МИНОБРНАУКИ РОССИИ

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 №6 дисциплина: «Технология надежности»

тема: «Без отказов (с очередью)»

Выполнил: ст. группы ВТ-32 Воскобойников И. С.

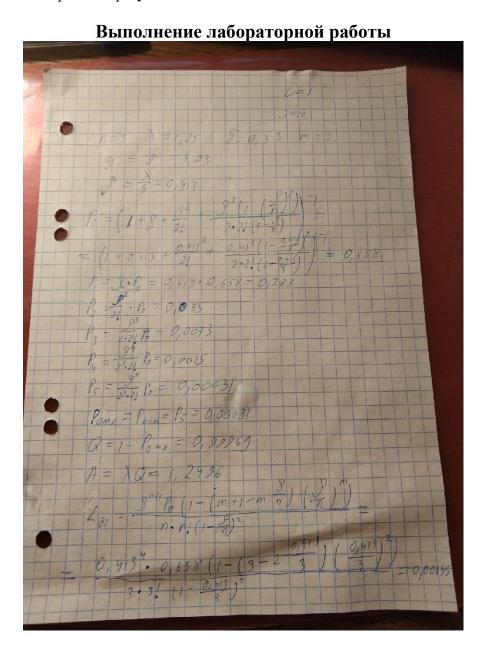
Проверил: Кабалянц П.С.

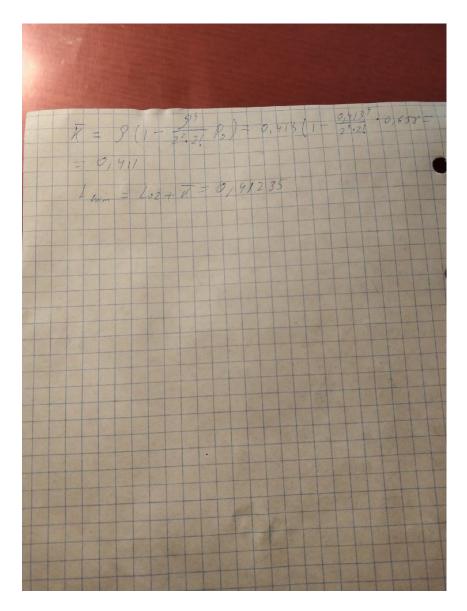
Задание

На сервере есть п каналов передачи сообщений. Среднее время обработки сообщений τ. На сервер поступают сообщения в среднем количестве λ сообщений в минуту. Для хранения сообщений в очереди на сервере выделено место для (n+1) сообщения.

Определить основные характеристики сервера - вероятность очереди, среднее число занятых каналов, средняя длина очереди, среднее число сообщений на сервере; предполагается, что сообщение не получает отказ при занятости всех каналов и очереди длины не больше (n+1). Параметры варианта определяются по формулам: n=3+[(i+j)/8], $\lambda=1+i/4$, $\tau=5/(5+j)$. Здесь квадратные скобки означают взятие целой части, а i,j - последние цифры зачетки.

2) Написать программу, которая имитирует поведение сервера и вычисляет его основные характеристики. Сравните результаты.





Программная реализация

```
import random
import math
i = 1
j = 10
n = 3
m = 2
lmb = 1.25
tau = 0.333
y = 3.03
ro = 0.413
A = 1.2496
Q = 0.99969
Potk = 0.00031
k = 0.411
Lqueue = 0.00135
Lsys = 0.41235
def wait_queue(queue, delta):
    for \overline{i} in range(0, len(queue)):
```

```
queue[i][1] += delta
def free channel(channels):
    for i in range (0, n):
        if channels[i] == 0:
            return i
    return -1
def message proc(channels, delta):
    for i in range(0, len(channels)):
        if (channels[i] > delta):
            channels[i] -= delta
        else:
            channels[i] = 0
def send message (messages, channels, queue):
     aqt = 0
     qm = 0
     unpm = 0
     pos = free channel(channels)
     if pos != -1:
          if len(queue) > 0:
               tmp = queue.pop(0)
               newMessage = tmp[0]
               am += 1
               aqt += tmp[1]
          else:
               newMessage = messages.pop(0)
          channels[pos] = newMessage
     else:
          if len(queue) < m:</pre>
               queue.append([messages.pop(0), 0])
          else:
               unpm += 1
               messages.clear()
     return [aqt, qm, unpm]
channels = []
for i in range(0, n):
    channels.append(0)
queue = []
maxTime = 100000
busyChannels = 0
unProcMessage = 0
totalMessage = 0
avgQueueTime = 0
queueMessage = 0
avgQueueLen = 0
messages = []
t = 0.01
avgTime = 0
for currentTime in range(0, int(maxTime / t)):
    message_proc(channels, t)
    wait queue(queue, t)
    if (random.random() < 1 - math.exp(-1 * lmb * t)):
```

```
messages.append(0.3 + tau - random.random()/2)
        avgTime += messages[0]
        totalMessage += len(messages)
        take mes = send_message(messages, channels, queue)
        avgQueueTime += take mes[0]
        queueMessage += take mes[1]
        unProcMessage += take mes[2]
       messages.clear()
    busyChannels += n - channels.count(0)
    avgQueueLen += len(queue)
loadedChannels = busyChannels / (maxTime / t)
averageTime = avgTime / totalMessage
print("Теоретические значения:")
print("Абсолютная пропускная способность: A = {}".format(A))
print("Относительная пропускная способность: Q = {}".format(Q))
print("Вероятность отказа в обработке: Р отк = {}".format(Potk))
print("Среднее число занятых каналов: `k = {}".format(k))
print("Среднее число заявок в очереди: L оч = {}".format(Lqueue))
print("Среднее число заявок в СМО: L сист = {}".format(Lsys))
print("\nПрограммные значения:")
print("Cepsep paGotaet {} минут, t = {} минуты".format(maxTime, t))
print("Всего получено сообщений: {}".format(totalMessage))
print("Пропущено сообщений: {}".format(unProcMessage))
print("Абсолютная пропускная способность: A = {}".format((totalMessage -
unProcMessage) / maxTime))
print("Относительная пропускная способность: Q = {}".format((totalMessage -
unProcMessage) / totalMessage))
print("Beposthoctb otkasa npu ofpafotke: P_otk = {}".format(unProcMessage /
totalMessage))
print("Среднее число занятых каналов при этом составило: k =
{}".format(busyChannels / (maxTime / t)))
print("Среднее число заявок в очереди: L оч = {}".format(avgQueueLen /
(maxTime / t)))
```

Результат работы программы

```
Теоретические значения:
Абсолютная пропускная способность: А = 1.2496
Относительная пропускная способность: Q = 0.99969
Вероятность отказа в обработке: Р_отк = 0.00031
Среднее число занятых каналов: `k = 0.411
Среднее число заявок в очереди: L_оч = 0.00135
Среднее число заявок в СМО: L_сист = 0.41235
Программные значения:
Сервер работает 10000 минут, t = 0.01 минуты
Всего получено сообщений: 12563
Пропущено сообщений: 4
Абсолютная пропускная способность: А = 1.2459
Относительная пропускная способность: Q = 0.9996816047122502
Вероятность отказа при обработке: Р_отк = 0.0003183952877497413
Среднее число занятых каналов при этом составило: k = 0.414127
Среднее число заявок в очереди: L_оч = 0.014999
```