



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

ФАКУЛЬТЕТ ИУ «Информатика и системы управления»

КАФЕДРА ИУ-7 «Программное обеспечение ЭВМ и информационные технологии»

Отчет
по лабораторной работе № 7

Дисциплина: Моделирование

Студент группы ИУ7-73Б

(Подпись, дата)

Паламарчук А.Н.

(Фамилия И.О.)

Преподаватель

(Подпись, дата)

Рудаков И.В.

(Фамилия И.О.)

2025 г.

Задание

Разработать программное обеспечение для моделирования работы системы массового обслуживания. Программа должна определять минимальную длину очереди, при которой не происходит потеря сообщений. Закон генерации сообщений задается пользователем. Закон обслуживания сообщений задается пользователем.

Предусмотреть возможность возврата определенного процента обслуженных заявок обратно в очередь для повторной обработки. Реализовать моделирование двумя способами: событийный принцип, пошаговый принцип.

Теоретическая часть

Распределения

Распределение	Момент времени t_i
Равномерное	$a + (b - a)R$
Экспоненциальное	$-\frac{1}{\lambda} \ln(1 - R)$
Нормальное	$\sigma_t \sqrt{\frac{12}{n}} \left(\sum_{i=1}^n R_i - \frac{n}{2} \right) + M_x$
Эрланга	$-\frac{1}{k\lambda} \sum_{i=1}^k \ln(1 - R_i)$

Пошаговый принцип (Δt)

Принцип заключается в последовательном анализе состояний всех блоков системы в момент $t + \Delta t$. При этом новое состояние блоков определяется в соответствии с их алгоритмическим описанием.

Недостаток: при недостаточно малом шаге Δt отдельные события в системе могут быть пропущены, а слишком малый шаг приводит к значительным временным затратам на моделирование.

Событийный принцип

Состояние отдельных устройств изменяются в дискретные моменты времени, совпадающие с моментами времени поступления сообщений в систему, временем окончания обработки задачи и т.д. При использовании событийного принципа состояние всех блоков системы анализируется лишь в момент проявления какого-либо события. Момент наступления следующего события определяется как минимальное значение из списка запланированных событий.

Результат работы

GPSS World Simulation Report - lab_07.19.1

Friday, November 21, 2025 16:27:42

START TIME	END TIME	BLOCKS	FACILITIES	STORAGES
0.000	1029.714	8	1	0

NAME	VALUE
ADDINQUEUE	2.000
FINISH	8.000
MY_ERLANG	10000.000
SERVER	10002.000
SYSTEMQ	10001.000

LABEL	LOC	BLOCK TYPE	ENTRY COUNT	CURRENT COUNT	RETRY
ADDINQUEUE	1	GENERATE	1000	0	0
	2	QUEUE	1117	0	0
	3	SEIZE	1117	0	0
	4	DEPART	1117	0	0
	5	ADVANCE	1117	0	0
	6	RELEASE	1117	0	0
FINISH	7	TRANSFER	1117	0	0
	8	TERMINATE	1000	0	0

FACILITY	ENTRIES	UTIL.	AVE. TIME	AVAIL.	OWNER	PEND	INTER	RETRY	DELAY
SERVER	1117	0.538	0.496	1	0	0	0	0	0

QUEUE	MAX CONT.	ENTRY	ENTRY(0)	AVE.CONT.	AVE.TIME	AVE.(-0)	RETRY
SYSTEMQ	4	0	1117	676	0.174	0.161	0.407 0

FEC XN	PRI	BDT	ASSEM	CURRENT	NEXT	PARAMETER	VALUE
1001	0	1030.565	1001	0	1		

Генератор заявок

 k (порядок) λ (интенсивность)

Обработчик заявок

 μ (мат. ожидание) σ (отклонение)

Обратная связь

Результаты

Метод	Длина очереди
Событийный	3
Пошаговый (шаг=0.01)	4

Код программы

```
; Генерация заявок
; Вариант 1: Равномерное [A, B]. Пример для A=0, B=10 -> 5,5
; GENERATE 5,5
; stream - номер генератора сл чисел (1)
; mode - сдвиг (0)
; mean - среднее (1.0)
; Вариант 2: Экспоненциальное [Lambda]. Пример Lambda=1 -> Среднее=1
; GENERATE (Exponential(1, 0, 1.0))
; stream - номер генератора сл чисел (1)
; mean - среднее (1.0)
; stddev - среднеквадратичное отклонение
; Вариант 3: Нормальное [Mu, Sigma]. Пример Mu=10, Sigma=2
; GENERATE (Normal(1, 10, 2))
; Настройки Эрланга для генератора
; Порядок k = 2. Общее Среднее = 1.0.
; Среднее для части = 1.0 / 2 = 0.5
MY_ERLANG FVARIABLE (Exponential(1, 0, 0.5) + Exponential(1, 0, 0.5))
; Вариант 4: Эрланга
GENERATE V$MY_ERLANG

AddInQueue QUEUE SystemQ      ; Вход в очередь, сбор статистики
      SEIZE Server              ; Занять прибор
      DEPART SystemQ           ; Выход из очереди (как только заняли прибор)

; Обслуживание
; Вариант 1: Равномерное [A, B]
; ADVANCE 5,5
; Вариант 2: Экспоненциальное [Lambda]. Пример Lambda=4 -> Среднее=0.25
; ADVANCE (Exponential(1, 0, 0.25))
; Вариант 3: Нормальное [Mu, Sigma]. Пример Mu=0.5, Sigma=0.1
ADVANCE (Normal(1, 0.5, 0.1))
; Вариант 4: Эрланга [k=2, Lambda=1.0].
; Среднее для одной фазы = 1 / (k * lambda) = 1 / (2 * 1) = 0.5
; Делаем цикл из k раз
; ASSIGN 1,2                  ; Записываем k=2 в параметр транзакта 1
; ErlLoop ADVANCE (Exponential(1, 0, 0.5))
; LOOP 1,ErlLoop              ; Повторяем k раз

; Освободить прибор
RELEASE Server
; TRANSFER Pick, A, B - вероятность Pick пойти в B, иначе в A.
TRANSFER 0.1,Finish,AddInQueue
; Удалить транзакт, уменьшить счетчик START на 1
Finish TERMINATE 1
START 1000
```