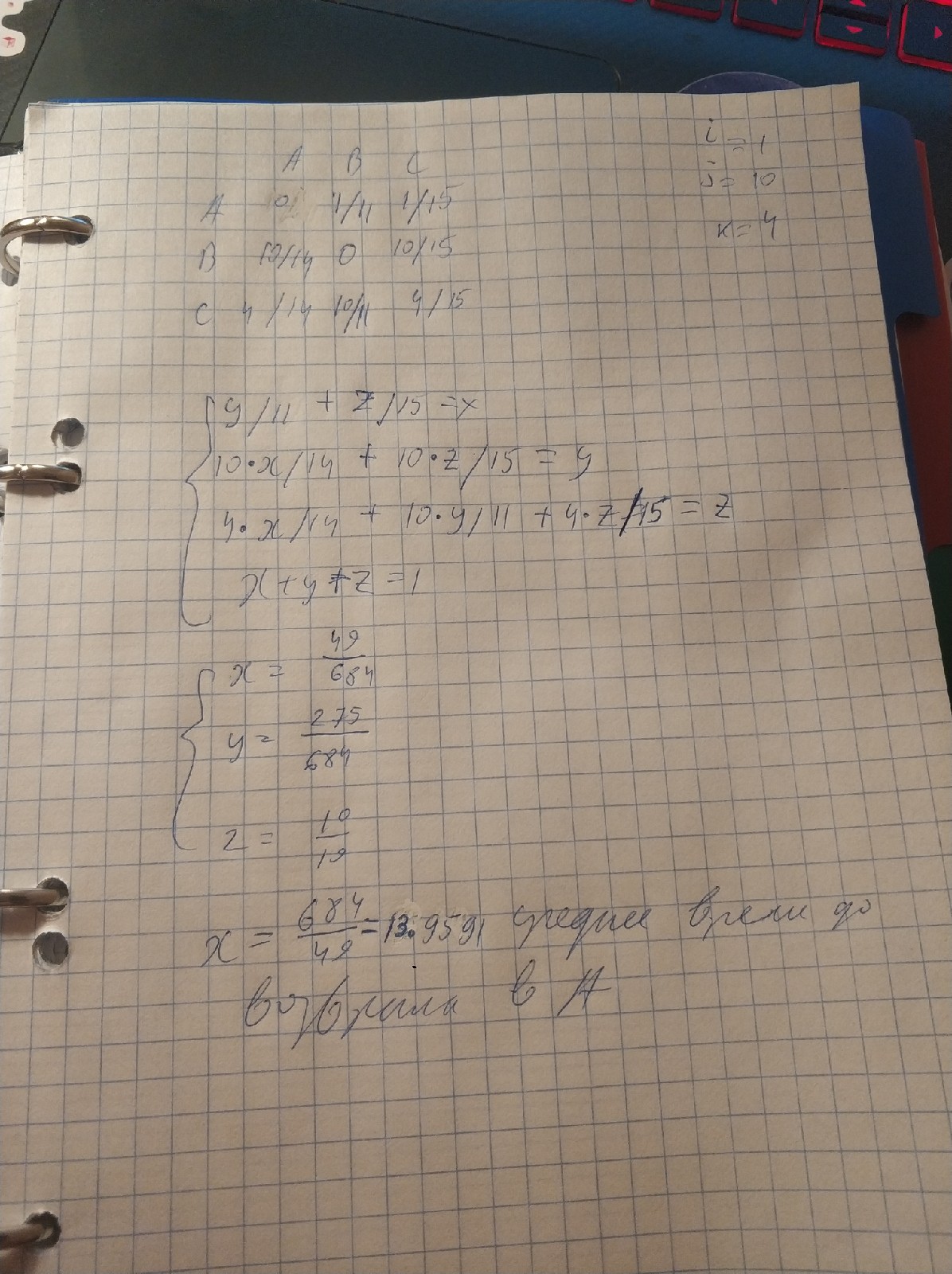
Проверил: Кабалянц П. С.

Белгород 2021 г.

**Задание:**

Имеется треугольник с вершинами A, B, C. В начальный момент времени муха находится в вершине A. Каждую секунду муха перелетает в другую вершину или остается на месте. Ее поведение задается графом марковского процесса (смотри приложенный рисунок). 1) Необходимо определить среднее время, через которое муха вернётся в вершину А. 2) Написать программу, которая имитирует поведение мухи и выводит среднее количество переходов до первого возвращения в точку А. Сравните результаты.





from random import random  
i = 1  
j = 10  
k = 4  
  
  
  
def step(state):  
 random\_value = random()  
  
 if state == **"A"**:  
 if random\_value < (10. / 14.):  
 state = **"B"** else:  
 state = **"C"** return state  
  
 if state == **"B"**:  
 if random\_value < (1. / 11.):  
 state = **"A"** else:  
 state = **"C"** return state  
  
 if state == **"C"**:  
 if random\_value < (1. / 15.):  
 state = **"A"** elif random\_value < (10. / 15.):  
 state = **"B"** else:  
 state = **"C"** return state  
  
 *# возвращает время до возврата в начальное положение*def run():  
 state = **'A'** count\_step = 0  
  
 while True:  
 count\_step += 1  
 state=step(state)  
  
 if (state == **'A'**):  
 return count\_step  
  
  
n = 100000  
  
generate\_result = [run() for \_ in range(n)]  
avg\_ret\_A = sum(generate\_result)/len(generate\_result)  
print(**f"Эмпирическое среднее время до возврата в А =** {avg\_ret\_A}**"**)  
print(**"Теоретическое среднее время до возврата в А = 13.9591"**)  
print(**f"Разница между полученными данными и посчитанными =** {avg\_ret\_A - 13.9591}**"**)  
print(**"Разница не большая значит все расчеты были выполненны верно."**)

