



Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет
имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ «Информатика и системы управления»

КАФЕДРА «Программное обеспечение ЭВМ и информационные технологии»

ОТЧЕТ

к лабораторной работе №7

По курсу: «Моделирование»

Студент

ИУ7И-76Б

(Группа)

Нгуен Ф. С.

(Подпись, дата)

(И.О. Фамилия)

Преподаватель

Рудаков И.В.

(Подпись, дата)

(И.О. Фамилия)

Москва, 2021 г.

Оглавление

| | | |
|-------------|-------------------------------------|----------|
| <i>I.</i> | <i>Задача.....</i> | <i>3</i> |
| <i>II.</i> | <i>Теоретическая часть.....</i> | <i>3</i> |
| <i>III.</i> | <i>Экспериментальная часть.....</i> | <i>4</i> |
| <i>IV.</i> | <i>Код программы.....</i> | <i>5</i> |

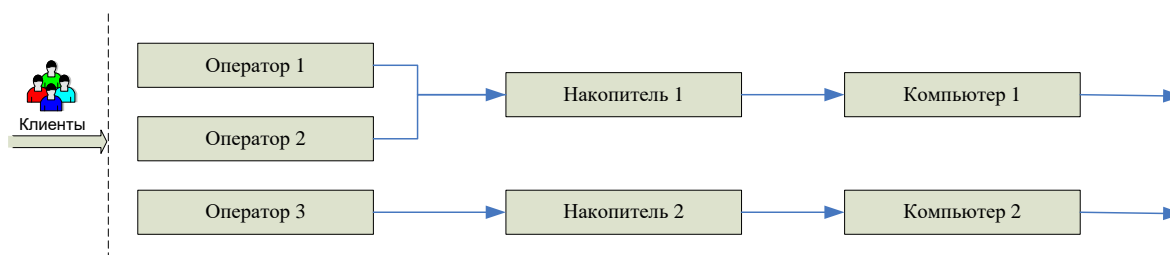
I. Задача

В информационный центр приходят клиенты через интервал времени 10 ± 2 минуты. Если все три имеющихся оператора заняты, клиенту отказывают в обслуживании. Операторы имеют разную производительность и могут обеспечивать обслуживание среднего запроса пользователя за 20 ± 5 ; 40 ± 10 ; 40 ± 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.

II. Теоретическая часть

Необходимо создать концептуальную модель в терминах СМО, определить эндогенные и экзогенные переменные и уравнения модели.

За единицу системного времени выбрать 0,01 минуты.



В процессе взаимодействия клиентов с информационным центром возможно:

1) Режим нормального обслуживания, т.е. клиент выбирает одного из свободных операторов, отдавая предпочтение тому у которого меньше номер.

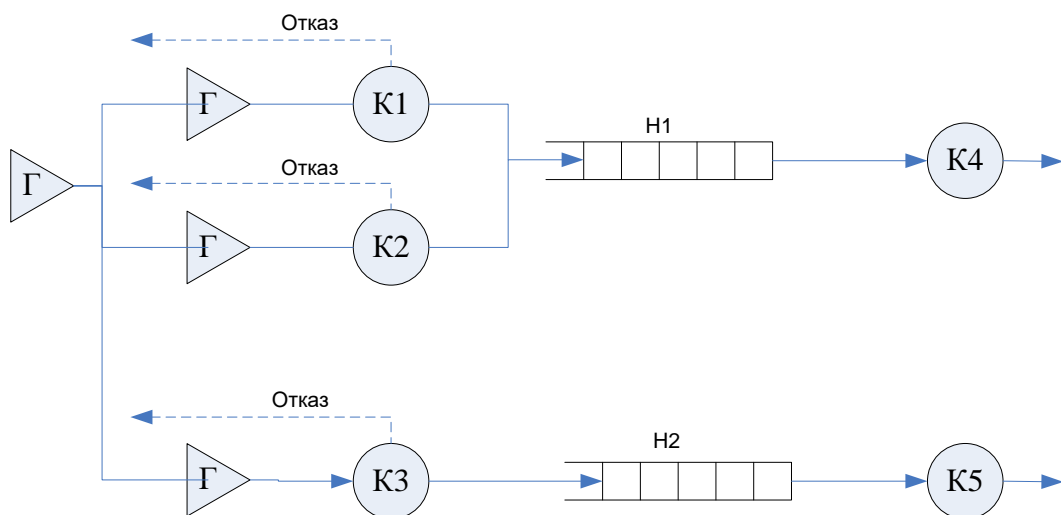
2) Режим отказа в обслуживании клиента, когда все операторы заняты

Переменные и уравнения имитационной модели.

Эндогенные переменные: время обработки задания i -ым оператором, время решения этого задания j -ым компьютером.

Экзогенные переменные: число обслуженных клиентов и число клиентов получивших отказ.

На рисунке представлена концептуальная модель в терминах СМО.



Вероятность отказа:

| | |
|---|-----|
| $P = \frac{C_{\text{откл}}}{C_{\text{откл}} + C_{\text{обсл}}}$ | (1) |
|---|-----|

III. Экспериментальная часть

GPSS World Simulation Report - lab7.20.1

Sunday, December 17, 2021 18:17:41

| | | | | |
|------------|----------|--------|------------|----------|
| START TIME | END TIME | BLOCKS | FACILITIES | STORAGES |
| 0.000 | 3058.002 | 34 | 5 | 0 |

| NAME | VALUE |
|------------------|-----------|
| BLOCK_END | 31.000 |
| BLOCK_OPERATOR_1 | 2.000 |
| BLOCK_OPERATOR_2 | 7.000 |
| BLOCK_OPERATOR_3 | 12.000 |
| BLOCK_PC_1 | 17.000 |
| BLOCK_PC_2 | 23.000 |
| BLOCK_PROCESSED | 29.000 |
| BLOCK_REFUSED | 30.000 |
| COUNT_PROCESSED | 10005.000 |
| COUNT_REJECT | 10006.000 |
| PROB_REJECT | 10007.000 |
| QUEUE_PC_1 | 10003.000 |
| QUEUE_PC_2 | 10008.000 |
| USE_OPERATOR_1 | 10000.000 |
| USE_OPERATOR_2 | 10001.000 |
| USE_OPERATOR_3 | 10002.000 |
| USE_PC_1 | 10004.000 |
| USE_PC_2 | 10009.000 |

| LABEL | LOC | BLOCK TYPE | ENTRY COUNT | CURRENT COUNT | RETRY |
|------------------|-----|------------|-------------|---------------|-------|
| BLOCK_OPERATOR_1 | 1 | GENERATE | 300 | 0 | 0 |
| | 2 | GATE | 300 | 0 | 0 |
| | 3 | SEIZE | 121 | 0 | 0 |
| | 4 | ADVANCE | 121 | 0 | 0 |
| | 5 | RELEASE | 121 | 0 | 0 |
| BLOCK_OPERATOR_2 | 6 | TRANSFER | 121 | 0 | 0 |
| | 7 | GATE | 179 | 0 | 0 |
| | 8 | SEIZE | 59 | 0 | 0 |
| | 9 | ADVANCE | 59 | 0 | 0 |
| | 10 | RELEASE | 59 | 0 | 0 |
| BLOCK_OPERATOR_3 | 11 | TRANSFER | 59 | 0 | 0 |
| | 12 | GATE | 120 | 0 | 0 |
| | 13 | SEIZE | 51 | 0 | 0 |
| | 14 | ADVANCE | 51 | 0 | 0 |
| | 15 | RELEASE | 51 | 0 | 0 |
| BLOCK_PC_1 | 16 | TRANSFER | 51 | 0 | 0 |
| | 17 | QUEUE | 180 | 0 | 0 |
| | 18 | SEIZE | 180 | 0 | 0 |
| | 19 | DEPART | 180 | 0 | 0 |
| | 20 | ADVANCE | 180 | 0 | 0 |
| BLOCK_PC_2 | 21 | RELEASE | 180 | 0 | 0 |
| | 22 | TRANSFER | 180 | 0 | 0 |
| | 23 | QUEUE | 51 | 0 | 0 |
| | 24 | SEIZE | 51 | 0 | 0 |
| | 25 | DEPART | 51 | 0 | 0 |
| BLOCK_PROCESSED | 26 | ADVANCE | 51 | 0 | 0 |
| | 27 | RELEASE | 51 | 0 | 0 |
| | 28 | TRANSFER | 51 | 0 | 0 |
| | 29 | TRANSFER | 231 | 0 | 0 |
| | 30 | TRANSFER | 69 | 0 | 0 |
| BLOCK_END | 31 | SAVEVALUE | 300 | 0 | 0 |
| | 32 | SAVEVALUE | 300 | 0 | 0 |
| | 33 | SAVEVALUE | 300 | 0 | 0 |
| | 34 | TERMINATE | 300 | 0 | 0 |

| FACILITY | ENTRIES | UTIL. | AVE. TIME | AVAIL. | OWNER | PEND | INTER | RETRY | DELAY |
|----------------|---------|-------|-----------|--------|-------|------|-------|-------|-------|
| USE_OPERATOR_1 | 121 | 0.788 | 19.924 | 1 | 0 | 0 | 0 | 0 | 0 |
| USE_OPERATOR_2 | 59 | 0.772 | 40.036 | 1 | 0 | 0 | 0 | 0 | 0 |
| USE_OPERATOR_3 | 51 | 0.711 | 42.640 | 1 | 0 | 0 | 0 | 0 | 0 |
| USE_PC_1 | 180 | 0.883 | 15.000 | 1 | 0 | 0 | 0 | 0 | 0 |
| USE_PC_2 | 51 | 0.500 | 30.000 | 1 | 0 | 0 | 0 | 0 | 0 |

| QUEUE | MAX | CONT. | ENTRY | ENTRY(0) | AVE.CONT. | AVE.TIME | AVE.(-0) | RETRY |
|------------|-----|-------|-------|----------|-----------|----------|----------|-------|
| QUEUE_PC_1 | 2 | 0 | 180 | 61 | 0.279 | 4.737 | 7.165 | 0 |
| QUEUE_PC_2 | 1 | 0 | 51 | 48 | 0.004 | 0.212 | 3.598 | 0 |

| SAVEVALUE | RETRY | VALUE |
|-----------------|-------|---------|
| COUNT_PROCESSED | 0 | 231.000 |
| COUNT_REJECT | 0 | 69.000 |
| PROB_REJECT | 0 | 0.230 |

IV. Код программы

SIMULATE

GENERATE 10,2,,300, ; Генерирует заявки каждые 10+-2 единиц времени
;значение общего числа транзактов = 300

```

; Первый оператор, если занят - переход ко второму оператору
; NU - оператор не занят
BLOCK_OPERATOR_1  GATE NU      USE_OPERATOR_1,BLOCK_OPERATOR_2
                  SEIZE       USE_OPERATOR_1      ; Заявка поступает в оператор 1
                  ADVANCE     20,5                ; Оператор 1 обслуживает запрос 20+-5 единиц времени
                  RELEASE     USE_OPERATOR_1      ; Освободить оператор 1
                  TRANSFER    ,BLOCK_PC_1,,        ; Перейти к блоку BLOCK_PC_1

; Второй оператор
; Если занят - переход к третьему оператору
BLOCK_OPERATOR_2  GATE NU      USE_OPERATOR_2,BLOCK_OPERATOR_3
                  SEIZE       USE_OPERATOR_2      ; Заявка поступает в оператор 2
                  ADVANCE     40,10               ; Оператор 2 обслуживает запрос 40+-10 единиц времени
                  RELEASE     USE_OPERATOR_2      ; Освободить оператор 2
                  TRANSFER    ,BLOCK_PC_1,,        ; Перейти к блоку BLOCK_PC_1

; Третий оператор
; Если занят - переход к блоку отказов REFUSED
BLOCK_OPERATOR_3  GATE NU      USE_OPERATOR_3,BLOCK_REFUSED
                  SEIZE       USE_OPERATOR_3      ; Заявка поступает в оператор 3
                  ADVANCE     40,20               ; Оператор 3 обслуживает запрос 40+-20 единиц времени
                  RELEASE     USE_OPERATOR_3      ; Освободить оператор 3
                  TRANSFER    ,BLOCK_PC_2,,        ; Перейти к блоку BLOCK_PC_2

; Первый компьютер
BLOCK_PC_1  QUEUE QUEUE_PC_1      ; Добавить заявку в очередь PC 1
            SEIZE  USE_PC_1        ; Заявка поступает в PC 1
            DEPART QUEUE_PC_1      ; Извлечь заявку из очереди PC 1
            ADVANCE 15,            ; PC 1 обслуживает запрос 15 единиц времени
            RELEASE USE_PC_1        ; Освободить PC 1
            TRANSFER ,BLOCK_PROCESSED,, ; Заявка обслужена: переход к блоку BLOCK_PROCESSED

; Второй компьютер
BLOCK_PC_2  QUEUE QUEUE_PC_2      ; Добавить заявку в очередь PC 2
            SEIZE  USE_PC_2        ; Заявка поступает в PC 2
            DEPART QUEUE_PC_2      ; Извлечь заявку из очереди PC 2
            ADVANCE 30,            ; PC 2 обслуживает запрос 30 единиц времени
            RELEASE USE_PC_2        ; Освободить PC 2
            TRANSFER ,BLOCK_PROCESSED,, ; Заявка обслужена: переход к блоку BLOCK_PROCESSED

; Обработанные заявки
BLOCK_PROCESSED TRANSFER ,BLOCK_END,, ; Переход к блоку BLOCK_END

; Отказы
BLOCK_REFUSED TRANSFER ,BLOCK_END,, ; Переход к блоку BLOCK_END

; Найденные значения
BLOCK_END      SAVEVALUE COUNT_PROCESSED,N$BLOCK_PROCESSED ; Количество обработанных заявок
                SAVEVALUE COUNT_REJECT,N$BLOCK_REFUSED    ; Количество отказов
                SAVEVALUE PROB_REJECT,((N$BLOCK_REFUSED) / (N$BLOCK_END)) ; Вероятность отказа

TERMINATE 1
START 300

```