

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Рязанский государственный радиотехнический университет
имени В.Ф. Уткина»

Кафедра ЭВМ

Отчет о лабораторной работе №2
«Методы работы с моделями»
по дисциплине «Моделирование»

Выполнили:

ст. гр. 245

бригада №4

Сокол Илья

Лапин Кирилл

Проверил:

доц. каф. ЭВМ

Саблина В.А.

Рязань 2025

Цель работы: изучение методов работы с моделями СМО на языке GPSS, способов задания условия окончания моделирования, формирования отчетов, переопределения параметров модели; изучение влияния квантования времени обслуживания и назначения приоритетов на качество работы СМО; изучение замкнутой СМО.

Практическая часть

1. СМО с квантованием времени обслуживания

Ознакомимся с моделью СМО с квантованием по времени. Данная модель имеет следующее графическое представление (рисунок 1.1):

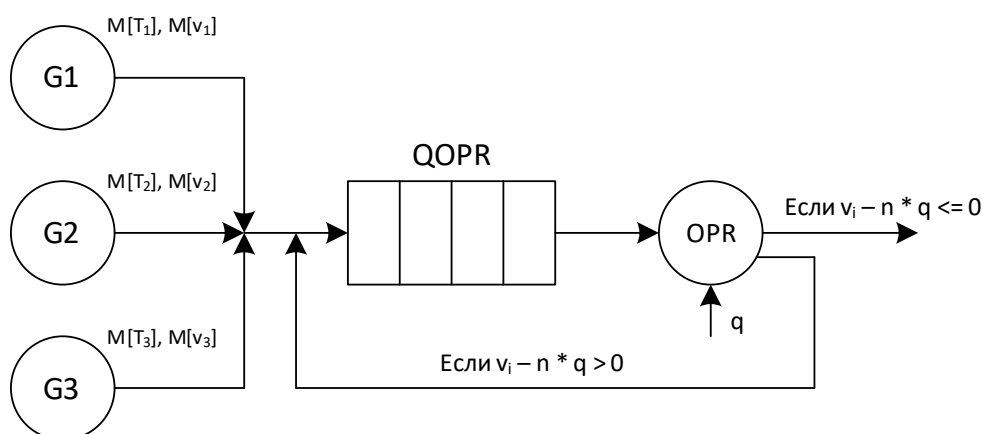


Рисунок 1.1 – Графическое представление модели СМО с квантованием по времени

В соответствии с исходными данными из таблицы 1.1 отредактируем приведенную программу, описывающую работу модели СМО с квантованием по времени.

Таблица 1.1 – Исходные данные для СМО с квантованием по времени

Вариант	$M[T_1]$	$M[T_2]$	$M[T_3]$	$M[v_1]$	$M[v_2]$	$M[v_3]$
4	15	50	100	5	10	40

Ниже представлен листинг полученной программы:

```
; Задание таблицы табулирования времени пребывания транзакта в системе
TIME TABLE M1,50,50,10

; Задание (кусочно-линейное) функции экспоненты EXP1
EXP1 FUNCTION RN1,C24
0,0/.1,.104/.2,.222/.3,.335/.4,.509/.5,.69/.6,.915
.7,1.2/.75,1.38/.8,1.6/.84,1.85/.88,2.12/.9,2.3
.92,2.52/.94,2.81/.95,2.99/.96,3.2/.97,3.5/.98,3.9
.99,4.6/.995,5.3/.998,6.2/.999,7/.9998,8
```

```

; Квант процессорного времени q=1
QUANT VARIABLE 1

; Генератор заявок с приоритетом 1, M[T1]=15, M[v1]=5
GENERATE 15, FN$EXP1,,,1          ; Генерируем заявки с приоритетом 1
                                   ; через интервалы времени
                                   ; (EXP1, M[T1]=15)

ASSIGN 1,5,EXP1                    ; Присваиваем первому
                                   ; параметру транзакта
                                   ; время обслуживания (EXP1, M[v1]=5)

TRANSFER ,MET1                     ; Пересылаем транзакт в очередь
                                   ; (безусловная передача транзакта
                                   ; на метку MET1)

; Генератор заявок с приоритетом 1, M[T2]=50, M[v2]=10
GENERATE 50, FN$EXP1,,,1          ; Генерируем заявки с приоритетом 1
                                   ; через интервалы времени
                                   ; (EXP1, M[T2]=50)

ASSIGN 1,10,EXP1                   ; Присваиваем первому
                                   ; параметру транзакта
                                   ; время обслуживания (EXP1, M[v2]=10)

TRANSFER ,MET1                     ; Пересылаем транзакт в очередь

; Генератор заявок с приоритетом 1, M[T3]=100, M[v3]=40
GENERATE 100, FN$EXP1,,,1         ; Генерируем заявки с приоритетом 1
                                   ; через интервалы времени
                                   ; (EXP1, M[T3]=100)

ASSIGN 1,40,EXP1                   ; Присваиваем первому
                                   ; параметру транзакта
                                   ; время обслуживания (EXP1, M[v3]=40)

MET1      QUEUE QOPR                ; Вход транзакта в очередь
          SEIZE OPR                  ; Занимаем устройство OPR
          DEPART QOPR               ; Выход транзакта из очереди

          TEST L V$QUANT,P1,MET2    ; Если кванта не хватило
                                   ; q < остатка vi, то
          ADVANCE V$QUANT            ; задерживаем заявку на время кванта q
                                   ; иначе на метку MET2

          RELEASE OPR                ; Освобождаем устройство OPR

          ASSIGN 1-,V$QUANT          ; Вычитаем из времени обслуживания
                                   ; заявки vi квант времени q

          TRANSFER ,MET1             ; Передаем недообслуженный транзакт
                                   ; в очередь

FIN       TABULATE TIME              ; Занести значение времени
                                   ; пребывания транзакта в таблицу TIME

          TERMINATE 1                ; Регистрация обслуженной заявки
                                   ; (увеличение счетчика обслуженных
                                   ; заявок и удаление заявки)

MET2      ADVANCE P1                 ; Если кванта хватило, то задержка
                                   ; на остаток времени обслуживания

```

RELEASE OPR	; Освобождаем устройство OPR
TRANSFER ,FIN	; Конец обслуживания заявки

Запустим программу на выполнение при помощи команды START 200000, задав значение счетчика завершений равным 200000. В результате выполнения программы получим отчет, представленный на рисунке 1.2.

START TIME	END TIME	BLOCKS	FACILITIES	STORAGES
0.000	2063271.991	21	1	0

NAME	VALUE
EXP1	10001.000
FIN	17.000
MET1	9.000
MET2	19.000
OPR	10004.000
QOPR	10003.000
QUANT	10002.000
TIME	10000.000

LABEL	LOC	BLOCK TYPE	ENTRY COUNT	CURRENT	COUNT	RETRY
	1	GENERATE	137760		0	0
	2	ASSIGN	137760		0	0
	3	TRANSFER	137760		0	0
	4	GENERATE	41591		0	0
	5	ASSIGN	41591		0	0
	6	TRANSFER	41591		0	0
	7	GENERATE	20660		0	0
	8	ASSIGN	20660		0	0
MET1	9	QUEUE	2034160		10	0
	10	SEIZE	2034150		1	0
	11	DEPART	2034149		0	0
	12	TEST	2034149		0	0
	13	ADVANCE	1834149		0	0
	14	RELEASE	1834149		0	0
	15	ASSIGN	1834149		0	0
	16	TRANSFER	1834149		0	0
FIN	17	TABULATE	200000		0	0
	18	TERMINATE	200000		0	0
MET2	19	ADVANCE	200000		0	0
	20	RELEASE	200000		0	0
	21	TRANSFER	200000		0	0

FACILITY	ENTRIES	UTIL.	AVE. TIME	AVAIL.	OWNER	PEND	INTER	RETRY	DELAY
OPR	2034150	0.936	0.950	1	200004	0	0	0	10

QUEUE	MAX	CONT.	ENTRY	ENTRY(0)	AVE. CONT.	AVE. TIME	AVE. (-0)	RETRY
QOPR	80	11	2034160	121922	13.575	13.769	14.647	0

TABLE	MEAN	STD.DEV.	RANGE	RETRY	FREQUENCY	CUM.%
TIME	149.687	362.635		0		
			50.000		100629	50.31
			100.000		34441	67.53
			150.000		18285	76.68
			200.000		11032	82.19
			250.000		7413	85.90
			300.000		5233	88.52
			350.000		3867	90.45
			400.000		2884	91.89
			450.000		2227	93.01
					13989	100.00

CEC XN	PRI	M1	ASSEM	CURRENT	NEXT	PARAMETER	VALUE
200004	1	2063166.116	200004	10	11	1	0.545

FEC XN	PRI	BDT	ASSEM	CURRENT	NEXT	PARAMETER	VALUE
199990	1	2063272.644	199990	0	7		
200014	1	2063275.266	200014	0	1		
200011	1	2063296.030	200011	0	4		

Рисунок 1.2 – Результат выполнения программы модели СМО с квантованием по времени

Рассчитаем теоретическое значение коэффициента загрузки обслуживающего аппарата ρ по формуле:

$$\rho = \frac{M[v_1]}{M[T_1]} + \frac{M[v_2]}{M[T_2]} + \frac{M[v_3]}{M[T_3]} = \frac{5}{15} + \frac{10}{50} + \frac{40}{100} \approx 0.933$$

Экспериментально было получено, что коэффициент загрузки обслуживающего аппарата равен 0.936. Таким образом, теоретическое и экспериментальное значения коэффициента загрузки обслуживающего аппарата совпадают с точностью до сотых.

Снимем зависимость среднего времени пребывания заявки в системе от величины кванта $q = 1, 4, 16, 64$ (таблица 1.2) и построим график (рисунок 1.3).

Таблица 1.2 – Зависимость среднего времени пребывания заявки в системе от величины кванта

Величина кванта q	Среднее время пребывания заявки в системе
1	149.687
4	160.406
16	203.655
64	282.741

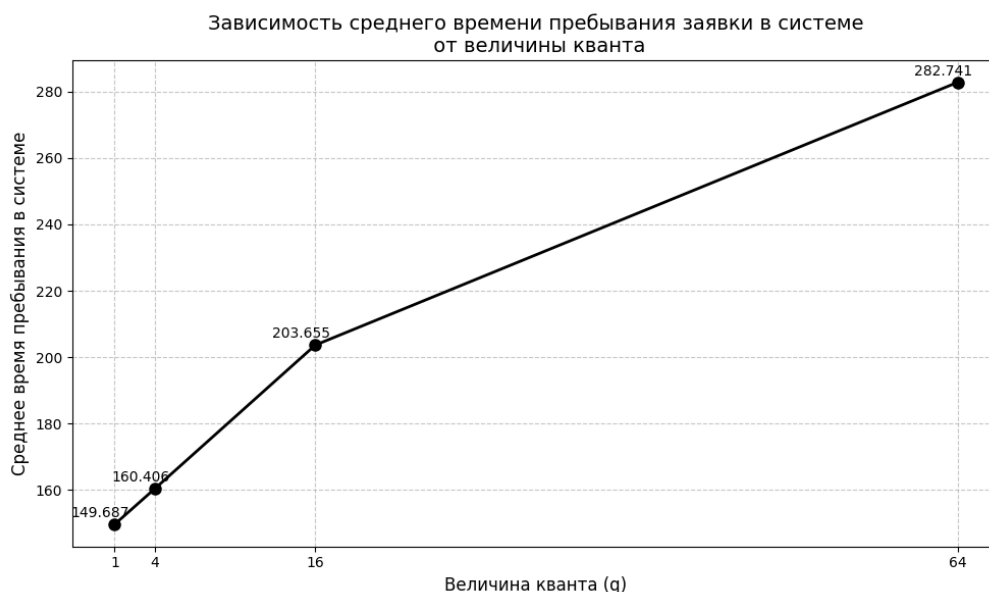


Рисунок 1.3 – График зависимости среднего времени пребывания заявки в системе от величины кванта

При маленькой величина кванта система часто переключается между заявками, что позволяет быстро обслуживать короткие задачи и сразу же возвращать в очередь длинные, но уже с уменьшенным оставшимся временем обслуживания. Это обеспечивает низкое среднее время ожидания для всех заявок и высокую пропускную способность системы. Однако с ростом величины кванта длинные заявки надолго захватывают обслуживающий аппарат, вынуждая короткие заявки проводить в очереди больше времени. Это приводит к увеличению времени ожидания для всех заявок и к росту среднего времени пребывания заявок в системе.

Выключим в модели квантование времени обслуживания, задав значение кванта равным 400. В результате получим среднее время пребывания заявки в системе, представленное на рисунке 1.4.

TABLE TIME	MEAN	STD.DEV.	RANGE	RETRY	FREQUENCY	CUM.%
	309.292	304.520		0		
			-	50.000	39213	19.61
			50.000 -	100.000	21839	30.53
			100.000 -	150.000	18625	39.84
			150.000 -	200.000	16110	47.89
			200.000 -	250.000	13671	54.73
			250.000 -	300.000	12102	60.78
			300.000 -	350.000	10077	65.82
			350.000 -	400.000	8661	70.15
			400.000 -	450.000	7653	73.98
			450.000 -		52049	100.00

Рисунок 1.4 – Среднее время пребывания заявки в системе с выключенным квантованием

Таким образом, при отключенном квантовании времени обслуживания среднее время пребывания заявки в системе составляет 309.292, что больше всех значений полученных ранее (даже значения при $q = 64$).

Квантование дает такой эффект, так как позволяет эффективно разделять длительные задачи на отдельные «сегменты», что обеспечивает более справедливое распределение ресурсов обслуживающего аппарата. Благодаря этому короткие заявки, требующие меньше времени на обработку, получают возможность быстрее проходить через систему, не дожидаясь завершения длительных операций.

Установим $q = 1$ и назначим приоритеты так, чтобы самый высокий приоритет был у заявок с меньшей трудоемкостью. В результате выполнения такой программы выводится следующий отчет (рисунок 1.5):

START TIME		END TIME	BLOCKS	FACILITIES	STORAGES
0.000		2063233.845	21	1	0

NAME		VALUE
EXP1		10001.000
FIN		17.000
MET1		9.000
MET2		19.000
OPR		10004.000
QOPR		10003.000
QUANT		10002.000
TIME		10000.000

LABEL	LOC	BLOCK TYPE	ENTRY COUNT	CURRENT COUNT	RETRY
MET1	1	GENERATE	137758	0	0
	2	ASSIGN	137758	0	0
	3	TRANSFER	137758	0	0
	4	GENERATE	41591	0	0
	5	ASSIGN	41591	0	0
	6	TRANSFER	41591	0	0
	7	GENERATE	20660	0	0
	8	ASSIGN	20660	0	0
	9	QUEUE	2034120	8	0
	10	SEIZE	2034112	1	0
	11	DEPART	2034111	0	0
	12	TEST	2034111	0	0
	13	ADVANCE	1834111	0	0
	14	RELEASE	1834111	0	0
	15	ASSIGN	1834111	0	0
	16	TRANSFER	1834111	0	0
FIN	17	TABULATE	200000	0	0
MET2	18	TERMINATE	200000	0	0
	19	ADVANCE	200000	0	0
	20	RELEASE	200000	0	0
	21	TRANSFER	200000	0	0

FACILITY	ENTRIES	UTIL.	AVE. TIME	AVAIL.	OWNER	PEND	INTER	RETRY	DELAY
OPR	2034112	0.936	0.950	1	200010	0	0	0	8

QUEUE	MAX CONT.	ENTRY	ENTRY(0)	AVE.CONT.	AVE.TIME	AVE.(-0)	RETRY
QOPR	50	9	2034120	137326	7.684	7.794	8.359

TABLE	MEAN	STD.DEV.	RANGE	RETRY	FREQUENCY	CUM.%
TIME	88.906	408.321		0		
			50.000 -	50.000	171814	85.91
			50.000 -	100.000	10039	90.93
			100.000 -	150.000	3174	92.51
			150.000 -	200.000	1758	93.39
			200.000 -	250.000	1362	94.07
			250.000 -	300.000	977	94.56
			300.000 -	350.000	831	94.98
			350.000 -	400.000	745	95.35
			400.000 -	450.000	659	95.68
			450.000 -		8641	100.00

CEC XN	PRI	M1	ASSEM	CURRENT	NEXT	PARAMETER	VALUE
200010	3	2063228.481	200010	10	11	1	1.137

FEC XN	PRI	BDT	ASSEM	CURRENT	NEXT	PARAMETER	VALUE
200012	3	2063252.955	200012	0	1		
199990	1	2063272.644	199990	0	7		
200011	2	2063296.030	200011	0	4		

Рисунок 1.5 – Результат выполнения программы с корректно назначенными приоритетами

Таким образом, в случае корректного назначения приоритетов и $q = 1$ среднее время пребывания заявки в системе (равное 80.906) в несколько раз

меньше среднего времени пребывания заявки в системе при одинаковых приоритетах всех заявок и $q = 1$ (равного 149.687).

2. Замкнутая сетевая модель СМО

Ознакомимся с замкнутой сетевой моделью СМО. Данная модель имеет следующее графическое представление (рисунок 2.1):

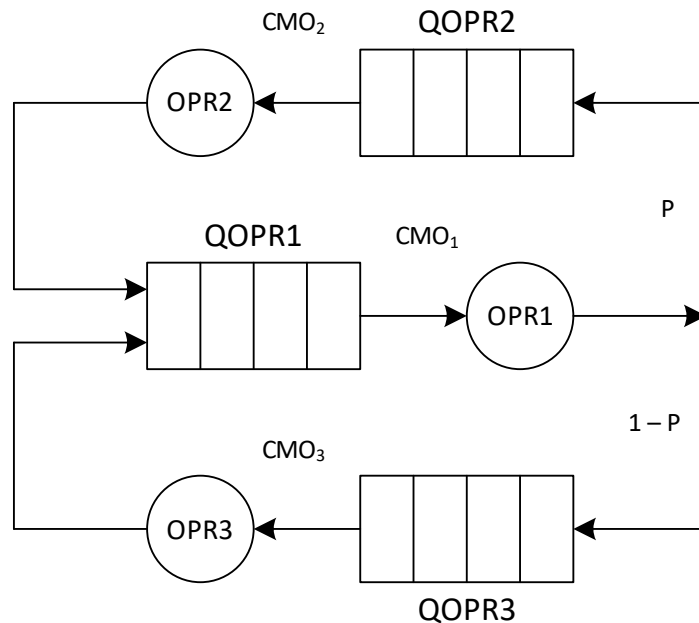


Рисунок 2.1 – Графическое представление замкнутой сетевой модели СМО

В соответствии с исходными данными из таблицы 2.1 отредактируем приведенную программу, описывающую работу замкнутой сетевой модели СМО.

Таблица 2.1 – Исходные данные для замкнутой сетевой модели СМО

Вариант	$M[v_1]$	$M[v_2]$	$M[v_3]$	P
4	100	200	50	0.8

Ниже приведен листинг полученной программы:

```
; Задание функции EXP1 табличным способом
EXP1 FUNCTION RN1,C24
0,0/.1,.104/.2,.222/.3,.335/.4,.509/.5,.69/.6,.915
.7,1.2/.75,1.38/.8,1.6/.84,1.85/.88,2.12/.9,2.3
.92,2.52/.94,2.81/.95,2.99/.96,3.2/.97,3.5/.98,3.9
.99,4.6/.995,5.3/.998,6.2/.999,7/.9998,8

; Генерируем 3 заявки, которые поступают
; в очередь устройства OPR1
GENERATE ,,,3

MET1      QUEUE QOPR1
          SEIZE OPR1
          ; Занимаем устройство OPR1
```



```

DEPART QOPR1
ADVANCE 100, FN$EXP1      ; Обслуживаем заявку в OPR1
RELEASE OPR1              ; Освобождаем устройство OPR1

TRANSFER 0.200,,MET3      ; Передаем заявку в устройство OPR3
                           ; с вероятностью 0.2 и в устройство
                           ; OPR2 с вероятностью 0.8

QUEUE QOPR2
SEIZE OPR2                ; Занимаем устройство OPR2
DEPART QOPR2
ADVANCE 200, FN$EXP1      ; Обслуживаем заявку в OPR2
RELEASE OPR2              ; Освобождаем устройство OPR2

TRANSFER ,MET1            ; Передаем заявку обратно в
                           ; очередь устройства OPR1

MET3  QUEUE QOPR3
SEIZE OPR3                ; Занимаем устройство OPR3
DEPART QOPR3
ADVANCE 50, FN$EXP1       ; Обслуживаем заявку в OPR3
RELEASE OPR3              ; Освобождаем устройство OPR3

TRANSFER ,MET1            ; Возвращаем заявку в
                           ; очередь устройства OPR1

GENERATE 100000           ; Задание времени моделирования
TERMINATE 1

```

Запустим программу на выполнение при помощи команды START 1. В результате выполнения программы получим отчет, представленный на рисунке 2.2.

START TIME		END TIME	BLOCKS	FACILITIES	STORAGES
0.000		100000.000	21	3	0

NAME	VALUE
EXP1	10000.000
MET1	2.000
MET3	14.000
OPR1	10002.000
OPR2	10004.000
OPR3	10006.000
QOPR1	10001.000
QOPR2	10003.000
QOPR3	10005.000

LABEL	LOC	BLOCK TYPE	ENTRY COUNT	CURRENT	COUNT	RETRY
MET1	1	GENERATE	3		0	0
	2	QUEUE	568		0	0
	3	SEIZE	568		0	0
	4	DEPART	568		0	0
	5	ADVANCE	568		0	0
	6	RELEASE	568		0	0
	7	TRANSFER	568		0	0
	8	QUEUE	453		2	0
	9	SEIZE	451		0	0
	10	DEPART	451		0	0
	11	ADVANCE	451		1	0
	12	RELEASE	450		0	0
	13	TRANSFER	450		0	0
MET3	14	QUEUE	115		0	0
	15	SEIZE	115		0	0
	16	DEPART	115		0	0
	17	ADVANCE	115		0	0
	18	RELEASE	115		0	0
	19	TRANSFER	115		0	0
	20	GENERATE	1		0	0
	21	TERMINATE	1		0	0

FACILITY	ENTRIES	UTIL.	AVE. TIME	AVAIL.	OWNER	PEND	INTER	RETRY	DELAY
OPR1	568	0.587	103.391	1		0	0	0	0
OPR2	451	0.856	189.730	1		3	0	0	2
OPR3	115	0.053	45.761	1		0	0	0	0

QUEUE	MAX	CONT.	ENTRY	ENTRY (0)	AVE. CONT.	AVE. TIME	AVE. (-0)	RETRY
QOPR1	2	0	568	274	0.445	78.394	151.454	0
QOPR2	2	2	453	105	1.053	232.452	302.588	0
QOPR3	2	0	115	105	0.006	5.344	61.461	0

FEC	XN	PRI	BDT	ASSEM	CURRENT	NEXT	PARAMETER	VALUE
3	0		100031.820	3	11	12		
5	0		200000.000	5	0	20		

Рисунок 2.2 – Результат выполнения программы замкнутой сетевой модели

СМО

Зафиксируем для каждой СМО загрузки, среднюю длину очереди и среднее время ожидания (таблица 2.2).

Таблица 2.2 – Экспериментальные результаты для 3 заявок

	СМО₁	СМО₂	СМО₃
Загрузка	0.587	0.856	0.053
Средняя длина очереди	0.445	1.053	0.006
Среднее время ожидания	78.394	232.452	5.344

Повторим моделирование для 6, 9 и 12 заявок в модели и зафиксируем экспериментальные результаты для каждой СМО в сводной таблице 2.3.

Таблица 2.3 – Сводная таблица экспериментальных результатов

	Количество заявок в модели	Загрузка	Средняя длина очереди	Среднее время ожидания
СМО₁	3	0.587	0.445	78.394
	6	0.673	0.997	152.372
	9	0.644	1.190	191.012
	12	0.634	1.152	183.117
СМО₂	3	0.856	1.053	232.452
	6	0.968	3.292	629.477
	9	0.981	6.121	1209.673
	12	0.995	9.155	1763.893
СМО₃	3	0.053	0.006	5.344
	6	0.065	0.005	4.183
	9	0.061	0.003	2.655
	12	0.060	0.005	4.382

Построим графики изменения рассмотренных параметров как функций числа заявок в модели (рисунок 2.3).

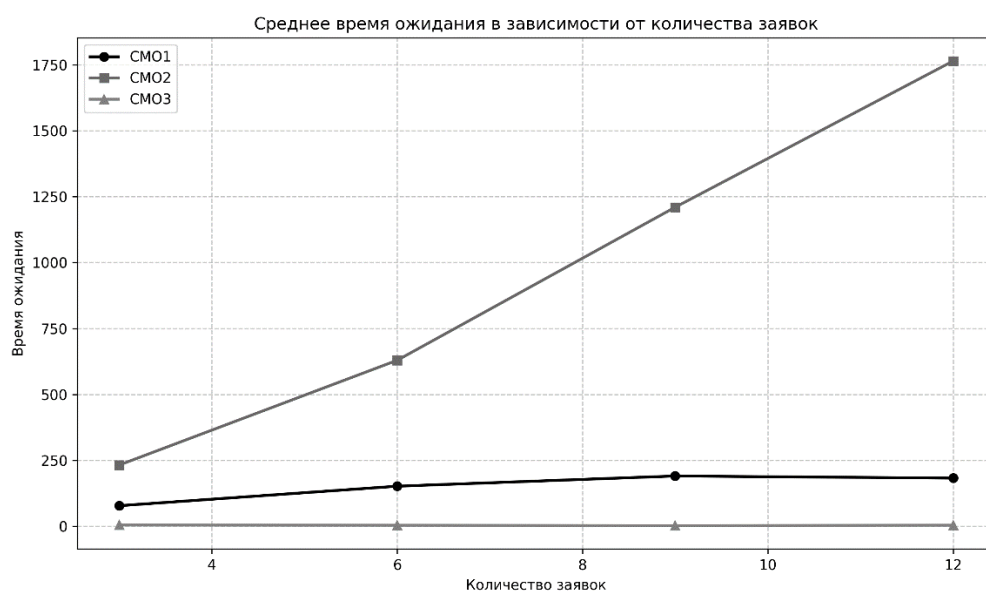
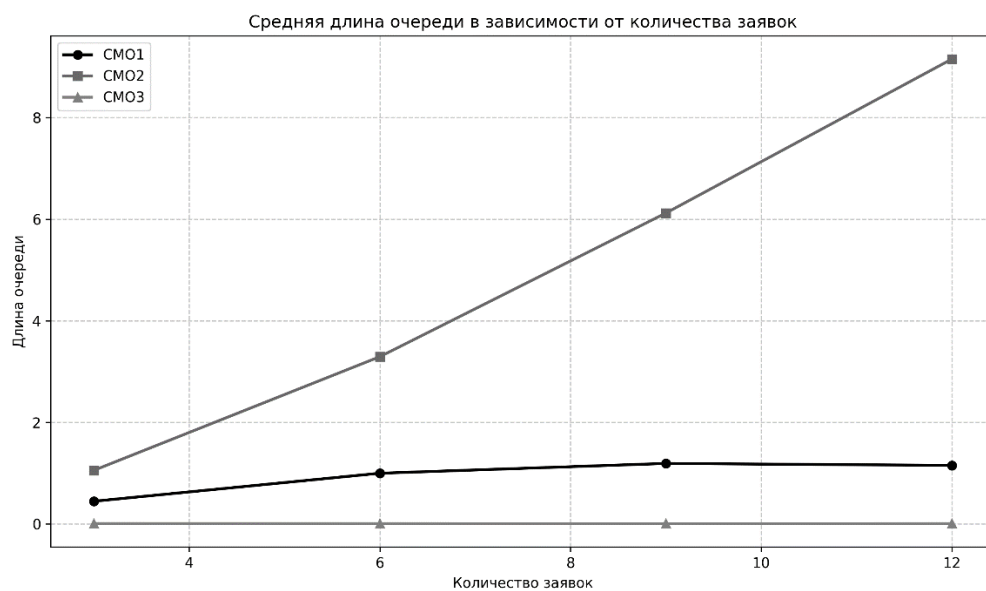
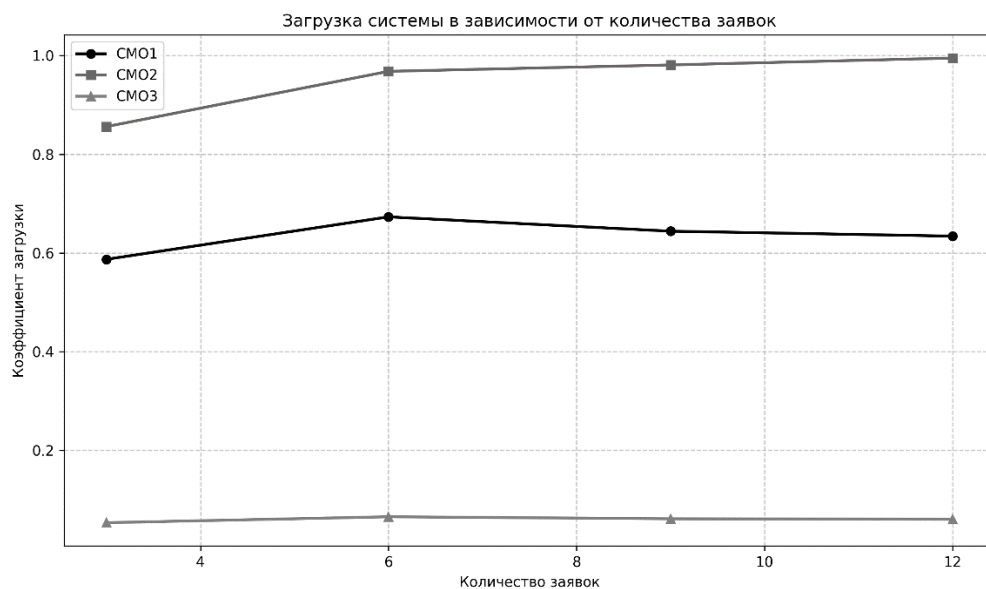


Рисунок 2.3 – Графики зависимостей параметров от числа заявок

Наиболее критичной ситуация складывается для СМО₂, которая явно является "узким местом" всей системы. Уже при минимальной нагрузке (3 заявки) её загрузка достигает 0.856. С ростом количества заявок до 12 этот показатель приближается к единице (0.995), что означает практически полную загрузку обслуживающего аппарата. Такая перегрузка закономерно приводит к росту очереди - с 1.053 до 9.155 заявок, и соответствующему увеличению времени ожидания с 232.452 до 1763.893 единиц времени. Характер роста этих показателей указывает на то, что СМО₂ не справляется с поступающей нагрузкой.

В отличие от СМО₂, СМО₁ демонстрирует более стабильное поведение. Её загрузка колеблется в диапазоне 0.587-0.673, достигая максимума при 6 заявках и несколько снижаясь при дальнейшем увеличении нагрузки. Средняя длина очереди растёт умеренными темпами с 0.445 до 1.190-1.152 заявок, а время ожидания увеличивается с 78.394 до 183.117-191.012 единиц. Такой характер изменения параметров свидетельствует о том, что СМО₁ способна адаптироваться к возрастающей нагрузке.

Наиболее благополучная ситуация наблюдается в СМО₃, которая остается практически незагруженной на всех режимах работы. Её загрузка не превышает 6.5%, длина очереди колеблется в пределах 0.003-0.006 заявок, а время ожидания не превышает 5.344 единиц. Эти показатели указывают на то, что данный ресурс системы используется крайне неэффективно и представляет собой резерв производительности.

Вывод

Были изучены методы работы с моделями СМО на языке GPSS, способы задания условия окончания моделирования, формирование отчетов, переопределение параметров модели; было изучено влияние квантования времени обслуживания и назначения приоритетов на качество работы СМО; была изучена замкнутая СМО.