

*Государственное образовательное учреждение высшего профессионального образования  
«Московский Государственный Технический Университет имени Н. Э. Баумана»*

## ОТЧЕТ

По лабораторной работе №5  
«Система массового обслуживания»

По курсу «Очередь заявок»

Студент: Белова А.А.

Группа: ИУ7-75

Преподаватель: Рудаков И.В.

Москва, 2018

### **Задание**

В информационный центр приходят клиенты через интервал времени  $10 \pm 3$  минуты. Если все три имеющихся оператора заняты, клиенту отказывают в обслуживании. Операторы имеют разную производительность и могут обеспечивать обслуживание среднего запроса пользователя за  $20 \pm 5$ ;  $40 \pm 10$ ;  $40 \pm 20$ . Клиенты стремятся занять свободного оператора с максимальной производительностью. Полученные запросы сдаются в накопитель. Откуда выбираются на обработку. На первый компьютер запросы от 1 и 2-ого операторов, на второй – запросы от 3-его. Время обработки запросов первым и 2-м компьютером равны соответственно 15 и 30 мин. Промоделировать процесс обработки 300 запросов.

Вероятность отказа  $P = (C_{отк}) / (C_{отк} + C_{обсл})$

### **Реализация**

```

op0 = Operator(op0_m, op0_d)
op1 = Operator(op0_m, op0_d)
op2 = Operator(op0_m, op0_d)
comp0 = Computer(comp0_m)
comp1 = Computer(comp0_m)
stuff = [client_gen, op0, op1, op2, comp0, comp1]
current_time = 0
lost = 0
success = 0
start = True
while client_gen.max_req > 0:
    # Find next event
    print(client_gen.max_req)
    current_time, index = client_gen.time_to_start, 0
    for i, device in enumerate(stuff):
        if 0 < device.time_to_start < current_time:
            current_time = device.time_to_start
            index = i
    if index == 0:
        req = client_gen.get_from_queue()
        client_gen.gen_new_time()
        result = op0.add_to_queue(current_time)
        if result:
            continue
        result = op1.add_to_queue(current_time)
        if result:
            continue
        result = op2.add_to_queue(current_time)
        if result:
            continue
        lost += 1
    elif index == 1:
        res = op0.get_from_queue()
        if res == 0:
            continue
        res = comp0.add_to_queue(current_time)
        if res == False:
            lost += 1
    elif index == 2:
        res = op1.get_from_queue()
        if res == 0:
            continue
        res = comp0.add_to_queue(current_time)
        if res == False:
            lost += 1

    elif index == 3:
        res = op2.get_from_queue()
        if res == 0:
            continue
        res = comp1.add_to_queue(current_time)
        if res == False:
            lost += 1

    elif index == 4:
        success += comp0.get_from_queue()

```

## Интерфейс и результаты работы программы:

The screenshot shows a software window titled "лаб" (lab) with a light blue border. The interface is organized into several sections with input fields and a "старт" (start) button.

**Заявки:** (Requests)

	10	delta	2
--	----	-------	---

**Операторы:** (Operators)

0	20	delta	5
1	40	delta	10
2	40	delta	20

**Компьютеры:** (Computers)

0	15
1	30

**Results:**

Всего заявок	300
Вероятность отказа	0.2067
Пропущено	62

**Buttons:** A "старт" (start) button is located at the bottom center of the window.

Рисунок 1. Пример работы программы