

Лабораторная работа №1. ИЗУЧЕНИЕ ПРАВИЛ ФОРМИРОВАНИЯ ТРАНЗАКТОВ В GPSS WORLD

Цель работы – освоить правила формирования и уничтожения динамических объектов – транзактов в моделях системы GPSS.

Задание: 1) отладить программу, обеспечивающую генерацию транзактов на заданном отрезке времени, и определить количество сгенерированных транзактов за указанное в примере время моделирования; 2) написать и отладить программу для заданного варианта №1 и выполнить с ней заданные исследования.

Подготовка к выполнению работы

Для выполнения данной лабораторной работы необходимо:

- изучить теоретические основы программирования на языке GPSS;
- написать программу, обеспечивающую генерацию транзактов в соответствии с формулировкой задачи в системе моделирования GPSS;
- ввести текст программы и запустить модель на выполнение;
- вывести результаты моделирования в виде отчета и объяснить их.

Выполнение работы

1. Запустите систему моделирования GPSS.
2. В окне текстового редактора введите текст программы, который приводится ниже:
GENERATE 10,5,200,,1 ; генерация транзактов на отрезке [5,15]
TERMINATE ; удаление транзактов из модели
GENERATE 200 ; время работы модели
TERMINATE 1 ; удаление по одному транзакту из модели
3. Реализуйте 10 прогонов модели.
4. Измените, системное время на ± 50 ед. и выполните по 10 прогонов модели. Ранее представленный текст программы позволяет решить задачу для первого условия задания. Все остальные варианты решения задания получаются при изменении начального времени генерации транзактов, следовательно, первая строка будет иметь следующие варианты записей: **GENERATE 10,5,250,,1** и **GENERATE 10,5,150,,1**.
5. Получите и обработайте статистику количества генерируемых транзактов.
6. Перейдите к выполнению заданного варианта.

Отчет по лабораторной работе

В отчете по лабораторной работе необходимо привести:

- формулировку цели работы;
- постановку задачи в соответствии с вариантом задания;
- листинг заданного варианта программы на языке GPSS;
- результаты моделирования в виде отчетов;
- выводы по работе;
- варианты индивидуальных заданий приведены в табл. 1.

Варианты заданий

Таблица 1

№	Описание работы	№	Описание работы
1	Определить число сгенерированных транзактов. Изменить $T_{нач}$ на ± 50 . Время моделирования – 200.	10	Определить число транзактов в $T=175$ с приоритетами соответственно 1 и 0. GENERATE 10

	Запустить модель 10 раз. GENERATE 10,5,200,,1		GENERATE 20,10,50,,1 Запустить модель 10 раз.
2	Определить число сгенерированных транзактов. Записать оператор GENERATE , генерирующий транзакты на отрезке [6,9]. Время генерации транзактов – 100.	11	Определить число транзактов в T=175 с приоритетами соответственно 1 и 0. GENERATE 10,5 GENERATE 20,5,100,,1 Запустить модель 10 раз.
3	Задать генерацию транзактов в интервалах [4,8] и [4,9]. Определить число транзактов в T = 50. Запустить модель 10 раз.	12	Задать генерацию транзактов в интервале [30,53]. Определить число транзактов в T =500. Запустить модель 10 раз.
4	Определить число транзактов в T=185 с приоритетами соответственно 0,1,2. GENERATE 5,,,1 GENERATE 10 GENERATE 1,,60,,2	13	Определить число сгенерированных транзактов. Изменить T _{нач} на +10 при каждом запуске. Запустить модель 5 раз. GENERATE 20,10,100,,2
5	Определить число транзактов в T=160 с приоритетами соответственно 2.4.6. GENERATE 10,,,2 GENERATE 6,,,6 GENERATE 1,,40,,4	14	Определить число сгенерированных транзактов. Записать блок GENERATE , генерирующий транзакты на отрезке [20,5]. Время генерации транзактов - 200.
6	Определить число транзактов в T=200 с приоритетами соответственно 0,7,13. GENERATE 30 GENERATE 6,,,7 GENERATE 1,,480,,13	15	Задать генерацию транзактов в интервалах [10,9] и [10,7]. Определить число транзактов в T=100. Запустить модель 10 раз.
7	Задать генерацию транзактов в интервалах [10,20] и [20,41]. Определить число транзактов в T = 500. Запустить модель 10 раз.	16	Определить число транзактов в T=180 с приоритетами соответственно 0,1,2. GENERATE 10,,,1 GENERATE 2,,10,2 GENERATE 20
8	Задать генерацию транзактов в интервалах [21,30] и [30,40]. Определить число транзактов в T =200. Запустить модель 20 раз.	17	На временных интервалах [20,33] и [30,51] задать генерацию транзактов. Определить число транзактов в T = 300. Запустить модель 10 раз.
9	Задать генерацию транзактов в интервале [9,20]. Определить число транзактов в T =250. Запустить модель 20 раз.	18	Задать генерацию транзактов в интервале [6,15]. Определить число транзактов в T =125. Запустить модель 20 раз.
10	Определить число транзактов в T=175 с приоритетами соответственно 1 и 0. GENERATE 10 GENERATE 20,10,50,,1 Запустить модель 10 раз.	19	Определить число сгенерированных транзактов. Изменить атрибут A на +/- 3. Запустить модель 10 раз. GENERATE 8,4,100
11	Определить число транзактов в T=175 с приоритетами соответственно 1 и 0. GENERATE 10,5 GENERATE 20,5,100,,1 Запустить модель 10 раз.	20	На временных интервалах [10,20] и [40,65] задать генерацию транзактов. Определить число транзактов в T = 200. Запустить модель 10 раз.

Лабораторная работа № 2. МОДЕЛИРОВАНИЕ РАБОТЫ ОДНОКАНАЛЬНЫХ СИСТЕМ МАССОВОГО ОБСЛУЖИВАНИЯ В GPSS WORLD

Цель работы: освоить методику построения имитационных моделей и их программирование на языке GPSS World на примере одноканальных систем массового

обслуживания. Получить навыки исследование имитационных моделей, обработки и анализа результатов моделирования.

Основы теории

Система массового обслуживания (СМО) – это совокупность обслуживающих устройств и очередей, в которых накапливаются заявки на обслуживание, поступающие на вход системы, когда эти устройства заняты. Тип СМО определяется характером потока заявок, количеством обслуживающих устройств, дисциплин постановки в очередь и выбора из нее и правил, по которым осуществляется обслуживание.

Входящий поток – это последовательность заявок, поступающих на вход СМО. Однородный ординарный поток заявок описывается законом распределения длин интервалов времени между появлением последовательных заявок.

Дисциплины постановки в очередь и выбора из нее определяют порядок постановки заявок в очередь, если заняты устройства обслуживания, и порядок выбора из очереди, если освобождается обслуживающее устройство. Дисциплина FIFO означает постановку заявок в очередь в порядке их поступления. Организация очереди по правилу LIFO предполагает, что на обслуживание первыми выбираются последние заявки из очереди. Выбор заявок из очереди может быть и случайным. Если входной поток неоднородный, то возможна организация выбора из очереди по параметрам заявок (например, мужчины в очереди пропускают женщин вперед).

Очереди могут характеризоваться *ограничениями по допустимой длине* (ограниченным и неограниченным числом мест) или *по правилам пребывания в ней* (с конечным и бесконечным временем).

Процесс обслуживания характеризуется длительностью обслуживания, количеством заявок, которые могут обслуживаться одновременно, и дисциплиной обслуживания.

В общем случае длительность обслуживания – случайная величина; ее задают законом распределения. Обслуживание может осуществляться одним устройством или несколькими устройствами одновременно. В соответствии с установившейся в теории СМО терминологией обслуживающие устройства называют **каналами**, поэтому указанные выше СМО называют, соответственно, одноканальными и многоканальными.

Дисциплины обслуживания определяют:

- при каких условиях прекращается обслуживание заявок;
- как выбирается для обслуживания следующая заявка;
- что делать с частично обслуженной заявкой.

Различают дисциплины обслуживания **бесприоритетные** и **приоритетные**. При неприоритетном обслуживании порядок обслуживания определяется дисциплиной выбора из очереди. При приоритетном обслуживании каждая заявка характеризуется некоторым параметром, который определяет её правило предпочтения при выборе из очереди. Этот параметр может задаваться в числовом виде или в виде функции, которая зависит от времени пребывания в системе.

Выходящий поток – это поток заявок, который покидает систему, причем в нем могут быть как обслуженные, так и не обслуженные заявки. Распределение заявок в выходящем потоке во времени зависит от плотности входящего потока и характеристик работы каналов [1-3].

Подготовка к выполнению работы

Для выполнения лабораторной работы необходимо:

- изучить теоретические основы программирования на языке GPSS;
- выполнить все этапы разработки имитационной модели;

- написать, отладить и реализовать программу, обеспечивающую генерацию транзактов в соответствии с формулировкой задачи;
- выполнить расчет заданных характеристик системы по её аналитической модели;
- обработать результаты моделирования и провести их анализ; сравнивая результаты имитационного моделирования с результатами, полученными расчетом по аналитической модели.

Разработка модели

На первом этапе работы изучается содержательное описание системы с целью её формализации. На втором этапе готовится формализованная схема процесса, на третьем этапе разрабатывается имитационный алгоритм.

Содержательное описание системы

В магазине работает один человек, выполняющий функции продавца и кассира. Покупатели заходят в магазин, знакомятся с товаром и занимают очередь. Кассир обслуживает покупателя в порядке очереди.

Интервал времени Δt между приходом покупателей, время знакомства с товаром t_1 и время обслуживания t_2 – случайные величины с равномерным законом распределения.

Задание: 1) оцените занятость работника на протяжении **4 часов** при следующих исходных данных: $\Delta t \in R[2,12]$; $t_1 \in R[2,8]$; $t_2 \in R[1,5]$; 2) измените, модель таким образом, чтобы получить информацию об очереди, образующейся перед кассой; 3) оформите результаты в виде таблицы и графика распределения времени обслуживания покупателей кассой.

Подготовка формализованной схемы процесса

На данном этапе содержательное описание излагается в терминах выбранной математической схемы (в данном случае – СМО).

С точки зрения теории СМО приход покупателей – это входящий поток заявок на обслуживание. Поскольку нет иных указаний, можно считать его однородным и ординарным. ПРОДАВЕЦ-КАССИР – это обслуживающее устройство, или канал. Если заявка претендует на обслуживание каналом, а он занят, то заявка становится в ОЧЕРЕДЬ (рис. 1.1).

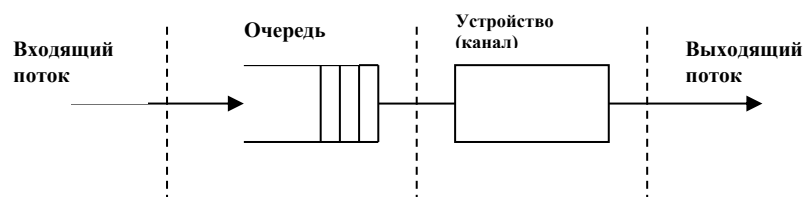


Рис. 1.1.

Разработка имитационного моделирующего алгоритма

Имитационный алгоритм является моделью системы: его структура воспроизводит временные и причинно-следственные отношения, имеющие место в исследуемой системе.

Поскольку предполагается программирование моделирующего алгоритма на языке GPSS, алгоритм должен формулироваться в терминах, принятых в этом языке. Это блоки и транзакты.

Можно применить следующую методику построения алгоритма.

Алгоритм должен отражать основные этапы «жизненного цикла» ПОКУПАТЕЛЯ - транзакта, т.е. его действия от момента его прихода в магазин до момента выхода из магазина.

Для генерации входящего потока транзактов используем оператор **GENERATE** (*Генерировать*), для данного примера – в следующем виде:

GENERATE 7,5 ; приход покупателей

На следующем этапе ПОКУПАТЕЛЬ задерживается в течение определенного интервала времени для знакомства с товаром в магазине. Для моделирования этого процесса используется оператор **ADVANCE** (*Задержать*):

ADVANCE 5,3 ; знакомство с товаром

Следующее событие для покупателя имеет условный характер: 1) если продавец свободен, то он обслуживает покупателя; 2) если продавец занят, то покупатель становится в очередь.

Рассмотрим первый случай.

Для данного примера программа записывается в следующем виде:

SEIZE PROD ; обращение к продавцу

Подпрограмма ОБСЛУЖИВАНИЕ транзакта в языке GPSS интерпретируется оператором **ADVANCE**, который создает уведомление о событии в списке будущих событий для последующей передачи управления подпрограмме освобождения ресурса транзактом.

Для данного примера оператор записывается в следующем виде:

ADVANCE 3,2 ; покупка товаров

Подпрограмма ОСВОБОЖДЕНИЕ, инициированная оператором **RELEASE** (*Освободить*), освобождает ранее занятый канал и 1) изменяет состояние канала на «свободный», после чего передает управление подпрограмме УНИЧТОЖЕНИЕ транзактов; 2) проверяет, есть ли транзакты в ОЧЕРЕДИ к каналу. Если они есть, то очередной транзакт выбирается из ОЧЕРЕДИ и для него формируется ЗАПРОС-НАЗНАЧЕНИЕ.

Подпрограмма УНИЧТОЖЕНИЕ транзактов, которой в языке GPSS соответствует оператор **TERMINATE** (*Завершить*), удаляет из модели транзакты, которые более не участвуют в работе системы, необходима для уничтожения структуры данных каждого транзакта, т.е. для освобождения памяти, выделенной под транзакт. Если транзакты не уничтожать, то со временем они переполняют память компьютера.

Для данного примера запишем операторы в следующем виде:

RELEASE PROD ; освобождение продавца
TERMINATE ; уход покупателей

В языке GPSS длительность прогона модели можно ввести двумя способами:

1) завершить моделирование после того, как модель покинет заданное число транзактов определенного типа; 2) завершить моделирование по истечению заданного интервала времени.

В первом способе, в операторе **START** (*Начать*) операнду **A** присваивается начальное значение заданного числа транзактов, оно записывается в ячейку памяти ЭВМ – счетчик завершения. Во всех операторах **TERMINATE**, через которые транзакты заданного типа покидают модель, операнду **A** присваивается значение «1» или любое другое, отличное от нуля (соответственно содержательному значению транзактов); в других операторах **TERMINATE** значение операнда **A** используется по умолчанию (**A=0**), так как значение счетчика завершения не будет зависеть от этих операторов. В процессе моделирования транзакты попадают в оператор **TERMINATE** и уменьшают значение счетчика на величину указанную в операнде **A**. Моделирование заканчивается, когда значение счетчика становится равным нулю или отрицательному значению.

Во втором способе (как, в частности, и для данного примера), первоначально задается единица модельного времени (например, 1 минута) и системное время работы

моделируемой система (для данного примера - 240 минут), для чего в модель вводится таймер-сегмент, состоящий из двух операторов:

GENERATE 240 ; время работы модели

TERMINATE 1 ; уменьшение общего времени моделирования на 1 мин

Во всех других операторах **TERMINATE** в модели значение операнда **A** используется по умолчанию (**A=0**). Это означает, что прекращение моделирования, определяемое счетчиком завершения, не будет зависеть от других операторов **TERMINATE**; в команде **START** операнд **A** должен равняться «1»:

START 1 ; пропускаются по одному покупателю

Таким образом, в процессе моделирования завершение движения транзактов в других операторах **TERMINATE** не влияет на счетчик завершения. В момент времени 240 транзакт выйдет из оператора **GENERATE** и сразу же перейдет в оператор **TERMINATE**. Содержимое счетчика завершения уменьшится на единицу, и интерпретатор завершит моделирование.

Подготовка имитационной модели к работе

1. Запустите систему моделирования GPSS.
2. В окне текстового редактора введите текст программы, который приводится ниже:

; GPSSW File PRODOVEC.GPS

Моделирование работы магазина

;Имитация прихода покупателей

GENERATE 7,5 ; приход покупателей

ADVANCE 5,3 ; знакомство с товаром

;Имитация работы магазина

SEIZE PROD ; обращение к продавцу

ADVANCE 3,2 ; покупка товаров

RELEASE PROD ; освобождение продавца

TERMINATE ; уход покупателей

;Задание времени моделирования

GENERATE 240 ; время работы модели

TERMINATE 1 ;уменьшение общего времени моделирования на 1 мин

START 1 ; система пропускает по одному покупателю

3. Запустите программу моделирования на выполнение.
4. Если программа завершена успешно, то выведите результаты моделирования в виде отчета. Распечатка выходных данных для рассматриваемой задачи приведена ниже.

GPSS World Simulation Report - Untitled Model 2.2.1

START TIME	END TIME	BLOCKS	FACILITIES	STORAGES
0.000	240.000	8	1	0

LABEL	LOC	BLOCK TYPE	ENTRY	COUNT	CURRENT	COUNT	RETRY
	1	GENERATE		32		0	0
	2	ADVANCE		32		1	0
	3	SEIZE		31		0	0
	4	ADVANCE		31		0	0
	5	RELEASE		31		0	0
	6	TERMINATE		31		0	0
	7	GENERATE		1		0	0
	8	TERMINATE		1		0	0

FACILITY	ENTRIES	UTIL.	AVE. TIME	AVAIL.	OWNER	PEND	INTER	RETRY	DELAY
PROD	31	0.384	2.969	1	0	0	0	0	0

FEC XN	PRI	BDT	ASSEM	CURRENT	NEXT	PARAMETER	VALUE
34	0	241.166	34	0	1		
33	0	243.017	33	2	3		
35	0	480.000	35	0	7		

GPSS World Simulation Report - Untitled Model 2.2.1

START TIME	END TIME	BLOCKS	FACILITIES	STORAGES
0.000	240.000	8	1	0

LABEL	LOC	BLOCK TYPE	ENTRY COUNT	CURRENT COUNT	RETRY
	1	GENERATE	32	0	0
	2	ADVANCE	32	1	0
	3	SEIZE	31	0	0
	4	ADVANCE	31	0	0
	5	RELEASE	31	0	0
	6	TERMINATE	31	0	0
	7	GENERATE	1	0	0
	8	TERMINATE	1	0	0

FACILITY	ENTRIES	UTIL.	AVE. TIME	AVAIL.	OWNER	PEND	INTER	RETRY	DELAY
PROD	31	0.384	2.969	1	0	0	0	0	0

FEC XN	PRI	BDT	ASSEM	CURRENT	NEXT	PARAMETER	VALUE
34	0	241.166	34	0	1		
33	0	243.017	33	2	3		
35	0	480.000	35	0	7		

Выходные статистические данные для рассмотренного примера содержат следующую информацию:

- 1) в двух верхних строках указывается общая информация о результатах работы модели:
START TIME (*Начальное время*) – модельное время в момент начала моделирования - **0**;
END TIME (*Время окончания*) – конечное время моделирования - **240**;
BLOCKS (*Число блоков*) – количество операторов, использованных в текущей модели - **8**;
FACILITIES (*Число каналов обслуживания*) – количество каналов, использованных в модели - **1**;
STORAGES (*Число накопителей*) – количество многоканальных устройств – **0**;
- 2) в последующих строках перечисляется информация об операторах модели и количестве входов в них транзактов. При этом каждый оператор имеет свой числовой номер:
LABEL (*Метка*) – алфавитно-цифровое имя данного оператора (**если оно задано**);
LOC (*Позиция*) – числовой номер позиции данного оператора в тексте модели - **с 1 по 8**;
BLOCK TYPE (*Тип блока*) – тип оператора в GPSS;
ENTRY COUNT (*Входные транзакты*) – количество транзактов вошедших в данный оператор, за время моделирования - **32**;
CURRENT COUNT (*Текущие транзакты*) – количество транзактов, находящихся в данном операторе к моменту завершения времени моделирования - **1**;
RETRY (*Повтор*) – количество транзактов, ожидающих специального условия для повторного входа, зависящего от состояния данного оператора – **0** (оператор не занят);
- 3) ниже приводится информация о результатах моделирования устройств (каналов) обслуживания:
FACILITY (*Каналы обслуживания*) – приводится имя и номер канала обслуживания - **PROD**;
ENTRIES (*Число входов*) – количество раз, когда устройство было занято за время моделирования - **31**;
UTIL. (*Коэффициент использования*) – доля времени моделирования, в течение которого устройство было занято – **0,384**;
AVE.TIME (*Среднее время обслуживания*) – время занятия устройства одним транзактом в течение времени моделирования – **2,969**;
AVAIL (*Доступность*) – состояние устройства в конце моделирования - **1** – устройство доступно (0 – недоступно);

OWNER (*Возможное число входов*) – номер транзакта, который занимает устройство - **0** (устройство не занято);

PEND (*Зависший*) – количество транзактов, ожидающих выполнения с прерыванием других транзактов - **0**;

INTER (*Прервать*) – количество транзактов, прерванных на данный момент – **0**;

RETRY (*Повтор*) – количество транзактов, ожидающих выполнения специального условия на повторное выполнение, зависящее от состояния данного устройства – **0**;

DELAY (*Отказано*) – количество транзактов, ожидающих занятия устройства (входят также транзакты, ожидающие занятия устройства в режиме прерывания) – **0**;

4) конечная строка содержит информацию о списках текущих и будущих событий:

FEC XN (*Список будущих событий*) – номер каждого транзакта, находящегося в списке будущих событий – **34; 33; 35**;

PRI (*Приоритет*) – приоритет транзакта - **0**;

BDT (*Таблица модельных событий*) – определяются моменты модельного времени, когда последние транзакты покинут список будущих событий – **241,166; 243,017; 480,000**;

ASSEM (*Семейство*) – номера транзактов - **34; 33; 35**;

CURRENT (*Текущий*) – номера блоков, в которых находятся транзакты в конце моделирования - **0; 2; 0**;

NEXT (*Следующий*) – номера блоков, в которые должны войти транзакт - **1; 3; 7**;

PARAMETER (*Параметр*) – имя или номер параметра транзакта;

VALUE (*Значение*) – значение параметра.

5. По приведенной статистической информации можно сделать соответствующие выводы:

1) среднее время обслуживания покупателей составляет – **2,97** мин.;

2) общее число обслуженных покупателей – **31**;

3) средняя загрузка продавца (в %) в течение 4 часов составила $\approx 40\%$.

Рассмотрим второй случай.

Для того чтобы получить информацию об очереди, образующейся перед кассой, следует применить оператор **QUEUE** (*Очередь*), который в совокупности с оператором **DEPART** (*Выйти*) собирает статистическую информацию о работе моделируемой очереди.

В данном примере это будет выглядеть так:

QUEUE OCH ; включение в очередь

.....

DEPART OCH ; выход из очереди

Текст программы после редактирования приводится ниже:

; GPSSW File PRODOVEC.GPS

Моделирование работы магазина

; Имитация прихода покупателей

GENERATE 7,5 ; приход покупателей

ADVANCE 5,3 ; знакомство с товаром

; Имитация работы магазина

QUEUE OCH ; включение в очередь

SEIZE PROD ; обращение к продавцу

DEPART OCH ; выход из очереди

ADVANCE 3,2 ; покупка товаров

RELEASE PROD ; освобождение продавца

TERMINATE ; уход покупателей

; Задание времени моделирования

GENERATE 240 ; время работы модели
TERMINATE 1 ; уменьшение общего времени моделирования на 1 мин
START 1 ; система пропускает по одному покупателю

В этой модели момент включения каждого транзакта в очередь **ОСН** совпадает с моментом его обращения к оператору **SEIZE**, т.к. оператор **QUEUE** выполняется в модельном времени мгновенно. Каждый транзакт находится в очереди до тех пор, пока не займет устройство **PROD**. Момент занятия устройства совпадает с моментом выхода транзакта из очереди. В данный момент очередь **ОСН** имеет естественную интерпретацию как очередь посетителей к продавцу, а длина очереди интерпретируется как число посетителей в очереди.

Дополнительные статистические данные для данного примера содержат следующую информацию:

QUEUE	MAX	CONT.	ENTRY	ENTRY(0)	AVE.CONT.	AVE.TIME	AVE.(-0)	RETRY
ОСН	1	0	31	27	0.032	0.248	1.924	0

Расшифровка полученной информации:

QUEUE (*Очередь*) – номер или имя очереди – **PROD**;

MAX (*Максимальное содержание*) – максимальная длина очереди в течение периода моделирования - **1**;

CONT. (*Текущее содержание*) – текущее содержимое очереди в конце процесса моделирования - **0**;

ENTRY (*Число входов*) – общее количество входов транзактов в очередь в течение времени моделирования - **31**;

ENTRY(0) (*Число нулевых входов*) – общее количество входов транзактов в очередь с нулевым временем ожидания - **27**;

AVE.CONT. (*Среднее содержимое*) – среднее значение содержимого очереди в течение времени моделирования – **0,032**;

AVE.TIME (*Среднее время*) – среднее время пребывания одного транзакта в очереди с учетом всех входов в очередь – **0,248**;

AVE.(-0) (*Среднее время без нулевых входов*) – среднее время пребывания одного транзакта в очереди без учета «нулевых» входов в очередь – **1,924**;

RETRY (*Повторные входы*) – количество транзактов, ожидающих выполнения специального условия, зависящего от состояния очереди - **0**.

По приведенной статистической информации можно сделать следующие выводы:

- 1) максимальная длина очереди в течении 4 часов работы магазина – **1 покупатель**;
- 2) среднее время пребывания покупателя в очереди – **0,248 мин.**

Для того чтобы сформировать таблицу с информацией о посещении покупателями данного магазина, воспользуемся оператором **TABLE** (*Таблица*). Оператор табулирования **TABULATE** (*Табулировать*), выполняющий сбор данных по времени и числу обслуживания в системе, записывается с тем же именем таблицы, которое было определено в операторе **TABLE**. Для данного примера можно записать:

MAG TABLE M1,2,2,5 ; формирование таблицы

.....

TABULATE MAG ; табулировать магазин

Текст программы после соответствующего редактирования приводится ниже:

; GPSSW File PRODOVEC.GPS

Моделирование работы магазина

```

*****
; Имитация прихода покупателей
MAG  TABLE M1,2,2,5 ; формирование таблицы
      GENERATE 7,5 ; приход покупателей
      ADVANCE 5,3 ; знакомство с товаром
;Имитация работы магазина
      QUEUE OCH ; включение в очередь
      SEIZE PROD ; обращение к продавцу
      DEPART OCH ; выход из очереди
      ADVANCE 3,2 ; покупка товаров
      RELEASE PROD ; освобождение продавца
      TABULATE MAG ; табулировать магазин
      TERMINATE ; уход покупателей
; Задание времени моделирования
      GENERATE 240 ; время работы модели
      TERMINATE 1 ; уменьшение общего времени моделирования на 1 мин
      START 1 ; система пропускает по одному покупателю

```

Дополнительные выходные статистические данные для данного примера содержат следующую информацию:

TABLE	MEAN	STD.DEV.	RANGE	RETRY	FREQUENCY	CUM.%
MAG	8.181	2.687		0		
			2.000 -	4.000	2	6.45
			4.000 -	6.000	7	29.03
			6.000 -	8.000	4	41.94
			8.000 -		18	100.00

Информация о таблицах:

TABLE (*Таблица*) – имя или номер таблицы или Q – таблицы – **MAG**;
MEAN (*Средняя*) – среднее значение табулируемого аргумента – **8,181**;
STD.DEV. (*Среднеквадратическое отклонение*) – оценка среднеквадратического отклонения – **2,687**;
RANGE (*Область*) – нижний и верхний пределы частотного класса – **2-4**;
RETRY (*Повторные входы*) – количество транзактов, ожидающих выполнения специального условия, зависящего от состояния данной таблицы - **0**;
FREQUENCY (*Частота*) – суммарная величина, которая формируется при попадании табулируемого аргумента в указанные границы - **2**;
CUM % (*Суммарный процент*) – величина частоты в процентах к общему количеству значений табулируемого аргумента – **6,45**.

Для большей наглядности выведем график того, как меняется длина очереди к продавцу за весь период моделирования. Для данного примера он будет выглядеть как на рис. 1.2.

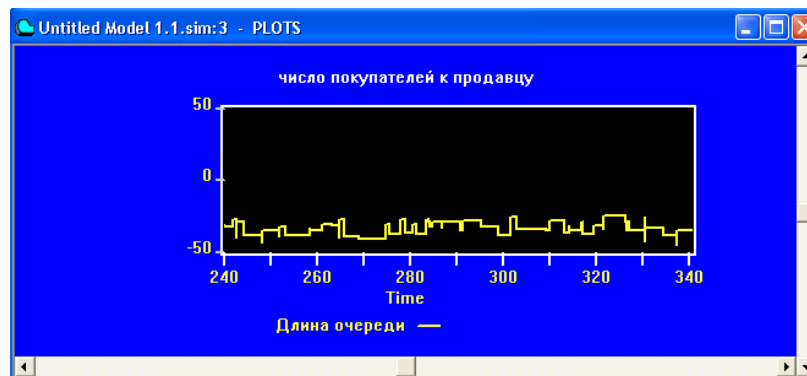


Рис. 1.2.

Помимо данных, представляемых в отчете в виде таблицы, при наличии оператора **TABLE**, можно вывести информацию в виде гистограммы, иллюстрирующей распределение времени обслуживания транзакта. Для данного примера она будет выглядеть как показано на рис. 1.3.

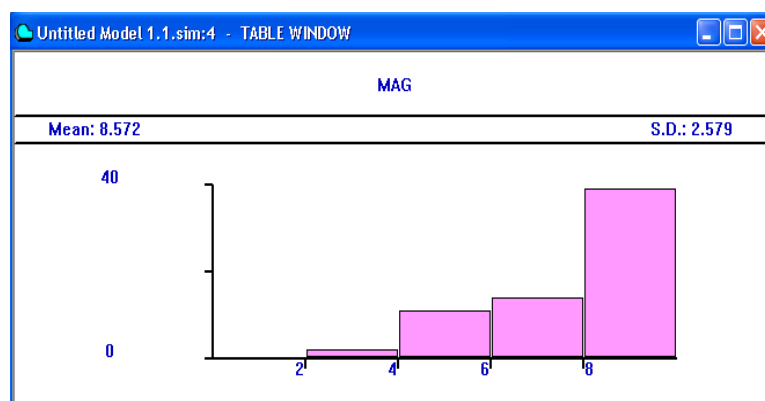


Рис. 1.3.

Расчет характеристик системы по ее аналитической модели

Функционирование СМО можно представить как последовательную смену ее возможных состояний, каждое из которых характеризуется количеством заявок (покупателей, рабочих, машин, неполадок) в системе. Поскольку эта величина имеет случайный характер, наличие n заявок задается вероятностью - P_n . Вероятность P_0 характеризует состояние, когда в системе нет заявок и канал обслуживания простаивает.

Важными параметрами функционирования СМО являются среднее число заявок, находящихся в системе - N_c , и средняя длина очереди - N_o . Исходными параметрами, характеризующими СМО, являются:

- число каналов обслуживания (касс, компьютеров, ремонтных бригад) - M ;
- интенсивность поступления заявок на обслуживание, т.е. среднее число входящих заявок в единицу времени - λ ;
- интенсивность обслуживания заявок каналом - μ .

Интенсивность поступления заявок на обслуживание определяется как величина, обратная среднему времени между поступлениями двух смежных заявок (t_p): $\lambda = 1/t_p$, для данного примера $\lambda = 1/7 = 0,143$ мин⁻¹.

Интенсивность обслуживания заявок определяется как величина, обратная среднему времени обслуживания одной заявки (t_o): $\mu = 1/t_o$, для данного примера $\mu = 1/3 = 0,333$ мин⁻¹.

Вероятность простоя канала определяется как $P_o = 1 - \rho$, где ρ – коэффициент загрузки системы $\rho = \lambda/\mu$, для данного примера $\rho = 0,143/0,333 = 0,429 \text{ мин}^{-1}$. Тогда $P_o = 1 - 0,429 = 0,571$. Загрузка, меньшая 1, говорит о том, что очередь к каналу не будет бесконечно большой.

Среднее число заявок N_c , находящихся в системе определяется из формулы

$$N_c = \rho / (1 - \rho) = 0,429 / (1 - 0,429) = 0,751.$$

Среднее число заявок, находящихся в очереди N_o , будет вычислено так:

$$N_o = \rho N_c = 0,429 \cdot 0,751 = 0,322.$$

Среднее время ожидания обслуживания определяется

$$t_{ож} = N_o / \mu = 3 \cdot 0,322 = 0,966 \text{ мин}^{-1}.$$

Среднее время пребывания заявки в системе вычисляется

$$t_c = 1 / \mu (1 - \rho) = 3 / 0,571 = 5,253 \text{ мин}^{-1}.$$

Отчет по лабораторной работе

В отчете по лабораторной работе необходимо привести:

- формулировку цели работы;
- постановку задачи в соответствии с вариантом задания;
- аналитический расчет заданных характеристик;
- листинг программы на языке GPSS;
- результаты моделирования в виде отчетов,
- результаты моделирования в виде графика и гистограммы;
- выводы по работе.

Варианты индивидуальных заданий

В парикмахерскую с одним креслом приходят клиенты; интервалы времени между ними случайны и распределены равномерно в интервале $[a \pm b]$ минут. Время обслуживания также распределено равномерно на интервале $[c \pm d]$ минут. Клиенты, приходящие в парикмахерскую, обслуживаются в порядке очереди «первый пришел – первым обслужен».

Задание: необходимо разработать и отладить имитационную модель парикмахерской в системе моделирования GPSS; модель должна обеспечить получение статистических данных об очереди. Исходные данные приведены в табл. 2.

Таблица 2

№	Среднее значение времени генерации транзактов a	Разброс времени генерации транзактов b	Среднее значение времени обслуживания устройства c	Разброс времени обслуживания устройства d	Время моделирования	Параметры для построения таблицы равномерного распределения
1	9	3	6	3	4	[0,80] шаг 5
2	11	4	7	3	4	[0,70] шаг 5
3	15	6	8	3	4	[0,60] шаг 5
4	32	10	9	3	5	[0,50] шаг 5
5	25	12	10	4	5	[0,40] шаг 5
6	18	5	11	4	5	[0,80] шаг 5
7	27	15	12	4	6	[0,70] шаг 5
8	35	15	13	4	6	[0,60] шаг 5
9	25	13	14	5	6	[0,50] шаг 5
10	35	14	15	5	7	[0,40] шаг 5
11	13	7	16	5	7	[0,90] шаг 5
12	19	8	17	5	7	[0,80] шаг 5
13	27	16	18	6	8	[0,70] шаг 5
14	36	17	19	6	8	[0,60] шаг 5
15	35	10	20	6	8	[0,50] шаг 5
16	20	5	12	6	9	[0,40] шаг 5

17	35	11	11	7	9	[0,90] шаг 5
18	27	13	10	7	9	[0,80] шаг 5
19	25	12	9	7	10	[0,70] шаг 5
20	19	11	7	5	10	[0,60] шаг 5

Лабораторная работа № 3. МОДЕЛИРОВАНИЕ РАБОТЫ МНОГОКАНАЛЬНЫХ СИСТЕМ МАССОВОГО ОБСЛУЖИВАНИЯ В СИСТЕМЕ GPSS WORLD

Цель работы: Освоить методику построения имитационных моделей и их программирование на языке GPSS World на примере многоканальных систем массового обслуживания. Получить навыки исследования имитационных моделей обработки и анализа результатов моделирования.

Задание: написать и отладить программу, моделирующую систему массового обслуживания.

Содержательное описание

Переговорный пункт имеет одно помещение с тремя кабинками для переговоров. Если посетитель застает свободной хотя бы одну кабинку, то он оплачивает будущий разговор в кассе, затем ждет соединения и ведет переговоры, после чего покидает переговорный пункт. Если посетитель застает все кабинки занятыми, он ждет освобождения одной из них у входа в переговорный пункт.

Интервал времени между приходами посетителей – случайный, распределенный по равномерному закону, составляет $1,5 \pm 0,4$ мин. Время ожидания соединения – $3,5 \pm 1,1$ мин., время разговора посетителя по телефону – $4,4 \pm 1,35$ мин. Длительность разговора и время ожидания соединения – случайные величины, подчиняются равномерному распределению.

Требуется:

- оценить занятость переговорного пункта на протяжении 8 часов;
- определить максимальное, среднее и текущее число посетителей на переговорном пункте;
- оценить среднее время обслуживания посетителей (привести таблицу и гистограмму);
- получить информацию об очереди, образующейся на переговорном пункте.

Подготовка к выполнению работы

Для выполнения лабораторной работы необходимо:

- изучить теоретические основы программирования на языке GPSS;
- выполнить все этапы разработки имитационной модели;
- написать, отладить и реализовать программу, обеспечивающую генерацию транзактов в соответствии с формулировкой задачи в системе моделирования GPSS;
- вывести результаты моделирования в виде отчета и гистограммы и провести их анализ.

Создание имитационной модели

На первом этапе работы изучается содержательное описание системы с целью её формализации. На втором этапе готовится формализованная схема процесса, на третьем этапе разрабатывается имитационный алгоритм.

Разработка формализованной схемы процесса

На этом этапе содержательное описание излагается в терминах выбранной математической схемы (в данном случае – СМО).

С точки зрения теории СМО приход посетителей – это входящий поток заявок на обслуживание. Поскольку нет иных указаний, можно считать его однородным и ординарным. Телефонные кабины переговорного пункта – это обслуживающие устройства, или каналы. Если заявка претендует на обслуживание каналом, а он занят, то заявка становится в ОЧЕРЕДЬ (рис. 1.4).

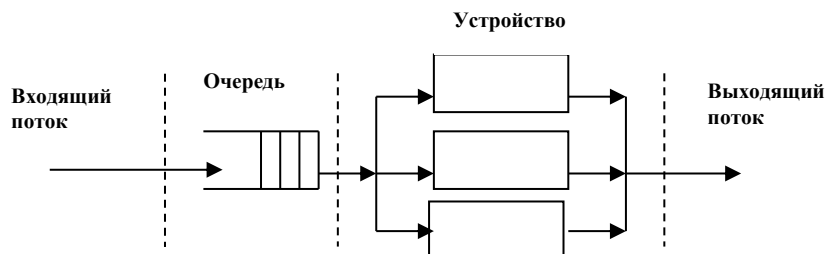


Рис. 1.4.

Разработка имитационного моделирующего алгоритма

Применим следующую методику построения алгоритма, отражающего основные этапы «жизненного цикла» посетителя, т.е. его действий от момента прихода на переговорный пункт до момента выхода из переговорного пункта.

Прежде всего следует указать основной параметр функционирования переговорного пункта, а именно – количество кабин. Для этого воспользуемся оператором **STORAGE** (*Накопитель*), который в данном примере для накопителя под символьным именем **Punkt** будет выглядеть так:

Punkt STORAGE 3 ; емкость переговорного пункта

Затем, используя оператор **TABLE**, сформируем таблицы с информацией об использовании посетителями переговорного пункта:

Transit TABLE M1,.5,1,30 ; формирование таблицы

В основной части алгоритма будем моделировать поток посетителей и работу переговорного пункта. Так как посетители (транзакты) непрерывно приходят на переговорный пункт (требуют обслуживания) на протяжении некоторого периода времени наблюдения за системой (это интервал, в течение которого моделируется система), то для генерации потока транзактов используем оператор **GENERATE**:

GENERATE 1.85,1 ; приход посетителей

Посетитель, пришедший на переговорный пункт, сначала определяет количество присутствующих посетителей переговорного пункта. Если их в пункте уже три, то пункт занят, и новый посетитель ожидает его освобождения. Если же число присутствующих посетителей меньше трех, то вновь прибывший идет к кассе для оплаты. Эту ситуацию можно промоделировать с помощью оператора **GATE**, который логическим ключом **SNF** (*Storage Not Full - Накопитель не полон*) проверяет, какой случай имеет место в данной реализации:

Povtor GATE SNF Punkt,Zanyt; регулируется поток посетителей в зависимости от состояния переговорного пункта

Если накопитель под символьным именем **Punkt** не полон, то посетитель входит на переговорный пункт. Это моделируется оператором **ENTER** (*Войти*):

ENTER Punkt ; войти на переговорный пункт

Далее посетитель встает в очередь (если она есть) для оплаты телефонного разговора. Это задается оператором **QUEUE**, который в совокупности с оператором

DEPART собирает статическую информацию о работе моделируемой очереди. Для данного примера запишем:

QUEUE Ocher_kassir ; включение в очередь

Посетитель может выйти из очереди только тогда, когда освободится кассир. Для моделирования этого обстоятельства вводится оператор **SEIZE**, который определяет занятость данного ресурса; при его освобождении очередной посетитель выходит из очереди и обслуживается. Это будет записано так:

SEIZE Kassir ; обращение к кассиру

Выход посетителей из очереди фиксируется оператором **DEPART** с соответствующим названием очереди:

DEPART Ocher_kassir ; выход из очереди

На следующем этапе посетитель задерживается на определенный интервал модельного времени для оплаты разговора. Для моделирования этого процесса используем оператор **ADVANCE**:

ADVANCE 1.5,0.4 ; оплата за разговор

После обслуживания кассиром посетитель идет к телефону для ведения переговоров. Однако перед этим системе должно быть послано сообщение об освобождении данного ресурса (кассира). Это делается с помощью оператора **RELEASE**, который в данной задаче записывается:

RELEASE Kassir ; освобождение кассира

После обслуживания в кассе посетитель направляется к кабине с телефонным аппаратом и ведет переговоры. В языке GPSS это действие интерпретируется оператором **ADVANCE**, который создает уведомление о событии в списке будущих событий для последующей передачи управления подпрограмме освобождения ресурса транзактом. Для данного примера оператор записывается в следующем виде:

ADVANCE 4.4,1.35 ; время разговора посетителей

После проведения переговоров посетитель освобождает телефонный аппарат и уходит из переговорного пункта (покидает систему). Это действие может быть представлено оператором освобождения накопителя **LEAVE** (*Оставить*) под символьным именем **Punkt**:

LEAVE Punkt ; оставить переговорный пункт

Оператор табулирования, выполняющий сбор данных по времени и числу обслуживаний в системе, записывается как **TABULATE** с тем же именем таблицы, которое было определено в операторе **TABLE**. Для данного примера можно записать:

TABULATE Transit ; табулировать переговорный пункт

После этого транзакт (посетитель) покидает модель (переговорный пункт), что моделируется оператором **TERMINATE**:

TERMINATE ; уход посетителей

Если все телефоны переговорного пункта заняты, то посетитель ожидает освобождения одного из них, данный процесс можно промоделировать оператором **ADVANCE** в следующем виде:

Zanyt ADVANCE 3.5,1.1 ; время ожидания освобождения телефона

Поскольку СМО многоканальная, то необходимо использовать оператор **TRANSFER** (*Передать*) для возможности направления посетителей к незанятому каналу обслуживания (телефонному аппарату). Для данного примера запишем это условие в следующем виде:

TRANSFER ,Povtor ; переход к оператору с меткой Povtor

Временной режим моделирования системы задается совокупностью операторов:

GENERATE 480 ; время работы модели

TERMINATE 1 ; уменьшение общего времени моделирования на 1 мин

START 1 ; пропускаются по одному посетителю

Подготовка имитационной модели к работе

1. Запустите систему моделирования GPSS.
2. В окне текстового редактора введите текст программы, который приводится ниже:

```
; GPSSW File PRODOVEC.GPS  
*****  
*Моделирование работы переговорного пункта*  
*****  
; Вместимость переговорного пункта  
Punkt STORAGE 3 ; емкость переговорного пункта  
Transit TABLE M1,5,1,30 ; формирование таблицы  
; Имитация прихода посетителей  
GENERATE 1.85,1 ; приход посетителей  
Povtor GATE SNF Punkt,Zanyt ; регулируется поток посетителей  
; Имитация работы переговорного пункта  
ENTER Punkt ; войти на переговорный пункт  
QUEUE Ocher_kassir ; включение в очередь  
SEIZE Kassir ; обращение к кассиру  
DEPART Ocher_kassir ; выход из очереди  
ADVANCE 1.5,0.4 ; оплата за разговор  
RELEASE Kassir ; освобождение кассира  
ADVANCE 4.4,1.35 ; время разговора посетителей  
LEAVE Punkt ; оставить переговорный пункт  
TABULATE Transit ; табулировать переговорный пункт  
TERMINATE ; уход посетителей  
Zanyt ADVANCE 3.5,1.1 ; время ожидания освобождения телефона  
TRANSFER ,Povtor ; переход к оператору с меткой Povtor  
; Задание времени моделирования  
GENERATE 480 ; время работы модели  
TERMINATE 1 ; уменьшение общего времени моделирования на 1 мин  
START 1 ; пропускаются по одному посетителю
```

3. Запустите программу моделирования на ее выполнение.
4. Если программа завершена успешно, то выведите результаты моделирования в виде отчета. Распечатка выходных данных для рассматриваемой задачи приведена ниже.

```
GPSS World Simulation Report - Untitled Model 1.4.1  
START TIME          END TIME  BLOCKS  FACILITIES  STORAGES  
      0.000          480.000    16         1         1  
  
LABEL      LOC  BLOCK TYPE      ENTRY COUNT  CURRENT  COUNT  RETRY  
POVTOR      1    GENERATE          259          0         0  
            2    GATE            2992          0         0  
            3    ENTER            221          0         0  
            4    QUEUE            221          0         0  
            5    SEIZE            221          0         0  
            6    DEPART            221          0         0  
            7    ADVANCE            221          1         0  
            8    RELEASE            220          0         0  
            9    ADVANCE            220          2         0  
           10    LEAVE            218          0         0  
           11    TABULATE            218          0         0  
           12    TERMINATE            218          0         0  
ZANYT       13    ADVANCE            2771         38         0  
            14    TRANSFER            2733          0         0  
            15    GENERATE             1          0         0  
            16    TERMINATE             1          0         0
```


FACILITY	ENTRIES	UTIL.	AVE. TIME	AVAIL.	OWNER	PEND	INTER	RETRY	DELAY
KASSIR	221	0.695	1.510	1	155	0	0	0	0

QUEUE	MAX	CONT.	ENTRY	ENTRY(0)	AVE.CONT.	AVE.TIME	AVE.(-0)	RETRY
OCHER_KASSIR	2	0	221	127	0.161	0.350	0.823	0

STORAGE	CAP.	REM.	MIN.	MAX.	ENTRIES	AVL.	AVE.C.	UTIL.	RETRY	DELAY
PUNKT	3	0	0	3	221	1	2.860	0.953	0	0

TABLE	MEAN	STD.DEV.	ANGE	RETRY	FREQUENCY	CUM.%
TRANSIT	39.306	40.043			0	
			4.050 -	5.050	4	1.83
			5.050 -	6.050	11	6.88
			6.050 -	7.050	10	11.47
			7.050 -	8.050	9	15.60
			8.050 -	9.050	8	19.27
			9.050 -	10.050	8	22.94
			10.050 -	11.050	8	26.61
			11.050 -	12.050	4	28.44
			12.050 -	13.050	4	30.28
			13.050 -	14.050	4	32.11
			14.050 -	15.050	6	34.86
			15.050 -	16.050	4	36.70
			16.050 -	17.050	4	38.53
			17.050 -	18.050	2	39.45
			18.050 -	19.050	5	41.74
			19.050 -	20.050	5	44.04
			20.050 -	21.050	2	44.95
			21.050 -	22.050	3	46.33
			22.050 -	23.050	2	47.25
			23.050 -	24.050	1	47.71
			24.050 -	25.050	1	48.17
			25.050 -	26.050	6	50.92
			26.050 -	27.050	1	51.38
			27.050 -	28.050	2	52.29
			28.050 -		104	100.00

FEC XN	PRI	BDT	ASSEM	CURRENT	NEXT	PARAMETER	VALUE
243	0	480.078	243	13	14		
201	0	480.138	201	13	14		
254	0	480.148	254	13	14		
247	0	480.238	247	13	14		
249	0	480.243	249	13	14		
229	0	480.262	229	13	14		
180	0	480.346	180	13	14		

... и т.д.

5. Помимо данных, представляемых в отчете в виде таблицы, выведем информацию в виде гистограммы, иллюстрирующей распределение времени обслуживания транзакта. Для данного примера она будет выглядеть так, как показано на рис. 1.5.

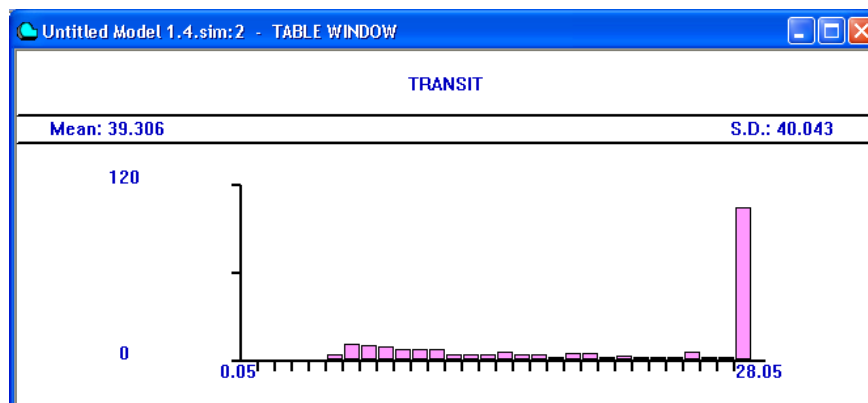


Рис. 1.5.

6. Выводы по работе: за моделируемый интервал времени работы системы (8 часов) было сгенерировано 259 транзактов: столько посетителей пришло ко входу в переговорный пункт. Количество посетителей, оплативших переговоры, составило 221 человек. Однако только 218 посетителей переговоры провели. Максимальная длина очереди к кассиру составила 2 посетителя. Загрузка устройства «касса» (коэффициент использования устройства) составила 0,695. Среднее время обслуживания устройством «касса» одного посетителя в течение времени моделирования составило 1,51 мин. Максимальное число использованных кабин переговорного пункта за период моделирования – 3. Среднее значение количества занятых кабин переговорного пункта за время моделирования – 2,86. Коэффициент использования кабин переговорного пункта составил 0,953.

Отчет по лабораторной работе

В отчете по лабораторной работе необходимо привести:

- формулировку цели работы;
- постановку задачи в соответствии с вариантом задания;
- листинг программы на языке GPSS;
- результаты моделирования в виде отчетов;
- результаты моделирования в виде графика и гистограммы;
- выводы по работе.

Варианты индивидуальных заданий

Морские суда двух типов прибывают в порт, где происходит их разгрузка. В порту есть два буксира, обеспечивающих ввод и вывод кораблей из порта. К первому типу судов относятся корабли малого тоннажа, которые требуют использования одного буксира. Корабли второго типа имеют большие размеры, и для входа и выхода из порта требуется два буксира. Из-за различия размеров двух типов кораблей необходимы и причалы различного размера. Кроме того, корабли имеют различное время погрузки - разгрузки.

Необходимо построить модель системы, в которой можно оценить время ожидания кораблями каждого типа входа в порт (время ожидания входа в порт включает время ожидания освобождения причала и буксира), в течение рабочей недели (5 дней при 8 часовом рабочем дне). Корабль, ожидающий освобождения причала, не обслуживается буксиром до тех пор, пока не будет предоставлен нужный причал. Корабль второго типа не займет буксир до тех пор, пока ему не будут доступны оба буксира.

Исходные данные по вариантам приведены в табл. 3.

Таблица 3

№	Тип корабля									
	1					2				
	Интервал прибыти я, мин.	Время входа в порт, мин	Кол-во доступны х	Время погрузки- разгрузки	Время выхода из порта.	Интервал прибыти я, мин	Время входа в порт, мин	Кол-во доступны х	Время погрузки- разгрузки	Время выхода из порта.
1	130±30	30±7	6	12±2	20±5	390±60	45±12	3	18±4	35±10
2	100±20	20±10	5	10±3	25±6	300±60	40±10	4	15±5	35±7
3	230±30	40±8	3	15±5	22±4	290±90	35±9	5	20±3	25±12
4	130±30	30±7	6	12±2	20±5	390±60	45±12	3	18±4	35±10
5	155±25	40±9	4	18±5	30±8	400±60	35±15	2	19±6	45±15
6	130±40	55±7	5	19±3	22±5	290±60	45±15	4	15±4	25±11
7	220±20	40±5	4	12±5	15±5	370±50	44±10	3	22±5	35±5

8	170±60	33±7	2	16±2	20±7	380±70	35±12	4	16±8	25±7
9	110±30	20±8	5	15±3	27±5	390±60	25±15	3	17±4	33±9
10	130±20	30±7	6	19±5	28±9	377±50	41±11	2	19±5	39±11
11	150±33	36±8	4	18±3	18±6	290±40	44±17	5	28±9	37±10
12	160±40	38±6	3	22±8	22±5	390±60	35±16	6	22±4	25±8
13	160±50	41±9	2	17±2	29±8	360±50	33±12	3	23±9	34±12
14	135±20	20±5	5	15±4	20±6	490±30	55±19	2	18±3	36±13
15	139±29	34±6	6	16±3	29±5	299±39	45±17	4	15±5	25±10
16	170±30	35±17	4	21±7	19±4	375±25	40±12	3	17±4	37±14
17	190±60	30±5	3	31±12	15±3	380±60	49±18	5	19±6	35±9
18	130±40	35±7	2	12±2	23±5	370±45	44±16	6	15±3	38±11
19	120±20	38±9	7	23±7	26±6	395±35	45±15	2	18±8	33±8
20	145±35	36±8	4	12±2	22±4	376±50	40±10	4	28±9	29±7

Контрольные вопросы

1. Для решения, каких задач предназначен GPSS World?
2. Какие типы переменных используются в языке GPSS World?
3. Что такое динамические объекты системы GPSS World? Приведите параметры динамических объектов.
4. Как задается время в процессе имитационного моделирования?
5. Перечислите основные операторы языка GPSS World.
6. Дайте определение СМО и расшифруйте основные понятия.
7. В чем особенность моделирования одноканальных СМО?
8. Как можно задать операторами GPSS World многоканальную СМО?
9. Перечислите этапы моделирования в системе GPSS World.
10. Как производится ввод и редактирование текста в системе GPSS World?
11. Как можно отобразить результаты моделирования в системе GPSS World?

Лабораторная работа №4.**Тема:** *Определение оптимального времени поступления заявок в заданной конфигурации ВС.***Цель работы:** *Изучить средства GPSS для создания имитационной модели позволяющей определить оптимальное время поступления запросов на обработку для заданной в варианте конфигурации схемы.***Краткие сведения****Задание переменных пользователем**

Переменные представляют собой сложные выражения, которые включают константы, системные числовые атрибуты (СЧА), арифметические и логические операции, библиотечные арифметические функции.

GPSS предоставляет пользователю возможность иметь свои переменные для хранения нужных данных при моделировании как числовых, так и строковых данных. Переменные пользователя создаются с помощью команды **EQU** или присваивающих **PLUS** - процедур. Оператор **EQU** (ЭКВИВАЛЕНТНОСТЬ) предназначен для присвоения числовых значений именам, которые используются в модели. Оператор имеет следующий формат:

Name EQU (ЭКВИВАЛЕНТНОСТЬ) A

Name – имя объекта;

EQU – определяет эквивалентность;

A – выражение.

Когда интерпретатор обрабатывает оператор **EQU**, он вычисляет выражение, заданное операндом **A**, после чего создает или переопределяет имя переменной. Имени присваивается результат вычисленного выражения. Полученное значение заменяет ссылки на это имя в операндах или выражениях, используемых в моделях.

Рассмотрим примеры использования данного операнда:

VER1	EQU	2.65
STROKA	EQU	11
STOLBEZ	EQU	8
NAME	EQU	"NAME"

в приведенных трех примерах переменным пользователя **VER1**, **STROKA**, **STOLBEZ** присваиваются числовые значения, а переменной **NAME** - строка. С помощью переменных пользователя имена объектов GPSS можно заменять номерами. Для этого следует предварительно, как показано в примерах, определить переменные пользователя командой **EQU**.

Также имена, заданные командой **EQU**, но в модели использующиеся как метки объектов, считаются метками.

При проведении эксперимента с имитационной моделью факторы, влияние которых на искомый предполагается исследовать, обязательно должны быть переменными показателями.

Проверка числовых выражений

Выходящие из любого блока транзакты, поступают в следующий блок, при этом в более сложных моделях возникает необходимость направления транзактов к другим блокам в зависимости от некоторых условий. Эту возможность могут обеспечить операторы изменения маршрутов транзактов, одним из которых является оператор **TEST** (ПРОВЕРИТЬ).

Данный оператор описывает условие, которое проверяет при входе в него транзакта, и определяет номер следующего блока для вошедшего транзакта в зависимости от того, выполняется требуемое условие или нет. Блок **TEST** может функционировать в двух режимах:

1. отказа во входе;
2. разрешении во входе и альтернативном выходе.

Формат записи имеет следующий вид:

Name TEST (ПРОВЕРИТЬ) X A,B[,C]

Name – имя метки;

X – вспомогательный операнд, который содержит условие проверки соотношения между СЧА и может принимать следующие значения: **L** (меньше); **LE** (меньше или равно); **E** (равно); **NE** (не равно); **GE** (больше или равно); **G** (больше).

Операнд **A** и **B** – содержат **A** – первый и соответственно **B** – второй из сравниваемых СЧА. Если проверяемое условие **AxB** выполняется, то блок **TEST** пропускает транзакт в следующий блок. Если же это условие не выполняется, то транзакт переходит к блоку, указанному в поле **C**. а если оно пусто то задерживается перед блоком **TEST**.

C – имя блока, в который переходит транзакт при условии, что ответ на вопрос, подразумеваемый оператором отношения, отрицательный.

Рассмотрим примеры использования данного операнда:

TEST LE Q1,Q2

проверяющий транзакт будет задержан в предыдущем блоке до тех пор, пока длина первой очереди не станет меньше или равна длине второй очереди;

TEST L Q\$LINE,5,OUT

направляет транзакты в блок с именем **OUT**, если текущая длина очереди **LINE** больше или равна **5**.

Изменение значений параметров

В процессе моделирования интерпретатор автоматически регистрирует и корректирует информацию, касающуюся различных элементов, используемых в модели. Большая часть информации доступна только интерпретатору и используется для сбора статистической информации о работе модели. Однако к некоторым атрибутам объектов может обращаться и программист, управляя процессом моделирования в зависимости от их значений.

Рассмотрим несколько примеров зависимости функционирования элементов модели от системных атрибутов, т.е. СЧА.

1. Интенсивность работы некоторого устройства зависит от длины очереди. Для определения времени обслуживания при каждом поступлении транзакта на обслуживание необходимо знать значение такого системного атрибута, как длина очереди.
2. Интенсивность обслуживания некоторого устройства зависит от общей продолжительности функционирования. Время обслуживания – функция, которая зависит от времени прошедшего от начала работы.
3. Предположим имеются два устройства и диспетчер, который распределяет работы между ними таким образом, чтобы загрузка устройств была равномерной. Для этого в пункте диспетчеризации необходимо иметь информацию о коэффициентах загрузки устройств и возможность выбора пути перемещения транзакта в зависимости от этих двух величин.

Условно атрибуты можно поделить на две категории:

- 1) атрибуты системы;
- 2) атрибуты транзактов.

Атрибуты системы – это параметры, которые описывают состояние объектов модели. Такие количественные показатели, как «текущая длина очереди», «коэффициент загрузки устройства» являются типичными системными атрибутами.

Транзакты также могут иметь некоторые числовые характеристики, например «уровень приоритета», кроме того, транзакт снабжается некоторым числом параметров.

Параметры транзактов – это свойства транзакта, определяемые пользователем. Множество параметров транзакта – набор стандартных числовых атрибутов, которые принадлежат транзакту. Параметры транзакта являются локальными переменными, которые доступны только данному транзакту. В процессе перемещения транзакта по модели, его параметры могут задаваться и модифицироваться в соответствии с логикой работы модели.

Блок **ASSIGN (НАЗНАЧИТЬ)** является основным средством для задания или изменения значения параметров транзактов. Формат записи данного оператора имеет вид:

Name ASSIGN (НАЗНАЧИТЬ) A[+,-],B[,C]

Name – имя метки;

A[+,-] – данный операнд задает номер или имя модифицируемого или задаваемого параметра, которому присваивается значение. Блок может быть, использован как в режиме замеще-

ния значения параметров (начальное значение всех параметров транзактов равно 0), так и в режиме увеличения и уменьшения. Режимы увеличения и уменьшения определяются введением знаков «плюс (+)» и «минус (-)» перед запятой, которая разделяет операнды **A** и **B**. В режиме увеличения предшествующее значение параметра увеличивается на значение, стоящее в операнде **B**. В режиме уменьшения оно уменьшается на величину, стоящую в операнде **B**.

B – операнд определяет значение, которое следует добавить, вычесть или которым следует заменить значение в параметре, заданном операндом **A**.

[,C] – задает номер модификатора-функции, при использовании данного операнда значение операнда **B** умножается на значение функции, указанной в операнде **C**.

Пример использования данного оператора:

ASSIGN COUNT +,1

данный блок добавляет **1** к текущему значению параметра с именем **COUNT**.

При каждом входе транзакта в модель интерпретатор фиксирует для него текущее значение времени. Это значение времени называется отметкой времени. Она может быть интерпретирована как время «рождения» транзакта или время входа транзакта в модель. В явном виде отметка времени недоступна, однако существует СЧА, который тесно связан со значением времени входа транзакта в модель. Его имя **M1**, а значение определяется так:

Текущее значение	Значение времени
M1 = таймера абсолютного -	входа транзакта
времени	в модель

Значение **M1** для каждого транзакта изменяется в процессе моделирования. Сразу после входа транзакта в модель **M1=0**, через 10 единиц модельного времени **M1=10** и т.д.

Для записи текущего модельного времени в заданный параметр транзакта или в активный транзакт служит оператор **MARK (ОТМЕТИТЬ)**, такую запись называют отметкой транзакта, а данный оператор имеет следующий формат записи:

Name MARK (ОТМЕТИТЬ) [A]

Name – имя метки;

A – номер параметра транзакта, в который записывается значение абсолютного модельного времени при входе этого транзакта в блок **MARK**, при этом данный операнд может быть именем, положительным целым числом, выражением в скобках или СЧА. Если операнд **A** отсутствует, то отметка времени заменяется текущим значением абсолютного времени.

Пример использования данного оператора:

MARK
MARK Vxod

в первом случае операнд **A** не используется и по умолчанию вошедшему в этот блок транзакту устанавливается время входа в систему, равное абсолютному модельному времени; во втором случае операнд **A** используется, поэтому значение абсолютного модельного времени (СЧА) заносится в параметр с именем **Vxod** вошедшего транзакта, если этого параметра нет в модели, то он создается.

Задание функции

В общем случае источником «случайности» в моделях являются одна или несколько функций, при обращении к которым можно получить случайные значения из выборки равномерно распределенных в интервале чисел.

Для применения в GPSS функция должна быть задана табличными значениями. Каждая функция определяется перед началом моделирования с помощью оператора **определения FUNCTION (ФУНКЦИЯ)**, имеющего следующий формат:

Name FUNCTION (ФУНКЦИЯ) A,B

Name – имя функции, используемое для ссылок на нее;

A – стандартный числовой атрибут, являющийся аргументом функции, может быть именем, положительным целым числом, строкой, выражением в скобках;

B – состоит из одной буквы определяющей тип функции и целого положительного числа, задающего количество пар возможных значений аргумента функции (число точек в таблице).

Существует пять типов функций:

C – непрерывная числовая; **D** – дискретная числовая; **E** – дискретная атрибутивная; **L** – списковая числовая; **M** – списковая атрибутивная.

Помимо этого существует еще один тип функции, которая применяется в блоках **GENERATE** и **ADVANCE** в качестве операнда **B**. Она называется модификатор функции. Значение этой функции вычисляется с двойной точностью и умножается на значение операнда **A**. Полученный результат используется как требуемая временная задержка в данном блоке.

Пример: Пусть среднее значение интервалов поступления **T** в пуассоновском потоке требований равно 2 часам, единица времени в модели равна 1 мин, тогда поступление заявок моделируется блоком:

GENERATE 120, FN\$XPDIS.

Если необходимо моделировать задержку, распределенную по экспоненциальному закону со средним значением времени 345 мин, то для этого используется блок:

ADVANCE 345, FN\$XPDIS.

Значения, заданные командой **FUNCTION**, хранятся в форме чисел с плавающей запятой двойной точности.

При описании любой из функций с помощью языка GPSS происходит интерполяция. Для дискретных функций – это кусочно-постоянная интерполяция, для непрерывных – линейная интерполяция. Координаты функции, задаваемые парами, являются узлами интерполяции.

Рассмотрим непрерывную числовую функцию, таблица которой содержит **3** точки, значит поле **B** должно иметь значение **C3** (функция непрерывна и для ее описания используется 3 пары аргументов-функций). При использовании непрерывной функции для генерирования случайных чисел ее аргументом может быть один из генераторов случайных чисел **RNj** (всего в данном языке существует 8 датчиков (генераторов) равномерно распределенных случайных чисел, которые обозначаются RN1-RN8. Эти датчики выдают равновероятные целочисленные значения из диапазона 0-999. Если датчик используется в качестве аргумента функции, то он выдает вещественные числа от 0 до 0,999999). Так, оператор для определения функции показательного распределения может иметь следующий вид: **EXP FUNCTION RN1,C3**

Таблица с координатами точек функции располагается в строках, следующих непосредственно за оператором **FUNCTION**. Эти строки не должны иметь поля нумерации. Каждая точка таблицы задается парой **Xi** (значение аргумента) и **Yi** (значение функции), отделяемых друг от друга запятой. Пары координат отделяются друг от друга символом «/» и располагаются на произвольном количестве строк. Последовательность значений аргумента **Xi** должна быть строго возрастающей. Поскольку язык оперирует с целочисленными величинами, то любое значение функции округляется до целого числа.

Пример дискретной функции содержащей три пары аргументов - функции:

SERV FUNCTION RN1,D5

.15,2/.35,5/.6,8/.82,9/1,12

Пример функции **XPDIS** с экспоненциальным распределением и интенсивностью $\lambda=1$:

XPDIS FUNCTION RN2,C24

0,0/.1,.104/.2,.222/.3,.355/.4,.509/.5,.69/.6,.915/.7,1.2/.75,1.38/.8,1.6/.84,1.83/.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

Задание к лабораторной работе №4

Постановка задачи: Базовая схема состоит из **M** терминалов пользователей, канала прямой передачи данных и канала обратной передачи данных. По сигналу пользователя данные могут считываться из некоторой базы данных и отсылаться на хранение в эту базу данных. Рассчитать производительность ЭВМ, длины очередей к каналам и ЭВМ и определить оптимальное время поступления запросов на обработку для заданных в варианте конфигурации схемы, характеристик обслуживания ЭВМ и времени работы любого из каналов. Обязательны таблицы времени обработки транзактов на ЭВМ.

Оптимальное время поступления заявок должно обеспечивать наивысшую загрузку ЭВМ и минимальную очередь на обработку заданий, в пределах 2 – 4 заявок.

Т – терминал пользователя;
КБД – канал базы данных;
ЭВМ – процессор и память;
БД – база данных;
К – канал прямой и обратной передачи данных;
КП – канал прямой передачи сообщений;
КО – канал обратной передачи сообщений.

μ - интенсивность обслуживания сообщений с пуассоновским законом распределения времени обработки заявок.

P1, P2 – интервал обработки заявок с равномерным законом обслуживания в ЭВМ;

K1, K2 – интервал времени работы любого из каналов.

Порядок выполнения лабораторной работы

1. Написать программу модели.
2. Промоделировать систему с разными значениями $t_{вх}$.
3. Построить зависимость $\rho=f(t_{вх})$ и $Q=f(t_{вх})$.
4. Выполнить ручной расчёт загрузки устройств, длин очередей и производительности системы.
5. Построить гистограммы времён обработки транзактов для каждого типа заявок.
6. Проанализировать полученные результаты моделирования и определить оптимальное время поступления заявок для заданной конфигурации схемы.

Пример выполненной лабораторной работы

Вариант задания: Задана схема из двух терминалов, канала передачи сообщений на ЭВМ и от ЭВМ, ЭВМ, канала передачи сообщений к БД и БД. Время передачи сообщений на ЭВМ равномерно распределено на отрезке [10,15].

Построение имитационной модели процесса

Листинг программы:

***GPSS TERMINAL_1_re.GPS**

* Моделирование работы вычислительной сети *

; Начальные условия

PROBA EQU 480 ; блок для установления времени t вх
 T_TAB TABLE MP4,100,100,5 ; формат таблицы для моделирования

; Сегмент имитации моделирования транзактов

GENERATE PROBA ; время пребывания транзакта в ЭВМ
 ASSIGN 1,1 ; определение номера терминала
 TRANSFER ,METKA ; задается позиция блока в которую переходит транзакт
 GENERATE PROBA ; время пребывания транзакта в ЭВМ
 ASSIGN 1,2 ; определение номера терминала

; Сегмент имитации моделирования транзактов к КАНАЛУ

METKA QUEUE QKAN ; создается очередь к КАНАЛУ
 SEIZE KAN ; определяется занятость КАНАЛА
 DEPART QKAN ; выход транзакта из очереди к КАНАЛУ
 ADVANCE 65,25 ; обслуживание в КАНАЛЕ
 RELEASE KAN ; освобождение КАНАЛА

; Сегмент имитации моделирования транзактов к ЭВМ

MARK 4 ; отметка времени входа транзакта в очередь к ЭВМ
 QUEUE QEBM ; создается очередь к ЭВМ
 SEIZE EBM ; определяется занятость ЭВМ
 DEPART QEBM ; выход транзакта из очереди к ЭВМ
 ADVANCE 125,25 ; обслуживание в ЭВМ
 RELEASE EBM ; освобождение ЭВМ
 TABULATE T_TAB ; первая точка табуляции времени
 ; пребывания транзакта в очереди к ЭВМ

; Сегмент имитации моделирования транзактов в КБД

QUEUE QKBD ; создается очередь к КБД

	SEIZE	KBD	; определяется занятость КБД
	DEPART	QKBD	; выход транзакта из очереди к КБД
	ADVANCE	65,25	; обслуживание в КБД
	RELEASE	KBD	; освобождение КБД
; Сегмент имитации моделирования транзактов к БД			
	QUEUE	BD	; создается очередь к БД
	SEIZE	BD	; определяется занятость БД
	DEPART	BD	; выход транзакта из очереди к БД
	ADVANCE	0	; обслуживание в БД
	RELEASE	BD	; освобождение БД
	QUEUE	QBD	; создается очередь к БД
	SEIZE	KBD	; определяется занятость КБД
	DEPART	QBD	; выход транзакта из очереди к БД
	ADVANCE	65,35	; обслуживание транзактов
	RELEASE	KBD	; освобождение КБД
	MARK	4	; отметка вхождения в очередь транзакт
			; при движении от базы данных
	QUEUE	QEBM	; создается очередь к ЭВМ
	SEIZE	EBM	; определяется занятость ЭВМ
	DEPART	QEBM	; выход транзактов из очереди к ЭВМ
	ADVANCE	125,25	; обслуживание в ЭВМ
	RELEASE	EBM	; освобождение ЭВМ
	QUEUE	QKAN	; стать в очередь к КАНАЛУ
	SEIZE	KAN	; занять КАНАЛ для передачи транзактов
	DEPART	QKAN	; выход транзакта из очереди
	ADVANCE	65,25	; обслуживание транзактов
	RELEASE	KAN	; освобождение КАНАЛА
	TABULATE	T_TAB	; табуляция времени пребывания заносится в таблицу
	TEST E	P1,1,TERM2	; транзакты направляются в ЭВМ, если переменные P1=1
	SAVEVALUE	PER1+,1	; позволяет определить количество обработанных транзактов с каждого из терминалов пользователей
	TRANSFER	,METKA1	; транзакт переходит в позицию блока с именем METKA1
TERM2	SAVEVALUE	PER2+,1	; определяется количество обработанных транзактов
METKA1	TERMINATE		;
; Сегмент задания времени моделирования			
	GENERATE	10000	; выбранное время моделирования
	TERMINATE	1	; выход из системы

Запустите программу на выполнение и проанализируйте результаты моделирования.

Результаты моделирования

Отчет по результатам моделирования:

GPSS World Simulation Report - Untitled Model 1.1.1

START TIME	END TIME	BLOCKS	FACILITIES	STORAGES
0.000	10000.000	51	4	0

NAME	VALUE
BD	10008.000
EBM	10005.000
KAN	10003.000
KBD	10007.000
METKA	6.000
METKA1	49.000
PER1	10010.000
PER2	10011.000
PROBA	480.000
QBD	10009.000
QEBM	10004.000
QKAN	10002.000
QKBD	10006.000
TERM2	48.000
T_TAB	10001.000

LABEL	LOC	BLOCK	TYPE	ENTRY COUNT	CURRENT COUNT	RETRY
-------	-----	-------	------	-------------	---------------	-------

METKA	1	GENERATE	20	0	0				
	2	ASSIGN	20	0	0				
	3	TRANSFER	20	0	0				
	4	GENERATE	20	0	0				
	5	ASSIGN	20	0	0				
	6	QUEUE	40	0	0				
	7	SEIZE	40	0	0				
	8	DEPART	40	0	0				
	9	ADVANCE	40	0	0				
	10	RELEASE	40	0	0				
	11	MARK	40	0	0				
	12	QUEUE	40	1	0				
	13	SEIZE	39	0	0				
	14	DEPART	39	0	0				
	15	ADVANCE	39	0	0				
	16	RELEASE	39	0	0				
	17	TABULATE	39	0	0				
	18	QUEUE	39	0	0				
	19	SEIZE	39	0	0				
	20	DEPART	39	0	0				
	21	ADVANCE	39	1	0				
	22	RELEASE	38	0	0				
	23	QUEUE	38	0	0				
	24	SEIZE	38	0	0				
	25	DEPART	38	0	0				
	26	ADVANCE	38	0	0				
	27	RELEASE	38	0	0				
	28	QUEUE	38	0	0				
	29	SEIZE	38	0	0				
	30	DEPART	38	0	0				
	31	ADVANCE	38	0	0				
	32	RELEASE	38	0	0				
	33	MARK	38	0	0				
	34	QUEUE	38	1	0				
	35	SEIZE	37	0	0				
	36	DEPART	37	0	0				
	37	ADVANCE	37	1	0				
	38	RELEASE	36	0	0				
	39	QUEUE	36	0	0				
	40	SEIZE	36	0	0				
	41	DEPART	36	0	0				
	42	ADVANCE	36	0	0				
	43	RELEASE	36	0	0				
	44	TABULATE	36	0	0				
	45	TEST	36	0	0				
	46	SAVEVALUE	18	0	0				
	47	TRANSFER	18	0	0				
TERM2	48	SAVEVALUE	18	0	0				
METKA1	49	TERMINATE	36	0	0				
	50	GENERATE	1	0	0				
	51	TERMINATE	1	0	0				
FACILITY	ENTRIES	UTIL.	AVE. TIME	AVAIL.	OWNER	PEND	INTER	RETRY	DELAY
KAN	76	0.486	63.975	1	0	0	0	0	0
EBM	76	0.936	123.188	1	38	0	0	0	2
KBD	77	0.478	62.085	1	40	0	0	0	0
BD	38	0.000	0.000	1	0	0	0	0	0
QUEUE	MAX	CONT.	ENTRY	ENTRY(0)	AVE.CONT.	AVE.TIME	AVE.(-0)	RETRY	
QKAN	2	0	76	39	0.212	27.920	57.350	0	
QEBM	4	2	78	6	1.226	157.238	170.341	0	
QKBD	1	0	39	31	0.017	4.351	21.213	0	
BD	1	0	38	38	0.000	0.000	0.000	0	
QBD	1	0	38	38	0.000	0.000	0.000	0	
TABLE	MEAN	STD.DEV.	RANGE		RETRY FREQUENCY		CUM.%		
T_TAB	314.767	113.433			0				

100.000	-	200.000	10	13.33
200.000	-	300.000	24	45.33
300.000	-	400.000	24	77.33
400.000	-	—	17	100.00

SAVEVALUE	RETRY	VALUE
PER1	0	18.000
PER2	0	18.000

FEC	XN	PRI	BDT	ASSEM	CURRENT	NEXT	PARAMETER	VALUE
40	0		10019.946	40	21	22	1	1.000
							4	9645.135
38	0		10079.160	38	37	38	1	1.000
							4	9665.276
42	0		10080.000	42	0	1		
43	0		10080.000	43	0	4		
44	0		20000.000	44	0	50		

Обработка результатов моделирования

Гистограмма результатов моделирования:

Полученные в отчете данные отобразим на гистограмме процесса моделирования, которая позволит судить о времени обработки транзактов для каждого типа заявок (рис. 1).

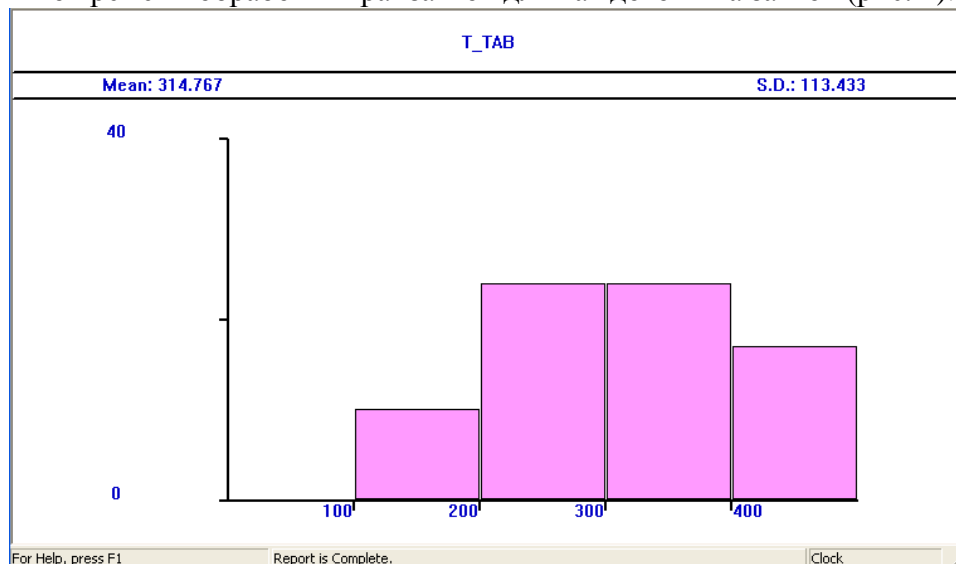


Рис. 1. Окно гистограммы имитационной модели

Вывод по результатам моделирования

Выводы: На основании проведенного моделирования заданной конфигурации ВС и обработки результатов имитации видно, что оптимальным временем для поступления заявок является $t_{вх}$, равное 480 единицам времени. При этом среднем времени поступления заявок длина очереди равна 2 единицам, а загрузка ЭВМ – порядка 0,93.

Табуляция времён обработки заявок на ЭВМ показала, что среднее время обработки при выбранном времени генерации заявок равно 314.767, из которого только 123.188 единицы времени – собственно обработка заявок, а остальное время расходуется на пребывание в очереди.

Замечания по структуре: быстродействие ЭВМ должно быть выше, чтобы в целом производительность системы была выше.

В отчете по лабораторной работе необходимо привести:

- постановку задачи;
- листинг программы;
- результаты моделирования при разных значениях $t_{вх}$ (можно рассмотреть две величины);
- вывод гистограмм для каждого случая обработки заявок;
- построенные графики зависимостей $\rho=f(t_{вх})$ и $Q=f(t_{вх})$;

- выполненный ручной расчёт загрузки устройств, длин очередей и производительности системы;
- объяснение (выводы) результатов решения задачи: проанализировав полученные результаты моделирования и определив оптимальное время поступления заявок для заданной конфигурации схемы.

Варианты индивидуальных заданий к лабораторной работе №4.

№	μ	P1	P2	K1	K2
1	0,1			5	8
2		6	18	4	9
3	0,01			6	8
4		20	40	10	12
5	0,02			5	8
6		4	8	3	6
7	0,04			10	15
8		10	15	5	9
9	0,01			4	8
10	0,05			6	8
11		10	15	4	9
12	0,08			5	12
13	0,04			8	11
14	0,01			12	15
15	0,05			3	5
16		8	10	4	7
17	0,05			3	5
18		8	10	2	6
19	0,02			7	10
20		15	25	6	12
21		10	30	13	15
22		20	40	4	7
23	0,02			5	12
24	0,04			3	6
25	0,01			2	7
26		10	30	4	8
27	0,04			5	9
28		40	60	9	14
29	0,2			8	12
30		30	40	6	10

Примечания:

1. Если в варианте задана величина μ , то это означает, что время обработки описывается экспоненциальной функцией распределения.
2. Время моделирование подбирается в процессе моделирования, исходя из решения варианта задания.

Теоретические расчеты заданных характеристик системы:

Интенсивность поступления транзактов в КАНАЛ: $\lambda_{вх.КАН} = 1 / T_z$;

Интенсивность обработки транзактов в КАНАЛЕ: $\mu_{КАН} = 1 / T_{обр}$;

Коэффициент загрузки КАНАЛА: $\rho_{КАН} = \lambda_{вх.КАН} / \mu_{КАН} > 1$, загрузка меньше 1 говорит о том, что к КАНАЛУ не будет очереди;

Длина очереди к КАНАЛУ обслуживания: $L_{средняя КАН} = (\lambda_{вх} - \lambda_{вых}) * T_{мод}$.

Лабораторная работа №5.

Тема: Моделирование работы производственных систем.

Цель работы: научиться использовать GPSS язык для исследования процедур имитационного моделирования сложных технических объектов, представленных как систем массового обслуживания (СМО).

Краткие сведения:

Имитационная модель СМО представляет собой алгоритм, отражающий поведение СМО, т.е. отражающий изменения состояния СМО во времени при заданных потоках заявок, поступающих на входы системы. Параметры входных потоков заявок - внешние параметры СМО. Выходными параметрами являются величины, характеризующие свойства системы - качество ее функционирования. Примеры выходных параметров: **производительность** СМО - среднее число заявок, обслуживаемых в единицу времени; **коэффициенты загрузки оборудования** - отношение времен обслуживания к общему времени в каждом ОА (обслуживающем аппарате – транзакте); **среднее время обслуживания одной заявки**. Основное свойство ОА, учитываемое в модели СМО, - это затраты времени на обслуживание, поэтому внутренними параметрами в модели СМО являются величины, характеризующие это свойство ОА. Обычно время обслуживания рассматривается как случайная величина и в качестве внутренних параметров фигурируют параметры законов распределения этой величины.

Имитационное моделирование позволяет исследовать СМО при различных типах входных потоков и интенсивностях поступления заявок на входы, при вариациях параметров ОА, при различных дисциплинах обслуживания заявок. **Дисциплина обслуживания** - правило, по которому заявки поступают из очередей на обслуживание. Величина, характеризующее право на первоочередное обслуживание, называется **приоритетом**. В моделях СМО заявки, приходящие на вход занятого ОА, образуют очереди, отдельные для заявок каждого приоритета. При освобождении ОА на обслуживание принимается заявка из непустой очереди с наиболее высоким приоритетом.

Управление потоком транзактов

Для управления потоком транзактов с помощью логических операторов используется блок **GATE (ВПУСТИТЬ)** – который позволяет задерживать или изменять маршруты транзактов в зависимости от состояния аппаратных объектов модели, данный оператор имеет следующий формат:

Name GATE (ВПУСТИТЬ) X A,B

Name – имя метки;

X – вспомогательный операнд, который содержит код состояния проверяемого аппаратного объекта. Операнд **X** может принимать следующие значения: 1) Логические операторы, связанные с устройствами: **U** (устройство заданное в операнде **A** занято), **NU** (устройство свободно), **I** (устройство заданное в операнде **A** обслуживает прерыванием), **NI** (устройство не прервано), **FV** (устройство заданное в операнде **A** доступно), **FNV** (не доступно); 2) Логические операторы связанные с МКУ: **SE** (МКУ заданное в операнде **A** пустое), **SNE** (не пустое), **SF** (МКУ заданное в операнде **A** заполнено), **SNF** (не заполнено), **SV** (МКУ заданное в операнде **A** доступно), **SNV** (не доступно); 3) Логические операторы связанные с ключами: **LS** (логический ключ заданный в операнде **A** включен), **LR** (выключен).

A – в данном поле указывается имя или номер объекта для которого производится проверка;

B – данный операнд содержит номер следующего блока для входящего транзакта, если логический оператор имеет значение «ложь».

Рассмотрим пример использования данного блока:

QUEUE	LINE
GATE SV	LINE1
DEPART	LINE

в данном случае транзакт помещается в список задержки, если МКУ LINE1 не доступно в тот момент, когда транзакт пытается войти в блок GATE. Когда МКУ становится доступным, все транзакты выводятся из списка и делают попытку войти в МКУ.

Прерывание работы устройства

Для моделирования захвата (прерывания) устройства используются блоки **PREEMPT (ЗАХВАТИТЬ)** и **RETURN (ВЕРНУТЬ)**. С помощью этих блоков удобно описывать такие ситуации, когда обслуживающий прибор временно прекращает обработку поступившего в него транзакта и переходит в режим обслуживания «срочного» (приоритетного) транзакта и, обслужив его, возвращается к обслуживанию первого менее приоритетного транзакта. Блок **PREEMPT** позволяет транзакту в зависимости от условий, заданных в операндах блока, занять устройство, он имеет следующий формат:

Name PREEMPT (ЗАХВАТИТЬ) A,B,C,D,E

Name – метка используемая для ссылки;

A – имя или номер устройства подлежащего захвату;

B – в данном поле кодируется условие захвата;

C – указывается имя блока, в который будет направлен прерванный транзакт;

D – указывается номер или имя параметра прерванного транзакта, в который записывается время, оставшееся этому транзакту до завершения обслуживания на устройстве;

E – при отсутствии данного операнда, прерванный транзакт сохраняет право на автоматическое восстановление на устройстве по окончании захвата, если же в этом поле указано RE, то транзакт теряет такое право.

Блок **RETURN (ВЕРНУТЬ)** используется в паре с блоком **PREEMPT**, если транзакт захватил устройство посредством блока **PREEMPT**, то освободить его он может только в блоке **RETURN**. Данный блок имеет следующий формат:

Name RETURN (ВЕРНУТЬ) A

Name – метка используемая для ссылки;

A – имя или номер устройства подлежащего освобождению от захвата.

Задание к лабораторной работе №5

Тема. Оценка эффективности работы системы с учетом отказов

Постановка задачи: Необходимо рассмотреть функционирование системы имеющей три вида объектов (например, приборы). Объекты поступают в систему на обслуживание с интенсивностью λ . Поток поступления объектов на обслуживание – пуассоновский: $\lambda=0.1$ 1/ед. времени. Объекты обслуживаются в канале обслуживания с интенсивностью μ . Распределение времени обслуживания объектов – экспоненциальное $\mu=0.05$ 1/ед. времени. Одновременно все объекты находятся под действием пуассоновских потоков отказов с соответствующими интенсивностями: $\lambda_1=0.01$ 1/ед. времени, $\lambda_2=0.008$ 1/ед. времени, $\lambda_3=0.0125$ 1/ед. времени. Отказавшие объекты немедленно начинают ремонтироваться (восстанавливаться). При этом для восстановления отказавших объектов имеются два пункта восстановления (ПВ). Распределение времени восстановления объектов на каждом ПВ имеет экспоненциальный характер. Для первого ПВ интенсивность восстановления $\mu_1=0.033$ 1/ед. времени. Для второго ПВ интенсивность восстановления $\mu_2=0.025$ 1/ед. времени. Требуется оценить эффективность работы такой системы, т.е. определить:

- коэффициент использования всех объектов;
- среднее время восстановления объектов;
- коэффициент использования пунктов восстановления и др.

Выявление особенностей системы и теоретические расчеты заданных характеристик

Рассматриваемые в системе потоки отказов характеризуются следующими особенностями:

1) отказы объекта происходят по одному, т.е. вероятность отказа двух и более элементов объекта в один момент времени очень мала, и ею можно пренебречь (поток требований ординарный);

- 2) вероятность наступления последующих отказов объекта в любой момент времени не зависит от предыдущих отказов – поток отказов без последствия;
- 3) поток отказов стационарный.

Потоки отказов, в которых имеются все вышеперечисленные особенности, называются простейшими.

Интенсивность отказа объекта определяется как величина, обратная времени наступления очередного отказа объекта (t_o): $\lambda=1/t_p$.

Рассчитаем время наступления очередного отказа для трех объектов: - для первого прибора $t_{o1}=1/\lambda_1=1/0.01=100$; - для второго прибора $t_{o2}=1/\lambda_2=1/0.008=125$; - для третьего прибора $t_{o3}=1/\lambda_3=1/0.0125=80$.

Среднее время между входами двух идущих один за другим объектов в систему на обслуживание определяется как: $t_{вх}=1/\lambda=1/0.1=10$.

Интенсивность восстановления объекта определяется как величина, обратная времени восстановления (ремонта) очередного объекта (t_v): $\mu=1/t_v$.

Рассчитаем время восстановления объектов для двух пунктов восстановления: - для первого пункта $t_{в1}=1/\mu_1=1/0.033=30$; - для второго пункта $t_{в2}=1/\mu_2=1/0.025=40$.

Среднее время обслуживания объекта определяется, исходя из интенсивности обслуживания объектов как: $t_{обс}=1/\mu=1/0.05=20$.

Построение имитационной модели процесса:

Листинг программы:

* GPSSW Program File EX2.

* Моделирование работы системы с учетом отказов *

; Первый сегмент модели генерирует поток отказов различных объектов

```
GENERATE      (Exponential(1,0,100)),,1
PREEMPT      Object1,PR,MET4,,RE
ASSIGN       2,Object1
TRANSFER     ,COM2
GENERATE      (Exponential(1,0,125)),,1
PREEMPT      Object2,PR,MET4,,RE
ASSIGN       2,Object2
TRANSFER     ,COM2
GENERATE      (Exponential(1,0,80)),,1
PREEMPT      Object3,PR,MET4,,RE
ASSIGN       2,Object3
TRANSFER     ,COM2
```

; Второй сегмент модели обеспечивает обслуживание не отказавших объектов

```
MET1  GENERATE      (Exponential(1,0,10))
      GATE NU      Object1,MET2
      SEIZE        Object1
      ASSIGN       1,Object1
      TRANSFER     ,COM1
MET2  GATE NU      Object2,MET3
      SEIZE        Object2
      ASSIGN       1,Object2
      TRANSFER     ,COM1
MET3  GATE NU      Object3,MET4
      SEIZE        Object3
      ASSIGN       1,Object3
COM1  ADVANCE      (Exponential (1,0,20))
      RELEASE     P1
      TERMINATE    1
MET4  SAVEVALUE    2+,1
      TERMINATE    1
```

; Третий сегмент модели проводит восстановление отказавших объектов

```
COM2  TRANSFER     BOTH,REM1,REM2
REM1  QUEUE        Punkt_1
      SEIZE        Punkt_1
      DEPART       Punkt_1
```


	ADVANCE	(Exponential (1,0,30))
	RELEASE	Punkt_1
	TRANSFER	,COM3
REM2	QUEUE	Punkt_2
	SEIZE	Punkt_2
	DEPART	Punkt_2
	ADVANCE	(Exponential (1,0,40))
	RELEASE	Punkt_2
COM3	RETURN	P2
	SAVEVALUE	1+,M1
	TERMINATE	
	START	5000

Результаты моделирования:

Отчет, по результатам моделирования реализованный в среде GPSS World представить в отчете по лабораторной работе №5.

Построение гистограмм по результатам моделирования:

Гистограммы по результатам моделирования представить в отчете по лабораторной работе №5.

Выводы по результатам моделирования:

Вывод по результатам моделирования представить в отчете по лабораторной работе №5.

Примечание: проанализируйте полученные результаты моделирования и определите оптимальное время поступления заявок для заданной конфигурации схемы. Проведите эксперименты и приведите предложения по улучшению выходных параметров моделируемой системы.

Индивидуальное задание к лабораторной работе №5

Тема: Моделирование работы автозаправочной станции.

Постановка задачи: необходимо промоделировать работу автозаправочной станции (АЗС), которая имеет две заправочные колонки. Известны следующие параметры работы АЗС:

- поток автомобилей, поступающих на заправку, подчиняется экспоненциальному распределению вероятностей с параметрами $\lambda=0$ и $\beta=6.5$;
- время заправки на первой колонке составляет $a \pm b$ мин, а на второй - $c \pm d$ мин (таблица 1);
- автомобиль подъезжает к колонке, которая не занята обслуживанием другого автомобиля.

Требуется промоделировать работу АЗС в течение заданного времени моделирования и определить параметры функционирования АЗС:

- коэффициент загрузки каждой колонки;
- среднее время обслуживания в каждой колонке;
- максимальное, среднее и текущее число автомобилей в очереди к каждой колонке;
- среднее время нахождения автомобиля в каждой очереди и др.

Таблица 1

№	a	b	c	d	Tm (час)
1	10	2,5	10	5	8
2	15	6	18	4	9
3	12	3	11	6	5
4	20	5	40	10	12
5	22	7	10	5	8
6	11	4	8	3	6
7	14	3	30	10	15
8	22	10	15	5	9
9	13	5	11	4	18
10	15	3	16	6	23
11	20	10	15	4	36
12	18	4	16	5	12

13	14	3	18	8	11
14	10	2	12	2	15
15	15	5	13	3	5
16	18	8	10	4	7
17	15	2	15	3	4
18	17	8	10	2	6
19	12	4	13	5	24
20	22	5	25	6	12
21	24	10	30	13	15
22	18	2	14	4	17
23	12	3	13	5	12
24	14	4	13	3	16
25	17	5	14	2	7
26	11	1	13	4	21
27	14	7	15	5	19
28	18	4	16	9	14
29	12	5	13	8	12
30	19	3	14	6	10

Примечание:

Входной поток автомобилей, подъезжающих к заправке с интервалом времени, соответствует экспоненциальному закону. Интервал времени рассчитывается по следующей формуле:

$F(x) = (1/\beta) e^{-(x-\lambda)/\beta}$, $\beta > 0$ в противном случае его значение равно 0. В качестве единицы измерения времени принять 1 мин.

В отчете по лабораторной работе необходимо привести:

- Постановку задачи;
- Листинг программы согласно варианту задания;
- Результаты моделирования;
- Вывод гистограмм;
- Объяснение результатов решения задачи (для ПРИМЕРА 1 проанализируйте полученные результаты моделирования и определите оптимальное время поступления заявок для заданной конфигурации схемы. Приведите предложения по улучшению выходных параметров участка цепи).