

МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ
УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ
«БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ТРАНСПОРТА»

Кафедра «Прикладная математика»

Д. Н. ШЕВЧЕНКО, И. Н. КРАВЧЕНЯ

ИМИТАЦИОННОЕ МОДЕЛИРОВАНИЕ НА GPSS

**Учебно-методическое пособие для студентов
технических специальностей**

Гомель 2007

УДК 519.8 (075.8)
ББК 22.1
ШЗ7

Рецензент – зав. кафедрой «Математические проблемы управления» д-р техн. наук, профессор И. В. Максимей (УО «ГГУ им. Ф. Скорины»).

Шевченко, Д. Н.

ШЗ7 Имитационное моделирование на GPSS : учеб.-метод. пособие для студентов технических специальностей / Д. Н. Шевченко, И. Н. Кравченя ; М-во образования Респ. Беларусь, Белорус. гос. ун-т трансп. – Гомель : БелГУТ, 2007. – 97 с.
ISBN 978-985-468-301-0

Содержит начальные разделы теории имитационного моделирования, предусмотренные образовательным стандартом по специальности РД РБ о2100.5.009-98. Рассмотрены вопросы построения и исследования имитационных моделей с помощью системы моделирования общего назначения GPSS World. Приведены примеры и задания для самостоятельной работы.

Предназначено для студентов технических специальностей. Может быть использовано при выполнении курсовых и дипломных проектов студентами технических специальностей, аспирантами и научными работниками, занимающимися анализом технических систем, оптимизацией технологических процессов и процессов перевозок.

УДК 519.8 (075.8)
ББК 22.1

ISBN 978-985-468-301-0

© Шевченко Д. Н., Кравченя И. Н., 2007
© Оформление. УО «БелГУТ», 2007

ОГЛАВЛЕНИЕ

Введение. Место имитационного моделирования в исследовании систем	5
1 Имитационное моделирование на ЭВМ	8
1.1 Сущность имитационного моделирования	8
1.2 Возможности имитационного моделирования на ЭВМ.....	9
1.3 Модельное время. Способы изменения модельного времени	11
1.4 Способы организации квазипараллелизма компонентов ИМ.....	12
1.5 Технология построения и использования ИМ	13
1.5.1 Содержательное описание объекта моделирования.....	13
1.5.2 Построение концептуальной модели.....	14
1.5.3 Формальное описание объекта моделирования.....	14
1.5.4 Составление модели на языке моделирования	14
1.5.5 Испытание имитационной модели	15
1.5.6 Исследование свойств имитационной модели.....	15
1.5.7 Планирование имитационного эксперимента.....	16
1.5.8 Анализ результатов моделирования	16
2 Имитационное моделирование на GPSS	17
2.1 Назначение и особенности GPSS	17
2.2 Запуск и начало работы в GPSS	17
2.3 Объекты GPSS. Системные и стандартные числовые атрибуты объектов.....	18
2.4 Создание имитационной модели на языке GPSS.....	23
2.4.1 Операторы GPSS. Правила записи операторов языка GPSS	23
2.4.2 Команды управления моделированием (SIMULATE, START, CLEAR, RESET, HALT, CONTINUE, STEP, STOP).....	24
2.4.3 Работа с транзактами (блоки GENERATE, TERMINATE, PRIORITY, ASSIGN, SPLIT, ASSEMBLE, MARK)	25
2.4.4 Работа с обслуживающими устройствами (блоки SEIZE, RELEASE, PREEMPT, RETURN, ENTER, LEAVE).....	28
2.4.5 Работа с очередями (блоки QUEUE, DEPART). Пример имитационной модели билетной кассы.....	30
2.4.6 Работа со стандартными числовыми характеристиками объектов GPSS	31
2.4.7 Управление перемещением транзактов. Работа с логическими ключами (блоки TRANSFER, TEST, GATE, LOGIC, LOOP, MATCH).....	32
2.4.8 Работа с переменными и функциями (операторы VARIABLE, FVARIABLE, BVARIABLE, FUNCTION)	35
2.4.9 Работа с константами, ячейками, матрицами сохраняемых величин и таблицами (операторы EQU, INITIAL, MATRIX, TABLE, блоки SAVEVALUE, MSAVEVALUE, TABULATE)	37
2.5 Моделирование случайных величин в GPSS	40
2.5.1 Моделирование непрерывных случайных величин	40
2.5.2 Моделирование дискретных случайных величин	44
2.5.3 Моделирование случайных величин с произвольным законом распределения.....	44
2.5.4 Моделирование случайных событий.....	47
2.6 Трансляция и запуск имитационной модели в GPSS	48
2.7 Средства отладки имитационных моделей	49
2.8 Анализ результатов моделирования в GPSS.....	55
3 Примеры построения и исследования моделей технологических систем.	
Задания для самостоятельной работы	59
3.1 Пример моделирования двух стратегий обслуживания на АЗС.....	59
3.2 Пример моделирования процесса обслуживания клиентов в железнодорожной билетной кассе	65
3.3 Задания для самостоятельной работы	74
Приложение А Системные и стандартные числовые атрибуты	90
Приложение Б Блоки GPSS	92
Список литературы.....	96
Предметный указатель	97

ВВЕДЕНИЕ. МЕСТО ИМИТАЦИОННОГО МОДЕЛИРОВАНИЯ В ИССЛЕДОВАНИИ СИСТЕМ

Во многих задачах практики непосредственное изучение объектов (технических систем или технологических процессов) затруднено из-за сложности объекта, высокой стоимости или длительности исследования, отсутствия объекта (на этапе разработки объект еще отсутствует в природе), сложности задания необходимых условий функционирования объекта (например, определение характеристик процессов перевозок в чрезвычайных ситуациях) и других причин.

В таких случаях для изучения объектов используется *моделирование* – метод научного исследования, заключающийся в замене исходного объекта его моделью, изучением модели и обобщением полученных при анализе характеристик на сам объект.

В технике наиболее распространенными являются два вида моделирования: физическое и математическое (рисунок 1).

Физическое моделирование состоит в изучении объекта посредством анализа некоторого макета, сохраняющего физическую природу объекта. Примером является модель летательного аппарата, исследуемая в аэродинамической трубе.

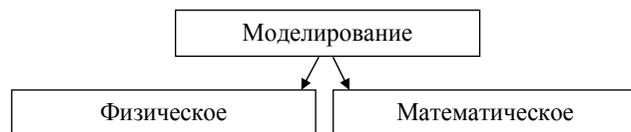


Рисунок 1 – Классификация методов моделирования в зависимости от типа используемой модели

Математические модели описывают исследуемые объекты с помощью математических формул, логических условий или алгоритмов. В свою очередь они классифицируются по следующим основным признакам (рисунок 2).

- По характеру изменения состояний объекта:
 - *дискретные* модели, в которых объект изменяет свое состояние в фиксированные моменты времени, а на интервалах между ними не изменяет

состояние. Например, состояние систем массового обслуживания, определяемое количеством заявок в системе, изменяется в моменты окончания обслуживания и поступления новых заявок;

- *непрерывные* модели, в которых состояние объекта изменяется в каждый момент времени моделирования. Например, изменение температуры в моделях термодинамики и др.

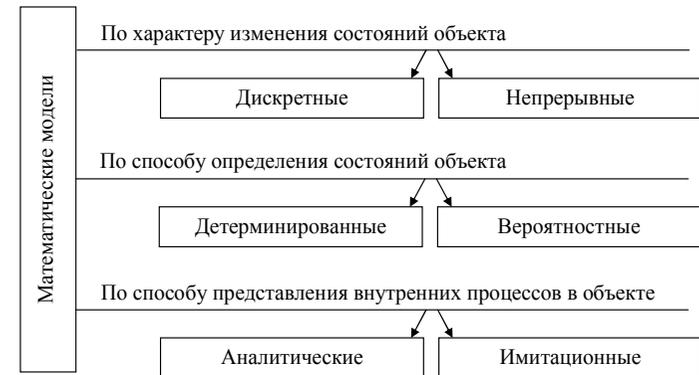


Рисунок 2 – Классификация математических моделей

- По способу определения состояний исследуемого объекта:
 - *детерминированные*, когда состояние объекта в заданный момент времени однозначно определяется начальными условиями и входными воздействиями на объект;
 - *вероятностные* (стохастические), когда состояние объекта в заданный момент времени невозможно определить однозначно, возможно определение только распределения вероятностей возможных состояний объекта при заданных распределениях вероятностей начальных условий и входной информации. Например, работоспособное состояние автомобиля в некоторый момент времени определяется длительностью его эксплуатации, качеством обслуживания, однако имеет вероятностную природу (время безотказной работы одинаковых автомобилей различно).
- По способу представления внутренних процессов в объекте:
 - *аналитические модели*;
 - *имитационные модели*.

Для *аналитического моделирования* характерно то, что процессы функционирования элементов объекта записываются в виде некоторых математических соотношений (алгебраических, интегродифференциальных и т. п.) или логических условий. Например, время t движения поезда со скоростью v на перегоне длины можно определить (не выполняя непосредственного из-

мерения) выражением $t = s/v$. Другой пример: описание траектории движения маятника с помощью дифференциального уравнения.

Для *имитационного моделирования* характерно воспроизведение (как правило, на ЭВМ) процесса функционирования элементов объекта во времени с сохранением их взаимосвязей, последовательности и алгоритмов.

Примером имитационной модели (ИМ) является график движения поездов, где наглядно указаны моменты прибытия и отправления поездов, периоды следования поездов по перегонам, периоды ожидания и пропуска встречных поездов. С усложнением изучаемых объектов в качестве инструмента для построения и исследования ИМ используют ЭВМ. При этом на компьютер возлагается как работа по воспроизведению процесса функционирования изучаемой ИМ, так и по проведению экспериментов с ней.

1 ИМИТАЦИОННОЕ МОДЕЛИРОВАНИЕ НА ЭВМ

1.1 Сущность имитационного моделирования

В случае, когда процессы в изучаемой системе столь сложны и многообразны, что аналитические модели становятся слишком грубым приближением к действительности, возможным выходом является имитационное моделирование.

Например, для исследования эффективности работы билетной кассы и определения необходимого количества кассиров возможно применение аналитических методов теории массового обслуживания. Но применение этих методов значительно усложняется для систем:

- обслуживающих заявки в несколько этапов;
- работающих в нестационарном режиме, с технологическими перерывами;
- обслуживающих заявки с различными приоритетами и закономерностями времени обслуживания (например, приобретение билета на международный рейс требует большего времени) и т. п.

Возможным подходом решения данной задачи является имитационное моделирование, при котором процесс функционирования билетной кассы воспроизводится на ЭВМ, причем элементарные явления, составляющие этот процесс, имитируются с сохранением логической структуры и последовательности протекания. В процессе имитации фиксируются определенные события и состояния, по которым вычисляются характеристики качества функционирования системы.

В частности, имитация процесса функционирования билетной кассы заключается:

- в имитации поступления в кассу посетителей через случайные промежутки времени (в соответствии с существующими вероятностными закономерностями);
- постановке посетителей на обслуживание или в очередь, в зависимости от свободности / занятости кассы в момент поступления требования;
- имитации обслуживания посетителей кассиром в течение случайных интервалов времени;
- удалении из кассы обслуженных посетителей и постановке на обслуживание посетителей, ожидающих в очереди;

- сборе статистики о времени пребывания посетителей в очереди и на обслуживании, длине очереди, времени загрузки кассира и других характеристиках функционирования кассы.

В более сложных случаях при имитации можно учитывать поступление нескольких групп посетителей, требующих различных приоритетов (ветераны, студенты) и длительности обслуживания (покупатели билетов на пригородные или международные рейсы), наступление технологических перерывов и прочих штатных ситуаций.

Многokrратно воспроизводя процесс функционирования билетной кассы, накапливают статистический материал, который позволяет судить об эффективности технологического процесса (количество поступивших и обслуженных покупателей, средняя длина очереди, среднее время ожидания, коэффициент загрузки кассира и прочее) и об его оптимизации (изменение количества кассиров, автоматизация их работы, изменение расписания технологических перерывов и т. п.).

1.2 Возможности имитационного моделирования на ЭВМ

Хотя имитацию процесса функционирования билетной кассы (и других объектов) теоретически можно выполнять «на бумаге», количество данных, которые должны сохраняться и обрабатываться при моделировании (время поступления, длительность ожидания и обслуживания каждого посетителя), диктует необходимость применения ЭВМ. Для имитационного моделирования на ЭВМ необходима реализация следующих видов алгоритмов:

- имитации во времени процесса функционирования элементов исследуемого объекта;
- обеспечения взаимодействия элементов исследуемого объекта и объединения их в единый процесс;
- генерации случайных факторов с требуемыми вероятностными характеристиками;
- статистической обработки и графической презентации результатов имитационного эксперимента.

Развитие компьютерной техники решило проблемы с моделированием большого количества элементов систем и их взаимодействия, а также со сбором статистической информации о функционировании модели системы. Поэтому в настоящее время имитационное моделирование позволяет рассматривать исследуемые системы, практически любой сложности, на любом уровне детализации. При этом в имитационной модели (ИМ) можно реализовать практически любой алгоритм управленческой деятельности или поведения системы.

Имитационное моделирование на ЭВМ позволяет получать наглядную картину поведения системы, рассматривать различные варианты модели, отвечающие различным сторонам функционирования системы и возможным структурным преобразованиям, получать значения необходимых количественных характеристик. Поэтому имитационное моделирование в настоящее время получает все большее распространение в исследовании сложных технических систем и технологических процессов. Целесообразность применения имитационного моделирования становится очевидной при наличии следующих условий:

- не существует законченной математической постановки задачи либо еще не разработаны аналитические методы решения сформулированной задачи;
 - аналитические методы имеются, но математические процедуры столь сложны и трудоемки, что имитационное моделирование дает более простой способ решения задачи;
 - кроме оценки определенных параметров, требуется осуществить наблюдение за ходом процесса функционирования системы в течение некоторого времени. При этом имитационное моделирование дает возможность полностью контролировать время изучения системы, поскольку явление может быть замедлено или ускорено по желанию;
 - необходимо использование ИМ в качестве тренажера при подготовке специалистов. При этом ИМ может применяться для приобретения новых навыков в управлении системой и освоения правил принятия решений.
- Важными ограничениями имитационного моделирования является то, что:
- оно не предоставляет непосредственного решения математических задач, что характерно для аналитических методов. Оно служит в качестве средства для анализа поведения системы в условиях, которые определяются экспериментатором;
 - разработка хорошей ИМ часто обходится дороже создания аналитической модели и требует наличия квалифицированных специалистов и больших затрат времени;
 - при использовании ИМ применяются многочисленные методы статистического анализа данных, что усложняет исследование.

Преодоление перечисленных выше ограничений лежит на пути создания программно-технологического инструментария, позволяющего автоматизировать этапы построения ИМ систем и тем самым ускорить сроки их исследования.

1.3 Модельное время. Способы изменения модельного времени

Отметим две особенности функционирования ЭВМ, которые приходится учитывать при разработке ИМ систем:

1 Исследуемые системы состоят из множества элементов. Все элементы системы функционируют одновременно. Однако в большинстве ЭВМ параллельное выполнение нескольких программ, имитирующих поведение отдельных элементов системы, невозможно.

2 Компьютеры, как цифровые устройства, способны имитировать поведение системы только в дискретном времени. То есть все события, возникающие в системе, должны иметь привязку к дискретной шкале времени с заданным минимальным делением (например, микросекунда, минута или сутки).

Чтобы обеспечить имитацию параллельных (одновременных) процессов функционирования элементов системы в ИМ используется специальная переменная t_m , называемая *модельным временем*. С помощью переменной t_m организуются синхронизация всех событий и процессов в ИМ системы, т. е. реализуется квазипараллельная работа компонентов ИМ. Приставка «квази» отражает последовательный характер обслуживания в ИМ компонентов и событий, одновременно возникающих в различных элементах реальной системы.

Модельное время t_m следует отличать от других типов времени, используемых при моделировании систем, таких как: t_p – *реальное время* системы, функционирование которой имитируется; t_s – *машинное время* имитации, отражающее затраты ресурса времени ЭВМ на организацию имитационного моделирования. Так, на современной ЭВМ в течение одной минуты ($t_s = 1$ мин) можно промоделировать годовую работу ($t_p = 1$ год) билетной кассы.

Возможны два способа изменения модельного времени: «шагом до следующего события» и «фиксированным шагом».

При использовании продвижения модельного времени «шагом до следующего события» модельное время в исходном состоянии устанавливается в 0. Определяется время возникновения будущих событий. После этого часы модельного времени переходят на время возникновения ближайшего события, и в этот момент обновляется состояние системы с учетом произошедшего события, а также сведения о времени возникновения будущих событий. Процесс продвижения t_m от времени возникновения одного события до времени возникновения других событий продолжается до тех пор, пока не будет выполнено некоторое указанное заранее условие останова.

В *дискретных моделях*, где все изменения состояния (все события) происходят мгновенно (например, поступление посетителя в очередь, начало

обслуживания), периоды бездействия системы просто пропускаются, и модельное время изменяется от момента возникновения одного события к моменту выполнения другого.

В *непрерывных моделях* (например, моделях гидравлических процессов, процессов механического износа деталей), где изменения состояний системы происходят плавно, продвижение модельного времени выполняют «фиксированным шагом», изменяя t_m на константу Δt_m малую по сравнению с длительностью изменения состояния системы. При этом динамика модели является дискретным приближением реальных процессов (вследствие необходимости представить непрерывный процесс в дискретной ЭВМ).

1.4 Способы организации квазипараллелизма компонентов ИМ

Вследствие последовательного характера обработки информации в большинстве ЭВМ параллельные процессы, происходящие в исследуемой системе, преобразуются в последовательные процессы. Например, при имитационном моделировании билетной кассы параллельные процессы, связанные с имитацией поступления пассажиров, имитацией обслуживания пассажиров в кассе и с контролем состояния очереди, фактически ЭВМ обрабатывает последовательно. Процедура преобразования параллельных процессов в последовательный называется организацией *квазипараллелизма*.

В зависимости от способов описания ИМ применяют следующие основные способы организации квазипараллелизма: событийный, процессный, транзактный, агрегатный и основанный на просмотре активностей.

Событийный способ организации квазипараллелизма используется, когда элементы изучаемой системы выполняют одни и те же функциональные действия, которые приводят к одним и тем же событиям. Множество событий можно разбить на небольшое число типов событий. Для каждого типа событий определена последовательность действий, приводящая к изменению состояния системы, а также определены условия перехода от одного события к другому для всех типов событий.

Агрегатный способ организации квазипараллелизма используется, когда имеет место тесное взаимодействие между функциональными действиями элементов системы. При агрегатном способе все элементы исследуемой системы представляют собой агрегаты, обменивающиеся сигналами. Выходной сигнал от одного агрегата является входным сигналом для другого. Моделирование поведения агрегата – это последовательная цепь переходов из одного состояния в другое под воздействием поступающих сигналов.

Способ, основанный на просмотре активностей, применяется, когда все действия для элементов исследуемой системы различны и приводят к наступлению различных событий. При этом каждое действие характеризуется набором условий его выполнения. Моделирующий алгоритм, основанный на просмотре активностей, реализует просмотр всех наборов условий, а также обрабатывает активности, условия для которых выполняются, т.е. моделирует время выполнения соответствующего действия и реализует само действие.

Процессный способ сочетает в себе черты событийного способа и способа, основанного на просмотре активностей. Он применяется, когда поведение элементов исследуемой системы может быть описано фиксированными для некоторого класса систем последовательностями событий и действий, так называемыми процессами.

Транзактный способ организации квазипараллелизма – развитие процессного способа для моделирования систем массового обслуживания. Инициаторами появления событий в ИМ являются транзакты – динамические объекты, отождествляемые с заявками на обслуживание, которые перемещаются между элементами системы массового обслуживания. Для описания ИМ создается фиксированный набор блоков, связанных с обработкой и обслуживанием транзактов. С их помощью происходит уничтожение и создание транзактов, задержка их на некоторый период времени, управление движением транзактов, занятие и освобождение различных типов ресурсов системы. Связь между обслуживающими приборами устанавливается с помощью системы очередей и способов извлечения из них транзактов.

Одну и ту же систему принципиально можно представить любым из указанных способов. Однако построенные на их основе модели будут отличаться размерами и количеством ресурсов, затраченных на их создание, испытание и использование.

1.5 Технология построения и использования ИМ

1.5.1 Содержательное описание объекта моделирования

Процесс построения и исследования ИМ систем целесообразно выполнять поэтапно. При этом рекомендуется придерживаться следующего порядка.

На первом этапе осуществляется постановка задачи и формулировка целей исследования:

- выбираются показатели эффективности функционирования исследуемой системы, отражающие цели моделирования;
- определяются управляющие параметры и контролируемые переменные;

- выполняется описание внешней среды, с которой взаимодействует объект исследования;

- составляется список возможных ограничений модели.

Иллюстрация этапов разработки и исследования ИМ представлена в разд. 3.

1.5.2 Построение концептуальной модели

На основании содержательного описания исследуемой системы осуществляется переход к концептуальной модели. Концептуальная модель представляет собой упрощенное математическое или алгоритмическое описание исследуемой системы.

На этапе построения концептуальной модели проводится разбиение (декомпозиция) системы на элементы, допускающие удобное математическое или алгоритмическое описание, определяются связи между элементами.

В состав концептуальной модели входят:

- уточнённое содержательное описание объекта моделирования;
- список управляющих параметров и контролируемых переменных;
- критерии эффективности функционирования вариантов системы;
- список используемых методов обработки результатов имитации и перечисление способов представления результатов моделирования.

1.5.3 Формальное описание объекта моделирования

На данном этапе выбирается способ формализации объекта исследования (событийный, процессный, транзактный, агрегатный или основанный на просмотре активностей).

Формальное описание исследуемой системы проводится следующим образом: уточнение декомпозиции системы, алгоритмизация компонентов модели.

Примеры формального описания объектов моделирования представлены в разд. 3.

1.5.4 Составление модели на языке моделирования

На этом этапе выбирается тип ЭВМ, средство автоматизации моделирования, примерные затраты памяти и времени на создание ИМ.

Разрабатывается алгоритм моделирования, приводится схема ИМ в терминах алгоритма моделирования или выбранного средства моделирования и описывается программная реализация модели.

Дается описание каждого блока программы с комментариями к ним. Для оценки правильности программной реализации ИМ проводится пробный

эксперимент с целью проверки правильности функционирования программы. Приводятся данные по тестированию модели.

Несмотря на то, что при разработке ИМ может использоваться любой из универсальных языков программирования (например, Паскаль, Си и др.), на практике предпочтение часто отдается специализированному предметно-ориентированному программному обеспечению автоматизации имитационного моделирования. Подходы, основанные на использовании универсальных языков и средств автоматизации имитационного моделирования, имеют как свои достоинства, так и недостатки.

1.5.5 Испытание имитационной модели

Испытание ИМ включает два этапа: верификацию и проверку адекватности.

На этапе *верификации* необходимо убедиться в правильности алгоритма функционирования ИМ, его соответствии замыслу моделирования. Необходимо получить гарантию того, что созданная ИМ во всех ситуациях будет правильно отражать поведение исследуемой системы. С помощью верификации устанавливается верность логической структуры модели.

Обычно верификация выполняется в ходе комплексной отладки программы ИМ на реальных или тестовых исходных данных. Важно при этом тщательно отладить и проверить поведение программы той части модели, которая прогнозирует работу системы.

Под *адекватностью* ИМ объекту исследования понимают совпадение с заданной точностью значений характеристик функционирования ИМ и реального объекта.

При отсутствии адекватности проводят калибровку ИМ (корректируют параметры компонентов модели и алгоритмы их функционирования). При наличии ошибок во взаимодействии компонентов модели возвращаются к этапу создания ИМ. Возможно, что в ходе формализации были слишком упрощены процессы и исключен из рассмотрения ряд важных сторон функционирования исследуемой системы, что привело к неадекватности модели.

1.5.6 Исследование свойств имитационной модели

На данном этапе оцениваются точность имитации, необходимый объем выборки, длина реализация прогона ИМ, устойчивость результатов моделирования, чувствительность критериев качества к изменению параметров ИМ, стационарность режима моделирования и др.

Точность имитации представляет собой оценку влияния случайных факторов на функционирование ИМ системы.

Устойчивость результатов моделирования характеризуется сходимостью контролируемого отклика ИМ к определённой величине при изменениях параметров модели.

Стационарность режима моделирования характеризует собой некоторое установившееся равновесие процессов в модели, когда дальнейшая имитация бессмысленна, поскольку новой информации из ИМ исследователь не получит, и продолжение имитации приведёт к увеличению затрат машинного времени. Поэтому необходимо разработать процедуру проверки достижения стационарного режима имитации.

Чувствительность ИМ представляется величиной минимального приращения выбранного критерия качества, вычисляемого по результатам моделирования, при варьировании параметров моделирования на всём диапазоне их изменения.

1.5.7 Планирование имитационного эксперимента

Этап эксплуатации ИМ начинается с составления плана эксперимента, позволяющего исследователю получить максимум информации при минимальных затратах времени и средств. Составляется статистическое обоснование плана эксперимента. Планирование эксперимента представляет собой процедуру выбора числа и условий проведения опытов, необходимых и достаточных для решения поставленной задачи с требуемой точностью. Стремятся минимизировать общее число опытов с одновременным варьированием параметров модели. Выбирают такую стратегию имитационного эксперимента, которая позволяет обоснованно принимать решения после каждой серии экспериментов с ИМ.

1.5.8 Анализ результатов моделирования

Получив результаты моделирования, необходимо приступить к интерпретации результатов. Возможно, что на данном этапе будут обнаружены ошибки либо при создании модели, либо при формализации объекта моделирования. В этих случаях осуществляется возврат на соответствующий этап построения ИМ.

Результатом этапа интерпретации данных являются рекомендации по оптимизации или модификации исследуемой системы. На их основе исследователи приступают к принятию решений. При интерпретации результатов большое значение имеют возможности выбранного средства моделирования по визуализации и статистическому анализу данных.

2 ИМИТАЦИОННОЕ МОДЕЛИРОВАНИЕ НА GPSS

2.1 Назначение и особенности GPSS

Пакет GPSS (General Purpose Simulation System – система моделирования общего назначения) предназначен для имитационного моделирования дискретных систем и входит в число наиболее распространенных и используемых на практике средств автоматизации имитационного моделирования. Одна из последних версий пакета GPSS для персональных компьютеров, работающих под управлением операционной системы Windows, называется GPSS World. Она будет рассмотрена в пособии.

Пакет GPSS реализует собственный язык имитационного моделирования, в основу которого положен транзактный способ организации квазипараллелизма и способ изменения модельного времени «шагом до следующего события».

Имитационная модель в GPSS представляет собой последовательность текстовых строк, каждая из которых определяет правила создания, перемещения, задержки и удаления транзактов.

2.2 Запуск и начало работы в GPSS

Запуск GPSS. Для запуска пакета GPSS в главном меню операционной системы Windows (кнопка «ПУСК» или «START») необходимо выбрать приложение «GPSS World».

Создание новой модели. Для создания новой ИМ необходимо в главном меню GPSS World выбрать пункт «File → New», затем в диалоговом окне выбрать режим создания новой модели. В рабочем окне GPSS World появится новое текстовое окно с заголовком «Untitled Model 1», в котором записывается текст создаваемой ИМ (рисунок 3).

Сохранение модели. Для сохранения текста создаваемой ИМ необходимо в главном меню GPSS World выбрать пункт «File → Save» или «File → Save As» (рисунок 4). Во втором случае пользователю будет предложено задать новое имя для файла с текстом ИМ. Файл с текстом модели в GPSS имеет расширение «gps».

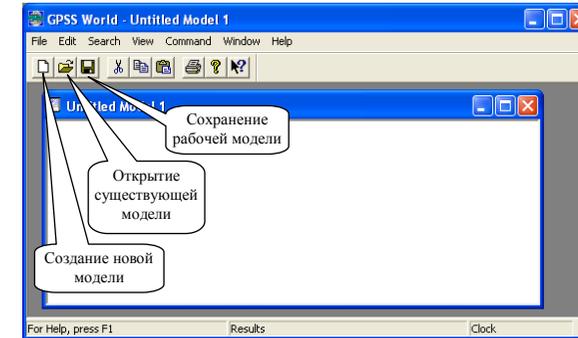


Рисунок 3 – Окно GPSS World

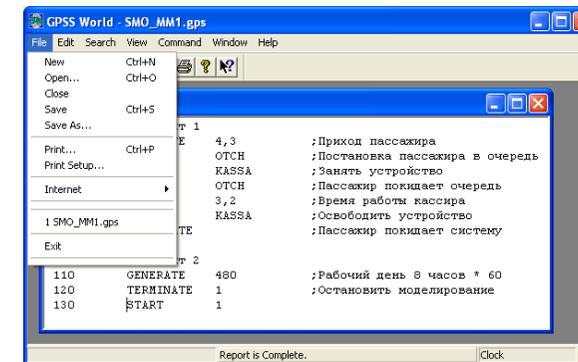


Рисунок 4 – Содержимое главного меню GPSS World

Использование существующей модели. Для открытия уже существующего файла с текстом ИМ необходимо в главном меню GPSS World выбрать пункт «File → Open», затем указать имя необходимого файла в стандартном окне Windows. Три перечисленные операции с файлами могут инициироваться кнопками на панели инструментов GPSS World (см. рисунок 3).

2.3 Объекты GPSS. Системные и стандартные числовые атрибуты объектов

Для моделирования систем в GPSS выделяется конечное множество абстрактных компонентов, необходимых для описания элементов реальной системы (например, источников заявок на обслуживание, очередей, обслуживающих приборов и т. д.), и конечное множество стандартных операций,

описывающих связи между элементами. Выделенным множествам элементов и операций ставится в соответствие множество объектов GPSS, основные типы которых представлены в таблице 1.

Таблица 1 – Типы объектов GPSS

Тип объектов	Обозначение	Тип объектов	Обозначение
Транзакты	–	Функции	<i>FN</i>
Блоки	–	Очереди	<i>Q</i>
Одноканальные устройства	<i>F</i>	Таблицы	<i>T</i>
Многоканальные устройства	<i>S</i>	Ячейки	<i>X</i>
Логические ключи	<i>L</i>	Матрицы ячеек	<i>M</i>
Арифметические переменные	<i>V</i>	Списки пользователя	<i>C</i>
Логические (Булевы) переменные	<i>BV</i>	Группы	<i>G</i>

Все объекты в GPSS имеют свойства, называемые **стандартными числовыми атрибутами** (СЧА). А свойства ИМ в целом в GPSS называются **системными числовыми атрибутами**. Каждый объект GPSS имеет свой набор СЧА. Значения всех СЧА доступны пользователю, причем некоторые из них могут быть изменены. Значения системных числовых атрибутов доступны лишь для чтения.

К системным числовым атрибутам относятся:

RN_j – число, возвращаемое *j*-м датчиком случайных чисел. Это целое число изменяется в пределах от 0 до 999 включительно, кроме случаев его использования в качестве аргумента функции или элемента в переменной, когда его значение – дробь от 0 до 0,999999;

CI – текущее значение относительного модельного времени (автоматически изменяется GPSS и устанавливается в 0 командами CLEAR или RESET, см. п. 2.4.2);

ACI – текущее значение абсолютного модельного времени (автоматически изменяется GPSS и устанавливается в 0 командой CLEAR, см. п. 2.4.2);

TGI – текущее значение счетчика завершений (см. п. 2.4.2);

MI – время пребывания в системе обрабатываемого в данный момент транзакта.

Транзакты описывают единицы исследуемых потоков (заявки, требования на обслуживание), например, автомобили в очереди на бензоколонке; составы, ожидающие осмотра бригадой техосмотра; корабли, разгружаемые в порту; покупатели в магазине и т. д. Транзакты движутся от блока к блоку так, как движутся элементы, которые они представляют. Каждое продвижение транзакта инициирует в модели некоторые события (например, занятие кассы или увеличение длины очереди при поступлении нового посетителя). События обрабатываются GPSS в соответствующий момент модельного времени.

Содержательное значение транзактов определяет разработчик модели. Именно он устанавливает аналогию между транзактами и реальными динамическими элементами моделируемой системы.

Основными СЧА транзакта в GPSS являются параметры и приоритет.

Каждый транзакт может иметь (по желанию пользователя) до 1020 параметров одного из четырех форматов:

- «слово» – может принимать целые значения от минус 2147483647 до плюс 2147483647;
- «полуслово» – целые значения от минус 32768 до плюс 32768;
- «байт» – целые значения от минус 255 до плюс 255;
- «плавающая точка» – дробные значения от минус 16777215 до плюс 16777215.

По умолчанию значения всех параметров транзакта устанавливаются в ноль, в дальнейшем могут изменяться с помощью блока ASSIGN (см. п. 2.4.3):

P_j – значение *j*-го параметра текущего транзакта;

PF_j – значение *j*-го параметра текущего транзакта форматом «слово»;

PH_j – значение *j*-го параметра текущего транзакта форматом «полуслово»;

PB_j – значение *j*-го параметра текущего транзакта форматом «байт»;

PL_j – значение *j*-го параметра текущего транзакта форматом «плавающая точка»;

PR – приоритет транзакта, обрабатываемого в данный момент (может изменяться от 0 до 127 блоком PRIORITY, см. п. 2.4.3);

MP_j – значение, равное разности абсолютного модельного времени и значения *j*-го параметра текущего транзакта;

MB_j – флаг синхронизации: 1, если транзакт в блоке *j* принадлежит тому же семейству, что и текущий транзакт; 0 – в противном случае.

Блоки задают логику функционирования ИМ системы и определяют пути движения транзактов. Практически все изменения состояний ИМ (события) происходят в результате входа транзактов в блоки и выполнения блоками своих функций. Основные функции блоков следующие:

- создание (генерация) и уничтожение транзактов;
- изменение числовых атрибутов блоков и транзактов;
- задержка транзакта на определенный интервал времени;
- изменение маршрута движения транзакта.

У каждого блока имеется два СЧА:

I_j – количество транзактов, находящихся в блоке с номером *j* в текущий момент модельного времени;

N_j – общее количество транзактов, поступивших в блок с номером *j*.

Одноканальные устройства (Facility) описывают оборудование, которое в любой момент времени может быть занято только одним транзактом (одноканальные системы массового обслуживания), а также оборудование, на котором обслуживание одной заявки может быть прервано обслуживанием другой заявки (например, с более высоким приоритетом). Одноканальные устройства в GPSS обеспечивают сбор основной статистической информации о своем функционировании и имеют следующие СЧА:

Fj – текущее состояние устройства j . Равно 0, если устройство свободно, и 1, если устройство занято (может изменяться блоками SEIZE и RELEASE, см. п. 2.4.4);

FRj – коэффициент использования устройства в тысячных долях, т. е. если коэффициент равен 0,7, то FRj равно 700;

FCj – общее число транзактов, вошедших в устройство j ;

FTj – среднее время использования устройства одним транзактом.

Многоканальные устройства (Storage) описывают оборудование, которое может использоваться несколькими транзактами одновременно (многоканальные системы массового обслуживания). Кроме того, многоканальные устройства обеспечивают сбор основной статистической информации о своем функционировании и имеют СЧА:

Sj – текущее содержимое многоканального устройства j (может изменяться блоками ENTER и LEAVE, см. п. 2.4.4);

Rj – число свободных единиц многоканального устройства;

SRj – коэффициент использования многоканального устройства в тысячных долях. Например, если коэффициент равен 0,65, то SRj равен 650;

SAj – среднее содержимое многоканального устройства j (целая часть);

SMj – максимальное содержимое многоканального устройства j ;

SCj – общее число транзактов, вошедших в многоканальное устройство j ;

STj – среднее время пребывания транзактов в многоканальном устройстве j ;

SEj – признак пустоты многоканального устройства j : 1 – пусто, 0 – заполнено;

SFj – признак заполненности многоканального устройства j : 1 – заполнено, 0 – в противном случае.

Логические ключи используются для блокировки или изменения направления движения транзактов в зависимости от ранее наступивших в модели событий. Логические ключи имеют один СЧА:

LSj – состояние логического ключа с номером j : 1 – включен, 0 – выключен.

Арифметические переменные позволяют вычислять арифметические выражения, в том числе с числовыми атрибутами объектов. В выражениях могут быть использованы функции.

Логические переменные позволяют проверять несколько условий, исходя из состояний или значений СЧА объектов, например, для описания условий, определяющих движение транзактов.

Функции позволяют задавать функциональные зависимости между несколькими переменными, а также переменной и СЧА объектов.

Переменные и функции в GPSS имеют следующие СЧА:

Vj – значение арифметической переменной j ;

BVj – значение логической переменной j (1 – истина, 0 – ложь);

FNj – значение функции j (дробная часть отбрасывается за исключением использования в качестве аргумента другой функцией).

Очереди (Queue) обеспечивают сбор основной статистической информации о времени задержки транзактов из-за недоступности или занятости оборудования. Очереди имеют СЧА:

Qj – текущая длина очереди j (может изменяться блоками QUEUE и DEPART, см. п. 2.4.5);

QAj – средняя длина очереди j ;

QMj – максимальная длина очереди j ;

QCj – общее число входов в очередь j ;

QZj – число нулевых входов в очередь j ;

QTj – среднее время пребывания транзактов в очереди j , включая транзакты, прошедшие очередь без ожидания;

QXj – среднее время пребывания транзактов в очереди j , исключая транзакты, прошедшие очередь без ожидания.

Таблицы предназначены для сбора статистической информации о случайных величинах, заданных пользователем. Таблица состоит из частотных групп, в которые заносится число попаданий заданной случайной величины (переменной или СЧА). Для каждой таблицы автоматически вычисляется математическое ожидание и среднее квадратическое отклонение значений. Ниже приведены СЧА таблиц:

TBj – среднее арифметическое значение элементов таблицы j ;

TCj – общее число элементов в таблице j ;

TDj – среднее квадратическое отклонение элементов таблицы j .

Ячейки и матрицы сохраняемых величин используются для сохранения некоторой числовой информации. Например, значение, занесенное в ячейку, может представлять собой длину очереди в какой-то конкретной точке модели. Любой транзакт может произвести запись информации в эти

объекты либо считать информацию. Ячейки и матрицы сохраняемых величин имеют следующие СЧА:

X_j – содержимое ячейки j ;

$MX_j(a,b)$ – содержимое элемента матрицы ячеек j , расположенного в строке a и столбце b .

Списки пользователя позволяют организовать работу с очередями, дисциплина обслуживания в которых отличается от FIFO («первым пришел – первым обслужен»), описываемых объектом «очередь», а также помогают синхронизировать движение различных транзактов по ИМ. СЧА списка пользователя:

CH_j – текущее число транзактов в j -м списке пользователя;

CA_j – среднее число транзактов в j -м списке пользователя;

CM_j – максимальное число транзактов в j -м списке пользователя;

CC_j – общее число транзактов, которые поступали в j -м списке пользователя;

CT_j – среднее время пребывания транзактов в j -м списке пользователя.

Системные и стандартные числовые атрибуты приведены в приложении А.

2.4 Создание имитационной модели на языке GPSS

2.4.1 Операторы GPSS. Правила записи операторов языка GPSS

Программа имитационного моделирования на языке GPSS представляет собой последовательность операторов – текстовых строк описания:

- объектов исследуемой системы;
- блоков модели, которые имитируют функционирование элементов системы;
- команд управления моделированием.

Строка оператора GPSS состоит из следующих полей, разделяемых знаком «пробел»:

Номер строки	Поле метки	Поле операции	Поле операндов	Поле комментариев
			A, B, C, D, E, F, G, H	;

Поле «номер строки» необязательное. Содержимым поля может быть любое десятичное число из семи символов, в том числе и дробное. В последнем случае десятичная точка рассматривается как один из семи символов.

Содержимое поля метки зависит от типа оператора. В операторах описания объектов содержимым поля является имя объекта, в операторах блоков – метка, в управляющих операторах поле метки пусто.

Поле операции содержит символическое обозначение оператора.

Содержимое поля операндов для разных операторов отличается количеством (от 0 до 8) и назначением операндов. Если при записи оператора пропускаются необязательные параметры, то их отсутствие отмечается запятой.

Поле комментариев (необязательное) содержит информацию, поясняющую назначение оператора, отделяется от прочих полей точкой с запятой.

2.4.2 Команды управления моделированием (SIMULATE, START, CLEAR, RESET, HALT, CONTINUE, STEP, STOP)

Для задания условий моделирования, таких как продолжительность моделирования, число повторных прогонов ИМ, порядок и условия сбора статистики в GPSS World используются управляющие команды: SIMULATE, START, RESET, CLEAR, HALT, CONTINUE, STEP, STOP. Команды могут быть включены в текст ИМ или выбраны в пункте «Command» главного меню GPSS (см. рисунок 3). Поступившие команды выстраиваются в очередь и выполняются одна за другой, пока этот процесс не будет приостановлен или пока не будут выполнены все команды.

Команды SIMULATE и START указывают условия завершения процесса моделирования (моделирование в течение заданного интервала реального времени, или в течение заданного интервала модельного времени, или по окончании обслуживания заданного числа транзактов) и имеют следующий вид:

SIMULATE [A]

A – число минут реального времени, по истечении которого моделирование будет завершено и на экран будет выведена накопленная к данному моменту статистическая информация. Если команда отсутствует или поле A пусто, то завершение моделирования определяется другими условиями.

START A

A – начальное значение счетчика завершений (системного числового атрибута TGI, см. подразд. 2.3). Моделирование завершается, когда счетчик завершений примет нулевое или отрицательное значение. Значение счетчика завершений уменьшается при поступлении транзактов в блок TERMINATE на величину, указанную в блоке TERMINATE (см. п. 2.4.3).

Для проведения нескольких прогонов ИМ в течение одного выполнения программы моделирования (например, при многократном исследовании точной работы билетной кассы для оценки среднемесячных показателей ее функционирования) предназначены команды RESET и CLEAR.

Команда RESET обнуляет всю собранную статистику и значение относительного модельного времени CI (модельного времени одного прогона модели).

Команда **CLEAR** дополнительно обнуляет значение абсолютного модельного времени *ACI*, инициализирует генераторы случайных чисел и удаляет из модели все имеющиеся транзакты.

Команда **HALT** немедленно прерывает процесс моделирования, переводя его в приостановленное состояние и удаляя оставшиеся команды из очереди команд. Моделирование может быть продолжено позже командой **CONTINUE**.

Команда **STEP** задает условие прерывания процесса моделирования при прохождении транзактами ИМ заданного количества блоков. Команду **STEP** используют обычно для трассировки модели. Формат команды:

STEP A

A – положительное целое число пройденных транзактами блоков.

Модель, которая была запущена командой **STEP**, не может быть завершена при нулевом счетчике завершений *TGI*.

Команда **STOP** устанавливает или снимает условие останова моделирования. Формат команды следующий:

STOP [A],[B]

A – номер транзакта, удовлетворяющего условию останова;

B – номер или метка блока, удовлетворяющего условию останова.

Например:

`STOP 100,50` ; команда задает условие останова ИМ при входе транзакта с номером 100 в блок с номером 50

Для отключения условия останова необходимо выполнить команду

STOP „OFF

Команда **STOP** без операндов вызывает немедленный останов процесса моделирования, который можно продолжить командой **CONTINUE**. Команды **HALT**, **STEP**, **STOP** и **CONTINUE** широко используются для отладки ИМ.

2.4.3 Работа с транзактами (блоки GENERATE, TERMINATE, PRIORITY, ASSIGN, SPLIT, ASSEMBLE, MARK)

Для создания транзактов и моделирования их поступления в модель предназначен блок **GENERATE**, который имеет следующий формат:

GENERATE A,[B],[C],[D],[E],[F],[G],[H],[I]

A – среднее значение интервала времени между моделируемыми транзактами (по умолчанию – 0);

B – величина разброса возможных значений времени;

C – модельное время генерации первого транзакта;

D – максимальное количество моделируемых транзактов;

E – приоритет транзактов (по умолчанию – 0, т.е. самый низкий приоритет);

F,..., I – количество и формат параметров транзактов (по умолчанию – 12 параметров формата «полуслово»).

Например:

`GENERATE 10,2,5,,2` ; моделирование транзактов через интервалы времени, равномерно распределенные на отрезке [8,12] (или 10 ± 2)
; первый транзакт моделируется в момент модельного времени, равный 5 единиц
; общее количество моделируемых транзактов не ограничено
; все транзакты имеют приоритет 2 и 12 параметров

`GENERATE 75, FN$EXPON,,20,,3PB` ; моделирование транзактов через интервалы времени, имеющие экспоненциальное распределение со средним значением 75 единиц (моделирование других распределений рассмотрено в подразд. 2.5); первый транзакт моделируется в нулевой момент модельного времени; генерируется только 20 транзактов с нулевым приоритетом; каждый транзакт имеет по 3 параметра формата «полуслово», т.е. способных принимать значения от -255 до 255

Если транзакт начал свое движение, он передвигается от блока к блоку по пути, указанному блок-схемой (логикой работы модели). В тот момент, когда транзакт входит в блок, вызывается соответствующая этому блоку подпрограмма. Далее транзакт пытается войти в следующий блок. Его перемещение продолжается, пока очередной блок доступен или пока он не входит в блок, функцией которого является задержка транзакта (на определенное время) или удаление его из модели. В противном случае транзакт остается в том блоке, в котором в данное время находится (но позже будет повторять попытки движения) и в модели делается попытка перемещения другого транзакта.

Для задержки транзактов в течение заданного интервала модельного времени используется блок **ADVANCE**. Формат блока:

ADVANCE A,[B]

Операнды A и B аналогичны соответствующим операндам блока **GENERATE** (см. выше).

Например:

`ADVANCE 30,5` ; моделирование задержки транзактов в течение времени, которое имеет равномерное распределение на отрезке [25, 35] (или 30 ± 5)

ADVANCE 75 ; задержка транзактов в течение 75 единиц модельного времени

Для удаления транзактов из модели используется блок TERMINATE, который имеет следующий вид:

TERMINATE [A]

операнд A указывает число (по умолчанию – 0), на которое уменьшается содержимое счетчика завершений (системного числового атрибута TGI), значение которого задается командой START (см. п. 2.4.2).

Например:

TERMINATE ; уничтожение транзакта, поступившего в блок; значение счетчика завершений не изменяется
TERMINATE 1 ; уничтожение транзакта, поступившего в блок; значение счетчика уменьшается на 1 (при достижении нулевого или отрицательного значения счетчика завершений процесс моделирования прекращается)

Для изменения приоритета транзакта используется блок PRIORITY, который имеет формат:

PRIORITY A

Операнд A указывает новое значение приоритета транзакта, вошедшего в блок (от 0 до 127 включительно).

Например:

PRIORITY 3 ; присвоить транзакту приоритет 3

Для изменения значений параметров транзакта используется блок ASSIGN, который имеет следующий вид:

ASSIGN A,B,[C],[D]

A – номер изменяемого параметра с указанием режима изменения: накопление (+), вычитание (–), замещение (без дополнительных символов);

B – число, изменяющее значение параметра;

C – имя функции, применяемой для модификации значения параметра;

D – формат изменяемого параметра: PF, PH, PB или PL (по умолчанию – PH).

Например:

ASSIGN 3+,5,,PB ; увеличить значение параметра 3 форматом «байт» на 5 единиц
ASSIGN 3–,7,,PF ; уменьшить значение параметра 3 форматом «слово» на 5 единиц
ASSIGN 2–6,5.75,,PL ; задать параметрам 2–6 (форматом «плавающая точка») значение 5,75

Для создания копий транзакта используется блок SPLIT формата

SPLIT A,[B],[C]

A – число создаваемых копий;

B – метка блока, к которому отправляются копии исходного транзакта (сам исходный транзакт безусловно переходит в следующий после SPLIT блок);

C – номер параметра, используемого для присвоения копиям последовательных номеров.

Например:

SPLIT 1 ; создать одну копию исходного транзакта и направить (вместе с исходным транзактом) в следующий блок
SPLIT 2,LABEL1 ; создать две копии исходного транзакта и направить их в блок с меткой LABEL1

Для объединения определенного числа транзактов одного семейства используется блок ASSEMBLE A, который имеет следующий формат:

ASSEMBLE A

A – число n объединяемых транзактов.

Первый из транзактов, принадлежащих некоторому семейству, вошедший в блок ASSEMBLE, задерживается до прихода в блок n – 1 транзакта из данного семейства. После чего первый транзакт переходит к следующему блоку, а оставшиеся n – 1 транзакты уничтожаются.

Ранее указывалось, что время пребывания транзакта в модели определяется его СЧА M1. Для определения времени перемещения транзакта между двумя произвольными точками модели используется блок MARK, который имеет вид:

MARK A

A – номер параметра транзакта, в который записывается текущее значение абсолютного модельного времени AC1 (см. системные числовые атрибуты в подразд. 2.3).

Например:

MARK 2 ; сохранение текущего модельного времени во 2-м параметре транзакта
; далее в тексте модели располагают блок, определяющий время перемещения транзакта, равное разности текущего модельного времени и значения, хранимого во 2-м параметре транзакта

2.4.4 Работа с обслуживающими устройствами (блоки SEIZE, RELEASE, PREEMPT, RETURN, ENTER, LEAVE)

Для моделирования работы одноканальных устройств систем массового обслуживания в GPSS предназначены блоки SEIZE, RELEASE. Занятие транзактом одноканального устройства моделируется блоком SEIZE, а его освобождение – блоком RELEASE:

SEIZE A

RELEASE A

A – имя устройства, занимаемого (освобождаемого) транзактом;

Например:

```
SEIZE KASSA ; моделирование занятия кассы посетителем
ADVANCE 5 ; задержка транзакта на 5 единиц модельного времени,
; моделирующая обслуживание посетителя
RELEASE KASSA ; освобождение кассы посетителем
```

Для моделирования работы одноканальных устройств с прерываниями, т. е. с приостановкой обслуживания транзакта, ранее занявшего одноканальное устройство, и захвата устройства прерывающим транзактом используется блок

PREEMPT A,B,[C],D,E

A – имя устройства, работа которого прерывается входящим в блок транзактом;

B – режим прерывания (по умолчанию – обычный, PR – прерывание по приоритету);

C – метка блока, в который направляется транзакт, обслуживание которого было прервано. Прерванный транзакт покидает устройство, но претендует на право его использования (если не задан операнд E);

D – номер параметра прерванного транзакта, в который заносится остаток времени обслуживания;

E – если в поле записывается «RE», то транзакт, обслуживание которого было прервано, не претендует на завершение своего обслуживания в устройстве.

Прерванный транзакт (если отсутствует адрес в поле C) ожидает окончания прерывания устройства и возвращения на устройство для обслуживания. Если таковых транзактов несколько, то они возвращаются на устройство в порядке прерываний (либо приоритетов).

Транзакт, вошедший в блок RETURN, снимает прерывание на устройстве, вызванное вхождением данного транзакта в блок PREEMPT:

RETURN A

A – имя устройства, с которого снимается прерывание.

Прерывание может быть снято только тем транзактом, которым оно было сгенерировано.

Пары блоков SEIZE–RELEASE и PREEMPT–RETURN могут работать с одними и теми же устройствами. В зависимости от логики работы модели пользователь сам должен определить потребность работы устройств с прерываниями.

Для моделирования многоканальных устройств систем массового обслуживания в GPSS предназначены блоки ENTER и LEAVE. Занятие транзак-

том одного из приборов многоканального устройства моделируется блоком ENTER, а его освобождение – блоком LEAVE:

ENTER A,[B]

LEAVE A,[B]

A – имя многоканального устройства, занимаемого (освобождаемого) транзактом;

B – число занимаемых (освобождаемых) приборов многоканального устройства (по умолчанию – 1).

При работе с многоканальными устройствами необходимо предварительно указать количество приборов устройства с помощью оператора

A STORAGE B

A – имя многоканального устройства;

B – количество приборов многоканального устройства.

Например:

```
PUNKT STORAGE 4 ; описание многоканального устройства междугородный
; переговорный пункт имеет 4 телефонные кабины
...
ENTER PUNKT ; моделирование занятия одной из свободных телефон-
; ных кабин
ADVANCE 3,2 ; задержка транзакта на 3 ± 2 минуты, моделирующая
; обслуживание абонента
LEAVE PUNKT ; освобождение транзактом одной из телефонных кабин
; по завершении обслуживания
```

2.4.5 Работа с очередями (блоки QUEUE, DEPART).

Пример имитационной модели билетной кассы

Для сбора и обработки статистики по очередям используются блоки QUEUE и DEPART. Блок может быть установлен в любой точке программы, в котором предполагается возникновение очереди, и имеет следующий формат:

QUEUE A,[B]

A – имя очереди, в которую заносится транзакт;

B – число мест в очереди, занимаемых транзактом.

Блок DEPART освобождает требуемое число мест в очереди при вхождении в него транзакта. Формат блока:

DEPART A,[B]

Операнды A и B определяются аналогично блоку QUEUE.

Пример 1. Приведем пример текста ИМ билетной кассы с одним касси-

```
ром.
10 * Сегмент 1
15 * выполняет непосредственное моделирование работы кассы
20 GENERATE 4,3 ; поступление посетителей в
30 * помещение кассы через интервалы времени,
```

35 * имеющие равномерное распределение на интервале [1,7]
40 QUEUE OTCH ;постановка посетителя в очередь
50 SEIZE KASSA ;попытка занять кассу
60 DEPART OTCH ;посетитель покидает очередь
70 ADVANCE 3,2 ;задержка транзакта на время
73 * обслуживания посетителя в среднем 3 ± 2 минуты
75 * с равномерным законом распределения
80 RELEASE KASSA ;освободить устройство «KASSA»
90 TERMINATE ;пассажир покидает систему
95 * Сегмент 2
100 * задает условия остановки моделирования работы билетной кассы
110 GENERATE 480 ;генерация фиктивного транзакта,
115 * указывающего на окончание рабочего дня
118 * (8 час · 60 мин = 480 мин)
120 TERMINATE 1 ;остановить моделирование
130 *
140 START 1 ;запуск процедуры моделирования;
145 * счетчик завершений установить в единицу

2.4.6 Работа со стандартными числовыми характеристиками объектов GPSS

Все объекты ИМ в GPSS определяются СЧА. Имя СЧА объекта состоит из двух частей. Первая часть указывает групповое имя, идентифицирующее тип объекта и тип информации об объекте. Вторая часть идентифицирует конкретного члена группы. Групповое имя состоит из одной-двух букв, фиксированных для информации определенного типа. Например, *Q* – ссылка на текущее значение длины очереди; *QA* – целая часть среднего значения длины очереди и т. д.

Объекты GPSS могут быть идентифицированы с помощью числовых либо символьных имен. Если объект идентифицирован с помощью номера, то ссылка на его СЧА записывается как СЧА_{*j*}, где *j* – номер объекта (целое число). При символической идентификации объекта ссылка на его стандартный атрибут записывается как СЧА\$<имя>, где <имя> – символьное имя объекта.

Например:

FR2 – СЧА содержит (в тысячных долях) коэффициент загрузки обслуживающего устройства 2 (второго устройства, определенного в модели)
FR\$KASSA3 – СЧА содержит (в тысячных долях) коэффициент загрузки обслуживающего устройства с символическим именем KASSA3
Q1 – СЧА содержит длину первой очереди, определенной в модели
QA\$OTCHERED – СЧА хранит значение средней длины очереди с символическим именем OTCHERED
LS\$KEY1 – СЧА содержит состояние логического ключа с символическим именем KEY1
FN\$EXPON – СЧА содержит значение функции с именем EXPON

Кроме того, GPSS позволяет обращаться к СЧА объектов, имена или номера которых записаны в одном из параметров текущего транзакта.

Например:

Q*2 – СЧА содержит значение текущей длины очереди, порядковый номер которой записан во 2-м параметре обрабатываемого транзакта
R\$*4 – количество свободных приборов многоканального обслуживающего устройства, символическое имя которого записано в 4-м параметре обрабатываемого транзакта

Системные числовые атрибуты и СЧА могут использоваться в качестве операндов в практически любом блоке программы. Они обеспечивают пользователю доступ к характеристикам ИМ системы и ее компонентов, а также позволяют управлять процессом моделирования.

2.4.7 Управление перемещением транзактов.

Работа с логическими ключами (блоки TRANSFER, TEST, GATE, LOGIC, LOOP, MATCH)

Блок TRANSFER изменяет маршрут движения транзактов в зависимости от значения первого операнда (в основном или альтернативном направлении) при определенном состоянии оборудования: обслуживающих устройств, очереди, логических ключей и т.п.:

TRANSFER [A],B,[C],[D]

A – режим перехода;

B – метка первого альтернативного блока;

C – метка второго альтернативного блока;

D – константа, используемая для относительной переадресации транзактов.

Если операнд A блока TRANSFER отсутствует, то транзакт, поступивший в блок TRANSFER, безусловно отправляется в блок с меткой, указанной в операнде B.

Если операнд A – число от 0 до 1, то он определяет вероятность перехода транзакта по адресу C. При этом операнд B определяет альтернативный адрес.

Если операнд A равен «BOTH», то транзакт делает попытку перемещения в блок с меткой B. При невозможности войти в блок с адресом B, транзакт перемещается в блок с меткой, указанной в операнде C.

Если операнд A равен «ALL», то транзакт делает попытку перемещения в блок с меткой B. При невозможности войти в данный блок, транзакт делает попытку перемещения в блок, следующий за блоком с адресом B через D блоков, если и эта попытка безуспешна – то в блок, следующий за блоком с адресом B через 2D блоков. Адрес последнего блока, в который может перемещаться транзакт, записывается в операнде C.

Например:

TRANSFER ,LABEL1 ; перемещение транзакта к блоку с меткой LABEL1

TRANSFER 0.6,KASSA2,KASSA1 ; с вероятностью 0,6 транзакт перемещается к блоку с меткой KASSA1, с вероятностью 0,4 – к блоку с меткой KASSA2

TRANSFER BOTH,OTCH1,OTCH2 ; транзакт перемещается к блоку с меткой OTCH1, но в случае невозможности войти в блок с меткой OTCH1, транзакт перемещается к блоку с меткой OTCH2

TRANSFER BOTH,,KASSA2 ; транзакт делает попытку войти в следующий блок, если это ему не удастся, транзакт перемещается к блоку с меткой KASSA2

TRANSFER ALL,FIRST,LAST,3 ; транзакт перемещается к блоку с меткой FIRST, в случае невозможности войти в блок с меткой FIRST, транзакт делает попытку войти в блок, следующий за блоком с меткой FIRST через 3 блока. В случае неудачи – через 6, 9, 12,... блоков, пока не будет достигнут блок с меткой LAST

В GPSS дополнительно используется относительная адресация блоков, не имеющих меток:

TRANSFER ,LABEL1 ; перемещение транзакта к блоку с меткой LABEL1

TRANSFER ,*+5 ; перемещение транзакта к блоку, следующему за данным блоком через 5 блоков

TRANSFER ,LABEL1+2 ; перемещение транзакта к блоку, следующему за блоком с меткой LABEL1 через 5 блоков

Блок TEST определяет направление движения транзакта в зависимости от выполнения условия, заданного алгебраическим соотношением:

TEST XX A,B,[C]

XX – знак логической операции: L – меньше, G – больше, E – равно, LE – меньше или равно, GE – больше или равно, NE – не равно;

A, B – сравниваемые значения или СЧА;

C – метка блока, куда перемещается транзакт в случае невыполнения заданного условия.

При выполнении условия, записанного в блоке TEST, транзакт переходит в следующий блок. В противном случае он направляется в блок с меткой, содержащейся в операнде C. Если операнд C не задан, то транзакт задерживается в блоке TEST до выполнения условия.

Например:

TEST LE Q\$OTH1,Q\$OTH2,LABEL2 ; транзакт переходит к следующему блоку, если текущая длина очереди OTH1 меньше или равна длине очереди OTH2. В противном случае транзакт направляется в блок с меткой LABEL2

TEST G 10,\$\$KASSA ; транзакт переходит к следующему блоку, если 10 больше числа занятых приборов многоканального устройства с именем KASSA. В противном случае транзакт задерживается в данном блоке

Блок GATE разрешает движение транзактам (в основном или альтернативном направлении) при определенном состоянии оборудования: обслуживающих устройств, очереди, логических ключей и т. п. Блок имеет следующий формат:

GATE XXX A,[B]

XXX – логический указатель (определяется таблицей 2);

A – имя или номер оборудования;

B – метка альтернативного блока.

Таблица 2 – Значения логического указателя блока GATE

Тип объекта	Логический указатель	Значение
Одноканальное устройство	FV	Устройство занято
	FNV	Устройство не занято
	FI	Устройство обслуживает прерывание
	FNI	Устройство не обслуживает прерывание
Многоканальное устройство	SF	Устройство заполнено
	SNF	Устройство не заполнено
	SE	Устройство пусто
	SNE	Устройство не пусто
Логический ключ	LR	Логический переключатель сброшен
	LS	Логический переключатель установлен

При выполнении условия, записанного в логическом указателе XXX, транзакт переходит в следующий за GATE блок. В противном случае он направляется в блок с меткой, содержащейся в операнде B. Если операнд B не задан, то транзакт задерживается в блоке GATE до выполнения условия, определяемого указателем XXX.

Например:

GATE SNF ОТК,LABEL1 ; транзакт переходит к следующему блоку, если многоканальное устройство ОТК не заполнено. В противном случае транзакт направляется в блок с меткой LABEL1

GATE FNU ; транзакт перемещается в блок с меткой KASSA2, ОКНО1,KASSA2 ; если одноканальное устройство ОКНО1 занято

GATE LS KEY1 ; транзакт переходит к следующему блоку, если логический ключ KEY1 установлен в единицу. В противном случае транзакт задерживается в блоке GATE

Для изменения в ИМ значений логических ключей используется блок LOGIC, который имеет следующий формат:

LOGIC X A

X – указатель операции с логическим ключом: S – установить (единица), R – сбросить (обнулить), I – инвертировать;

A – имя или номер логического ключа.

Например:

```
LOGIC S KEY1 ; установить единичное значение ключа с именем KEY1
LOGIC R 2 ; обнулить значение второго ключа, определенного в модели
LOGIC I KEY2 ; инвертировать значение ключа с именем KEY2
```

Для организации циклов перемещения транзактов используется блок

LOOP A,B

A – номер параметра транзакта, используемого в качестве счетчика цикла с указанием формата: «слово» (PF), «полуслово» (PH), «байт» (PB);

B – метка блока, являющегося начальным в повторяющейся группе блоков.

При каждом вхождении транзакта в блок значение счетчика уменьшается на 1. Если оно становится равным нулю, транзакт переходит к следующему блоку. Допускается изменение параметра-счетчика внутри цикла.

Например:

```
ASSIGN 5,3,,PH ; присвоить 5-му параметру текущего транзакта значение 3 и формат «полуслово»
LABEL1 ADVANCE 15,10 ; выполнение задержки транзакта на 15±10 единиц модельного времени
LOOP PH5,LABEL1 ; организация цикла (многократной задержки транзакта), число повторений равно 3
```

Для синхронизации движения двух транзактов из семейства используются два сопряженных блока **MATCH**, например:

```
LABEL1 MATCH LABEL2 ; транзакт, вошедший в блок с меткой LABEL1, будет ожидать в этом блоке прихода транзакта того же семейства в блок с меткой LABEL2
...
LABEL2 MATCH LABEL1 ; транзакт, вошедший в блок с меткой LABEL2, будет ожидать в этом блоке прихода транзакта того же семейства в блок с меткой LABEL1
```

2.4.8 Работа с переменными и функциями (операторы VARIABLE, FVARIABLE, BVARIABLE, FUNCTION)

Переменные и функции, используемые в ИМ на GPSS, должны быть предварительно определены. Определение арифметических переменных, арифметических переменных с плавающей точкой и булевских переменных выполняется соответственно операторами:

```
N VARIABLE A
N FVARIABLE A
N BVARIABLE A
```

N – имя или номер переменной;

A – арифметическое или логическое выражение.

Выражениями, используемыми в арифметических и булевских переменных, являются комбинации математических операторов, стандартных функций, СЧА и констант, составленные по правилам математики. В таблице 3 представлены операторы и стандартные функции, используемые в выражениях, в порядке приоритетов их вычисления.

Таблица 3 – Операции и стандартные функции GPSS

Обозначение	Содержание	Обозначение	Содержание
<i>Операции отношения</i>		<i>Логические операции</i>	
'G'	Больше	'NOT'	Логическое отрицание
'L'	Меньше	'AND'	Логическое умножение
'E'	Равно	'OR'	Логическое сложение
'NE'	Не равно	<i>Стандартные функции</i>	
'LE'	Меньше или равно	ABS(●)	Абсолютное значение
'GE'	Больше или равно	ATN(●)	Арктангенс в радианах
<i>Арифметические операции</i>		COS(●)	Косинус в радианах
^	Возведение в степень	INT(●)	Целая часть
#	Умножение	EXP(●)	Экспонента
/	Деление	LOG(●)	Натуральный логарифм
\	Деление нацело	SIN(●)	Синус в радианах
@	Деление по модулю	SQR(●)	Квадратный корень
+	Сложение	TAN(●)	Тангенс в радианах
-	Вычитание		

Обращение к значениям переменных выполняется через их СЧА.

Например:

```
CountAll VARIABLE F$Shlyuz+Q$VerhOtch ; определение переменной, значение которой равно числу транзактов на обслуживающем приборе Shlyuz и в очереди VerhOtch
TimeBetweenTrains FVARIABLE ; определение вещественной переменной, зависящей от длины очереди Ochl
(ABS(1.4\Q$Ochl-3))
TimeSort FVARIABLE 2.88#(1-1/(2#Dir)) ; определение времени сортировки вагона
GoNiz BVARIABLE ; определение логической переменной, значение которой зависит от длины очереди Verh и значения логического ключа Pol
((Q$Verh'E'0)'OR'(LS$Pol'E'0))
...
ASSIGN 3,V$CountAll ; 3-му параметру текущего транзакта присваивается вновь вычисленное значение переменной CountAll
```

```

ADVANCE V$TimeSort           ; задержка транзакта на время
                             ; сортировки вагона, определяе-
                             ; мое переменной TimeSort
Test E BV$GoNiz,1           ; задержка транзакта пока
                             ; логическая переменная GoNiz
                             ; не примет значение 1 (истина)

```

Определение функций выполняется в GPSS с помощью оператора

N FUNCTION A,B

N – имя или номер функции;

A – аргумент функции;

B – указатель типа функции (в частности, D – для дискретной, C – для непрерывной) и числа точек табуляции. За оператором описания функции следует описание множества значений аргумента и функции, которые отделяются друг от друга символом «/».

Определение значения функции выполняется чтением ее СЧА с именем FN.

Например:

```

ProcentNaloga FUNCTION Zarplata,D3 ; определение дискретной функции
6.1,10/9.2,13/12.3,20             с тремя возможными значениями:
                                   10, 13, 20
Output FUNCTION V$Input,C3        ; определение непрерывной функ-
1.1,10/20.5,98.7/33.3,889.2      ции. Аргументом функции служит
                                   значение переменной Input. Про-
                                   межуточные значения определяются
                                   линейной аппроксимацией соседних
                                   значений
...
ASSIGN 1,FN$ProcentNaloga        ; 1-му параметру текущего тран-
                                   закта присваивается вновь вычис-
                                   ленное значение функции Procent-
                                   Naloga
ADVANCE FN$Output                ; задержка транзакта на время,
                                   определяемое функцией Output

```

Значения переменных и функций вычисляются автоматически (уточняются) GPSS всякий раз при поступлении транзактов в блоки, использующие данные переменные или функции.

2.4.9 Работа с константами, ячейками, матрицами сохраняемых величин и таблицами (операторы EQU, INITIAL, MATRIX, TABLE, блоки SAVEVALUE, MSAVEVALUE, TABULATE)

Для наглядности и удобства отладки программы ИМ на GPSS допускает принудительное присвоение числовых значений строковым переменным или именам объектов с помощью оператора

N EQU A

N – имя объекта GPSS или константы;

A – порядковый номер объекта GPSS или числовое значение константы.

Например:

```

Directions EQU 3              ; определение константы Directions, равной 3
KursDollars EQU 2.185        ; определение константы KursDollars и при-
                               ; сваивание ей значения 2,185.

```

GPSS позволяет использовать пользовательские переменные (ячейки и матрицы ячеек), значения которых сохраняются или изменяются в процессе моделирования. Матрица ячеек перед использованием определяется оператором

N MATRIX A,B,C

N – имя матрицы;

A – неиспользуемый параметр;

B – число строк матрицы;

C – число столбцов матрицы.

Значения ячеек, в том числе ячеек матрицы, необходимо проинициализировать в начале текста модели оператором

INITIAL A,B

A – СЧА ячейки;

B – начальное значение.

Например:

```

Initial X$Train1,15          ; определение ячейки Train1 и присваивание
                               ; ей начального значения 15
Initial X$Train2,46         ; определение ячейки Train2 и присваивание
                               ; ей начального значения 46
Initial X3,4.6              ; определение ячейки с номером 3 и присваи-
                               ; вание ей начального значения 4,6
Times Matrix ,3,2           ; определение матрицы ячеек Times с тремя
                               ; строками и 2 столбцами
Initial MX$Times(1,2),5    ; начальное значение ячейки матрицы Times в
                               ; 1-й строке и 2-м столбце равно 5

```

Для изменения значений ячеек используется оператор

SAVEVALUE A[+,-],B

A – номер или имя ячейки с указанием режима изменения: накопление (+), вычитание (-), замещение (без дополнительных символов);

B – величина, используемая для модификации значения ячейки.

Для изменения значений ячеек матрицы используется оператор

MSAVEVALUE A[+,-],B,C,D

A – имя матрицы ячеек с указанием режима изменения: накопление (+), вычитание (-), замещение (без дополнительных символов);

B – номер строки матрицы ячеек;

C – номер строки матрицы ячеек;

D – величина, используемая для модификации значения ячейки.

Например:

```
SAVEVALUE Train1, (P2+P3+P4) ; ячейке Train1 присваивается значение,
равное сумме значений 2-го и 3-го параметров транзакта, вошедшего в блок Savevalue
SAVEVALUE Train2-, X$Value ; значение ячейки Train2 уменьшается на
значение арифметической переменной Value
MSAVEVALUE Times, 1, 2, 0 ; ячейке в 1-й строке 2-го столбца матрицы
Times присваивается значение 0
```

Для накопления выборочных значений случайной величины и статистической обработки выборки используется объект GPSS-таблица, который является аналогом сгруппированного статистического ряда распределения (гистограммы) и описывается оператором

N TABLE A,B,C,D

A – СЧА, значение которого учитывается в таблице;

B – значение правой границы первого интервала сгруппированного статистического ряда распределения (целое число, рисунок 5);

C – ширина интервала сгруппированного статистического ряда распределения (целое число);

D – количество интервалов сгруппированного статистического ряда распределения (целое число, см. рисунок 5).

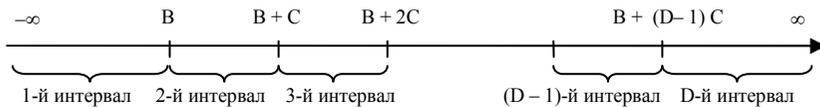


Рисунок 5 – Схема организации интервалов сгруппированного статистического ряда распределения

Выборочные значения попадают в таблицу при входе транзактов в блок TABULATE, формат которого следующий:

TABULATE A

A – имя таблицы, в которой табулируется СЧА, соответствующий оператору TABLE.

Например:

```
DlinaPoezda TABLE X$RealDlina, 10, 4, 20 ; определение таблицы
DlinaPoezda, обрабатывающей значения переменной
RealDlina. Минимальное значение таблицы 10, ширина
интервалов 2, количество интервалов 20
TimeBetween TABLE (AC1-X$Sostav), 4, 2, 20 ; определение таблицы Time-
Between, обрабатывающей значения разности текущего
```

модельного времени и переменной Sostav. Минимальное значение таблицы 4, ширина интервалов 2, количество интервалов 20

```
...
TABULATE DlinaPoezda ; добавление в таблицу
DlinaPoezda нового значения
TABULATE TimeBetween ; добавление в таблицу
TimeBetween нового значения
```

По результатам моделирования предоставляются частоты попадания значений исследуемого СЧА объекта в каждый интервал таблицы, а также основные числовые характеристики выборки (объем, среднее арифметическое значение и т. д.). Числовые характеристики выборок, записанных в таблицы, доступны и в процессе моделирования (см. СЧА таблиц в подразд. 2.3).

Описание блоков GPSS приведено в приложении Б.

2.5 Моделирование случайных величин в GPSS

2.5.1 Моделирование непрерывных случайных величин

Для моделирования случайных величин с заданными законами распределения в GPSS возможно использование библиотечных функций или задание требуемой функции распределения в табличном виде путем аппроксимации непрерывными функциями.

Встроенная библиотека GPSS содержит функции для моделирования случайных величин, имеющих следующие законы распределения:

- равномерное (Uniform);
- экспоненциальное (Exponential);
- гамма (Gamma);
- вейбулловское (Weibull);
- нормальное (Normal);
- логнормальное (LogNormal);
- биномиальное (Binomial);
- геометрическое (Geometric);
- дискретное равномерное (Discrete Uniform);
- пуассоновское (Poisson);
- логистическое (Logistic);
- логлапласово (LogLaplace);
- треугольное (Triangular).

Для моделирования случайной величины, имеющей **равномерное распределение**, используется библиотечная функция

UNIFORM(Stream,Min,Max)

Stream – номер генератора случайных чисел (от 1 до 7);

Min – наименьшее возможное значение;

Max – наибольшее возможное значение.

Например:

GENERATE (UNIFORM(4,2,8)); генерация транзактов через интервалы времени, равномерно распределенные на отрезке [2; 8], с использованием 4-го генератора случайных чисел

ADVANCE (UNIFORM(2,3,5)); задержка транзактов на время, имеющее равномерное распределение на отрезке [3; 5], с использованием 2-го генератора случайных чисел

Аналогичные действия выполняются при следующей записи блоков

GENERATE 5,3 ; генерация транзактов через интервалы времени, равномерно распределенные на отрезке [2; 8]

ADVANCE 4,1 ; задержка транзактов на время, имеющее равномерное распределение на отрезке [3; 5]

Для моделирования случайной величины, имеющей **экспоненциальное распределение**, используется библиотечная функция

EXPONENTIAL(Stream,Locate,Scale)

Stream – номер генератора случайных чисел (от 1 до 7);

Locate – величина сдвига (константа, добавляемая к значению моделируемой величины);

Scale – параметр формы распределения (математическое ожидание случайной величины при Locate =0).

Пример 2. Приведем пример использования функции генерации случайной величины, имеющей экспоненциальное распределение, при моделировании входящего потока транзактов (в данном случае – простейшего потока).

10 INITIAL X\$TimeOld,0 ; инициализация ячейки, сохраняющей время моделирования предыдущего транзакта

20 Inter FVARIABLE AC1-X\$TimeOld ; определение переменной, равной интервалу между текущим модельным временем и временем моделирования предыдущего транзакта

30 Distrib TABLE V\$Inter,0,1,20 ; определение таблицы с 20 интервалами, правая граница 1-го интервала равна 0, ширина интервалов равна 1

40 GENERATE (Exponential(1,0,4)); генерация транзактов через интервалы времени, имеющие экспоненциальное распределение с математиче-

ским ожиданием, равным 4, без сдвига ; добавление в таблицу нового значения времени между поступающими транзактами ; сохранение текущего модельного времени ; уменьшение счетчика завершений на 1 ; задание начального значения счетчика завершений и запуск моделирования

50 TABULATE Distrib ;

60 SAVEVALUE TimeOld,AC1 ;

70 TERMINATE 1 ;

80 START 5000 ;

В результате моделирования в таблице Distrib была накоплена выборка значений случайной величины, имеющей экспоненциальное распределение, объемом 5000 элементов. Гистограмма частот, соответствующая таблице Distrib, полученная средствами GPSS (см. подразд. 2.7), представлена на рисунке 6.

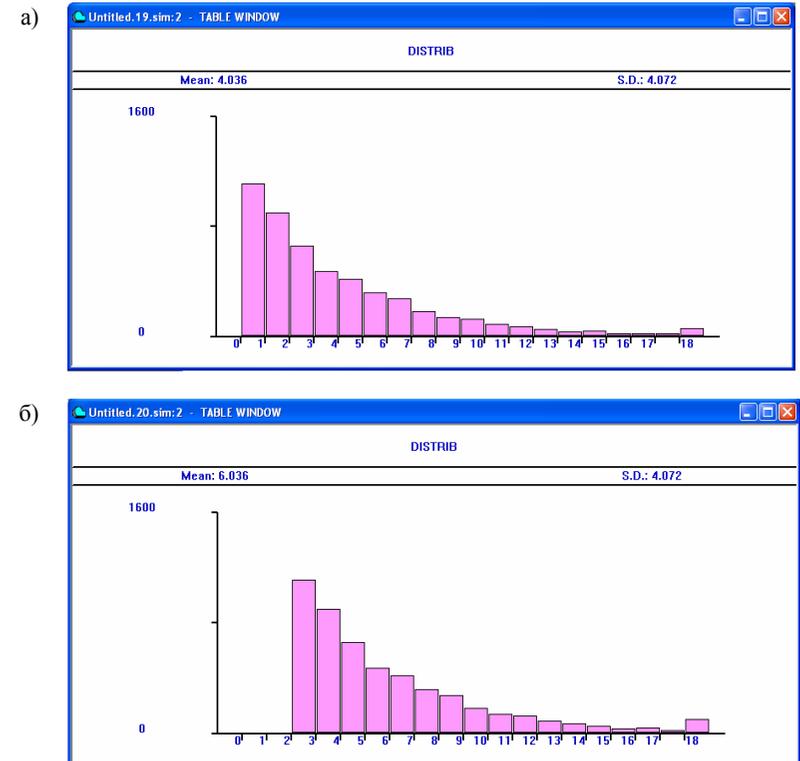


Рисунок 6 – Экранные формы гистограмм, соответствующих таблице Distrib: а – без сдвига (Locate = 0); б – со сдвигом (Locate = 2)

На рисунке 6, б представлена гистограмма частот времени между поступающими транзактами в случае, когда строка 40 ИМ в примере 2 заменена строкой

```
40 GENERATE (Exponential(1,2,4)); генерация транзактов через интервалы времени, имеющие экспоненциальное распределение с математическим ожиданием, равным 4, со сдвигом на 2 единицы вправо
```

Для моделирования случайной величины, имеющей **гамма-распределение**, используется библиотечная функция

GAMMA(Stream,Locate,Scale,Shape)

Stream – номер генератора случайных чисел (от 1 до 7);

Locate – величина сдвига (константа, добавляемая к значению моделируемой величины, см. выше);

Scale – параметр масштаба функции распределения;

Shape – параметр, определяющий форму кривой гамма-распределения.

Если аргумент Shape функции GAMMA – есть целое число, то гамма-распределение вырождается в **распределение Эрланга** k -го порядка, где $k = \text{Shape}$. А если $\text{Shape} = 1$, то гамма-распределение совпадает с **экспоненциальным распределением**.

Для моделирования случайной величины, имеющей **вейбулловское распределение** (распределение **Вейбулла-Гнеденко**), используется библиотечная функция

WEIBULL(Stream,Locate,Scale,Shape)

Stream – номер генератора случайных чисел (от 1 до 7);

Locate – величина сдвига (константа, добавляемая к значению моделируемой величины, см. выше);

Scale – параметр масштаба функции распределения;

Shape – параметр, определяющий форму распределения.

Вейбулловское распределение широко используется в теории надежности для описания времени безотказной работы систем на различных этапах их эксплуатации. Так в период приработки, когда интенсивность отказов систем уменьшается, $\text{Shape} < 1$; в период нормальной эксплуатации $\text{Shape} = 1$; в период старения, когда интенсивность отказов системы со временем возрастает, $\text{Shape} > 1$.

Для моделирования случайной величины, подчиняющейся **нормальному закону** распределения, используется библиотечная функция

NORMAL(Stream,Mean,StdDev)

Stream – номер генератора случайных чисел (от 1 до 7);

Mean – математическое ожидание;

StdDev – среднеквадратическое отклонение.

Для генерации случайной величины, имеющей **логнормальный закон распределения**, используется библиотечная функция

LOGNORMAL(Stream,Locate,Scale,Shape)

Stream – номер генератора случайных чисел (от 1 до 7);

Locate – величина сдвига (константа, добавляемая к значению моделируемой величины, см. выше);

Scale – параметр масштаба функции распределения;

Shape – параметр, определяющий форму распределения.

2.5.2 Моделирование дискретных случайных величин

Для моделирования дискретной случайной величины, имеющей **биномиальный закон** распределения, используется библиотечная функция

BINOMIAL(Stream,TrialCount,Probability)

Stream – номер генератора случайных чисел (от 1 до 7);

Locate – величина сдвига (константа, добавляемая к значению моделируемой величины, см. выше);

TrialCount – число испытаний Бернулли;

Probability – вероятность успеха в каждом испытании.

Для моделирования дискретной случайной величины, имеющей **пуассоновский закон** распределения, используется библиотечная функция

POISSON(Stream,Mean)

Stream – номер генератора случайных чисел (от 1 до 7);

Mean – математическое ожидание.

Пример 3. Приведем пример использования функции генерации случайной величины, имеющей пуассоновское распределение, при моделировании потока поездов с некоторым числом вагонов.

```
m_1 EQU 20 ; средний интервал между прибытием поездов
m_2 EQU 30 ; среднее количество вагонов в поезде
Generate (Exponential(1,0,m_1)) ; генерация поступления поезда
Split ((poisson(1,m_2))-1) ; разбиение отцепа на отдельные вагоны
; на данном этапе моделируются группы транзактов, образующие простейший поток. Число транзактов в каждой группе имеет пуассоновское распределение
```

2.5.3 Моделирование случайных величин с произвольным законом распределения

Законы распределения далеко не всех реальных случайных величин можно описать типовыми функциями распределения (например, время нахождения автомобиля под погрузкой, время техосмотра). Для моделирова-

ния случайной величины, имеющей произвольный закон распределения, необходимо непосредственно в тексте ИМ задать требуемую функцию распределения, используя оператор FUNCTION.

Пример 4. Рассмотрим моделирование дискретной случайной величины, характеризующей грузоподъемность поступающих на станцию автомобилей, которая описывается следующим рядом распределения (таблица 4).

Таблица 4 – Ряд распределения грузоподъемности автомобилей

Грузоподъемность, т	4	7,5	9	12	15	18
Относительная частота появления	0,2	0,3	0,1	0,2	0,15	0,05
Накопленная относительная частота	0,2	0,5	0,6	0,8	0,95	1,0

Представим текст ИМ, которая моделирует поступление автомобилей, соответствующей грузоподъемности, а также столбцовую диаграмму полученных экспериментальных значений (рисунок 7).

```
Distr FUNCTION RN1,D6 ; задание функции распределения
.2,4/.5,7/.6,9/.8,12/.95,15/1,18 дискретной случайной величины
Distrib TABLE P1,4,1,17
Generate (Exponential(1,0,20)) ; генерация транзактов (автомобилей)
ASSIGN 1, FN$Distr ; задание грузоподъемности автомобиля указанием соответствующего значения в 1-м параметре транзакта
TABULATE Distrib ; сохранение грузоподъемности автомобилей в таблице
TERMINATE 1
START 1000 ; моделирование 1000 автомобилей
```

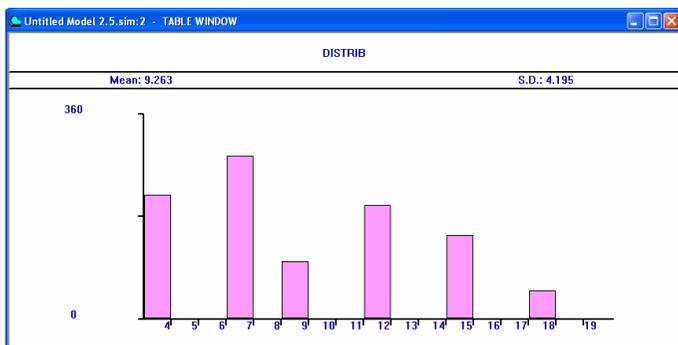


Рисунок 7 – Столбцовая диаграмма грузоподъемности автомобилей

Пример 5. Поясним принцип моделирования непрерывных случайных величин на примере экспоненциального распределения. Функцию распределения предварительно протабулируем для 24 точек и опишем с помощью задания значений непрерывной функции EXDIS. Для других значений (кроме 24 описанных) функция линейно интерполируется. Остальной текст ИМ аналогичен тексту из примера 2 (см. п. 2.5.1).

```
ExpDistr FUNCTION RN1,C24 ; задание функции экспоненциального распределения непрерывной случайной величины в 24 точках
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
10 INITIAL X$TimeOld,0 ; инициализация ячейки, сохраняющей время моделирования предыдущего транзакта
20 Inter FVARIABLE AC1-X$TimeOld ; определение переменной, равной интервалу между текущим модельным временем и временем моделирования предыдущего транзакта
30 Distrib TABLE V$Inter,0,2,20 ; определение таблицы с 20 интервалами, правая граница 1-го интервала равна 0, ширина интервалов равна 2
40 GENERATE 8, FN$ExpDistr ; генерация транзактов через интервалы времени, имеющие экспоненциальное распределение с математическим ожиданием, равным 8 единицам
50 TABULATE Distrib ; добавление в таблицу нового значения времени между поступающими транзактами
60 SAVEVALUE TimeOld,AC1 ; сохранение текущего модельного времени
70 TERMINATE 1 ; уменьшение счетчика завершений на 1
80 START 5000 ; задание начального значения счетчика завершений и запуск моделирования
```

Полученная в результате моделирования гистограмма значений генерируемой случайной величины представлена на рисунке 8.

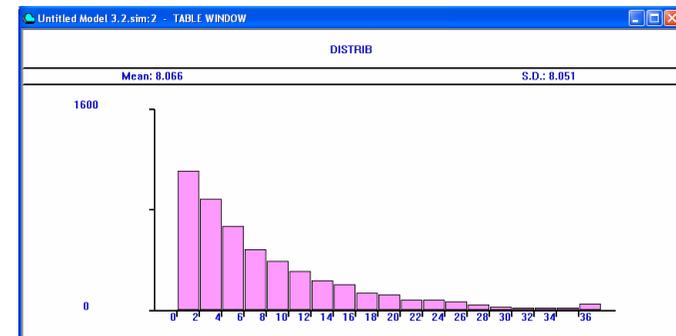


Рисунок 8 – Гистограмма частот случайной величины

2.5.4 Моделирование случайных событий

Часто условием перемещения транзакта в один из двух альтернативных блоков является наступление некоторого случайного события. В GPSS для этого предусмотрен блок TRANSFER (см. п. 2.4.7).

Пример 6. После технического осмотра в конце рабочего дня в среднем 20 % автобусов поступают для выполнения ремонта, а 80 % - в гараж.

```

...
TRANSFER .2,GARAG,REMONT ; с вероятностью 0,2 транзакты
(автобусы) перемещаются к метке
REMONT и с вероятностью 0,8 - к
метке GARAG
...
REMONT QUEUE OCH_REM ; постановка транзактов в очередь
для ремонта
...
GARAG QUEUE OCH_GAR ; постановка транзактов в очередь
гаража
...

```

Если в процессе моделирования необходимо переходить в различные блоки программы более, чем в двух направлениях, необходимо использовать дискретную переключающую функцию, разыгрывающую выбор одного из направлений. Переход осуществляется блоком TRANSFER в режиме безусловного перехода (см. п. 2.4.7).

Пример 7. После технического осмотра 5 % автомобилей поступают для выполнения сложного ремонта, 10 % – для ремонта средней сложности, 15 % – для мелкого ремонта, а 70 % – в гараж.

```

Perek1 FUNCTION RN4,D5 ; определение дискретной переключающей
0.05,Metka1/0.15,Metka2 функции
0.3,Metka3/1,Metka4

```

```

...
TRANSFER ,FN$Perek1; перемещение транзактов в одном из 4
направлений
...
Metka1 QUEUE OCH_REM1 ; постановка транзактов в очередь для
сложного ремонта (5%)
...
Metka2 QUEUE OCH_REM2 ; постановка транзактов в очередь для
ремонта (10%)
...
Metka3 QUEUE OCH_REM3 ; постановка транзактов в очередь для
мелкого ремонта (15%)
...
Metka4 QUEUE OCH_GAR ; постановка транзактов в очередь гаража
(70%)
...

```

2.6 Трансляция и запуск имитационной модели в GPSS

Подробную информацию о назначении блоков и примеры их использования можно получить в разделе справки GPSS (пункт «Help → Help Topics» главного меню) или при вставке блоков с помощью пункта главного меню «Edit → Insert GPSS Blocks...» (рисунок 9). В данном режиме пользователю предоставляется возможность указывать значения параметров блоков в диалоговом окне.

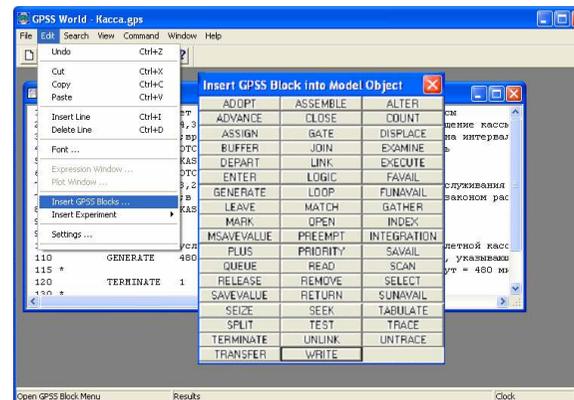


Рисунок 9 – Вставка блоков GPSS в диалоговом режиме

После написания текста ИМ на языке GPSS модель необходимо *оттранслировать*, т.е. преобразовать последовательность текстовых операторов в последовательность команд, написанную в машинных кодах компьютера. Для этого необходимо выбрать пункт создания модели «Command → Create Simulation» главного меню (рисунок 10).

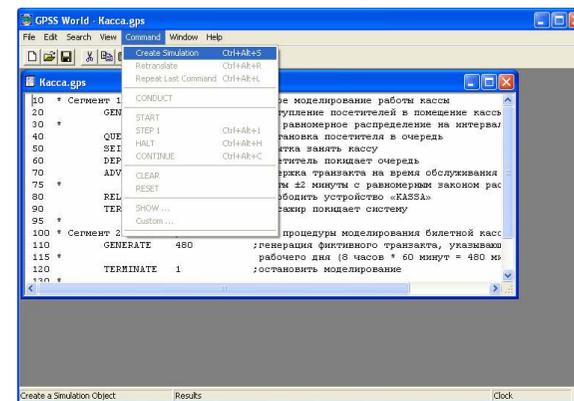


Рисунок 10 – Трансляция текста имитационной модели

Если при трансляции в тексте ИМ будут обнаружены некорректные директивы, значения параметров или синтаксические ошибки, то в появившемся окне документирования событий («JOURNAL») GPSS приведет список ошибочных строк. Для повторной трансляции модели после исправления ошибок используется пункт главного меню «Command → Retranslate».

После успешной трансляции GPSS сообщит о готовности модели к работе строкой «Ready» (рисунок 11), и пользователю станут доступны команды управления моделированием главного меню «Command» (см. рисунок 10), которые описаны в п. 2.4.2.

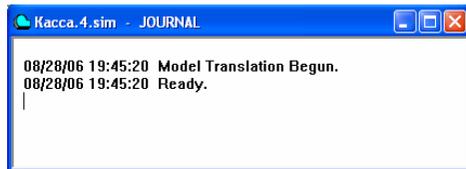


Рисунок 11 – Окно документирования событий GPSS

Для запуска процедуры моделирования в конце текста ИМ следует задать команду «START A» (см. п. 2.4.2) или выбрать пункт главного меню «Command → START» и в диалоговом окне указать начальное значение счетчика завершенных TGI.

2.7 Средства отладки имитационных моделей

GPSS имеет в своем составе развитые средства отладки ИМ, доступ к которым осуществляется из пункта главного меню «Window → Simulation Window» (рисунок 12).

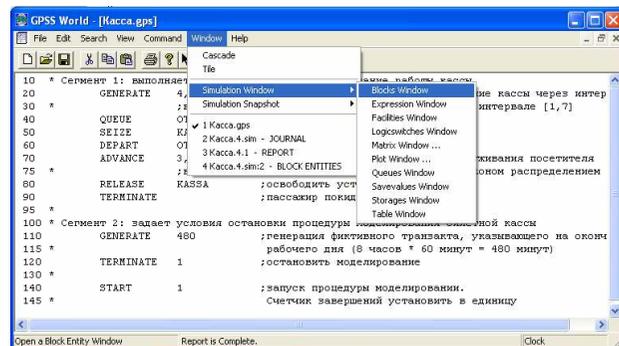


Рисунок 12 – Доступ к диалоговым окнам отладки ИМ

GPSS реализует пошаговую отладку модели с одновременным отображением **процесса перемещения транзактов между блоками ИМ** в окне «BLOCK ENTITIES». Для этого в главном меню необходимо выбрать пункт «Window → Simulation Window → Block Window» (см. рисунок 12).

Для управления процессом моделирования в панели инструментов окна «BLOCK ENTITIES» (рисунок 13) предусмотрены кнопки «Continue», «Halt» и «Step», назначение которых соответствует командам, рассмотренным в п. 2.4.2.

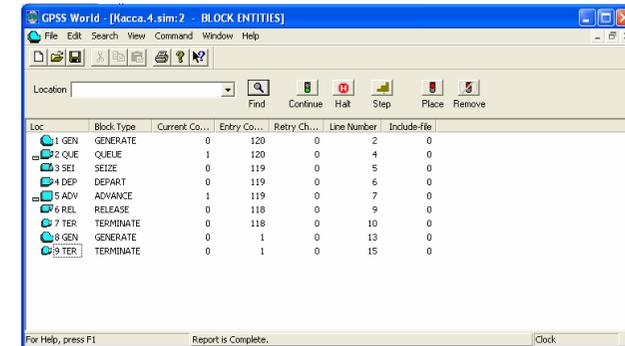


Рисунок 13 – Окно отображения блоков ИМ

В окне «BLOCK ENTITIES» (см. рисунок 13) указывается следующая информация о блоках ИМ:

Loc – указывает номер строки, связанный с блоком ИМ;

Block Type – тип блока GPSS;

Current Count – количество транзактов, находящихся в данном блоке и ожидающих выполнения некоторых условий;

Entry Count – количество транзактов, вошедших в данный блок, после последнего выполнения команд RESET или CLEAR или с начала процедуры моделирования;

Retry Chain – количество транзактов, ожидающих выполнения некоторых условий;

Line Number – номер строки ИМ, которая описывает блок.

Для отображения в модельном времени текущих значений **некоторых арифметических или логических выражений** в главном меню GPSS следует выбрать пункт «Window → Simulation Window → Expression Window» (см. рисунок 12). В поле «Label» появившегося диалогового окна «Edit Expression Window» (рисунок 14) следует указать комментарий, а в поле «Expression» – интересующее арифметическое или логическое выражение в соответствии с правилами, рассмотренными в п. 2.4.8.

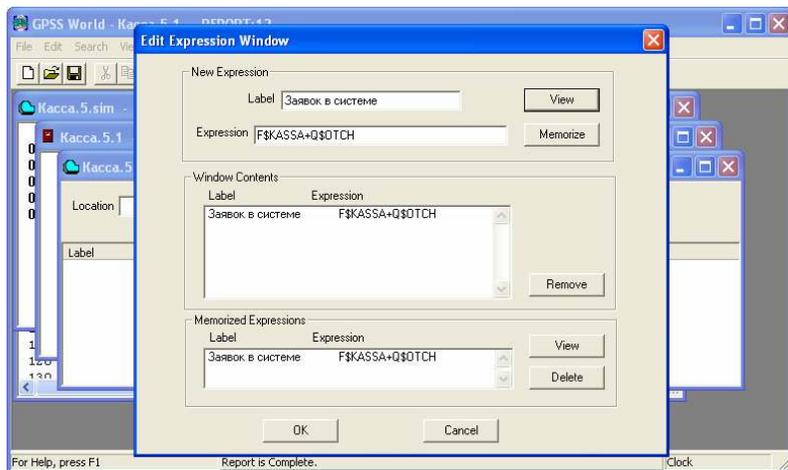


Рисунок 14 – Диалоговое окно задания арифметических или логических выражений

Записанное выражение следует добавить в список кнопкой «View» и сохранить (для возможного использования в будущем) кнопкой «Memorize» (см. рисунок 14). По окончании ввода следует нажать «OK». Текущие значения исследуемых арифметических или логических выражений отобразятся в окне «EXPRESSIONS» в виде списка (рисунок 15), где

Label – комментарий;

Expression – арифметическое или логическое выражение;

Value – текущее значение выражения.

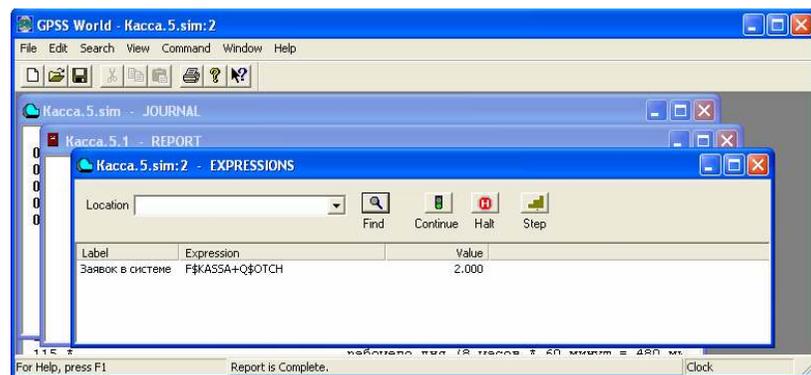


Рисунок 15 – Окно отображения текущих значений задания арифметических или логических выражений

Для управления процессом моделирования в панели инструментов окна «EXPRESSIONS» (см. рисунок 15) предусмотрены кнопки «Continue», «Halt» и «Step», назначение которых соответствует командам, рассмотренным в п. 2.4.2.

Для отображения в модельном времени текущих значений характеристик одноканальных обслуживающих устройств, использованных в ИМ, следует выбрать пункт «Window → Simulation Window → Facilities Window» главного меню GPSS (см. рисунок 12).

Приведем описание характеристик одноканальных обслуживающих устройств, указываемых в окне «FACILITY ENTITIES» (рисунок 16):

Facility – номер или имя одноканального устройства;

Utilization – часть периода моделирования, в течение которого устройство было занято (коэффициент загрузки);

Delay Chain – количество транзактов, ожидающих занятия устройства (включая транзакты, ожидающие освобождение устройства в режиме прерывания);

Acquisitions – количество транзактов, вошедших в устройство после последнего выполнения команды RESET или CLEAR или начала работы программы;

Available – состояние готовности устройства;

Ave. Time – среднее время занятости устройства одним транзактом в течение процедуры моделирования после последнего выполнения команд CLEAR или RESET;

Owner XN – номер последнего транзакта, занимавшего устройство (0 означает, что устройство не занималось);

Retry Chain – количество транзактов, ожидающих выполнения некоторых условий;

Pending Chain – количество транзактов, ожидающих устройство (находящееся в режиме прерывания);

Interrupt Chain – количество транзактов, обработка которых прервана на устройстве в данный момент модельного времени.

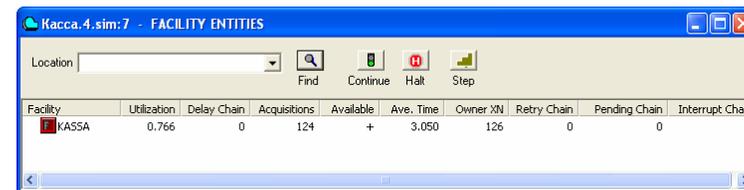


Рисунок 16 – Окно отображения текущих характеристик обслуживающих устройств

Для отображения в модельном времени текущих значений логических ключей, использованных в ИМ, следует выбрать пункт «Window → Simulation Window → Logicswitches Window» главного меню GPSS (см. рисунок 12).

В появившемся окне «LOGICSWITCH ENTITIES» (рисунок 17) указываются следующие текущие характеристики ключей:

Logicswitch – имя или номер объекта типа «логический ключ»;

Value – значение логического ключа;

Retry Chain – количество транзактов, ожидающих выполнения некоторых условий.

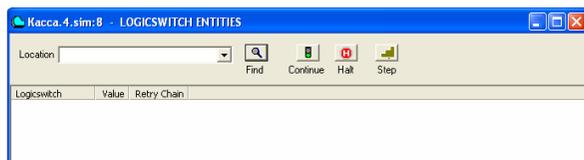


Рисунок 17 – Окно отображения текущих значений логических ключей

Для отображения в динамике значений некоторых выражений или СЧА объектов следует выбрать пункт построения временных диаграмм «Window → Simulation Window → Plot Window» главного меню GPSS (см. рисунок 12).

В появившемся диалоговом окне «Edit Plot Window» (рисунок 18) следует указать комментарий (поле «Label») и интересующее арифметическое или логическое выражение (поле «Expression»), затем добавить его в список кнопкой «Plot». В поле «Title» следует задать имя временной диаграммы, в поле «Time Range» – величину отображаемого модельного времени, в полях «Min Value» и «Max Value» – минимальное и максимальное значения отображаемой величины. По завершении нажать кнопку «OK».

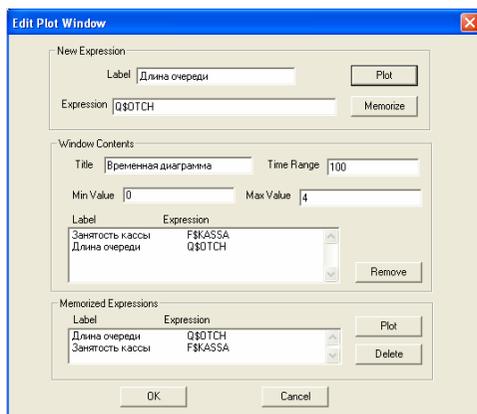


Рисунок 18 – Диалоговое окно задания временных диаграмм

В открывшемся окне могут одновременно отображаться несколько временных диаграмм (рисунок 19).

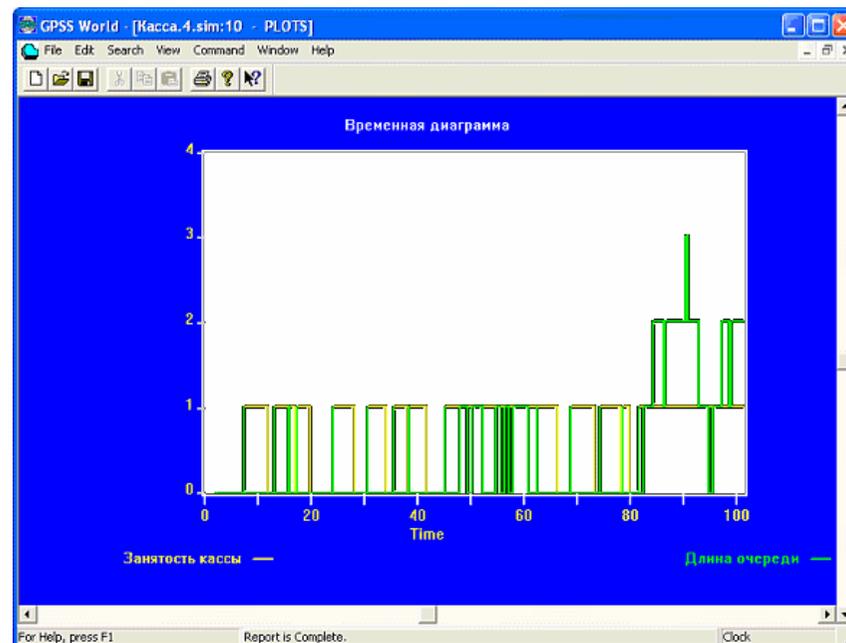


Рисунок 19 – Окно построения временных диаграмм



Рисунок 20 – Окно построения временных диаграмм

Для отображения гистограммы частот значений случайной величины, накопленной в одной из таблиц ИМ, в главном меню GPSS следует выбрать пункт «Window → Simulation Window → Table Window». Затем в диалоговом окне (рисунок 20) выбрать имя нужной таблицы. Примеры гистограмм GPSS приведены на рисунках 6–8.

Аналогичным образом из пункта главного меню «Window → Simulation Window» (см. рисунок 12) пользователь может получить доступ к текущим значениям ячеек матриц, таблиц и сохраняемых величин, к характеристикам очередей и многоканальных обслуживающих устройств.

2.8 Анализ результатов моделирования в GPSS

После завершения процедуры моделирования GPSS автоматически открывает окно отчета «REPORT» (рисунок 21), содержащее стандартную выходную информацию о результатах моделирования.

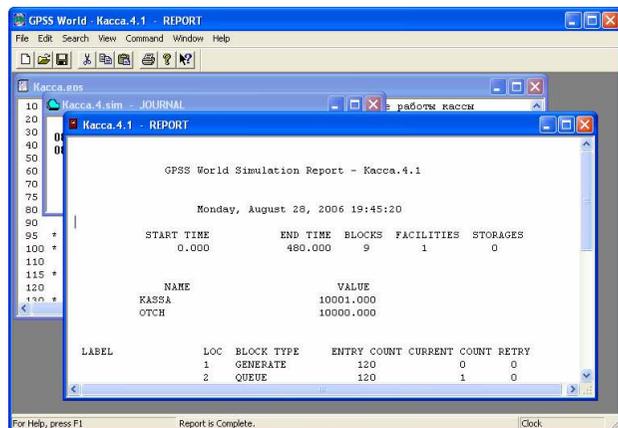


Рисунок 21 – Окно результатов моделирования в GPSS

Выходной файл статистики состоит из подразделов, содержащих стандартную статистику об объектах GPSS, используемых в данной модели (устройствах, очередях и т. д.). Начинается файл статистики с заголовка, который содержит имя модели, дату и время моделирования. Далее следует выходная информация, содержащая следующие основные сегменты вывода:

Информацию о результатах работы модели

START TIME	END TIME	BLOCKS	FACILITIES	STORAGEES
------------	----------	--------	------------	-----------

Здесь START TIME – абсолютное модельное время в момент начала моделирования (эквивалентно абсолютному модельному времени, после последнего применения команды RESET или CLEAR);

END TIME – абсолютное время или момент, когда счетчик завершений принял значение 0;

BLOCKS – количество блоков, использованных в текущей модели, к моменту завершения моделирования;

FACILITIES – количество одноканальных устройств, использованных в модели к моменту завершения моделирования;

STORAGEES – количество многоканальных устройств, использованных в текущей модели к моменту завершения моделирования.

Информация об именах

NAME	VALUE
------	-------

Здесь поле NAME содержит имена, используемые в программе модели;

VALUE – определяет числовое значение (номер), соответствующее имени, устанавливает начальный номер GPSS равным 10000.

Информация о блоках текущей модели

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

Здесь LABEL – метка оператора, связанного с блоком GPSS;

LOC – номер строки модели, связанной с блоком;

BLOCK TYPE – тип блока GPSS;

ENTRY COUNT – количество транзактов, вошедших в данный блок после последнего выполнения команд RESET или CLEAR или с начала процедуры моделирования;

CURRENT COUNT – количество транзактов, находящихся в данном блоке и ожидающих выполнения некоторых условий;

RETRY – количество транзактов, ожидающих выполнения некоторых условий.

Информация об одноканальных устройствах

FACILITY	ENTRIES	UTIL.	AVE. TIME	AVAIL.	OWNER	PEND	INTER	RETRY	DELAY
----------	---------	-------	-----------	--------	-------	------	-------	-------	-------

Здесь FACILITY – номер или имя одноканального устройства;

ENTRIES – количество транзактов, вошедших в устройство после последнего выполнения команды RESET или CLEAR или начала работы программы;

UTIL. – часть периода моделирования, в течение которого устройство было занято (коэффициент загрузки);

AVE. TIME – среднее время занятости устройства одним транзактом в течение процедуры моделирования после последнего выполнения команд CLEAR или RESET;

AVAILABLE – состояние готовности устройства в конце периода моделирования;

OWNER – номер последнего транзакта, занимавшего устройство (0 означает, что устройство не занималось);

PEND – количество транзактов, ожидающих устройство (находящееся в режиме прерывания);

INTER – количество транзактов, обработка которых прервана на устройстве в данный момент модельного времени;

RETRY – количество транзактов, ожидающих выполнения некоторых условий;

DELAY – количество транзактов, ожидающих занятия устройства (включая транзакты, ожидающие освобождение устройства в режиме прерывания).

Информация об очередях

QUEUE	MAX	CONT	ENTRIES	ENTRIES(0)	AVE.CONT	AVE.TIME	AVE.(-0)	RETRY
-------	-----	------	---------	------------	----------	----------	----------	-------

Здесь QUEUE – имя или номер объекта типа «очередь»;

MAX – максимальное содержимое объекта типа «очередь» в течение периода моделирования, который начинается с начала работы или с последней команды RESET или CLEAR;

CONT – текущее содержимое объекта типа «очередь» в момент завершения моделирования;

ENTRIES – общее количество входов в очередь в течение периода моделирования (счетчик входов);

ENTRIES(0) – общее количество входов в очередь с нулевым временем ожидания (счетчик «нулевых» входов);

AVE.CONT – среднее значение длины очереди;

AVE.TIME – среднее время, проведенное транзактом в очереди с учетом всех входов в очередь;

AVE.(-0) – среднее время, проведенное транзактом в очереди без учета «нулевых» входов в очередь;

RETRY – количество транзактов, ожидающих специальных условий, зависящих от состояния объекта типа «очередь».

Информация о многоканальных устройствах

STORAGE	CAP.	REMAIN	MIN	MAX	ENTRIES	AVL.	AVE.C.	UTIL.	RETRY	DELAY
---------	------	--------	-----	-----	---------	------	--------	-------	-------	-------

Поля имеют следующие назначения:

STORAGE – имя или номер многоканального устройства;

CAP. – количество каналов, заданное оператором STORAGE;

REMAIN – число свободных каналов в конце периода моделирования;

MIN – минимальное количество использовавшихся каналов за период моделирования;

MAX – максимальное количество использовавшихся каналов за период моделирования;

ENTRIES – количество входов в многоканальное устройство за период моделирования;

AVL. – состояние готовности многоканального устройства в конце периода моделирования;

AVE.C. – среднее число занятых каналов в устройстве за весь период моделирования;

UTIL. – часть периода моделирования, в течение которого многоканальное устройство использовалось;

RETRY – количество транзактов, ожидающих специальные условия, зависящие от состояния устройства;

DELAY – количество транзактов, ожидающих возможность входа в блок ENTER.

Информация о ячейках памяти

SAVEVALUE	VALUE	RETRY
-----------	-------	-------

Поля имеют назначения:

SAVEVALUE – имя или номер ячейки;

VALUE – значение ячейки в конце моделирования;

RETRY – количество транзактов, ожидающих наступления специальных условий, зависящих от состояния ячейки.

Список будущих событий

XN	PRI	BDT	ASSEM	CURRENT	NEXT	PARAMETER	VALUE
----	-----	-----	-------	---------	------	-----------	-------

XN – номер транзакта, ожидающего выполнения некоторого события;

PRI – приоритет транзакта;

BDT – время назначенного события, связанного с данным транзактом;

ASSEM – номер семейства транзактов;

CURRENT – номер блока, в котором находится транзакт (0 – если транзакт еще не вошел ни в один из блоков);

NEXT – номер блока, в который должен войти транзакт;

PARAMETER – номер или имя параметра транзакта;

VALUE – значение параметра.

3 ПРИМЕРЫ ПОСТРОЕНИЯ И ИССЛЕДОВАНИЯ МОДЕЛЕЙ ТЕХНОЛОГИЧЕСКИХ СИСТЕМ. ЗАДАНИЯ ДЛЯ САМОСТОЯТЕЛЬНОЙ РАБОТЫ

3.1 Пример моделирования двух стратегий обслуживания на АЗС

Для иллюстрации этапов разработки и исследования ИМ технических и технологических систем, представленных в предыдущих разделах, рассмотрим два примера, связанных с анализом эффективности функционирования автозаправочной станции (АЗС) и железнодорожной билетной кассы.

Представим **содержательное описание объекта моделирования** в соответствии с п. 1.5.1. Автозаправочная станция имеет 2 бензоколонки. Интервалы времени между поступлением автомобилей имеют экспоненциальное распределение со средним значением μ . Время заправки автомобилей имеет равномерное распределение на интервале $[a, b]$.

Предлагается две стратегии обслуживания прибывающих автомобилей:

1) автомобили образуют две очереди и обслуживаются соответствующими бензоколонками;

2) автомобили образуют одну общую очередь и обслуживаются освободившейся бензоколонкой.

Исходные данные: $\mu = 1,75$ мин, $a = 2$ мин, $b = 4$ мин.

Целью моделирования является определение:

- характеристик качества обслуживания автомобилей, в частности, средних длин очередей; среднего времени заправки автомобиля; среднего времени пребывания автомобиля на АЗС;

- наилучшей стратегии обслуживания автомобилей на автозаправочной станции;

- оптимального количества бензоколонок.

Параметрами моделирования (управляющими характеристиками) являются интенсивность поступления автомобилей на АЗС (интенсивность входящего потока), характеристики времени заправки автомобилей и количество бензоколонок. Переменными моделирования (контролируемыми характеристиками) объекта являются: количество поступивших и обслуженных автомобилей, коэффициенты загрузки бензоколонок, среднее время обслуживания автомобилей, средняя длина очереди, среднее время ожидания в очереди и т. п. В качестве показателей эффективности функционирования исследуемой системы, определяющих цели моделирования, будут выступать перечисленные выше характеристики.

Ограничения модели связаны:

- с условиями, накладываемыми на поток поступающих автомобилей (он предполагается простейшим, отсутствуют повторные заявки);
- упрощенным алгоритмом функционирования АЗС и выбора бензоколонки для обслуживания;
- отсутствием явлений, изменяющих закономерности времени обслуживания автомобилей (технологические перерывы, отказы и сбои оборудования, отсутствие топлива, ошибки при заправке и при денежных расчетах, количество и тип топлива).

Составим **концептуальную модель** работы автозаправочной станции в соответствии с п. 1.5.2. Элементами исследуемой системы являются: входящий поток автомобилей, две бензоколонки и очереди. На рисунке 22 приведены концептуальные модели АЗС, соответствующие двум стратегиям обслуживания автомобилей.

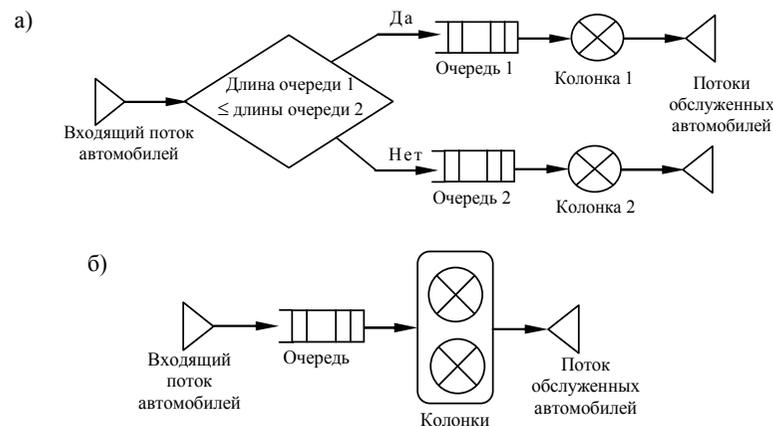


Рисунок 22 – Схема обслуживания автомобилей на АЗС:

a – автомобили образуют две очереди;

b – автомобили образуют одну общую очередь

В качестве критериев, используемых для сравнения стратегий обслуживания автомобилей, выберем:

- коэффициенты загрузки системы;
- максимальные и средние длины очередей;
- средние значения времени ожидания обслуживания.

Выполним этап **формализации** функционирования автозаправочной станции в соответствии с п. 1.5.3. Так как исследуемая система – система массового обслуживания, в которой все события инициируются перемеще-

нием автомобилей – динамических объектов (на территорию АЗС, в очередь, на обслуживание, с АЗС), то в данном случае целесообразно использовать транзактный способ организации квазипараллелизма компонентов модели.

Для формализации АЗС используем два типа элементов: обслуживающие устройства и очереди. Каждому поступающему автомобилю соответствует транзакт. Для создания или уничтожения транзактов, изменение маршрута их движения, задержки транзактов на некоторый интервал времени будем использовать дополнительные фиктивные блоки (рисунок 23).

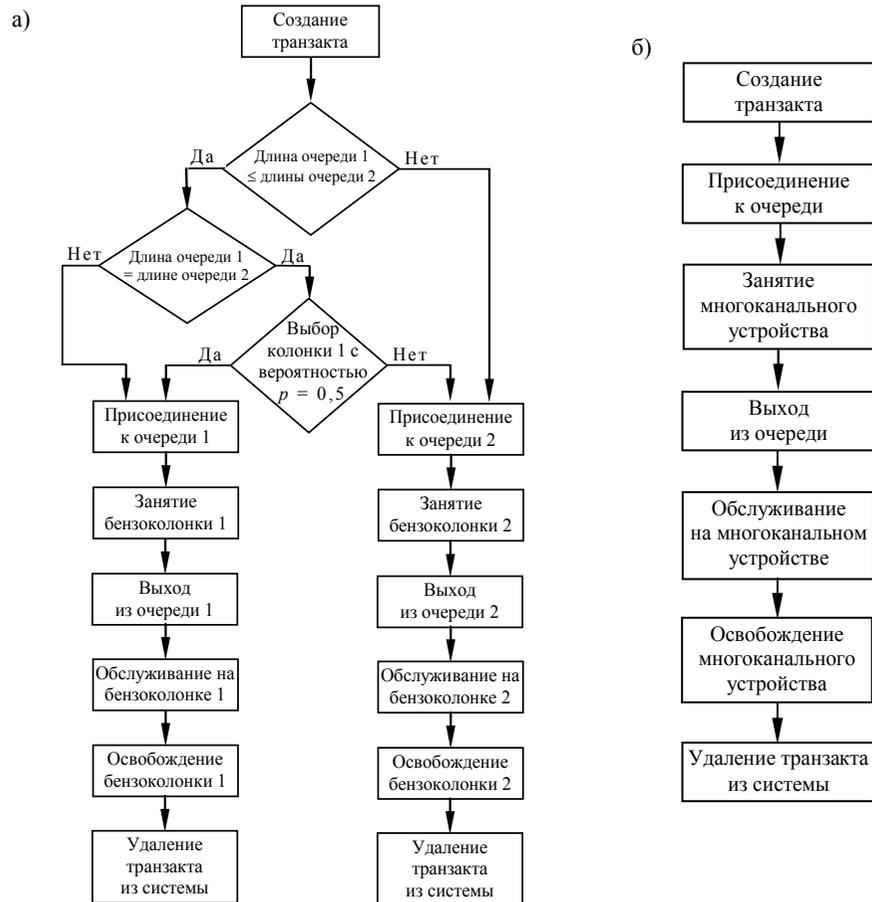


Рисунок 23 – Уточненная схема обслуживания автомобилей на АЗС:
 а – автомобили образуют две очереди;
 б – автомобили образуют одну общую очередь

Составим модель на языке моделирования в соответствии с п. 1.5.4.

В качестве средства автоматизации имитационного моделирования автозаправочной станции будем использовать GPSS World.

Приведем текст ИМ для первой стратегии обслуживания, когда прибывающие автомобили образуют две очереди и обслуживаются соответствующими бензоколонками:

```

GENERATE (Exponential(1,0,1.75)) ; прибытие автомобилей
TEST LE Q$Other1,Q$Other2,Obsl_2 ; длина очереди 1
* <= длины очереди 2
* TEST E Q$Other1,Q$Other2,Obsl_1 ; длина очереди 1
* = длине очереди 2
TRANSFER 0.5,Obsl_1,Obsl_2 ; длины очередей равны,
* выбираем колонку наудачу

* моделирование работы бензоколонки 1
Obsl_1 QUEUE Other1 ; присоединение к очереди 1
SEIZE Kolonka1 ; занятие бензоколонки 1
DEPART Other1 ; выход из очереди 1
ADVANCE 3,1 ; обслуживание на бензоколонке 1
RELEASE Kolonka1 ; освобождение бензоколонки 1
TERMINATE ; автомобиль покидает систему

* моделирование работы бензоколонки 2
Obsl_2 QUEUE Other2 ; присоединение к очереди 2
SEIZE Kolonka2 ; занятие бензоколонки 2
DEPART Other2 ; выход из очереди 2
ADVANCE 3,1 ; обслуживание на бензоколонке 2
RELEASE Kolonka2 ; освобождение бензоколонки 2
TERMINATE ; автомобиль покидает систему

* задание условия остановки процедуры моделирования АЗС
GENERATE 10080 ; генерация фиктивного транзакта,
* указывающего на окончание рабочего дня
* (7 дней · 24 часа · 60 мин = 10080 мин)
TERMINATE 1 ; остановить моделирование
START 1 ; запуск процедуры моделирования
  
```

В результате моделирования получен следующий отчет:

FACILITY	ENTRIES	UTIL.	AVE.TIME	AVAIL.	OWNER	PEND	INTER	RETRY	DELAY
KOLONKA2	2875	0.857	3.004	1	5767	0	0	0	0
KOLONKA1	2891	0.859	2.994	1	0	0	0	0	0

QUEUE	MAX	CONT.	ENTRY	ENTRY(0)	AVE.CONT.	AVE.TIME	AVE.(-0)	RETRY
OTHER1	11	0	2891	498	1.488	5.189	6.268	0
OTHER2	12	0	2875	496	1.513	5.304	6.410	0

Приведем текст ИМ для второй стратегии обслуживания, когда прибывающие автомобили образуют одну очередь и обслуживаются освободившейся бензоколонкой.

```

Kolonka STORAGE 2 ; 2 бензоколонки на АЗС
GENERATE (Exponential(1,0,1.75)) ; поступление автомобилей
QUEUE Other ; присоединение к очереди
ENTER Kolonka ; занятие бензоколонки
DEPART Other ; выход из очереди
ADVANCE 3,1 ; обслуживание на бензоколонке
LEAVE Kolonka ; освобождение бензоколонки
TERMINATE ; автомобиль покидает систему
* задание условия остановки процедуры моделирования АЗС
GENERATE 10080 ; генерация фиктивного транзакта,
* указывающего на окончание рабочего дня
(7 дней · 24 часа · 60 мин = 10080 мин)
*
TERMINATE 1 ; остановить моделирование
START 1 ; запуск процедуры моделирования

```

В результате моделирования получен следующий отчет:

```

QUEUE MAX CONT. ENTRY ENTRY(0) AVE.CONT. AVE.TIME AVE.(-0) RETRY
OTHER 18 0 5647 1376 1.895 3.382 4.471 0

```

```

STORAGE CAP. REM. MIN. MAX. ENTRIES AVL. AVE.C. UTIL. RETRY DELAY
KOLONKA 2 0 0 2 5647 1 1.681 0.840 0 0

```

Испытание имитационных моделей стратегий обслуживания АЗС в соответствии с п. 1.5.5 заключалось главным образом в верификации модели. При этом в ходе пошаговой отладки моделей было установлено соответствие алгоритмов их функционирования содержательному описанию АЗС, концептуальной модели и замыслу моделирования.

Адекватность созданных моделей реальным системам полностью обусловливается ограничениями и допущениями, сформулированными на этапе содержательного описания объекта моделирования. Поэтому адекватность моделей в данной учебной задаче не проверялась.

Этап **исследования свойств ИМ** (точности, устойчивости, чувствительности и пр.) пропущен, поскольку:

- алгоритмы функционирования ИМ уже верифицированы;
- поставленные цели моделирования не касаются свойств ИМ;
- обеспечение высокой точности и устойчивости результатов достигается выбором достаточно длительного исследуемого периода модельного времени.

В качестве **исследуемого периода модельного времени** выбрана 1 неделя круглосуточной работы АЗС. Такой период времени является достаточно представительным (за этот период промоделировано обслуживание нескольких тысяч автомобилей) и не требует больших затрат машинного времени.

Проведем **анализ результатов моделирования** в соответствии с п. 1.5.8. Для этого сведем полученные статистики моделирования в таблицу 5.

Таблица 5 – Результаты моделирования двух стратегий обслуживания на АЗС

Показатель	Стратегия 1			Стратегия 2
	Бензоколонка 1	Бензоколонка 2	В целом	
Поступило автомобилей	2891	2875	5766	5647
Обслужено автомобилей	2891	2874	5765	5645
Коэффициент загрузки	0,859	0,857	0,858	0,840
Максимальная длина очереди	11	12	23	18
Средняя длина очереди	1,488	1,513	3,001	1,895
Среднее время ожидания, мин	5,189	5,304	5,2465	3,382

По результатам моделирования можно сделать следующий вывод: наилучшей стратегией обслуживания автомобилей на АЗС является стратегия 2, когда прибывающие автомобили образуют общую очередь и обслуживаются освободившейся бензоколонкой, поскольку в этом случае при прочих равных характеристиках:

- меньше время ожидания автомобилей в общей очереди и, следовательно, меньше время пребывания автомобилей на АЗС;
- меньше средняя длина очереди к бензоколонкам (среднее число автомобилей, ожидающих обслуживания перед АЗС);
- меньше максимальная длина очереди (максимальное число автомобилей, одновременно ожидающих обслуживания) и, следовательно, требуются меньшие подъездные площади АЗС).

Для определения оптимального количества бензоколонок зададимся следующими нормами:

- коэффициент загрузки бензоколонок принадлежит интервалу от 0,5 до 0,9;
- среднее число автомобилей, одновременно находящихся на АЗС, не должно превышать 3;
- среднее время ожидания обслуживания не должно превышать 4 мин.

При исследовании будем использовать ИМ, реализующую 2-ю стратегию обслуживания. Варьировать количество бензоколонок на АЗС будем заменой количества приборов многоканального обслуживающего устройства Kolonka в первой строке модели. Например:

```

Kolonka STORAGE 3 ; 3 бензоколонки на АЗС

```

При исследовании АЗС с 1 бензоколонкой были получены следующие основные результаты:

```

QUEUE MAX CONT. ENTRY ENTRY(0) AVE.CONT. AVE.TIME AVE.(-0) RETRY
OTHER 2373 2373 5745 1 1204.438 2113.270 2113.638 0

```

```

STORAGE CAP. REM. MIN. MAX. ENTRIES AVL. AVE.C. UTIL. RETRY DELAY
KOLONKA 1 0 0 1 3372 1 1.000 1.000 0 2373

```

При исследовании АЗС с 3 бензоколонок – следующие результаты:

QUEUE OTHER	MAX 7	CONT. 4	ENTRY 5702	ENTRY (0) 4002	AVE.CONT. 0.228	AVE.TIME 0.404	AVE. (-0) 1.355	RETRY 0		
STORAGE KOLONK	CAP. 3	REM. 0	MIN. 0	MAX. 3	ENTRIES 5698	AVL. 1	AVE.C. 1.692	UTIL. 0.564	RETRY 0	DELAY 4

Результаты моделирования АЗС с различным числом бензоколонок сведены в таблицу 6.

Таблица 6 – Характеристики АЗС с различным числом бензоколонок

Количество бензоколонок	1	2	3	4
Коэффициент загрузки бензоколонок	1	0,840	0,564	0,424
Среднее число автомобилей, одновременно находящихся на АЗС	1 + + 1204,438 = = 1205,438	1,681 + + 1,895 = = 3,576	1,692 + + 0,228 = = 1,92	1,696 + + 0,053 = = 1,749
Среднее время ожидания обслуживания	2113,270	3,382	0,404	0,093

Из представленных результатов видно, что АЗС с одной бензоколонкой не справляется с поступающим потоком автомобилей. Для эффективного обслуживания автомобилей целесообразно использование 3 бензоколонок. Увеличение или уменьшение числа колонок целесообразно при изменении входящего потока автомобилей или характеристик времени их обслуживания.

3.2 Пример моделирования процесса обслуживания клиентов в железнодорожной билетной кассе

Представим **содержательное описание объекта моделирования** в соответствии с п. 1.5.1. В железнодорожной билетной кассе имеется 2 кассы, работающие круглосуточно. Каждая касса имеет по 4 получасовых технологических перерыва, варианты организации которых представлены в таблице 7. В начале технологического перерыва обслуживание заявки, уже находящейся в обработке, продолжается; а посетители, ожидающие обслуживания, перемещаются в конец соседней очереди.

Таблица 7 – Варианты организации технологических перерывов

Вариант 1		Вариант 2		Вариант 3	
Касса 1	Касса 2	Касса 1	Касса 2	Касса 1	Касса 2
$2^{00} - 2^{30}$	$5^{00} - 5^{30}$	$1^{00} - 1^{30}$	$4^{00} - 4^{30}$	$3^{00} - 3^{30}$	$0^{30} - 1^{00}$
$8^{00} - 8^{30}$	$11^{00} - 11^{30}$	$6^{00} - 6^{30}$	$10^{00} - 10^{30}$	$12^{00} - 12^{30}$	$6^{30} - 7^{00}$
$14^{00} - 14^{30}$	$17^{00} - 17^{30}$	$14^{00} - 14^{30}$	$16^{00} - 16^{30}$	$16^{00} - 16^{30}$	$12^{30} - 13^{00}$
$20^{00} - 20^{30}$	$23^{00} - 23^{30}$	$19^{00} - 19^{30}$	$22^{00} - 22^{30}$	$22^{00} - 20^{30}$	$18^{30} - 19^{00}$

Поток посетителей в кассы образован двумя потоками. Посетители 1-го типа обслуживаются только в случае, когда в очереди нет посетителей 2-го типа. Посетители 2-го типа (ветераны и т.д.) при обслуживании имеют более высокий приоритет. Интервалы времени между поступлением посетителей в кассу имеют экспоненциальное распределение со средним значением μ_1 – для посетителей 1-го типа, μ_2 – для посетителей 2-го типа. Поступающие посетители занимают любую из свободных касс или присоединяются к очереди, которая на текущий момент является кратчайшей. После обслуживания посетитель уходит из кассы.

Время обслуживания посетителей зависит от типа приобретаемого билета (на поезд пригородного или дальнего следования) и имеет равномерное распределение на отрезках $[a_1, b_1]$ и $[a_2, b_2]$ соответственно. Заявки на пригородные перевозки составляют в среднем α процентов в общей массе заявок.

Исходные данные: $\mu_1 = 4$ мин, $\mu_2 = 25$ мин, $a_1 = 4$ мин, $b_1 = 1$ мин, $a_2 = 6$ мин, $b_2 = 2$ мин, $\alpha = 60\%$.

Целью моделирования является определение:

- характеристик качества обслуживания посетителей, в частности, средних длин очередей; среднего времени обслуживания одного посетителя;
- эффективного расписания технологических перерывов;
- оптимального количества касс.

В качестве показателей эффективности функционирования исследуемой системы, определяющих цели моделирования, будут выступать перечисленные выше характеристики. Управляемыми переменными (параметрами) моделирования являются интенсивность поступления посетителей каждого типа в билетную кассу, а также варианты организации технологических перерывов. Контролируемыми характеристиками (переменными) модели являются статистики моделирования: количество поступивших и обслуженных посетителей, коэффициенты загрузки касс, средние длины очередей, среднее время ожидания в очереди, среднее время обслуживания посетителей (включая ожидание в очереди) и т. п.

Ограничения ИМ связаны:

- с условиями, накладываемыми на потоки заявок (они предполагаются простейшими, отсутствуют повторные заявки);
- упрощенным алгоритмом функционирования касс и выбора кассы посетителями;
- отсутствием явлений, изменяющих закономерности времени обслуживания посетителей (выдача справок, возвраты билетов, отсутствие билетов, ошибки при подготовке билетов и при денежных расчетах, отказы и сбои технического оборудования).

Составим **концептуальную модель** работы билетной кассы в соответствии с п. 1.5.2.

Элементами исследуемой системы являются потоки посетителей первого и второго типов, две кассы и две очереди (рисунок 24).

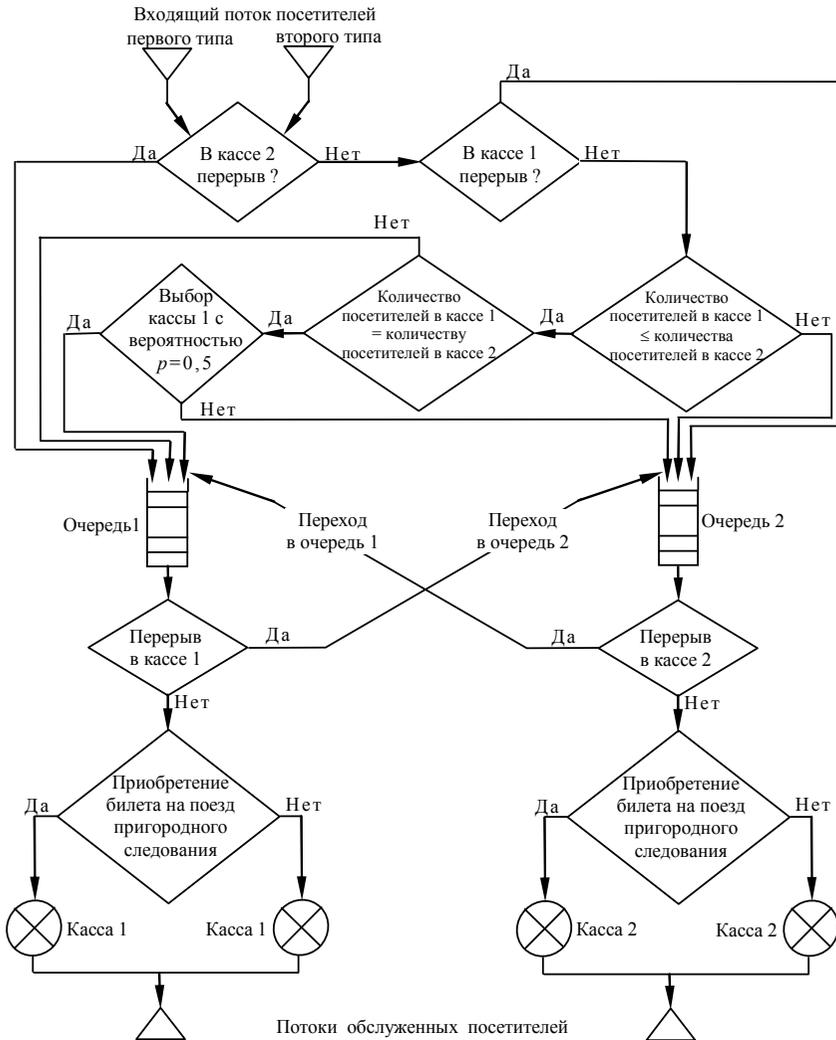


Рисунок 24 – Упрощенная схема обслуживания посетителей

В качестве критериев, используемых для сравнения времени начала технологических перерывов, будем рассматривать:

- количество поступивших и обслуженных посетителей;
- коэффициенты загрузки касс;

- максимальные и средние длины очередей;
- среднее время обслуживания (включая время ожидания).

Для определения эффективного количества касс зададимся следующими нормами:

- коэффициент загрузки касс принадлежит интервалу от 0,5 до 0,8;
- среднее число посетителей, одновременно находящихся в помещении касс, не превышает 6;
- среднее время ожидания обслуживания не превышает 5 мин.

Для организации технологических перерывов в билетной кассе используются фиктивные генераторы транзактов и устройства «Pereryv 1» и «Pereryv 2», которые работают следующим образом. В момент начала технологического перерыва генерируется транзакт, который задерживается на соответствующем устройстве «Pereryv» на время технологического перерыва (30 минут). По истечении этого времени транзакт поглощается. Для проверки на наличие технологического перерыва в билетной кассе необходимо проверить свободно ли соответствующее фиктивное устройство «Pereryv» (рисунок 25).

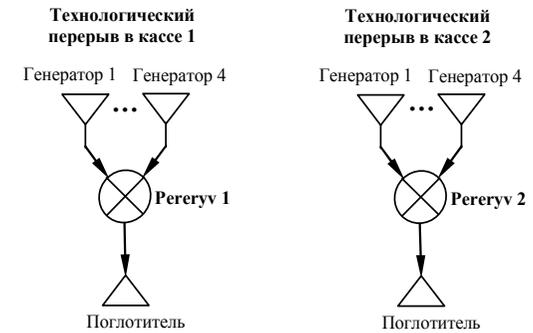


Рисунок 25 – Упрощенная схема организации технологических перерывов в билетной кассе

Выполним этап **формализации** функционирования билетной кассы в соответствии с п. 1.5.3.

Так как исследуемая система является системой массового обслуживания, в которой все события связаны с перемещением посетителей – динамических объектов (в помещение кассы, в очередь, в кассу на обслуживание, из кассы), то в данном случае целесообразно использовать транзактный способ организации квазипараллелизма компонентов модели.

Для формализации билетной кассы используем два типа элементов: обслуживающие устройства и очереди. Каждому поступающему посетителю соответствует свой транзакт.

Для создания или уничтожения транзактов, изменения маршрута их движения, задержки транзактов на некоторый интервал времени будем использовать дополнительные фиктивные блоки. Схема формализации кассы при транзактном способе имитации представлена на рисунке 26.

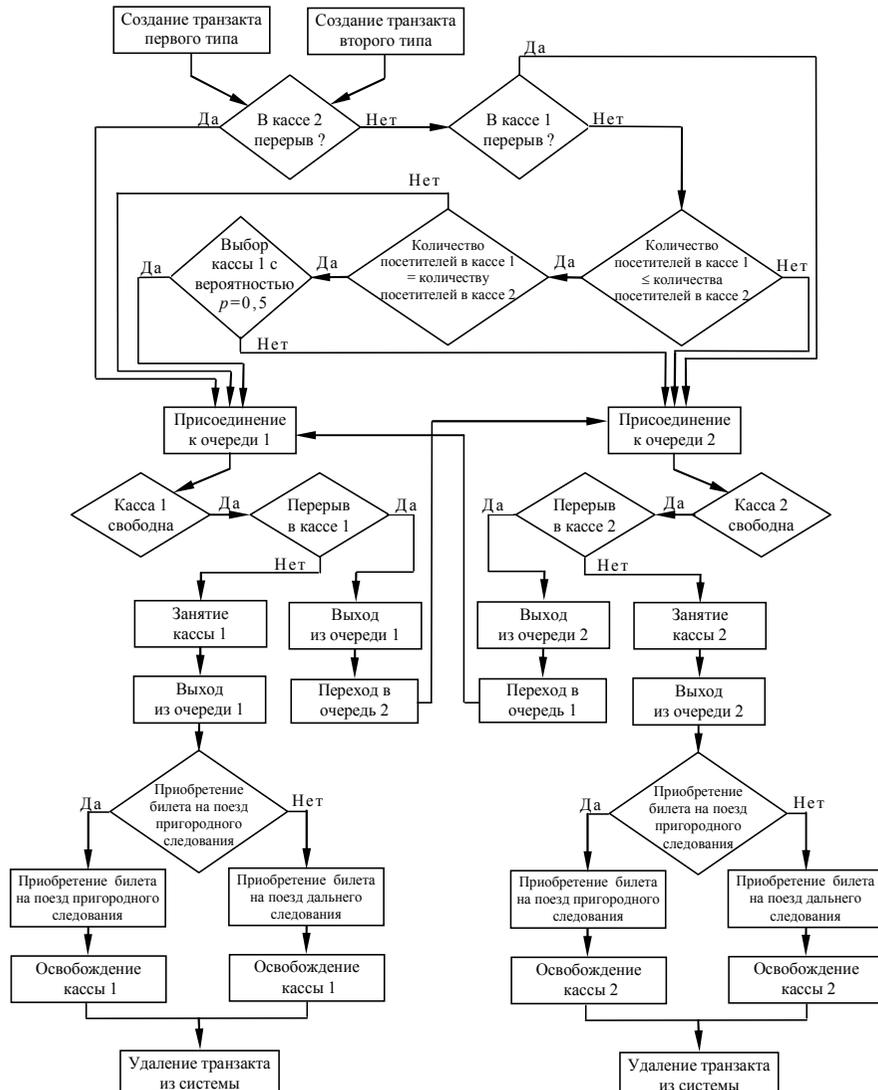


Рисунок 26 – Уточненная схема обслуживания посетителей

На рисунке 27 приведена уточненная схема организации технологических перерывов в билетной кассе.



Рисунок 27 – Подсистема моделирования технологических перерывов в билетной кассе

Составим модель на языке моделирования в соответствии с п. 1.5.4. В качестве средства автоматизации имитационного моделирования железнодорожной кассы будем использовать GPSS World.

Приведем текст ИМ.

```

* Определение переменных
Posetit1 Variable F$Kassir1+Q$Other1 ; количество посетителей в кассе 1
Posetit2 Variable F$Kassir2+Q$Other2 ; количество посетителей в кассе 2

* Поступление посетителей, распределение по кассам, постановка в очереди
Generate (Exponential(1,0,4)),,,,1 ; приход посетителя 1-го типа
Transfer ,Obslug ; переход на обслуживание
Generate (Exponential(1,0,25)),,,,2 ; приход посетителя 2-го типа
Obslug Test E F$Pereryv2,0,Kassa_1 ; в кассе 2 нет перерыва?
Test E F$Pereryv1,0,Kassa_2 ; в кассе 1 нет перерыва?
Test LE V$Posetit1,V$Posetit2,Kassa_2 ; количество посетителей
* в кассе 1 <= количества посетителей в кассе 2?
Test E V$Posetit1,V$Posetit2,Kassa_1 ; количество посетителей
* в кассе 1 = количеству посетителей в кассе 2?
Transfer 0.5,Kassa_1,Kassa_2 ; выбор кассы наудачу
* при равном количестве посетителей

* Обслуживание в кассе 1
Kassa_1 Queue Other1 ; присоединение к очереди 1
Test E F$Kassir1,0 ; касса 1 свободна?
Test E F$Pereryv1,1,Obsl1 ; есть ли технологический перерыв
* в кассе 1?
Depart Other1 ; выход из очереди 1
Transfer ,Kassa_2 ; переход на обслуживание в кассу 2
    
```

```

Obsl1  Seize Kassir1      ;занятие кассы 1
       Depart Other1    ;выход из очереди 1
       Transfer 0.6,DBilet1 ;переход для приобретения билета
       *                ;на поезд дальнего следования
       Advance 4,1      ;приобретение билета на пригородный поезд
       Transfer ,Uhod1  ;безусловный переход на выход

DBilet1 Advance 6,2     ;приобретение билета на дальний поезд
Uhod1  Release Kassir1 ;освобождение кассы 1
       Terminate       ;посетитель покидает кассу

* Обслуживание в кассе 2
Kassa_2 Queue Other2   ;присоединение к очереди 2
       Test E F$Kassir2,0 ;касса 2 свободна?
       Test E F$Pereryv2,1,Obsl2 ;есть ли технологический перерыв
       *                ;в кассе 2?
       Depart Other2    ;выход из очереди 2
       Transfer ,Kassa_1 ;переход на обслуживание в кассу 1

Obsl2  Seize Kassir2   ;занятие кассы 2
       Depart Other2    ;выход из очереди 2
       Transfer 0.6,DBilet2 ;переход для приобретения билета
       *                ;на поезд дальнего следования
       Advance 4,1      ;приобретение билета на пригородный поезд
       Transfer ,Uhod2  ;безусловный переход на выход

DBilet2 Advance 6,2     ;приобретение билета на дальний поезд
Uhod2  Release Kassir2 ;освобождение кассы 2
       Terminate       ;посетитель покидает кассу

* Организация технологического перерыва в кассе 1
10     Generate 1440,,120 ;создание транзакта в 2.00
       Transfer ,Met2     ;безусловный переход
20     Generate 1440,,480 ;создание транзакта в 8.00
       Transfer ,Met2     ;безусловный переход
30     Generate 1440,,840 ;создание транзакта в 14.00
       Transfer ,Met2     ;безусловный переход
40     Generate 1440,,1200 ;создание транзакта в 20.00
Met2   Seize Pereryv1    ;занятие фиктивного устройства
       Advance 30        ;задержка транзакта на 30 минут
       *                ;(время технологического перерыва)
       Release Pereryv1 ;освобождение фиктивного устройства
       Terminate        ;удаление транзакта из системы

* Организация технологического перерыва в кассе 2
50     Generate 1440,,300 ;создание транзакта в 5.00
       Transfer ,Met3     ;безусловный переход
60     Generate 1440,,660 ;создание транзакта в 11.00
       Transfer ,Met3     ;безусловный переход
70     Generate 1440,,1020 ;создание транзакта в 17.00
       Transfer ,Met3     ;безусловный переход
80     Generate 1440,,1380 ;создание транзакта в 23.00
Met3   Seize Pereryv2    ;занятие фиктивного устройства
       Advance 30        ;задержка транзакта на 30 минут
       *                ;(время технологического перерыва)
       Release Pereryv2 ;освобождение фиктивного устройства
       Terminate        ;удаление транзакта из системы

* Задание условия остановки процедуры моделирования
Generate 14400          ;генерация фиктивного транзакта,
*                       ;указывающего на окончание рабочего дня
*                       ;(10 дней · 24 часа · 60 мин = 14400 мин)
Terminate 1            ;остановить моделирование
START 1                ;запуск процедуры моделирования

```

В результате моделирования получен следующий отчет:

FACILITY	ENTRIES	UTIL.	AVE.TIME	AVAIL.	OWNER	PEND	INTER	RETRY	DELAY
PERERYV2	40	0.083	30.000	1	0	0	0	0	0
PERERYV1	40	0.083	30.000	1	0	0	0	0	0
KASSIR1	2020	0.675	4.813	1	4138	0	0	1	0
KASSIR2	2034	0.674	4.770	1	4143	0	0	2	0

QUEUE	MAX	CONT.	ENTRY	ENTRY(0)	AVE.CONT.	AVE.TIME	AVE.(-0)	RETRY
OTHER1	12	1	2047	730	0.794	5.587	8.683	0
OTHER2	12	2	2051	738	0.887	6.227	9.727	0

При изменении времени начала технологических перерывов (в соответствии с вариантами из таблицы 7) в тексте ИМ будут изменены следующие строки:

```

* Организация технологического перерыва в кассе 1
10     Generate 1440,,60 ;создание транзакта в 1.00
20     Generate 1440,,360 ;создание транзакта в 6.00
30     Generate 1440,,840 ;создание транзакта в 14.00
40     Generate 1440,,1140 ;создание транзакта в 19.00

* Организация технологического перерыва в кассе 2
50     Generate 1440,,240 ;создание транзакта в 4.00
60     Generate 1440,,600 ;создание транзакта в 10.00
70     Generate 1440,,960 ;создание транзакта в 16.00
80     Generate 1440,,1320 ;создание транзакта в 22.00

```

В результате моделирования получен следующий отчет:

FACILITY	ENTRIES	UTIL.	AVE.TIME	AVAIL.	OWNER	PEND	INTER	RETRY	DELAY
PERERYV2	40	0.083	30.000	1	0	0	0	0	0
PERERYV1	40	0.083	30.000	1	0	0	0	0	0
KASSIR1	2061	0.684	4.776	1	4187	0	0	0	0
KASSIR2	2036	0.675	4.777	1	4186	0	0	1	0

QUEUE	MAX	CONT.	ENTRY	ENTRY(0)	AVE.CONT.	AVE.TIME	AVE.(-0)	RETRY
OTHER1	12	0	2076	766	0.661	4.585	7.266	0
OTHER2	10	1	2053	792	0.644	4.519	7.357	0

Для третьего варианта организации технологических перерывов:

```

* Организация технологического перерыва в кассе 1
10     Generate 1440,,180 ;создание транзакта в 3.00
20     Generate 1440,,720 ;создание транзакта в 12.00
30     Generate 1440,,960 ;создание транзакта в 16.00
40     Generate 1440,,1320 ;создание транзакта в 22.00

* Организация технологического перерыва в кассе 2
50     Generate 1440,,30 ;создание транзакта в 0.30
60     Generate 1440,,390 ;создание транзакта в 6.30
70     Generate 1440,,750 ;создание транзакта в 12.30
80     Generate 1440,,1110 ;создание транзакта в 18.30

```

В результате моделирования получен следующий отчет:

FACILITY	ENTRIES	UTIL.	AVE. TIME	AVAIL.	OWNER	PEND	INTER	RETRY	DELAY
PERERYV2	40	0.083	30.000	1	0	0	0	0	0
PERERYV1	40	0.083	30.000	1	0	0	0	0	0

KASSIR1	2084	0.702	4.851	1	4271	0	0	0	0
KASSIR2	2096	0.702	4.826	1	4269	0	0	1	0
QUEUE	MAX	CONT.	ENTRY	ENTRY (0)	AVE.CONT.	AVE.TIME	AVE. (-0)	RETRY	
OTHER1	17	0	2104	712	0.849	5.812	8.785	0	
OTHER2	10	1	2125	712	0.817	5.537	8.327	0	

Испытание имитационной модели функционирования билетной кассы в соответствии с п. 1.5.5 заключалось главным образом в верификации модели. При этом в ходе пошаговой отладки модели было установлено соответствие алгоритма ее функционирования содержательному описанию, концептуальной модели и замыслу моделирования.

Адекватность модели билетной кассы реальной системе обусловлена только ограничениями и допущениями, сформулированными на этапе содержательного описания объекта моделирования. Поэтому адекватность моделей в данной учебной задаче не проверялась.

Этап **исследования свойств ИМ** (точности, устойчивости, чувствительности и пр.) пропущен, поскольку:

- алгоритмы функционирования ИМ уже верифицированы;
- поставленные цели моделирования не касаются свойств ИМ;
- обеспечение высокой точности и устойчивости результатов достигается выбором достаточно длительного исследуемого периода модельного времени.

В качестве **исследуемого периода модельного времени** выбран период в 10 дней круглосуточной работы кассы. Такой период времени является достаточно представительным (за этот период промоделировано обслуживание нескольких тысяч посетителей) и не требует больших затрат машинного времени.

Проведем **анализ результатов моделирования** в соответствии с п. 1.5.8. Для этого сведем полученные статистики моделирования в таблицу 8.

Таблица 8 – Результаты моделирования работы билетной кассы

Показатель	Варианты организации технологических перерывов								
	Вариант 1			Вариант 2			Вариант 3		
	Касса 1	Касса 2	В целом	Касса 1	Касса 2	В целом	Касса 1	Касса 2	В целом
Поступило посетителей	2020	2034	4054	2061	2037	4098	2084	2097	4181
Обслужено посетителей	2019	2033	4052	2060	2035	4095	2083	2095	4178
Коэффициент загрузки	0,675	0,674	0,6745	0,684	0,675	0,6795	0,702	0,702	0,702
Максимальная длина очереди	12	12	24	12	10	22	17	10	27
Средняя длина очереди	0,794	0,887	1,681	0,661	0,644	1,305	0,849	0,817	1,666
Среднее время ожидания, мин	5,587	6,227	5,907	4,585	4,519	4,552	5,812	5,537	5,6745

По результатам моделирования можно сделать следующий вывод: наиболее эффективным является второй вариант организации технологических перерывов, поскольку в этом случае при прочих равных характеристиках:

- меньше время ожидания посетителей в очереди и, следовательно, меньше время пребывания их в билетных кассах;
- меньше средняя и максимальная длина очереди.

Количество касс является эффективным, так как соответствует указанным выше нормам. При данном потоке посетителей в ИМ с одной кассой длина очереди неограниченно растет. Модель с тремя кассами не рассматривалась, поскольку алгоритм ее функционирования является достаточно сложным.

3.3 Задания для самостоятельной работы

Задание 1 Моделирование работы справочной службы вокзала

1 В справочной службе железнодорожного вокзала установлена n -канальная круглосуточная телефонная линия (т. е. одновременно могут обслуживаться n абонентов).

2 Длины интервалов времени между поступлениями вызовов случайны, независимы и подчинены экспоненциальному закону распределения со средним значением μ .

3 Время обслуживания каждого абонента имеет равномерное распределение на интервале $[a, b]$.

4 Если все операторы заняты, то абонент подключается к одному из m автоответчиков для ожидания. Если все автоответчики заняты, то абонент получает отказ.

Цель. Построить GPSS-модель работы справочной службы железнодорожного вокзала. Оценить вероятность занятости всех операторов справочной службы, среднее число занятых каналов и среднее время простоя каждого канала. Сделать предложения относительно необходимости изменения количества операторов и (или) автоответчиков.

Исходные данные:

а) $n = 5, a = 10 \text{ с}, b = 25 \text{ с}, \mu = 1 \text{ мин}, m = 2;$

б) $n = 7, a = 5 \text{ с}, b = 20 \text{ с}, \mu = 1 \text{ мин}, m = 3.$

Задание 2 Моделирование автомобильного движения на нерегулируемом Т-образном перекрестке

1 При движении налево на нерегулируемом Т-образном перекрестке с одной полосой движения в каждом направлении необходимо уступить дорогу встречным транспортным средствам. При движении прямо водитель проезжает перекресток без остановок.

2 Поток автомобилей в обоих направлениях является простейшим с интенсивностью λ , причем α , %, автомобилей осуществляет поворот налево.

3 Для предотвращения аварии время проезда перекрестка при повороте налево не должно превышать время подъезда к перекрестку встречного автомобиля. При этом время проезда перекрестка при повороте налево – случайная величина, подчиняющаяся равномерному закону распределения на интервале $[a, b]$.

Цель. Разработать GPSS-модель для моделирования поворота налево на нерегулируемом Т-образном перекрестке. Оценить целесообразность запрета поворота налево.

Исходные данные:

а) $\lambda = 4$ авт./ч, $\alpha = 30$ %, $a = 5$ с, $b = 15$ с;

б) $\lambda = 6$ авт./ч, $\alpha = 5$ %, $a = 10$ с, $b = 20$ с.

Задание 3 Моделирование движения автомобилей через железнодорожный переезд

1 Поток автомобилей через железнодорожный переезд по дороге с одной полосой движения в одном направлении является простейшим с интенсивностью λ .

2 Время проезда через переезд – случайная величина, равномерно распределенная на интервале $[a, b]$.

3 Движение через железнодорожный переезд запрещается в среднем через каждые t минут на время, равномерно распределенное на интервале $[m, h]$.

Цель. Разработать GPSS-модель для анализа движения автомобилей через железнодорожный переезд в одном направлении. Определить, насколько целесообразно увеличение числа полос для проезда через железнодорожный переезд.

Исходные данные:

а) $\lambda = 10$ авт./ч, $a = 5$ с, $b = 10$ с, $t = 30$ мин, $m = 3$ мин, $h = 5$ мин;

б) $\lambda = 15$ авт./ч, $a = 4$ с, $b = 8$ с, $t = 40$ мин, $m = 3$ мин, $h = 7$ мин.

Задание 4* Моделирование работы маршрутных такси

1 На некотором городском маршруте по кольцевому маршруту с M остановками работают N восьмиместных микроавтобусов.

2 Время движения между остановками – величина, равномерно распределенная на интервале $[a, b]$.

3 На каждую остановку в соответствии с экспоненциальным законом распределения со средним значением μ прибывают пассажиры и ждут микроавтобус.

4 Микроавтобус подъезжает к остановке и забирает столько пассажиров, сколько имеется свободных мест. Если свободных мест больше, чем пассажиров, то микроавтобус забирает всех. Если на остановке никто не выходит и в микроавтобусе нет свободных мест, он не останавливается.

5 Вероятность того, что пассажир проедет n_i остановок, равна p_i .

Цель. Разработать GPSS-модель для анализа работы маршрутных такси. Оценить загруженность микроавтобусов. Определить оптимальное число микроавтобусов для данного маршрута.

Исходные данные:

а) $M = 10$, $N = 5$, $a = 5$ мин, $b = 8$ мин, $\mu = 5$ мин, $n_1 = 3$, $p_1 = 0,1$, $n_2 = 4$, $p_2 = 0,15$, $n_3 = 5$, $p_3 = 0,2$, $n_4 = 6$, $p_4 = 0,25$, $n_5 = 7$, $p_5 = 0,3$;

б) $M = 12$, $N = 6$, $a = 4$ мин, $b = 7$ мин, $\mu = 4$ мин, $n_1 = 3$, $p_1 = 0,05$, $n_2 = 4$, $p_2 = 0,10$, $n_3 = 5$, $p_3 = 0,2$, $n_4 = 6$, $p_4 = 0,3$, $n_5 = 7$, $p_5 = 0,35$.

Задание 5 Моделирование функционирования парка отправления поездов

1 Поезда, поступающие в парк отправления, подвергаются техническому осмотру (ТО). Параллельно работают N бригад.

2 Поток поступающих в парк составов является пуассоновским с интенсивностью λ , а время ТО – случайная величина, равномерно распределенная на интервале $[a, b]$.

3 Парк отправления имеет M путей.

Цель. Построить GPSS-модель для анализа работы парка отправления поездов. Определить эффективность использования большего количества бригад ТО.

Исходные данные:

а) $N = 1$, $M = 7$, $\lambda = 3$ поездов/ч, $a = 15$ мин, $b = 25$ мин;

б) $N = 2$, $M = 5$, $\lambda = 4$ поездов/ч, $a = 10$ мин, $b = 20$ мин.

Задание 6 Моделирование станции технического обслуживания

1 На станцию технического обслуживания, которая состоит из бокса для ремонта и бокса для технического осмотра (ТО), поступают автомобили, образуя простейший поток событий с интенсивностью λ .

2 После прохождения ТО α , %, автомобилей требуют ремонта. Время ТО имеет равномерный закон распределения на интервале $[b \pm \varepsilon]$, а время ремонта – нормальный закон распределения со средним значением m и среднеквадратическим отклонением σ .

Цель. Построить GPSS-модель для анализа работы станции технического обслуживания. Дать рекомендации об увеличении количества боксов для ТО и ремонта автомобилей, а также о возможности обслуживания автомобилей по предварительной записи.

* Задания повышенной сложности.

Исходные данные:

а) $\lambda = 2$ авт./ч, $b = 45$ мин, $\varepsilon = 15$ мин, $m = 200$ мин, $\sigma = 50$ мин, $\alpha = 27\%$;

б) $\lambda = 1,5$ авт./ч, $b = 50$ мин, $\varepsilon = 20$ мин, $m = 150$ мин, $\sigma = 40$ мин, $\alpha = 38\%$.

Задание 7 Моделирование работы автозаправочной станции

1 На автозаправочной станции установлены 4 бензоколонки: одна колонка с низкооктановым бензином, две колонки с высокооктановым бензином и одна – с дизельным топливом.

2 Поступающие на АЗС автомобили образуют простейший поток событий с интенсивностью λ .

3 Среди поступающих автомобилей $\alpha, \%$, заправляются низкооктановым бензином, $\beta, \%$, – высокооктановым бензином и $\gamma, \%$, – дизельным топливом.

4 Время заправки автомобилей низкооктановым, высокооктановым бензином и дизельным топливом – случайные величины, имеющие равномерный закон распределения на интервалах $[m_i \pm h_i]$ ($i = \overline{1,3}$) соответственно.

Цель. Разработать GPSS-модель работы автозаправочной станции. Определить среднее время обслуживания автомобилей. Дать рекомендации для изменения числа бензоколонок с тем или иным видом топлива.

Исходные данные:

а) $\lambda = 10$ авт./ч, $\alpha = 30\%$, $\beta = 50\%$, $\gamma = 20\%$, $m_1 = 6$ мин, $h_1 = 1$ мин, $m_2 = 5,5$ мин, $h_2 = 0,5$ мин, $m_3 = 5$ мин, $h_3 = 1$ мин;

б) $\lambda = 15$ авт./ч, $\alpha = 20\%$, $\beta = 50\%$, $\gamma = 30\%$, $m_1 = 5$ мин, $h_1 = 1$ мин, $m_2 = 4,5$ мин, $h_2 = 1$ мин, $m_3 = 4,5$ мин, $h_3 = 0,5$ мин.

Задание 8 Моделирование процесса перевозки грузов

1 На стройке работает N грузовых автомобилей. В начале 8-часового рабочего дня они прибывают на стройку, образуя простейший поток событий с интенсивностью λ .

2 Время разгрузки автомобилей распределено равномерно на интервале $[m_1, h_1]$.

3 Одновременно могут разгружаться не более M автомобилей (другие ожидают в очереди).

4 Разгрузившись, автомобили выполняют следующий рейс и через время, равномерно распределённое на интервале $[m_2, h_2]$, они вновь прибывают для разгрузки на стройку.

Цель. Разработать GPSS-модель для анализа процесса перевозки. Определить оптимальное количество мест для разгрузки автомобилей, а также дать рекомендации об изменении числа задействованных автомобилей.

Исходные данные:

а) $N = 10$, $M = 2$, $\lambda = 1$ авт./ч, $m_1 = 10$ мин, $h_1 = 20$ мин, $m_2 = 60$ мин, $h_2 = 80$ мин;

б) $N = 12$, $M = 3$, $\lambda = 1,2$ авт./ч, $m_1 = 10$ мин, $h_1 = 15$ мин, $m_2 = 60$ мин, $h_2 = 90$ мин.

Задание 9 Моделирование процесса функционирования строительного узла

1 В строительном узле выполняется последовательно N работ по обслуживанию прибывающих автомобилей со строительными материалами.

2 Время выполнения каждой из работ распределено равномерно на интервале $[m_i, h_i]$ ($i = \overline{1, N}$).

3 Промежутки времени между моментами прибытия автомобилей имеют экспоненциальный закон распределения с параметром μ .

4 Одновременно могут разгружаться не более M автомобилей. При отсутствии места для разгрузки автомобили ожидают.

Цель. Разработать GPSS-модель для анализа процесса функционирования строительного узла. Определить среднее время обслуживания машины, количество обслуженных машин и среднее время ожидания в очереди. Дать рекомендации по оптимизации функционирования строительного узла.

Исходные данные:

а) $N = 3$, $M = 2$, $\mu = 60$ мин, $m_1 = 0$ мин, $h_1 = 5$ мин, $m_2 = 10$ мин, $h_2 = 15$ мин, $m_3 = 5$ мин, $h_3 = 10$ мин;

б) $N = 4$, $M = 1$, $\mu = 30$ мин, $m_1 = 0$ мин, $h_1 = 5$ мин, $m_2 = 8$ мин, $h_2 = 12$ мин, $m_3 = 4$ мин, $h_3 = 9$ мин, $m_4 = 3$ мин, $h_4 = 7$ мин.

Задание 10 Моделирование процесса стендовых испытаний

1 Стендовые испытания выполняются на двух испытательных стендах.

2 На стендовые испытания поступают изделия двух видов, соответственно α и $100 - \alpha, \%$.

3 Изделия 1-го типа проходят испытания на стенде 1, а изделия 2-го типа – на стенде 1 и 2. Изделия типа 2 поступают на стенд 1 в случае, если заняты стенд 2.

4 Изделия, поступающие на стендовые испытания, образуют простейший поток с интенсивностью λ .

5 Время испытания изделий имеет экспоненциальное распределение. Математическое ожидание времени обслуживания изделий 1-го типа на 1-м стенде – μ_1 , изделий типа 2 на 2-м стенде – μ_2 , а на 1-м стенде – μ_3 .

Цель. Разработать GPSS-модель для анализа процесса стендовых испытаний изделий. Определить среднюю длину очереди к стендам и коэффициент загрузки стендов.

Исходные данные:

а) $\alpha = 40\%$, $\lambda = 0,067$ изд./мин, $\mu_1 = 15$ мин, $\mu_2 = 25$ мин, $\mu_3 = 20$ мин;

б) $\alpha = 70\%$, $\lambda = 0,05$ изд./мин, $\mu_1 = 17$ мин, $\mu_2 = 25$ мин, $\mu_3 = 25$ мин.

Задание 11 Моделирование функционирования участка механообработки

1 Участок механообработки включает входной конвейер, транспортный робот, два однотипных независимо работающих станка с ЧПУ и выходной конвейер.

2 Время поступления деталей на обработку и время обработки детали на станке распределены равномерно на интервалах $[a \pm \delta]$ и $[b \pm \varepsilon]$ соответственно.

3 Робот переносит деталь с входного конвейера на свободный станок и затем со станка на выходной конвейер, время каждого переноса распределено равномерно на интервале $[c \pm v]$.

Цель. Разработать GPSS-модель для анализа процесса функционирования участка. Определить среднюю длину очереди, среднее время простоя робота, среднее время простоя станков.

Исходные данные, с:

а) $a = 160, \delta = 90, b = 280, \varepsilon = 30, c = 10, v = 2;$

б) $a = 200, \delta = 50, b = 300, \varepsilon = 40, c = 12, v = 2.$

Задание 12 Моделирование процесса дозаправки самолетов

1 С целью увеличения дальности беспосадочного полета производится дозаправка самолетов горючим в воздухе. В районе дозаправки постоянно дежурит m самолетов-дозаправщиков.

2 Время дозаправки самолетов - случайная величина, равномерно распределенная на интервале $[\tau_1, \tau_2]$. Временные промежутки между прибытиями самолетов в район дозаправки также случайны, независимы и распределены равномерно на интервале $[s_1, s_2]$.

3 Если все дозаправщики заняты, то самолет, нуждающийся в дозаправке, некоторое время может "ожидать" (совершая полет по кругу в районе дозаправки). Время ожидания дозаправки описывается случайной величиной, равномерно распределенной в интервале $[t_1, t_2]$.

4 Самолет, не дождавшийся дозаправки в воздухе, приземляется на запасной аэродром. Число самолетов, ожидающих дозаправки в воздухе, не ограничено.

Цель. Разработать GPSS-модель функционирования дозаправки в воздухе. Оценить среднее число занятых дозаправщиков, вероятность занятости произвольно выбранного дозаправщика, среднее время простоя дозаправщика. Определить оптимальное количество дозаправщиков.

Исходные данные:

а) $m = 4, \tau_1 = 10$ мин, $\tau_2 = 11$ мин, $s_1 = 5$ мин, $s_2 = 6$ мин, $t_1 = 0$ мин, $t_2 = 20$ мин;

б) $m = 3, \tau_1 = 14$ мин, $\tau_2 = 16$ мин, $s_1 = 6$ мин, $s_2 = 7$ мин, $t_1 = 5$ мин, $t_2 = 25$ мин.

Задание 13 Моделирование процесса обслуживания клиентов в кассе

1 В железнодорожной кассе имеется N кассиров. Приход клиентов в кассу описывается пуассоновским потоком с интенсивностью λ .

2 Время обслуживания клиентов – экспоненциальное со средним значением μ .

3 Если в момент входа клиента в кассу хотя бы один кассир свободен, клиент сразу же попадает к этому кассиру. В противном случае клиент присоединяется к любой очереди, которая на текущий момент является кратчайшей.

4 Обслуживание клиентов в очереди осуществляется по принципу "первым пришел – первым обслужился". После обслуживания клиент уходит из кассы.

Цель. Разработать GPSS-модель для анализа работы кассы. Определить оптимальное количество кассиров. Сравнить предлагаемую модель обслуживания клиентов с моделью с общей очередью.

Исходные данные:

а) $N = 2, \lambda = 40$ чел./ч, $\mu = 3$ мин;

б) $N = 3, \lambda = 50$ чел./ч, $\mu = 4$ мин.

Задание 14 Моделирование процесса обслуживания рыболовных траулеров

1 Группа из m рыболовных траулеров обслуживается одной плавучей базой, на которой имеется один причал. База принимает на переработку рыбу и обеспечивает траулер необходимыми материалами.

2 Длины временных промежутков между прибытиями кораблей на базу случайны, независимы и одинаково распределены по равномерному закону на интервале $[\tau_1, \tau_2]$. Время обслуживания траулера на базе – случайная величина, распределенная равномерно на интервале $[t_1, t_2]$.

3 Если причал на базе занят, то траулер становится в очередь на обслуживание.

Длина очереди практически не ограничена.

Цель. Разработать GPSS-модель обслуживания траулеров. Оценить среднюю длину очереди, среднее время простоя траулера, среднее время простоя базы, среднее время ожидания в очереди.

Исходные данные:

а) $m = 10, \tau_1 = 0$ сут, $\tau_2 = 2$ сут, $t_1 = 4$ ч, $t_2 = 6$ ч;

б) $m = 8, \tau_1 = 0$ сут, $\tau_2 = 2$ сут, $t_1 = 5$ ч, $t_2 = 9$ ч.

Задание 15 Моделирование двух стратегий обслуживания на АЗС

1 Автозаправочная станция (АЗС) имеет две бензоколонки, причем автомобили прибывают через каждые ξ , с, а время заправки их бензином составляет η , с. Случайные величины ξ, η имеют равномерный закон распределения на интервалах $[a \pm \varepsilon], [b \pm \delta]$ соответственно.

2 Можно предложить две стратегии обслуживания прибывающих автомобилей:

1) автомобили образуют две очереди и обслуживаются соответствующими бензоколонками;

2) автомобили образуют одну очередь и обслуживаются освободившейся бензоколонкой.

Цель. Разработать GPSS-модель для анализа двух стратегий обслуживания автомобилей. По результатам имитации выбрать лучшую стратегию обслуживания на автозаправочной станции.

Исходные данные, с:

а) $a = 60, \varepsilon = 30, b = 120, \delta = 45$;

б) $a = 50, \varepsilon = 20, b = 100, \delta = 30$.

Задание 16 Моделирование процесса функционирования автомойки

1 Моечная станция имеет N мест на стоянке для автомобилей. Если клиенты подъезжают и не застают свободного места для ожидания, они уезжают.

2 Поток автомобилей является пуассоновским с интенсивностью λ . Время мойки автомобиля распределено экспоненциально с интенсивностью μ .

Цель. Разработать GPSS-модель и использовать ее для выбора оптимального числа мест N на стоянке так, чтобы число уехавших машин было минимальным.

Исходные данные:

а) $\lambda = 5$ авт./ч, $\mu = 15$ мин, $3 < N < 7$;

б) $\lambda = 7$ авт./час, $\mu = 20$ мин, $4 < N < 8$.

Задание 17* Моделирование процесса обслуживания танкеров в порту

1 Порт используют для заливки танкеров сырой нефтью. Порт имеет возможность заливать одновременно до N танкеров, которые прибывают в порт через промежутки времени, распределенные по равномерному закону в интервале $[a_1, a_2]$.

2 В порту имеется один буксир. Если буксир свободен, время подхода к стоянке или отхода от нее занимает T часов. Если буксир занят, танкеры ожидают его.

3 Время обслуживания танкера в порту – случайная величина, распределенная равномерно на интервале $[t_1, t_2]$.

Цель. Построить GPSS-модель, имитирующую работу порта. Обеспечить в модели возможность определения времени пребывания танкеров в порту. Определить среднее время простоя буксира и целесообразность увеличения количества буксиров.

Исходные данные:

а) $N = 3, a_1 = 4$ ч, $a_2 = 18$ ч, $T = 1$ ч, $t_1 = 15$ ч, $t_2 = 25$ ч;

б) $N = 4, a_1 = 10$ ч, $a_2 = 16$ ч, $T = 1,2$ ч, $t_1 = 20$ ч, $t_2 = 32$ ч.

Задание 18 Моделирование участка с двумя обрабатывающими центрами

1 Производственный участок имеет два обрабатывающих центра (ОБРЦ-I и ОБРЦ-II), которые осуществляют механообработку деталей трех типов.

2 Детали поступают на обработку через каждые t мин, образуя пуассоновский поток. В этом потоке детали типов 1 и 2 появляются с вероятностями 0,25, детали типа 3 – с вероятностью 0,5.

3 Время обработки деталей типов 1 и 2 – τ_1 , деталей типа 3 – τ_2 .

4 ОБРЦ-I осуществляет обработку деталей типов 1 и 2, а ОБРЦ-II – деталей всех типов, причем детали типов 1 и 2 поступают в ОБРЦ-II в случае, если ОБРЦ-I занят.

5 На участке фиксируется количество обработанных деталей каждого типа.

Цель. Разработать GPSS-модель для анализа процесса функционирования производственного участка. Определить коэффициенты загрузки станков.

Исходные данные, мин:

а) $t = 6, \tau_1 = 10, \tau_2 = 8$;

б) $t = 10, \tau_1 = 6, \tau_2 = 5$.

Задание 19 Моделирование автосервиса с приоритетным обслуживанием

1 Участок ремонта кузовов автомобилей состоит из двух рабочих мест. После восстановления кузова автомобили поступают в окрасочную камеру.

2 Длины временных промежутков между поступлениями поврежденных автомобилей первой модели – случайные, равномерно распределенные величины на интервале $[\tau_1, \tau_2]$, второй модели – случайные, равномерно распределенные величины на интервале $[\lambda_1, \lambda_2]$. Время пребывания автомобиля первой модели на кузовном ремонте – случайная равномерно распределенная величина на интервале $[h_1, h_2]$, второй модели – случайная величина с экспоненциальным законом распределения со средним значением μ . Время окраски любого автомобиля – случайная величина, имеющая равномерное распределение на интервале $[s_1, s_2]$.

3 Модели первого типа при обслуживании имеют более высокий приоритет.

4 В случае, если ремонтные места и окрасочная камера заняты, автомобили дожидаются обслуживания в очередях, длины которых не ограничены.

Цель. Разработать GPSS-модель функционирования ремонтных работ. Оценить отдельно для 1-й и 2-й моделей среднее время, которое тратится на ремонт автомобиля (от момента поступления на ремонт до завершения окраски), среднее время ожидания в очередях.

Исходные данные:

а) $\tau_1 = 0$ ч, $\tau_2 = 6$ ч, $\lambda_1 = 0$ ч, $\lambda_2 = 2$ ч, $h_1 = 1$ ч, $h_2 = 3$ ч, $\mu = 3$ ч, $s_1 = 10$ мин, $s_2 = 12$ мин.

б) $\tau_1 = 0$ ч, $\tau_2 = 3$ ч, $\lambda_1 = 0$ ч, $\lambda_2 = 5$ ч, $h_1 = 2$ ч, $h_2 = 4$ ч, $\mu = 5$ ч, $s_1 = 13$ мин, $s_2 = 14$ мин.

Задание 20* Моделирование автоматизированной технологической линии

1 Технологическая линия включает источник деталей, два взаимосвязанных станка, накопитель, технологический модуль для окончательной обработки деталей, рабочее место комплектации палет и транспортный робот для их транспортировки на склад.

2 Время поступления деталей из источника распределено равномерно на интервале $[a \pm \delta]$, причем деталь поступает в минимальную из очередей к станкам.

3 Если деталь поступает на станок 1, то затем она поступает на станок 2. Если деталь поступает на станок 2, то затем она поступает на станок 1. Время работы деталей на станках 1, 2 распределено равномерно на интервалах $[b_1 \pm \varepsilon_1]$ и $[b_2 \pm \varepsilon_2]$ соответственно.

4 После цикла механообработки деталь попадает в накопитель (на 10 деталей). Из накопителя все детали одновременно передаются в технологический модуль для окончательной обработки. Время обработки в технологическом модуле распределено равномерно на интервале $[b_3 \pm \varepsilon_3]$.

5 Затем осуществляется укладка деталей в палеты по 10 шт. Транспортный робот отбирает по 2 палеты и транспортирует их на склад. Время транспортировки распределено равномерно на интервале $[c \pm \gamma]$.

Цель. Разработать GPSS-модель для анализа процесса функционирования автоматизированной технологической линии.

Исходные данные, с:

а) $a = 10$, $\delta = 2$, $b_1 = 10$, $\varepsilon_1 = 4$, $b_2 = 9$, $\varepsilon_2 = 3$, $b_3 = 6$, $\varepsilon_3 = 2$, $c = 16$, $\gamma = 4$;

б) $a = 15$, $\delta = 5$, $b_1 = 12$, $\varepsilon_1 = 3$, $b_2 = 10$, $\varepsilon_2 = 2$, $b_3 = 5$, $\varepsilon_3 = 1$, $c = 10$, $\gamma = 3$.

Задание 21 Моделирование процесса обработки детали

1 В обрабатывающий цех поступают детали, обработка которых осуществляется двумя станками.

2 Детали, поступающие на первичную обработку, образуют пуассоновский поток с интенсивностью λ .

3 Первый станок обрабатывает деталь в среднем t_1 , мин, и имеет a_1 , %, брака, второй – соответственно t_2 , мин, и a_2 , %, брака. Все бракованные детали возвращаются на повторную обработку на первый станок. Интервалы времени обработки обоих станков имеют экспоненциальное распределение.

Цель. Построить GPSS-модель процесса обработки для n деталей. Определить среднюю загрузку первого станка на вторичной обработке и вероятность появления отходов (деталей, попавших в раздел бракованных после вторичной обработки).

Исходные данные:

а) $\lambda = 1$ дет./ч, $t_1 = 40$ мин, $t_2 = 60$ мин, $a_1 = 4$ %, $a_2 = 8$ %, $n = 500$ дет.

б) $\tau = 2$ дет./ч, $t_1 = 20$ мин, $t_2 = 40$ мин, $a_1 = 3$ %, $a_2 = 2$ %, $n = 1000$ дет.

Задание 22 Моделирование станции технического контроля

1 Собранные телевизионные приемники проходят серию испытаний на станциях технического контроля. На последней станции проверяют регулировку установки кадров по вертикали. Отбракованный телевизор отправляют в цех наладки.

2 Предположим, что телевизоры попадают на последнюю станцию через промежутки времени, распределенные по равномерному закону в интервале $[a_1, b_1]$. На станции находятся два контролера. Каждому из них требуется на проверку телевизора время, равномерно распределенное в интервале $[a_2, b_2]$.

3 Примерно α % телевизоров проходят проверку успешно и попадают в цех упаковки. Остальные β % попадают в цех наладки, в котором находится один наладчик. Наладка занимает время, равномерно распределенное в интервале $[a_3, b_3]$.

Цель. Построить GPSS-модель для анализа работы указанного подразделения производственной линии.

Исходные данные:

а) $\alpha = 85$ %, $\beta = 15$ %, $a_1 = 3$ мин, $b_1 = 7$ мин, $a_2 = 6$ мин, $b_2 = 12$ мин, $a_3 = 20$ мин, $b_3 = 40$ мин;

б) $\alpha = 90$ %, $\beta = 10$ %, $a_1 = 6$ мин, $b_1 = 8$ мин, $a_2 = 8$ мин, $b_2 = 14$ мин, $a_3 = 30$ мин, $b_3 = 50$ мин.

Задание 23 Моделирование работы морского порта

1 Морские суда прибывают в порт каждые $[a \pm \delta]$ часов.

2 В порту имеется N причалов. Каждый корабль по длине занимает M причалов и находится в порту $[b \pm \varepsilon]$ часов.

Цель. Построить GPSS-модель для анализа работы морского порта. Определить оптимальное количество причалов для эффективной работы порта.

Исходные данные:

а) $a = 20$ ч, $\delta = 5$ ч, $b = 10$ ч, $\varepsilon = 3$ ч, $N = 10$, $M = 3$;

б) $a = 30$ ч, $\delta = 10$ ч, $b = 8$ ч, $\varepsilon = 4$ ч, $N = 6$, $M = 2$.

Задание 24 Моделирование процесса обработки и покраски детали

1 В обрабатывающий цех поступают детали, обработка которых осуществляется последовательно на N станках. Детали, поступающие на обработку, образуют пуассоновский поток со средним интервалом между поступлением τ .

2 Время обработки на каждом из станков распределено равномерно на интервале $[m_i \pm h_i]$ ($i = \overline{1, N}$).

3 После обработки детали поступают в окрасочную камеру. Одновременно могут проходить покраску две детали. Время покраски деталей распределено равномерно на интервале $[a \pm b]$.

Цель. Разработать GPSS-модель для анализа процесса обработки деталей. Определить среднее время обработки и покраски деталей.

Исходные данные:

а) $N = 3$, $\tau = 7$ мин, $m_1 = 5$ мин, $h_1 = 2$ мин, $m_2 = 7$ мин, $h_2 = 2$ мин, $m_3 = 3$ мин, $h_3 = 1$ мин, $a = 10$ мин, $b = 2$ мин;

б) $N = 4$, $\tau = 15$ мин, $m_1 = 10$ мин, $h_1 = 3$ мин, $m_2 = 3$ мин, $h_2 = 0,5$ мин, $m_3 = 8$ мин, $h_3 = 2$ мин, $m_4 = 3$ мин, $h_4 = 1$ мин, $a = 5$ мин, $b = 1$ мин.

Задание 25 Моделирование процесса обслуживания заказов

1 На складе обслуживают заказы трех видов. Интервалы времени между поступающими заказами подчиняются нормальному закону с параметрами (m, σ) .

2 Вероятность появления заказа первого вида составляет α , второго вида – β , третьего вида – γ .

3 Все заказы выполняются одним из двух рабочих за время $[a \pm b]$ минут.

Цель. Разработать GPSS-модель для анализа процесса обслуживания заказов. Определить оптимальное количество рабочих для обслуживания заказов.

Исходные данные:

а) $m = 12$, $\sigma = 2$ мин, $\alpha = 0,3$, $\beta = 0,55$, $\gamma = 0,15$, $a = 16$ мин, $b = 5$ мин;

б) $m = 20$, $\sigma = 5$ мин, $\alpha = 0,6$, $\beta = 0,1$, $\gamma = 0,3$, $a = 30$ мин, $b = 15$ мин.

Задание 26 Моделирование двух вариантов технологического процесса

1 Объект моделирования включает технологический модуль, транспортный робот и склад изделий.

2 Палеты с деталями поступают с технологического модуля через каждые $[a \pm \delta]$ минут. Транспортный робот отбирает одну палету, транспортирует ее на склад и возвращается назад, затрачивая $[b \pm \varepsilon]$ минут. В обоих случаях время распределено по равномерному закону.

Цель. Разработать GPSS-модель, реализовать две стратегии обслуживания: FIFO и LIFO. Выяснить, какая из стратегий является более предпочтительной.

Исходные данные, мин:

а) $a = 18$, $\delta = 5$, $b = 16$, $\varepsilon = 4$;

б) $a = 25$, $\delta = 5$, $b = 23$, $\varepsilon = 1$.

Задание 27 Моделирование станции технического обслуживания

1 На станцию технического обслуживания (СТО) прибывают автомобили для технического обслуживания (α , %, автомобилей) и ремонта (β , %, автомобилей). Время прибытия автомобилей на СТО – случайная величина, распределенная равномерно на интервале $[a, b]$.

2 На СТО есть m боксов для технического обслуживания и n боксов для ремонта автомобилей.

3 Время технического обслуживания и ремонта автомобилей – случайные величины, имеющие равномерный закон распределения на интервалах $[m_i, h_i]$ ($i = \overline{1, 2}$) соответственно.

4 После технического обслуживания γ , %, автомобилей поступают для выполнения ремонта.

Цель. Построить GPSS-модель для анализа работы станции технического обслуживания. Определить оптимальное число боксов для технического обслуживания и ремонта.

Исходные данные:

а) $a = 12$ мин, $b = 18$ мин, $\alpha = 36$ %, $\beta = 64$ %, $\gamma = 14$ %, $m = 2$, $n = 3$, $m_1 = 10$ мин, $h_1 = 55$ мин, $m_2 = 50$ мин, $h_2 = 110$ мин;

б) $a = 15$ мин, $b = 25$ мин, $\alpha = 50$ %, $\beta = 50$ %, $\gamma = 10$ %, $m = 3$, $n = 3$, $m_1 = 10$ мин, $h_1 = 55$ мин, $m_2 = 50$ мин, $h_2 = 110$ мин.

Задание 28 Моделирование работы транспортного цеха

1 Транспортный цех объединения, имеющий m грузовиков, обслуживает три филиала A , B и C . В момент начала работы грузовики находятся в филиале A . Грузовики перевозят изделия A в B и из B в C , возвращаясь потом в A без груза.

2 Погрузка в A занимает в среднем t_1 , переезд из A в B длится t_2 , разгрузка (погрузка) в B – t_3 , переезд в C – t_4 , разгрузка в C – t_5 , и переезд в A – t_6 (всюду распределение экспоненциальное).

Цель. Разработать GPSS-модель работы цеха. Определить оптимальное число грузовиков для обслуживания филиалов.

Исходные данные:

а) $m = 8$, $t_1 = 20$ мин, $t_2 = 30$ мин, $t_3 = 20$ мин, $t_4 = 30$ мин, $t_5 = 20$ мин, $t_6 = 20$ мин;

б) $m = 10$, $t_1 = 20$ мин, $t_2 = 30$ мин, $t_3 = 40$ мин, $t_4 = 30$ мин, $t_5 = 20$ мин, $t_6 = 30$ мин.

Задание 29 Моделирование процесса сборки и обжига деталей

1 Изготовление деталей определенного вида включает длительный процесс сборки, который заканчивается коротким периодом обжига в печи.

2 Поскольку эксплуатация печи обходится очень дорого, несколько сборщиков используют одну печь, в которой одновременно можно обжигать только одну деталь. Сборщик не может начать новую сборку, пока не вытащит из печи предыдущую деталь.

3 Сборщик работает в следующем режиме: собирает деталь, ожидает возможности использования печи по принципу FIFO, использует печь, переходит к сборке новой детали.

4 Время, необходимое для сборки и обжига деталей – случайные величины, имеющие равномерный закон распределения на интервалах $[m_i \pm h_i]$ ($i = \overline{1,2}$) соответственно.

Цель. Разработать GPSS-модель описанного процесса. Определить оптимальное число сборщиков, использующих одну печь, т. е. такое количество, которое позволяет обработать наибольшее количество деталей.

Исходные данные, мин:

а) $m_1 = 30, h_1 = 5, m_2 = 8, h_2 = 2;$

б) $m_1 = 10, h_1 = 3, m_2 = 6, h_2 = 1.$

Задание 30 Моделирование процесса дозаправки самолетов

1 Поток самолетов, требующих посадки в аэропорту, – пуассоновский с интенсивностью λ .

2 В аэропорту есть n посадочных полос. Самолет, совершив посадку на полосу, освобождает ее через $t_1 \pm t_2$ минут.

3 Если самолет, требующий посадки, застает все полосы занятыми, то он становится в «очередь» самолетов, ожидающих посадки.

4 Через $t_3 \pm t_4$ минут после затребования посадки самолет нуждается в дозаправке. После безуспешного ожидания самолет отправляется на посадку в другой аэропорт.

Цель. Разработать GPSS-модель функционирования дозаправки самолетов. Определить оптимальное количество посадочных полос.

Исходные данные:

а) $\lambda = 10$ сам./ч, $n = 2, t_1 = 35$ мин, $t_2 = 5$ мин, $t_3 = 140$ мин, $t_4 = 20$ мин.

б) $\lambda = 20$ сам./ч, $n = 4, t_1 = 90$ мин, $t_2 = 15$ мин, $t_3 = 150$ мин, $t_4 = 25$ мин.

Задание 31 Моделирование процесса работы склада

1 На складе работает один кладовщик. Он выдает запасные части механикам, обслуживающим станки и устанавливающим эти части на испорченных станках.

2 Время, необходимое для удовлетворения запроса, зависит от типа запасной части.

3 Запросы бывают двух категорий. Интервал времени прихода механиков с запросом первой категории $[a_1 \pm \delta_1]$ секунд, время обслуживания запросов первой категории $[b_1 \pm \varepsilon_1]$ секунд. Интервал времени прихода механиков с запросом второй категории $[a_2 \pm \delta_2]$ секунд, а время обслуживания запросов второй категории $[b_2 \pm \varepsilon_2]$ секунд.

4 Порядок обслуживания механиков кладовщиком такой: запросы первой категории обслуживаются только в том случае, когда в очереди нет ни одного запроса второй категории.

Цель. Построить GPSS-модель, имитирующую работу склада. Оценить время ожидания в очереди запросов механиков различных категорий.

Исходные данные, с:

а) $a_1 = 420$, $\delta_1 = 360$, $a_2 = 360$, $\delta_2 = 240$, $b_1 = 300$, $\varepsilon_1 = 90$, $b_2 = 100$, $\varepsilon_2 = 30$;

б) $a_1 = 250$, $\delta_1 = 180$, $a_2 = 370$, $\delta_2 = 100$, $b_1 = 90$, $\varepsilon_1 = 20$, $b_2 = 120$, $\varepsilon_2 = 10$.

Задание 32* Моделирование процесса погрузки (разгрузки) судов в порту

1 Морские суда двух типов прибывают в порт, где происходит их разгрузка. Длины временных промежутков между прибытиями судов первого типа распределены по равномерному закону на интервале $[a_1 \pm \delta_1]$, судов второго типа – на интервале $[a_2 \pm \delta_2]$.

2 В порту есть два буксира, обеспечивающих ввод и вывод кораблей из порта. Время входа в порт судов первого типа и время выхода из порта равномерно распределены на интервалах $[b_1 \pm \varepsilon_1]$ и $[b_2 \pm \varepsilon_2]$, судов второго типа равномерно распределены на интервалах $[b_3 \pm \varepsilon_3]$ и $[b_4 \pm \varepsilon_4]$.

3 К первому типу судов относятся корабли малого тоннажа, которые требуют использования одного буксира. Корабли второго типа имеют большие размеры, и для их ввода и вывода из порта требуются два буксира.

4 Для кораблей первого типа доступны p_1 причалов, для кораблей второго типа – p_2 причалов. 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.

5 Корабли имеют различное время погрузки (разгрузки). Время обслуживания судна первого типа в порту – случайная величина, распределенная равномерно на интервале $[c_1 \pm \gamma_1]$, второго типа – $[c_2 \pm \gamma_2]$.

Цель. Построить GPSS-модель, имитирующую работу порта, оценить время ожидания кораблями каждого типа входа в порт (время ожидания входа в порт включает время ожидания освобождения причала и буксира).

Исходные данные:

а) $a_1 = 130$ мин, $\delta_1 = 30$ мин, $a_2 = 390$ мин, $\delta_2 = 60$ мин, $b_1 = 30$ мин, $\varepsilon_1 = 7$ мин, $b_2 = 20$ мин, $\varepsilon_2 = 5$ мин, $b_3 = 45$ мин, $\varepsilon_3 = 12$ мин, $b_4 = 35$ мин, $\varepsilon_4 = 10$ мин, $p_1 = 6$, $p_2 = 3$, $c_1 = 12$ ч, $\gamma_1 = 2$ ч, $c_2 = 18$ ч, $\gamma_2 = 4$ ч;

б) $a_1 = 160$ мин, $\delta_1 = 20$ мин, $a_2 = 250$ мин, $\delta_2 = 50$ мин, $b_1 = 25$ мин, $\varepsilon_1 = 5$ мин, $b_2 = 27$ мин, $\varepsilon_2 = 7$ мин, $b_3 = 35$ мин, $\varepsilon_3 = 10$ мин, $b_4 = 43$ мин, $\varepsilon_4 = 13$ мин, $p_1 = 5$, $p_2 = 4$, $c_1 = 10$ ч, $\gamma_1 = 4$ ч, $c_2 = 20$ ч, $\gamma_2 = 3$ ч.

Задание 33* Моделирование процесса функционирования грузового аэропорта

1 Грузы прибывают для отправки в аэропорт в контейнерах. Длины временных промежутков между прибытиями двух контейнеров распределены равномерно на интервале $[\tau_1, \tau_2]$.

2 В грузовом аэропорту нет фиксированного расписания, а каждый самолет отправляется в момент его полной загрузки.

3 Для перевозки грузов из данного аэропорта в аэропорт назначения используют два типа самолетов: N_1 самолетов с грузоподъемностью m контейнеров и N_2 самолетов повышенной грузоподъемности $2m$ контейнеров. Время полета каждого самолета туда и обратно – случайная величина, распределенная равномерно на интервале $[\gamma_1, \gamma_2]$.

4 Самолеты повышенной грузоподъемности используются лишь при отсутствии самолетов первого типа.

5 Время погрузки каждого контейнера в самолет случайно и имеет равномерное распределение на интервале $[\rho_1, \rho_2]$.

Цель. Построить GPSS-модель, имитирующую работу грузового аэропорта, оценить среднее время простоя грузов, среднее время простоя самолетов.

Исходные данные:

а) $\tau_1 = 0,5$ мин, $\tau_2 = 1,5$ мин, $N_1 = 3$, $N_2 = 2$, $m = 80$, $\gamma_1 = 2$ ч, $\gamma_2 = 4$ мин, $\rho_1 = 1$ мин, $\rho_2 = 2$ мин;

б) $\tau_1 = 2$ мин, $\tau_2 = 6$ мин, $N_1 = 4$, $N_2 = 1$, $m = 70$, $\gamma_1 = 3$ ч, $\gamma_2 = 5$ мин, $\rho_1 = 2$ мин, $\rho_2 = 3$ мин.

ПРИЛОЖЕНИЕ А
(справочное)

Системные и стандартные числовые атрибуты

Таблица А.1 – Системные числовые атрибуты

Обозначение	Описание системных числовых атрибутов
RN _j	Число, возвращаемое <i>j</i> -м датчиком случайных чисел
C1	Текущее значение относительного модельного времени
AC1	Текущее значение абсолютного модельного времени
TG1	Текущее значение счетчика завершений
XN1	Номер активного транзакта (обрабатываемого в данный момент)
M1	Время пребывания в модели активного транзакта

Таблица А.2 – Стандартные числовые атрибуты (СЧА)

Обозначение	Описание СЧА
<i>СЧА транзактов</i>	
<i>P_j</i>	Значение <i>j</i> -го параметра текущего транзакта
<i>PF_j</i>	Значение <i>j</i> -го параметра текущего транзакта форматом «слово»
<i>PH_j</i>	Значение <i>j</i> -го параметра текущего транзакта форматом «полуслово»
<i>PB_j</i>	Значение <i>j</i> -го параметра текущего транзакта форматом «байт»
<i>PL_j</i>	Значение <i>j</i> -го параметра текущего транзакта форматом «плавающая точка»
<i>PR</i>	Приоритет активного транзакта
<i>MP_j</i>	Значение, равное разности абсолютного модельного времени, и значения <i>j</i> -го параметра текущего транзакта
<i>MB_j</i>	Флаг синхронизации: 1, если транзакт в блоке <i>j</i> принадлежит тому же семейству, что и текущий транзакт; 0 – в противном случае
<i>СЧА блоков</i>	
<i>W_j</i>	Количество транзактов, находящихся в блоке с номером <i>j</i> в текущий момент модельного времени
<i>N_j</i>	Общее количество транзактов, поступивших в блок с номером <i>j</i>
<i>СЧА одноканальных обслуживающих устройств</i>	
<i>F_j</i>	Текущее состояние устройства <i>j</i> : 1, если устройство занято; иначе – 0.
<i>FR_j</i>	Коэффициент использования устройства в тысячных долях
<i>FC_j</i>	Общее число транзактов, вошедших в устройство <i>j</i>
<i>FT_j</i>	Среднее время использования устройства одним транзактом
<i>СЧА многоканальных обслуживающих устройств</i>	
<i>S_j</i>	Текущее содержимое многоканального устройства <i>j</i>
<i>R_j</i>	Число свободных единиц многоканального устройства
<i>SR_j</i>	Коэффициент использования многоканального устройства в тысячных долях

Окончание таблицы А.2

Обозначение	Описание СЧА
<i>СЧА многоканальных обслуживающих устройств</i>	
<i>SA_j</i>	Среднее содержимое многоканального устройства <i>j</i> (целая часть)
<i>SM_j</i>	Максимальное содержимое многоканального устройства <i>j</i>
<i>SC_j</i>	Общее число транзактов, вошедших в многоканальное устройство <i>j</i>
<i>ST_j</i>	Среднее время пребывания транзактов в многоканальном устройстве <i>j</i>
<i>SE_j</i>	Признак пустоты многоканального устройства <i>j</i> : 1 – пусто, 0 – в противном случае
<i>SF_j</i>	Признак заполненности многоканального устройства <i>j</i> : 1 – заполнено, 0 – в противном случае
<i>СЧА логических ключей</i>	
<i>LS_j</i>	Состояние логического ключа с номером <i>j</i> : 1 – включен, 0 – выключен
<i>СЧА переменных и функций</i>	
<i>I_j</i>	Значение арифметической переменной <i>j</i>
<i>BV_j</i>	Значение логической переменной <i>j</i> (1 – истина, 0 – ложь)
<i>FN_j</i>	Значение функции <i>j</i> (дробная часть отбрасывается за исключением использования в качестве аргумента другой функцией)
<i>СЧА очередей</i>	
<i>Q_j</i>	Текущая длина очереди <i>j</i>
<i>QA_j</i>	Средняя длина очереди <i>j</i>
<i>QM_j</i>	Максимальная длина очереди <i>j</i>
<i>QC_j</i>	Общее число входов в очередь <i>j</i>
<i>QZ_j</i>	Число нулевых входов в очередь <i>j</i>
<i>QT_j</i>	Среднее время пребывания транзактов в очереди <i>j</i> (включая транзакты, прошедшие очередь без ожидания)
<i>QX_j</i>	Среднее время пребывания транзактов в очереди <i>j</i> (исключая транзакты, прошедшие очередь без ожидания)
<i>СЧА таблиц</i>	
<i>TB_j</i>	Среднее арифметическое значение элементов таблицы <i>j</i>
<i>TC_j</i>	Общее число элементов в таблице <i>j</i>
<i>TD_j</i>	Среднеквадратическое отклонение значения элементов таблицы <i>j</i>
<i>СЧА ячеек и матриц ячеек</i>	
<i>X_j</i>	Содержимое ячейки <i>j</i>
<i>MX_j(a, b)</i>	Содержимое элемента матрицы ячеек <i>j</i> , расположенного в строке <i>a</i> и столбце <i>b</i>
<i>СЧА списка пользователя</i>	
<i>CH_j</i>	Текущее число транзактов в <i>j</i> -м списке пользователя
<i>CA_j</i>	Среднее число транзактов в <i>j</i> -м списке пользователя
<i>CM_j</i>	Максимальное число транзактов в <i>j</i> -м списке пользователя
<i>CC_j</i>	Общее число транзактов, которые поступали в <i>j</i> -й список пользователя
<i>CT_j</i>	Среднее время пребывания транзактов в <i>j</i> -м списке пользователя

ПРИЛОЖЕНИЕ Б
(справочное)
Блоки GPSS

Таблица Б.1 – Блоки GPSS

Код и назначение	Операнд	Содержание операнда
ADVANCE Задержка транзакта	A, [B]	Среднее время Модификатор-интервал или модификатор-функция: <i>FNj, FN\$<имя>, FN*j</i>
ASSEMBLE Объединение транзактов	A	Число транзактов из одного семейства, объединяемых в один транзакт
ASSIGN Изменение значений параметров	A B C	Номер или диапазон изменения номеров параметров и указатель режима (накопления – «+», вычитания – «-», замещение – по умолчанию) Значение, на которое изменяется параметр (параметры). Номер модификатора-функции
COUNT XXX Определение количества объектов из заданного множества, удовлетворяющих заданному условию	A B C D [E]	Номер параметра-счетчика объектов Нижняя граница диапазона номеров объектов Верхняя граница диапазона номеров объектов Значение, сравнимое со значением СЧА в поле E Код СЧА анализируемых объектов (без номера и имени)
DEPART Уменьшение очереди	A [B]	Номер (имя) очереди Число единиц, на которое уменьшается длина очереди (не превосходит текущей длины очереди)
ENTER Занятие многоканального устройства	A [B]	Номер (имя) многоканального устройства Число занимаемых каналов
GATE XXX Выбор направления в зависимости от состояния объекта	A [B]	Номер (имя) устройства, ключа Номер (метка) альтернативного блока в случае невыполнения проверяемого условия
GENERATE Создание и запуск транзактов в модель с заданным законом времени поступления	[A] [B] [C] [D] [E]	Среднее время поступления транзактов Модификатор-интервал или модификатор-функция (см. ADVANCE) Время задержки первого транзакта Количество создаваемых блоком транзактов Приоритет транзактов
LEAVE Освобождение многоканального устройства	A [B]	Номер (имя) устройства Число освобождаемых транзактом каналов

Продолжение таблицы Б.1

Код и назначение	Операнд	Содержание операнда
LINK Перевод транзакта из списка текущих событий в список пользователей	A B [C]	Номер (имя) списка пользователя Режим упорядочивания списка; FIFO – «первым пришел – первым обслужился», LIFO – «последним пришел – первым обслужился», номер параметра Номер (метка) альтернативного блока
LOGIC X Изменения состояния ключа $X = \{S, R, I\}$	A	Номер (имя) логического ключа, состояние которого изменяется: S – включается, R – выключается, I – инвертируется
LOOP Организация циклов	A [B]	Номер (имя) параметра транзакта, используемого в качестве счетчика цикла Номер (метка) первого блока в цикле
MATCH Синхронизация транзактов	A	Номер (метка) сопряженного блока MATCH Метка исходного блока MATCH указывается в поле A сопряженного блока
PRIORITY Изменение приоритета транзакта	A	Новое значение приоритета транзакта
PREEMPT Прерывание обслуживания устройством одного обслуживания транзакта с целью захвата устройства прерывающим транзактом	A [B] [C] [D] [E]	Номер (имя) прерываемого устройства Задание режима прерывания: по приоритету – PR, обычный – операнд опущен Номер (метка) блока для прерванного транзакта Номер (имя) параметра прерванного транзакта для записи остатка времени RE – режим удаления прерванного транзакта с направлением по адресу в поле C
QUEUE Увеличение очереди	A [B]	Номер (имя) очереди Число единиц, на которое увеличивается длина очереди
RELEASE Освобождение устройства	A	Номер (имя) освобождаемого устройства транзактом, который его занимал
RETURN Снятие прерывания с устройства	A	Номер (имя) устройства, на котором снимается прерывание транзактом, осуществляющим прерывание
SAVEVALUE Изменение значений ячеек	A B	Номер (имя) ячейки и указатель режима: накопления – «+», вычитания – «-», замещение – по умолчанию Значение, на которое изменяется содержимое ячейки
SEIZE Занятие устройства	A	Номер (имя) занимаемого устройства

Окончание таблицы Б.1

Код и назначение	Операнд	Содержание операнда
SELECT XXX Выбор первого из множества объектов, удовлетворяющего заданному условию	A	Номер параметра-счетчика объектов
	B	Нижняя граница диапазона номеров объектов
	C	Верхняя граница диапазона номеров объектов
	[D]	Значение, сравнимое со значением СЧА в поле E
	[E]	Код СЧА анализируемых объектов
	[F]	Номер (метка) альтернативного блока в случае невыполнения проверяемого условия
SELECT MAX (MIN) Выбор первого из множества объектов, удовлетворяющего заданному условию	A	Номер параметра-счетчика объектов
	B	Нижняя граница диапазона номеров объектов
	C	Верхняя граница диапазона номеров объектов
	[D]	Значение, сравнимое со значением СЧА в поле E
	[E]	Код СЧА анализируемых объектов (без номера и имени)
	[F]	Номер (метка) альтернативного блока в случае невыполнения проверяемого условия MAX (MIN) – поиск объектов со значением указанного СЧА
SPLIT Создание копий исходного транзакта	A	Число создаваемых копий
	[B]	Метка блока, к которому направляется копия
	[C]	Номер (имя) параметра для присвоения копиям последовательных номеров
TABULATE Табуляция текущего значения аргумента таблицы	A	Номер (имя) таблицы
	[B]	Число добавляемых в соответствующий интервал таблицы единиц
TEST YY Выбор направления движения транзакта в зависимости от значений СЧА	A, B	Величины, сравниваемые с помощью операции отношения $YY = \{ L, LE, G, GE, E, NE \}$
	[C]	Номер (имя) альтернативного блока при невыполнении проверяемого условия
TRANSFER Выбор направления движения транзактов из множества возможных направлений	[A]	Режим выбора направления: «,» – режим безусловного перехода; «BOTH» – выбор двух направлений; «ALL» – выбор из более чем двух направлений; «K» – выбор с вероятностью K
	[B],[C]	Значение номеров (меток) блоков
	[D]	Константа, используемая для вычисления возможных адресов в режиме «ALL»
TERMINATE Удаление транзактов из модели	[A]	Число единиц, на которое уменьшается счетчик завершений (поле A команды START)
UNLINK Удаление транзактов из списка пользователя	A	Номер (имя) списка пользователя
	B	Номер (метка) блока следования транзактов, удаляемых из списка
	[C]	Счетчик числа удаляемых транзактов («ALL» – удаление всех транзактов)

Таблица Б.2 – Значения логического указателя блоков COUNT, GATE, SELECT

Тип объекта	Логический указатель	Значение
Одноканальное устройство	FV	Устройство занято
	FNV	Устройство не занято
	FI	Устройство обслуживает прерывание
	FNI	Устройство не обслуживает прерывание
Многоканальное устройство	SF	Устройство заполнено
	SNF	Устройство не заполнено
	SE	Устройство пусто
	SNE	Устройство не пусто
Логический ключ	LR	Логический переключатель сброшен
	LS	Логический переключатель установлен

СПИСОК ЛИТЕРАТУРЫ

1 **Задачи и модели исследования операций.** Ч. 3. Технология имитации на ЭВМ и принятие решений : учеб. пособие / И. В. Максимей [и др.]. – Гомель : БелГУТ, 1999. – 150 с.

2 **Кельтон, В.** Имитационное моделирование / В. Кельтон, А. Лоу. – 3-е изд. – СПб.: Питер, 2004. – 848 с.

3 **Максимей, И. В.** Имитационное моделирование на ЭВМ / И. В. Максимей. – М. : Радио и связь, 1988. – 232 с.

4 **Томашевский, В. Н.** Имитационное моделирование в среде GPSS / В. Н. Томашевский, Е. Г. Жданова. – М. : Бестселлер, 2003. – 416 с.

5 **Основы имитационного и статистического моделирования:** учеб. пособие / Ю. С. Харин [и др.]. – Минск : Дизайн ПРО, 1997. – 288 с.

6 **Шеннон, Р.** Имитационное моделирование систем – наука и искусство / Р. Шеннон. – М. : Мир, 1978. – 418 с.

7 **Шрайбер, Т. Дж.** Моделирование на GPSS / Т. Дж. Шрайбер. – М. : Машиностроение, 1980. – 593 с.

ПРЕДМЕТНЫЙ УКАЗАТЕЛЬ

Адекватность модели	15	Обслуживающее устройство	21
Блок GPSS	20	- многоканальное	21
		- одноканальное	21
Верификация	15	Оператор GPSS	23
Время	11	Очередь	22
- машинное	11	Стационарность	16
- модельное	11	Счетчик завершений	24
- реальное	11	Точность	15
Моделирование	5	Транзакт	19
- аналитическое	6	Устойчивость	16
- имитационное	7	Числовые атрибуты	19
- математическое	5	- стандартные	19
- физическое	5	- системные	19
Модель	5	Чувствительность	16
- вероятностная	6		
- детерминированная	6		
- дискретная	5		
- непрерывная	6		

МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ
УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ
«БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ТРАНСПОРТА»

Кафедра «Прикладная математика»

Д. Н. ШЕВЧЕНКО, И. Н. КРАВЧЕНЯ

ИМИТАЦИОННОЕ МОДЕЛИРОВАНИЕ НА GPSS

Учебно-методическое пособие для студентов
технических специальностей

*Одобрено методическими комиссиями факультетов
УПП и строительного*

Гомель 2007

Учебное издание

*ШЕВЧЕНКО Дмитрий Николаевич
КРАВЧЕНЯ Ирина Николаевна*

Имитационное моделирование на GPSS

Учебно-методическое пособие для студентов технических специальностей

Редактор Т. М. Ризевская
Технический редактор В. Н. Кучерова
Корректор М. П. Дежко
Компьютерный набор и верстка – Д. Н. Шевченко, И. Н. Кравченя

Подписано в печать 05.11.2007 г. Формат 60 × 84^{1/16}.
Бумага офсетная. Гарнитура Таймс. Печать на ризографе.
Усл. печ. л. 5,81. Уч.-изд. л. 5,97. Тираж 400 экз.
Зак. № . Изд. № 15

Издатель и полиграфическое исполнение
Белорусский государственный университет транспорта:
ЛИ № 02330/0133394 от 19.07.2004 г.
ЛП № 02330/0148780 от 30.04.2004 г.
246653, г. Гомель, ул. Кирова, 34.