



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 по лабораторной работе №7

Название Моделирование системы массового обслуживания на языке GPSS

Дисциплина Моделирование

Студент Зайцева А. А.

Группа ИУ7-72Б

Оценка (баллы) _____

Преподаватель Рудаков И. В.

Москва — 2022 г.

1 Задание

Реализовать лабораторную работу №4 на языке GPSS.

Задание к лабораторной работе №4.

Промоделировать работу системы массового обслуживания, определить минимальный размер буфера памяти, при котором не будет потерянных заявок.

Время появления заявок распределено по равномерному закону, время обработки заявки обслуживающим аппаратом – по закону Пуассона (вариант из лабораторной работы №1). С заданной вероятностью обработанная заявка возвращается обратно в очередь на обслуживание.

2 Теоретические сведения

Равномерное распределение

Функция плотности распределения $f(x)$ случайной величины X , имеющей равномерное распределение на отрезке $[a, b]$ ($X \sim R(a, b)$), где $a, b \in R$, имеет следующий вид:

$$f(x) = \begin{cases} \frac{1}{b-a}, & x \in [a, b] \\ 0, & \text{иначе.} \end{cases} \quad (1)$$

Соответствующая функция распределения $F(x) = \int_{-\infty}^x f(t)dt$ принимает вид:

$$F(x) = \begin{cases} 0, & x < a \\ \frac{x-a}{b-a}, & x \in [a, b] \\ 1, & x > b. \end{cases} \quad (2)$$

2.1 Распределение Пуассона

Дискретная случайная величина X имеет закон распределения Пуассона с параметром λ ($X \sim \Pi(\lambda)$), где $\lambda > 0$, если она принимает значения $0, 1, 2, \dots$ с вероятностями:

$$P(X = k) = e^{-\lambda} \frac{\lambda^k}{k!}, \quad k \in \{0, 1, 2, \dots\} \quad (3)$$

Соответствующая функция распределения принимает вид:

$$F(x) = P(X < x) = \sum_{k=0}^{x-1} P(X = k) = e^{-\lambda} \sum_{k=0}^{x-1} \frac{\lambda^k}{k!} \quad (4)$$

3 Результаты работы программы

Для исследования разработанная программа была выполнена при фиксированном количестве заявок $n_tasks = 1000$ и параметрах времени появления заявок (параметры $a = 0$ и $b = 10$ равномерного распределения), и параметрах $lambda_value$ и $p_reenter$, принимающих значения 4 или 10 и 0.1 или 0.5, соответственно.

Для каждого набора параметров минимальный размер буфера памяти, при котором не будет потерянных заявок, равен максимальному размеру очереди THEQUEUE.

Результаты работы программы приведены в листингах 1 – 4.

Листинг 1 – Результат работы программы при $lambda_value = 4$ и $p_reenter = 0.1$ (максимальный размер очереди – 11)

| | | | | | | | | | |
|---|------------|----------|--------|------------|----------|------------|-----------|-----------|-------------|
| 0 | START TIME | END TIME | BLOCKS | FACILITIES | STORAGES | | | | |
| 1 | 0.000 | 5124.517 | 8 | 1 | 0 | | | | |
| 2 | ... | | | | | | | | |
| 3 | FACILITY | ENTRIES | UTIL. | AVE. TIME | AVAIL. | OWNER | PEND | INTER | RETRY DELAY |
| 4 | 0A | 1111 | 0.873 | 4.028 | 1 | 0 | 0 | 0 | 0 |
| 5 | | | | | | | | | |
| 6 | QUEUE | MAX | CONT. | ENTRY | ENTRY(0) | AVE. CONT. | AVE. TIME | AVE. (-0) | RETRY |
| 7 | THEQUEUE | 11 | 0 | 1111 | 243 | 2.040 | 9.411 | 12.046 | 0 |

Листинг 2 – Результат работы программы при $\lambda_value = 4$ и $p_reenter = 0.5$ (максимальный размер очереди – 605)

| | | | | | | | | | |
|---|------------|----------|--------|------------|----------|-----------|----------|----------|-------------|
| 0 | START TIME | END TIME | BLOCKS | FACILITIES | STORAGES | | | | |
| 1 | 0.000 | 7983.387 | 8 | 1 | 0 | | | | |
| 2 | ... | | | | | | | | |
| 3 | FACILITY | ENTRIES | UTIL. | AVE. TIME | AVAIL. | OWNER | PEND | INTER | RETRY DELAY |
| 4 | OA | 2004 | 0.998 | 3.976 | 1 | 1237 | 0 | 0 | 0 604 |
| 5 | | | | | | | | | |
| 6 | QUEUE | MAX | CONT. | ENTRY | ENTRY(0) | AVE.CONT. | AVE.TIME | AVE.(-0) | RETRY |
| 7 | THEQUEUE | 605 | 605 | 2608 | 3 | 281.952 | 863.089 | 864.083 | 0 |

Листинг 3 – Результат работы программы при $\lambda_value = 10$ и $p_reenter = 0.1$ (максимальный размер очереди – 1239)

| | | | | | | | | | |
|---|------------|-----------|--------|------------|----------|-----------|----------|----------|-------------|
| 0 | START TIME | END TIME | BLOCKS | FACILITIES | STORAGES | | | | |
| 1 | 0.000 | 10930.424 | 8 | 1 | 0 | | | | |
| 2 | ... | | | | | | | | |
| 3 | FACILITY | ENTRIES | UTIL. | AVE. TIME | AVAIL. | OWNER | PEND | INTER | RETRY DELAY |
| 4 | OA | 1108 | 0.999 | 9.857 | 1 | 1054 | 0 | 0 | 0 1237 |
| 5 | | | | | | | | | |
| 6 | QUEUE | MAX | CONT. | ENTRY | ENTRY(0) | AVE.CONT. | AVE.TIME | AVE.(-0) | RETRY |
| 7 | THEQUEUE | 1239 | 1238 | 2345 | 1 | 631.423 | 2943.164 | 2944.420 | 0 |

Листинг 4 – Результат работы программы при $\lambda_value = 10$ и $p_reenter = 0.5$ (максимальный размер очереди – 3034)

| | | | | | | | | | |
|---|------------|-----------|--------|------------|----------|-----------|----------|----------|-------------|
| 0 | START TIME | END TIME | BLOCKS | FACILITIES | STORAGES | | | | |
| 1 | 0.000 | 19784.424 | 8 | 1 | 0 | | | | |
| 2 | ... | | | | | | | | |
| 3 | FACILITY | ENTRIES | UTIL. | AVE. TIME | AVAIL. | OWNER | PEND | INTER | RETRY DELAY |
| 4 | OA | 1997 | 1.000 | 9.903 | 1 | 1601 | 0 | 0 | 0 3033 |
| 5 | | | | | | | | | |
| 6 | QUEUE | MAX | CONT. | ENTRY | ENTRY(0) | AVE.CONT. | AVE.TIME | AVE.(-0) | RETRY |
| 7 | THEQUEUE | 3034 | 3034 | 5030 | 1 | 1529.323 | 6015.262 | 6016.458 | 0 |

Для наглядности результаты также представлены в таблице 1.

Для сравнения на рисунке 1 приведена таблица результатов, полученных при исследовании программы с теми же параметрами из лабораторной работы №4.

Порядки полученных обеими программами результатов схожи (ре-

Таблица 1 – Таблица с результатами исследования программы

| lambda | p | максимальный размер очереди |
|--------|-----|-----------------------------|
| 4 | 0.1 | 11 |
| 4 | 0.5 | 605 |
| 10 | 0.1 | 1239 |
| 10 | 0.5 | 3034 |

| lambda | вероятность повторного попадания в очередь | максимальный размер очереди |
|--------|--|---------------------------------|
| 4 | 0.1 | Событийный: 8, delta t: 8 |
| 4 | 0.5 | Событийный: 477, delta t: 480 |
| 10 | 0.1 | Событийный: 1040, delta t: 1039 |
| 10 | 0.5 | Событийный: 2652, delta t: 2653 |

Рисунок 1 – Таблица с результатами исследования программы

зультаты отличаются в силу случайности генерируемых данных).

Максимальная длина очереди растет по мере роста *lambda_value* (так как время обработки заявки растет) и *p_reenter* (так как все больше заявок попадают в очередь на обслуживание повторно).

4 Код программы

В листинге 5 приведен код программы.

Листинг 5 – Код программы

```

0  GENERATE      (UNIFORM(1,0,10))      ; Время генерации заявки R(0, 10)
1
2  AddInQueue    QUEUE TheQueue          ; Вход в очередь, увеличение длины очереди
3  SEIZE        OA                          ; Захват или ожидание OA
4  DEPART        TheQueue                  ; Выход из очереди, уменьшение длины очереди
5
6  ADVANCE       (POISSON(1,10))          ; Обслуживание заявки в OA время( P(lambda))
7  RELEASE       OA                        ; Обслуживание заявки в OA окончено
8  TRANSFER      0.1,Finish,AddInQueue    ; С заданной вероятностью заявка вновь попадает в очередь
9

```

```
10 Finish  TERMINATE    1                ; Окончание обслуживания заявки
11 START 1000                ; Количество заявок
```