

Министерство образования Республики Беларусь
Учреждение образования
«Белорусский государственный университет
информатики и радиоэлектроники»

Факультет компьютерного проектирования

Кафедра проектирования информационно-компьютерных систем

В. Ф. Алексеев, Г. А. Пискун, В. А. Перовщиков

**МОДЕЛИРОВАНИЕ И ОПТИМАЛЬНОЕ
ПРОЕКТИРОВАНИЕ ТЕХНИЧЕСКИХ СИСТЕМ:
ПОСОБИЕ К ПРАКТИЧЕСКИМ ЗАНЯТИЯМ**

*Рекомендовано УМО по образованию в области
приборостроения в качестве пособия для специальности
1-38 80 04 «Технология приборостроения»*

Минск БГУИР 2017

УДК 621.396.6-042.4(076)
ББК 32.844-02я73
А47

Р е ц е н з е н т ы:

кафедра экономической информатики учреждения образования
«Белорусский государственный экономический университет»
(протокол №11 от 25.05.2016);

директор ООО «ИЗОВАК Технологии»,
кандидат технических наук Н. А. Долгий

Алексеев, В. Ф.

А47 Моделирование и оптимальное проектирование технических систем: пособие к практическим занятиям : пособие / В. Ф. Алексеев, Г. А. Пискун, В. А. Перевощиков. – Минск : БГУИР, 2017. – 116 с. : ил.
ISBN 978-985-543-258-7.

Содержит 15 практических работ, в которых рассмотрены основные возможности среды моделирования GPSS World. Целью выполнения работ является изучение процесса проектирования и моделирования конструкторских и технологических решений при помощи программного продукта GPSS World.

Предназначено для магистрантов высших учебных заведений, может быть использовано студентами и инженерами, занимающимися вопросами моделирования и оптимального проектирования технических систем.

**УДК 621.396.6-042.4(076)
ББК 32.844-02я73**

ISBN 978-985-543-258-7

© Алексеев В. Ф., Пискун Г. А.,
Перевощиков В. А., 2017
© УО «Белорусский государственный
университет информатики
и радиоэлектроники», 2017

СОДЕРЖАНИЕ

Введение	4
Практическая работа №1. Программа GPSS World. Выполнение процесса моделирования.....	5
Практическая работа №2. Решение проблем. Текстовый редактор GPSS World.....	19
Практическая работа №3. Транзакты и блоки модели	26
Практическая работа №4. Очереди и Q-таблицы: блоки «QUEUE» и «DEPART»	34
Практическая работа №5. Команды GPSS World. Режим ручного моделирования	40
Практическая работа №6. Графические окна GPSS World	45
Практическая работа №7. Запись и чтение внешних файлов	60
Практическая работа №8. Библиотека процедур ANOVA.....	67
Практическая работа №9. Переменные. Отчеты. Ведение журнала сессий.....	76
Практическая работа №10. Непрерывное моделирование	82
Практическая работа №11. Язык программирования для моделирования PLUS.....	88
Практическая работа №12. Использование языка PLUS.....	94
Практическая работа №13. Отладка моделей.....	99
Практическая работа №14. Пользовательские эксперименты	104
Практическая работа №15. Генераторы экспериментов.....	111

ВВЕДЕНИЕ

Компьютерное моделирование является технологией предсказания того, как будет себя вести новая или измененная система. Оно имеет более широкий спектр применения, чем чисто математические методы, и на протяжении многих лет помогло своим пользователям сэкономить миллионы долларов. Даже наиболее сложные реально существующие системы могут быть смоделированы при помощи системы GPSS World. Перечень возможностей использования компьютерного моделирования довольно большой и при этом постоянно увеличивается.

Одним из первых языков моделирования, облегчающих процесс написания имитационных программ, был язык GPSS, созданный в виде конечного продукта Джеффри Гордоном в фирме IBM в 1962 г. Этот язык в свое время входил в первую десятку лучших языков программирования, опережая транслятор с языка ALGOL, и был реализован практически на всех типах ЭВМ. В настоящее время есть трансляторы для операционных систем DOS – GPSS/PC, для OS/2 и DOS – GPSS/H и для Windows – GPSS World. Изучение языка GPSS и создание моделей позволяют понять принципы разработки имитационных программ и научиться работать с имитационными моделями.

Система GPSS World прежде всего была предназначена для расширения возможностей пользователя. Она выносит все примитивы моделирования на поверхность пользовательского интерфейса, упрощая процесс визуализации и управления моделированием. Результатом стала возможность более быстрой разработки, тестирования и понимания моделей, чем прежде.

В данном пособии к практическим занятиям рассмотрены основные возможности среды моделирования GPSS World. Пособие разбито на 15 практических работ, при выполнении которых магистранты должны освоить навыки пользования системой GPSS World для построения и оптимизации моделей технических систем.

Практическая работа №1

ПРОГРАММА GPSS WORLD. ВЫПОЛНЕНИЕ ПРОЦЕССА МОДЕЛИРОВАНИЯ

Цель работы

Изучить интерфейс программы GPSS World и набор основных команд для управления процессом моделирования. Изучить имитационную модель в программе GPSS World.

1.1 Теоретическая часть

Система GPSS World, разработанная компанией Minuteman Software (США), – это мощная среда компьютерного моделирования общего назначения, адресованная профессионалам в области моделирования. Это комплексный моделирующий инструмент, охватывающий области как дискретного, так и непрерывного компьютерного моделирования, обладающий высочайшим уровнем интерактивности и визуального представления информации.

Использование GPSS World дает возможность оценить эффект конструкторских решений в чрезвычайно сложных системах реального мира.

GPSS World основан на оригинальном языке компьютерного моделирования GPSS, что означает General Purpose Simulation System – общецелевая система моделирования. В основном этот язык был разработан Джеффри Гордоном в 1960 г. в IBM и привнес множество важных концепций в каждую из коммерческих реализаций языков компьютерного моделирования дискретных событий, разработанных с тех пор. GPSS World – это прямое развитие языка моделирования GPSS/PC, одна из первых реализаций GPSS для персональных компьютеров. После своего появления в 1984 г. GPSS/PC и его последующие версии сохранили тысячам пользователей миллионы долларов. В настоящее время версия GPSS World для ОС Windows имеет расширенные возможности, включая пользовательскую среду с интегрированными функциями работы с Интернетом.

GPSS World разработан для оперативного получения достоверных результатов с наименьшими усилиями. В соответствии с этим в GPSS World хорошо проработана визуализация процесса моделирования, а также встроены элементы статистической обработки данных. Сильная сторона GPSS World – это его прозрачность для пользователя.

Прозрачность для пользователя ценна по трем причинам. Во-первых, опасно полагаться на непрозрачное моделирование типа «черный ящик», внутренние механизмы функционирования которого скрыты от пользователя, так как в этом случае нельзя быть уверенным, что оно подходит для какого-то конкретного случая, и невозможно гарантировать, что оно работает, как задумано. Во-вторых, удачные имитационные модели являются очень ценными и пригодны в течение длительного периода времени. Возможно, потребуется, чтобы но-

вые сотрудники ознакомились с внутренними процессами модели, а это почти невозможная задача, если модель не имеет высокого уровня прозрачности. В-третьих, одним из наиболее эффективных, но наименее известных преимуществ компьютерного имитационного моделирования является возможность проникновения в самую суть поведения системы, когда опытный профессионал в области моделирования может видеть внутреннюю динамику в наиболее важные моменты модельного времени процесса моделирования.

GPSS World был разработан с целью решить все эти проблемы и является объектно-ориентированным языком. Его возможности визуального представления информации позволяют наблюдать и фиксировать внутренние механизмы функционирования моделей. Его интерактивность позволяет одновременно исследовать процессы моделирования и управлять ими. С помощью встроенных средств анализа данных можно легко вычислить доверительные интервалы и провести дисперсионный анализ. Кроме того, теперь есть возможность автоматически создавать и выполнять сложные отсеивающие и оптимизирующие эксперименты.

GPSS World максимально использует возможности вашей вычислительной системы. Использование механизма виртуальной памяти позволяет моделям достигать размера миллиарда байтов. Вытесняющая многозадачность и многопоточность обеспечивают высокую скорость реакции на управляющие воздействия и дают возможность GPSS World одновременно выполнять множество задач. Это также означает, что система моделирования GPSS World может использовать вычислительные возможности, предоставляемые симметричными многопроцессорными архитектурами (SMP).

GPSS World сочетает в себе функции дискретного и непрерывного моделирования. Возможность перехода из дискретной фазы моделирования в непрерывную и обратно обеспечивает тесную связь с непрерывным моделированием. В непрерывной фазе могут быть установлены пороговые значения, управляющие созданием транзактов в дискретной фазе.

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, 19 из них описывают связи между элементами. Выделенным множествам элементов и операций ставится в соответствие множество объектов GPSS.

Все объекты в GPSS имеют свойства, называемые стандартными числовыми атрибутами (СЧА). А свойства модели 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; C1 – текущее значение относительного модельного времени (автоматически изменяется GPSS и устанавливается в 0 командами CLEAR или RESET); AC1 – текущее значение абсолютного модельного времени (автоматически изменяется GPSS и устанавливается в 0 командой CLEAR); TG1 – текущее значение счетчика завершений; M1 – 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.

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.

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).

Многоканальные устройства («**Storage**») описывают оборудование, которое может использоваться несколькими транзактами одновременно (многоканальные системы массового обслуживания).

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

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.

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

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 среднеквадратичное отклонение значений.

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 23 объекта либо считать их информацию.

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.

1.2 Практическая часть

Задание 1.1: изучить основные элементы интерфейса программы GPSS World.

Для запуска программы GPSS World требуется открыть меню «Пуск», подменю «Программы» или «Все программы», «GPSS World», либо воспользоваться значком программы на рабочем столе компьютера. Главное окно программы показано на рисунке 1.1.

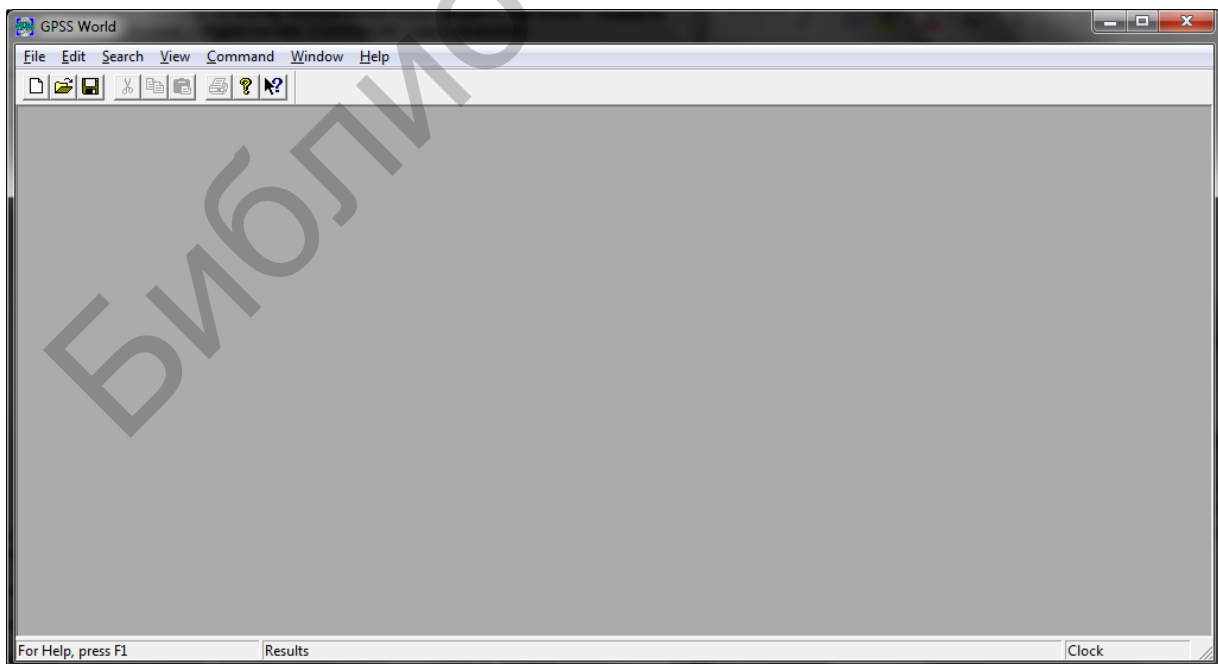


Рисунок 1.1 – Главное окно программы GPSS World

Для открытия файла модели нужно воспользоваться вкладкой «File» и командой «OPEN» либо сочетанием клавиш [Ctrl]+[O] (рисунок 1.2).

Чтобы открыть файл модели, требуется перейти в папку «Sample Models». Данная папка расположена в корне папки с программой GPSS World. При открытии отобразится список файлов с примерами, которые доступны для использования. Модель открывается двойным щелчком мыши. Выполним открытие модели «Sample1.GPS» (рисунок 1.3).

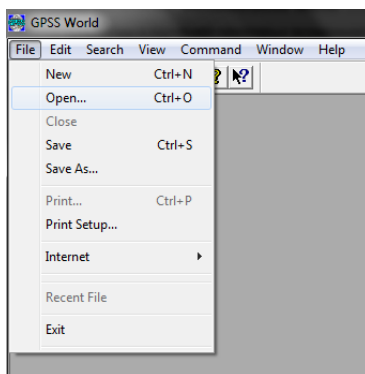


Рисунок 1.2 – Меню «File» главного окна

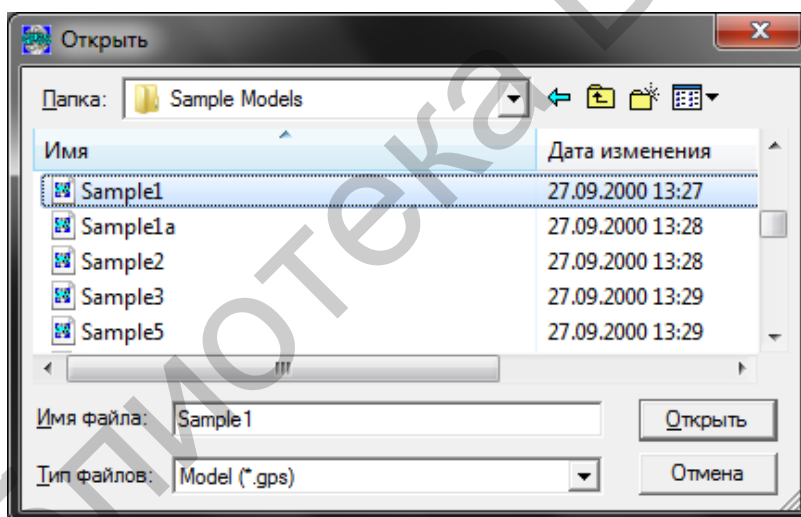


Рисунок 1.3 – Диалоговое окно открытия файла модели

Файл модели «Sample1.GPS» показан на рисунке 1.4. GPSS World использует порядковые номера строк, назначаемые полноэкранным текстовым редактором, для указания операторов, связанных с ошибками или другими условиями. Навигация по строкам модели осуществляется командами в меню «Search/Go to line».

Сохраните файл при помощи команды «File/Save as...».

Поменяйте имя файла модели на TMP (рисунок 1.5). После сохранения добавьте модель блоком «BUFFER», расположив его после блока «GENERATE». Для этого расположите текстовый курсор в любом месте строки «QUEUE Barber».

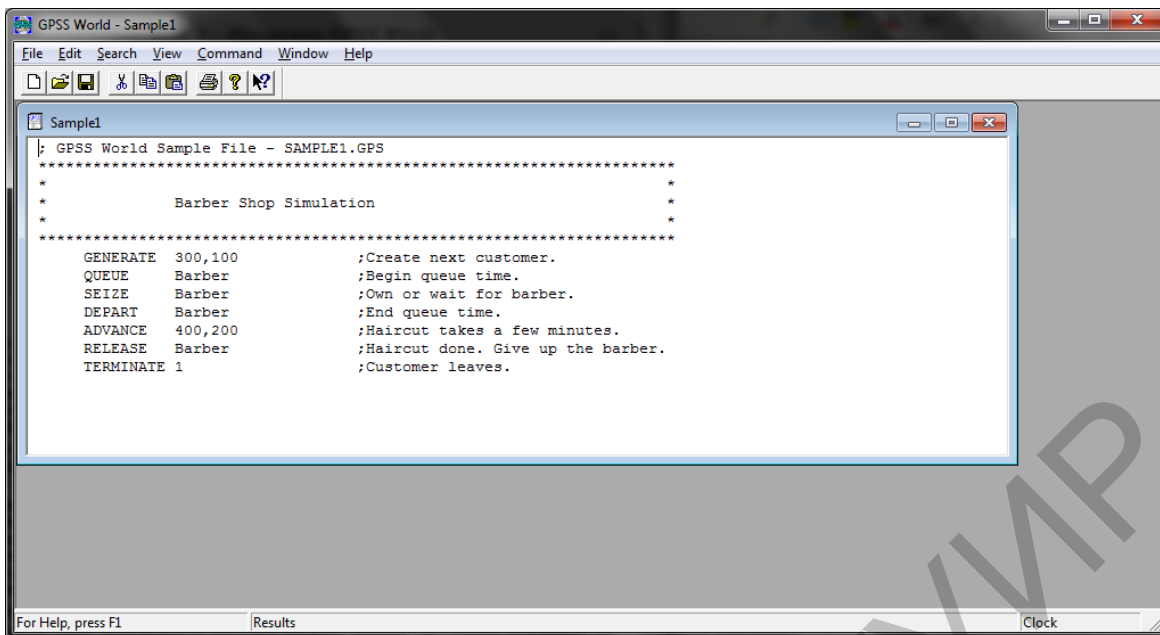


Рисунок 1.4 – Окно модели «Sample1.GPS», обзор текста

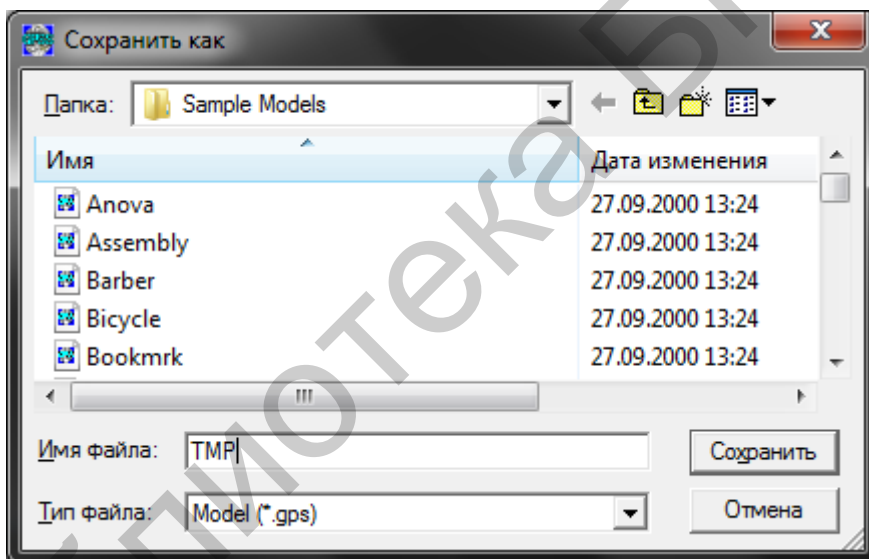


Рисунок 1.5 – Диалоговое окно «Сохранить как»

Далее выберите «Edit/Insert Line» или нажмите сочетание клавиш [Ctrl]+[I]. После этого поместите курсор в начало пустой строки.

Выравнивать строки и операторы можно при помощи табуляции или пробелов, однако в этом нет необходимости и воспользоваться этим стоит лишь для удобства последующего чтения. Наберите в пустой строке «BUFFER». Регистр букв также не имеет значения, однако это делается, чтобы выделить зарезервированные ключевые слова программы.

Теперь проверьте настройки модели. Для этого выделите «Edit/Settings». Появится страница «Simulation» журнала настроек модели. Не меняя настроек на странице «Simulation», выберите страницу «Reports» (рисунок 1.6).

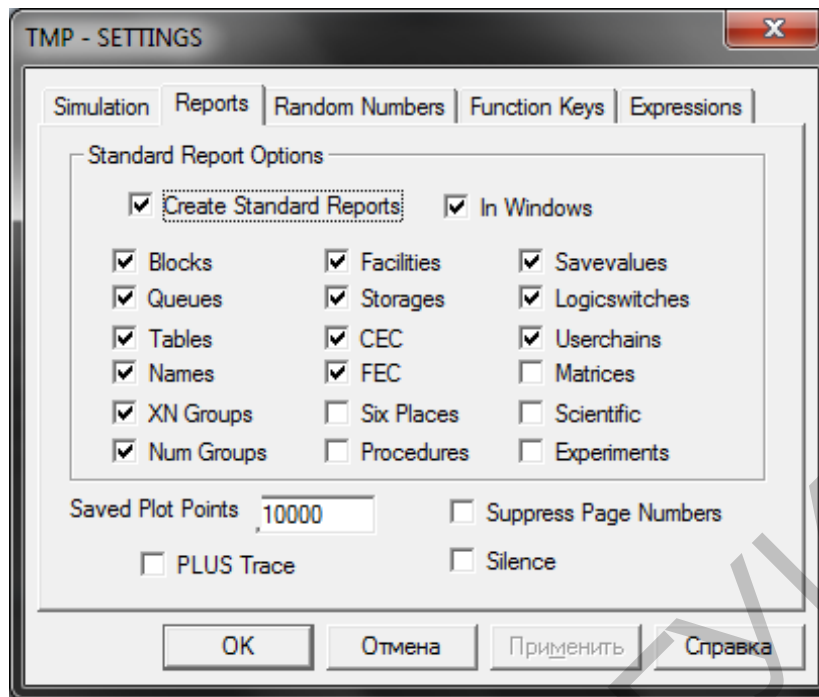


Рисунок 1.6 – Страница «Reports» журнала настроек модели

Страницы настроек модели «Random Numbers», «Function Keys», «Expressions» содержат опции, позволяющие назначить использование конкретных генераторов случайных чисел, связать необходимые операции с конкретными функциональными клавишами и соответственно добавить выражения. Чтобы выйти из журнала настроек, нажмите «OK».

Создайте процесс моделирования. Для этого требуется выбрать команду «Command/Create Simulation». Данное действие приведет к трансляции модели в процесс моделирования. Если в модели имеются синтаксические ошибки, то они будут обнаружены и выделены для исправления.

После запуска процесса моделирования можно изменить настройки в окне «Blocks». Для этого требуется выбрать меню «Window/Simulation Window/Blocks Window». После этого в окне, которое появится поверх окна модели и окна «Journal», будет отображаться подробный обзор блоков.

Окно блоков динамически обновляется в процессе моделирования. При помощи команды меню «View/Entity Details» можно скрывать и отображать подробности в окне блоков. Данная опция включена по умолчанию.

Запустите процесс моделирования и проследите за его выполнением с помощью окна «Blocks». Для этого введите интерактивную команду «START 300» при помощи меню «Command/START». В появившемся диалоговом окне измените значение 1 на 300 и нажмите «OK». Параметры в окне «Blocks» начнут быстро изменяться (рисунок 1.7). Важно помнить, что динамические окна всегда замедляют процесс выполнения моделирования.

После завершения моделирования откроется окно «Report» (рисунок 1.8).

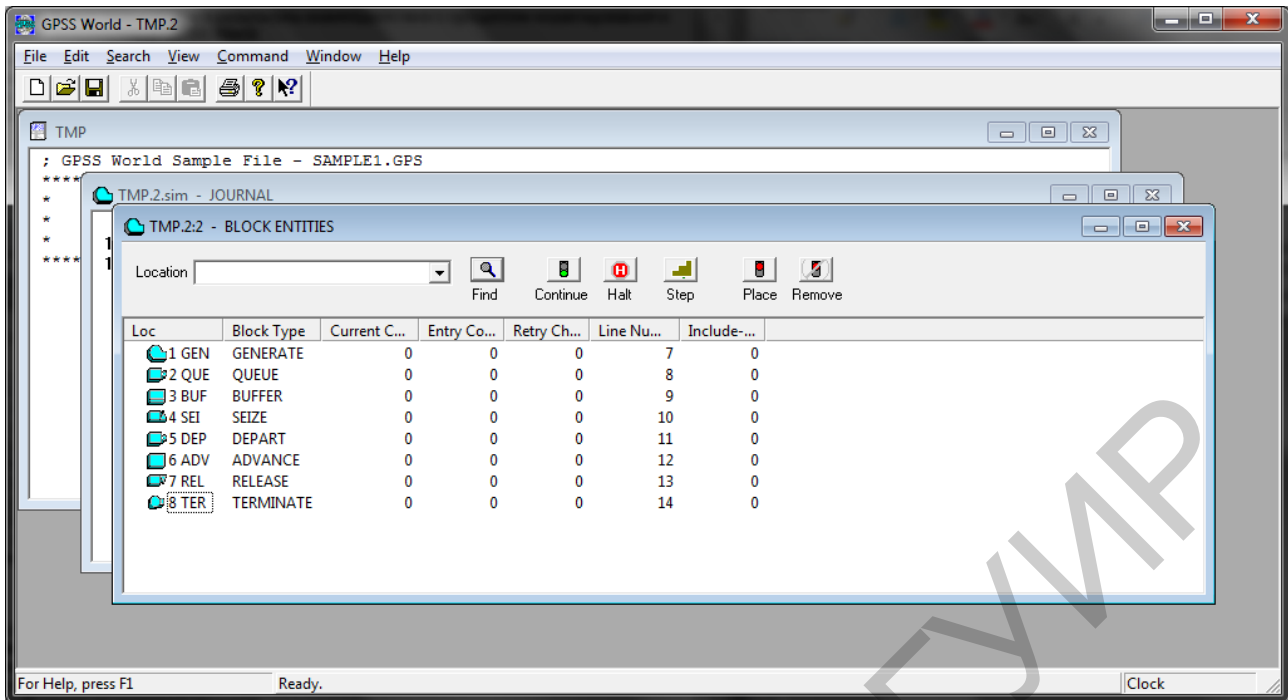


Рисунок 1.7 – Окно «Block Entities»

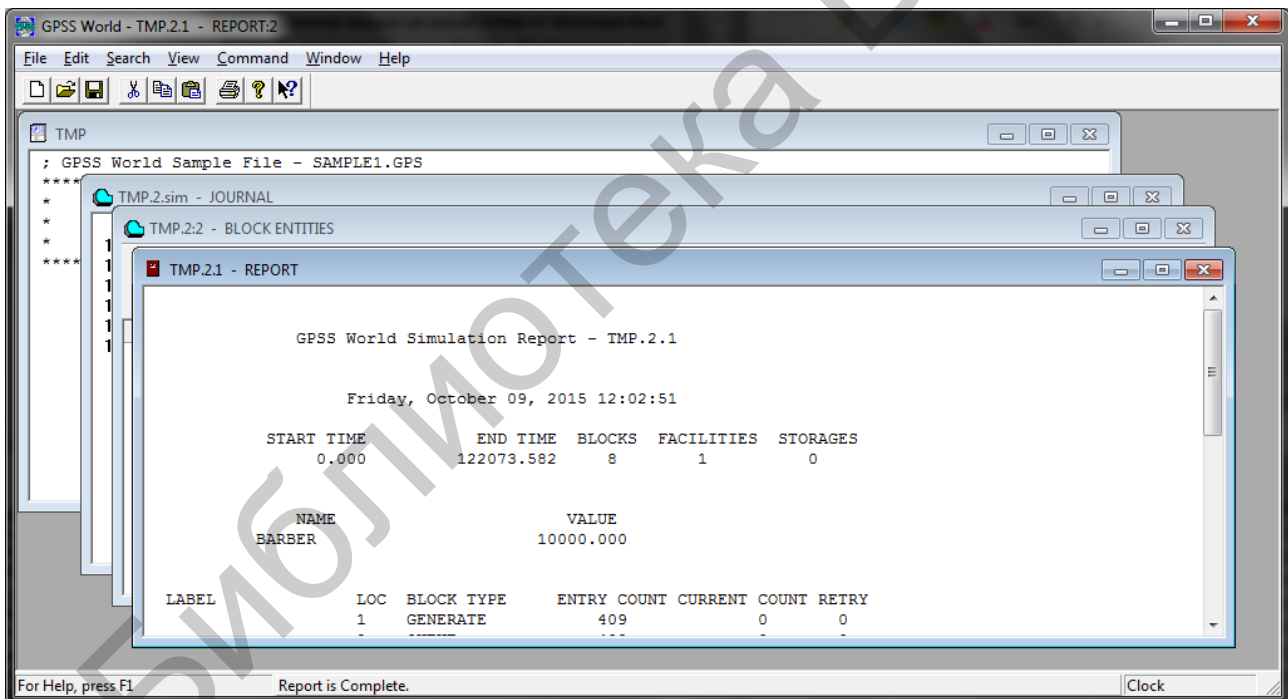


Рисунок 1.8 – Окно «Report» со стандартным отчетом

Полученный отчет можно распечатать, вызвав окно печати «File/Print».

Обратите внимание, что во время выполнения процесса моделирования необходимо оставлять открытыми только те окна, которые вы просматриваете.

Если вы закончили работу с окном и его функционал больше не нужен, рекомендуется его закрыть.

Если окно открыто или свернуто, ему передается непрерывная последовательность служебных сообщений, что, в свою очередь, увеличит время выполнения процесса моделирования.

Таким образом, программа GPSS World имеет в своем составе оконное управление интерфейсом, поддерживает стандартные команды приложений для Windows.

Объект программы GPSS World «Модель» определяется как последовательность операторов модели. Операторы модели – это операторы блоков, команды или PLUS-процедуры. При разработке модели стоит задача создать такую последовательность операторов модели, при которой поведение модели как можно более точно соответствовало бы поведению реальной системы.

Объект «Процесс моделирования» создается из объекта «Модель» посредством трансляции последовательности операторов модели. Трансляция выполняется с помощью команды меню «Command/Create «Simulation»». Уже после создания объекта «Процесс моделирования» возможно добавление дополнительных операторов модели.

Операторы, передаваемые существующему объекту «Процесс моделирования», называются интерактивными операторами. Примером интерактивного оператора может быть команда «START», которая передается после создания объекта «Процесс моделирования».

Все команды, за исключением «HALT», «SHOW» и «INCLUDE», после того как были приняты объектом «Процесс моделирования», помещаются в упорядоченную очередь команд. После этого они выполняются в том порядке, в котором были приняты. Эти команды называются несрочными командами.

«HALT» и «SHOW» называются срочными командами. Они исполняются независимо от того, какая операция производится в данный момент. Команда «HALT» представляет собой особый случай. Ее выполнение не только приостанавливает любой выполняющийся процесс моделирования, но также удаляет все несрочные команды, которые не начали выполняться.

GPSS обеспечивает широкое множество встроенных статистик, называемых стандартными числовыми атрибутами. Для их использования в операндах или выражениях достаточно сделать на них ссылку.

Задание 1.2: изучить порядок выполнения процесса имитационного моделирования.

Запустите программу, откройте модель «Sample1.GPS» и создайте объект «Процесс моделирования». Создайте окно графика «Plot» при помощи команды меню «Window/Simulation Window/Plot Window». Появится диалоговое окно. Введите информацию, показанную на рисунке 1.9, для вывода информации, отражающей количество ожидающих в очереди клиентов парикмахерской. После ввода информации нажмите кнопки «Plot» и «Memorize». Нажмите кнопку «ОК» и запустите моделирование командой «START 100». В процессе моделирования график будет динамически прорисовываться. Окончательный вид графика представлен на рисунке 1.10.

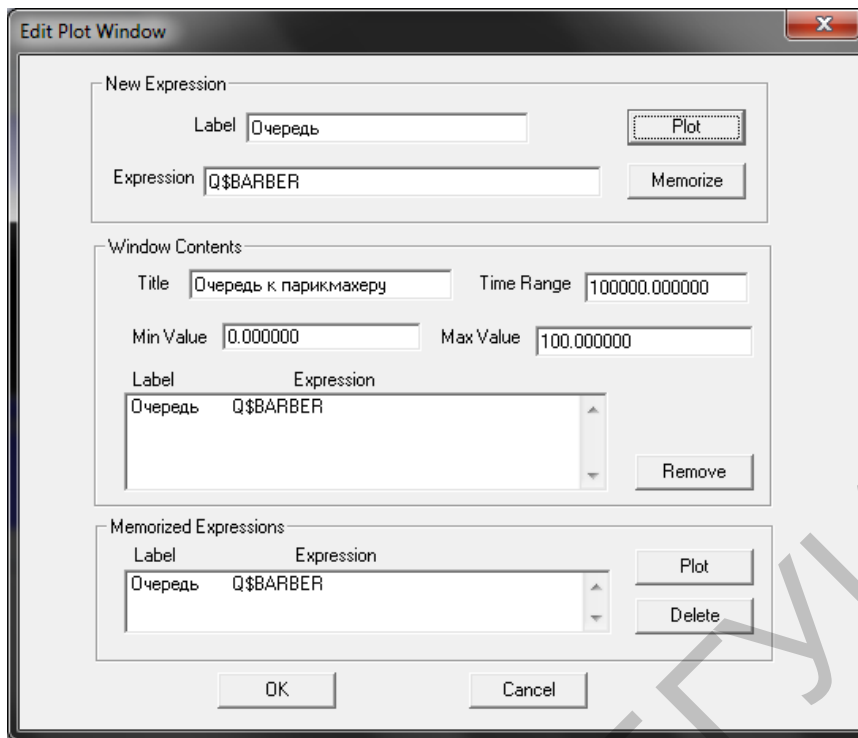


Рисунок 1.9 – Подготовка оперативного графика

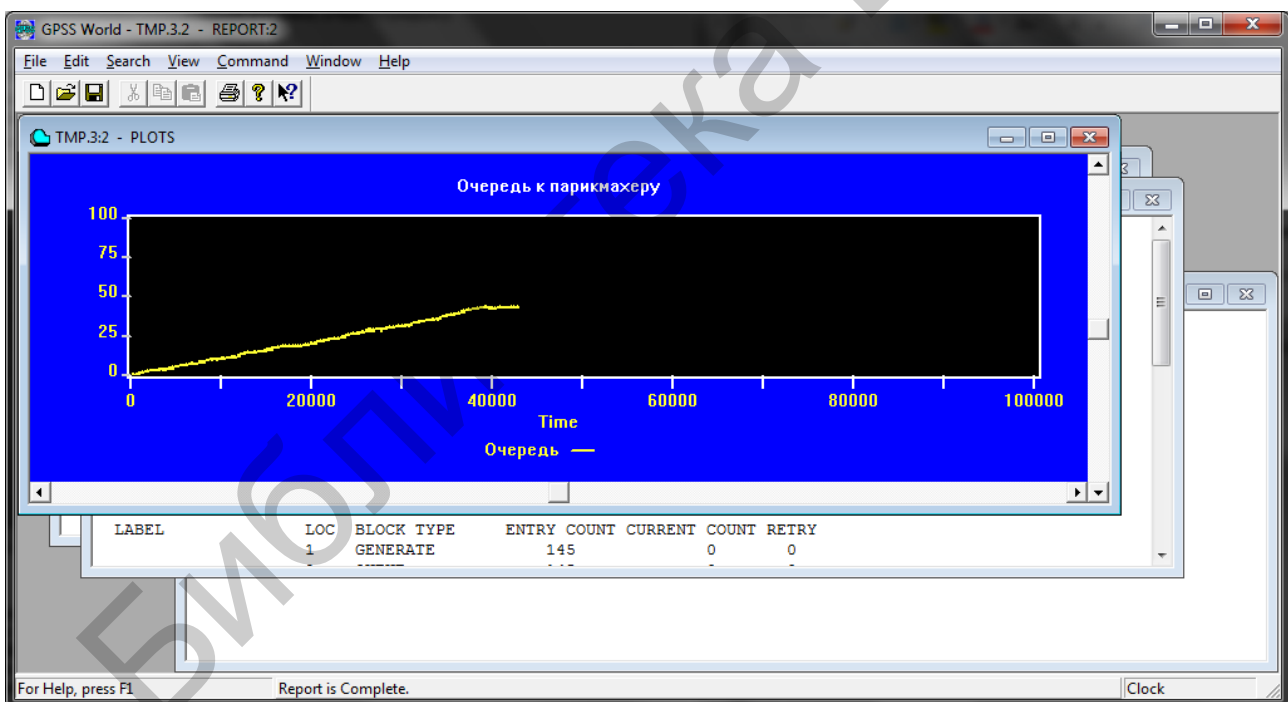


Рисунок 1.10 – Окно графика после завершения процесса моделирования

В процессе моделирования можно установить прерывание процесса моделирования клавишей [F4] (команда «HALT») и продолжить его выполнение клавишей [F2] (команда «CONTINUE»).

Полученный в результате моделирования график можно распечатать командой «Print».

Следует обратить внимание, что значение «Q\$BARBER» изменялось ступенчато в дискретные моменты времени.

После завершения процесса моделирования исследуйте полученные результаты при помощи команды «Show». Для этого выполните команду меню «Command/SHOW», в диалоговом окне введите «SHOW C1» и нажмите «ОК». Эта команда выводит в строке состояния относительное модельное время. Величина C1 равна модельному времени завершения процесса моделирования. Далее снова выполните команду «SHOW» с параметром «QM\$BARBER». Полученная величина показывает максимальное значение длины очереди с именем «BARBER».

GPSS World позволяет наблюдать за процессом моделирования с помощью различных способов. Каждый из основных типов объектов GPSS World имеет окно для наблюдения за его динамикой в процессе моделирования. Кроме того, имеются кадры процесса моделирования, которые также могут быть просмотрены в окнах или распечатаны. Также можно открыть окно «Expression», содержащее список выражений, для которых были созданы метки. Каждое из них представляет собой допустимое PLUS-выражение. Наиболее простые из них – это переменные или СЧА. Окна процесса моделирования обновляются динамически, а кадры процесса моделирования отображают состояние некоторого списка, группы или транзакта в определенный момент модельного времени.

Откройте окно «Expression» для модельного времени, длины очереди клиентов и номера активного транзакта. Для этого нужно выполнить команду меню «Window/Simulation Window/Expression Window». Отредактируйте окно «Expression» в поле «Label», наберите «Время», а в поле «Expression» – «AC1». Затем следует нажать кнопки «View» и «Memorize». В данном случае все происходит аналогично, как и в окне «Plot». При этом ранее созданные в этом же сеансе выражения сохраняются и выводятся в поле «Memorized Expression», откуда их можно выбрать кнопкой «View» (рисунок 1.11).

Создайте дополнительные выражения. Для этого в поле «Label» наберите «Активный транзакт», а в поле «Expression» – «XN1». Подтвердите выбор, нажав «View» и «Memorize». После этого введите имя «Очередь» и выражение «Q\$Barber». Подтвердите выбор кнопкой «ОК».

Откройте окно «Facilities», чтобы отследить состояние устройства GPSS, представляющего парикмахера. Для этого выберите меню «Window/Simulation Window/Facilities Window». Разместите окна «Facilities» и «Expression» так, чтобы было можно наблюдать их содержимое одновременно (рисунок 1.12).

Запустите процесс моделирования с параметром отказа от вывода итогового отчета. Для этого выберите команду «START», в параметрах укажите «START 10000,NP» и нажмите «ОК».

Пока идет процесс моделирования, можно наблюдать изменение важных статистик парикмахера в зависимости от условий, возникающих в процессе моделирования так же, как и значения специфических величин в окне «Expression».

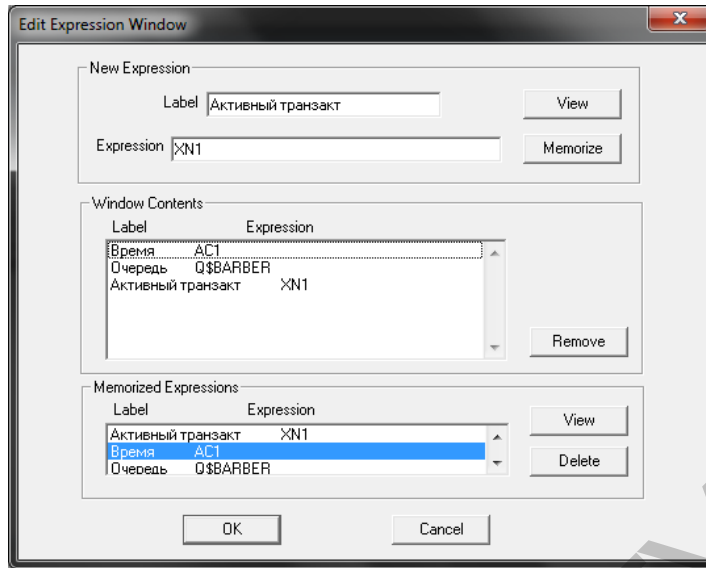


Рисунок 1.11 – Окно редактирования выражений «Edit Expression Window»

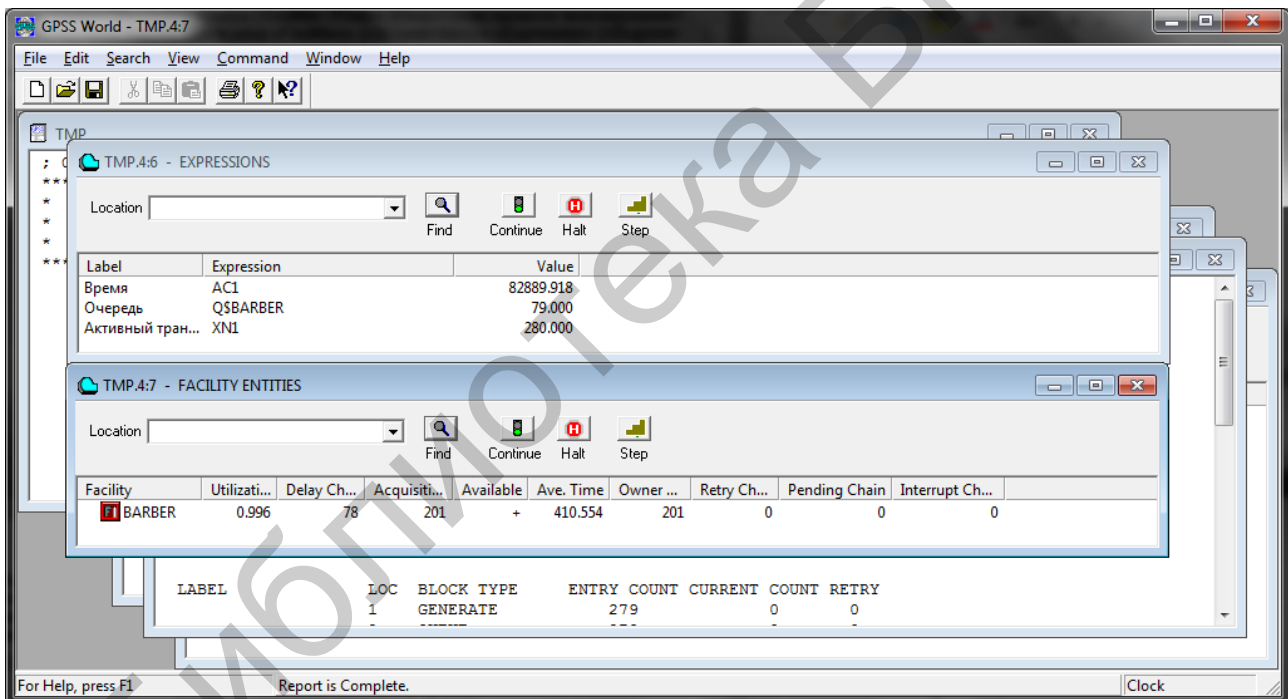


Рисунок 1.12 – Окна «Expressions» и «Facility Entities»

Задание 1.3: задать условие остановки на определенном блоке.

Для начала следует остановить процесс моделирования (клавиша [F4]) на отладочной панели инструментов окна «Facilities». Закройте окна «Facilities» и «Expression». Откройте окно «Blocks» («Window/Simulation Window/Blocks Window»). На блоке «DEPART» установите условие остановки, выделив его и нажав на кнопку «Place» на панели инструментов (рисунок 1.13).

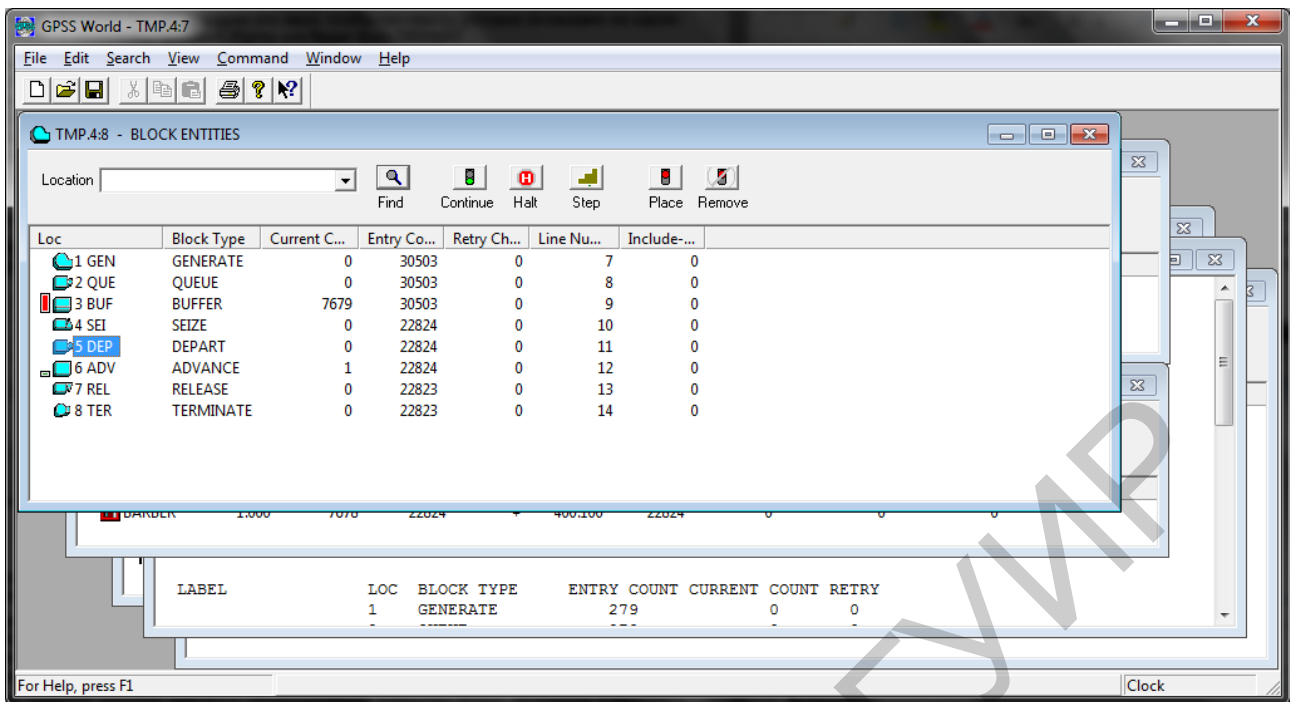


Рисунок 1.13 – Окно «Block Entities»

Запустите остановленный процесс моделирования командой «CONTINUE» (клавиша [F2]). Процесс моделирования остановится, когда какой-либо транзакт попытается войти в блок «DEPART».

Отправьте процессу моделирования оператор блока. Выберите меню «Command/Custom...» и в диалоговом окне введите «TRACE». В результате в активном транзакте будет установлен индикатор трассировки. Теперь после каждого входа этого транзакта в любой блок в окне «Journal» будет выдаваться трассировочное сообщение.

Проделанные операции представляют собой режим ручного моделирования, при котором транзакт принуждается ко входу в блок, который не является постоянной частью модели. Теперь требуется снять условие остановки, иначе каждый транзакт будет задерживаться перед блоком «DEPART». Перейдите в окно «Blocks». Выделите блок «DEPART» и нажмите кнопку «Remove», чтобы удалить из него условие остановки.

Продолжите процесс моделирования (клавиша [F2]). Трассировочные сообщения для данного транзакта будут появляться, пока он не войдет в блок «TERMINATE». Если требуется вывод трассировочных сообщений для всех транзактов в модели, необходимо поместить в модель блок «TRACE». Однако гораздо удобнее избирательно трассировать транзакты в частях модели, где они ведут себя не так, как предполагается.

Журнал системы может быть в любой момент распечатан и сохранен в формате «.txt».

1.3 Содержание отчета

1. Титульный лист.
2. Цель работы.
3. Краткие теоретические сведения.
4. Реализация решения задачи.
5. Выводы.
6. Список использованных источников.

1.4 Контрольные вопросы

1. Порядок создания, открытия и сохранения файлов модели в системе GPSS World.
2. Вызов окна настроек модели. Возможности настройки текущей модели.
3. Порядок введения команд «START», «SHOW», произвольной команды.
4. Порядок вызова графических окон в системе GPSS World.
5. Вывод отчетов по результатам процесса моделирования в системе GPSS World.
6. Настройка параметров вывода графического окна «Plot».
7. Настройка параметров вывода графического окна «Expressions».
8. Установка и удаление пользовательских условий прерывания на блоках модели.
9. «Горячие» клавиши стандартных функций по умолчанию.
10. Порядок вывода содержимого графических окон на печать.

Список использованных источников

1. Общецелевая система моделирования GPSS World [Электронный ресурс]. – 2015. – Режим доступа : <http://www.exponenta.ru/soft/others/gpss/gpss.asp>.
2. Шевченко, Д. Н. Имитационное моделирование на GPSS : учеб.-метод. пособие для студентов технических специальностей / Д. Н. Шевченко, И. Н. Кравченя. – Гомель : БелГУТ, 2007. – 97 с.
3. Сосновиков, Г. К. Компьютерное моделирование. Практикум по имитационному моделированию в среде GPSS World : учеб. пособие / Г. К. Сосновиков, Л. А. Воробейчиков ; под ред. Г. К. Сосновилова. – М. : Инфра-М, 2015. – 112 с.

Практическая работа №2

РЕШЕНИЕ ПРОБЛЕМ. ТЕКСТОВЫЙ РЕДАКТОР GPSS WORLD

Цель работы

Изучить возможности решения возникающих проблем при создании модели в GPSS World.

2.1 Теоретическая часть

Так или иначе в процессе моделирования можно столкнуться с различными проблемами, требующими корректно определить их причину и способы устранения.

Прежде всего, в программе существует встроенный механизм справки и поддержки, вызываемый клавишей [F1]. Также некоторые диалоговые окна имеют меню помощи «Help».

В систему GPSSW встроен текстовый редактор. Ввод или замена символов текста выполняется в месте расположения курсора, который периодически мигает, как, например, в текстовом редакторе Word. Для вставки или замены символов используется клавиша [Insert]. С помощью клавиш перемещения – со стрелками вверх, вниз, влево, вправо, расположенных в правой части клавиатуры, и мыши курсор можно перемещать в любое место текстового окна. Клавиша [Delete] удаляет символ, расположенный сразу за курсором. Клавиша [Backspace] удаляет символ, расположенный перед курсором. Клавиша [Home] перемещает курсор в начало строки, в которой находится курсор, а [End] – в конец текстовой строки. Для вставки пустой строки можно использовать два способа: нажать клавишу [Enter] или когда курсор находится в начале строки, на месте которой должна быть введена пустая строка, или в конце строки, после которой вводится пустая строка. Вставку пустой строки можно осуществить и с помощью пунктов меню.

Существуют специальные комбинации «горячих» клавиш для быстрого перемещения по текстовому объекту:

- [Ctrl]+[Alt]+[B] – следующая метка;
- [Ctrl]+[Alt]+[G] – следующая строка;
- [Ctrl]+[Alt]+[N] – следующая ошибка;
- [Ctrl]+[Alt]+[P] – предыдущая ошибка.

Для редактирования текста могут быть использованы следующие комбинации клавиш:

- [Ctrl]+[Alt]+[F] – найти и заменить;
- [Ctrl]+[I] – вставить строку;
- [Ctrl]+[D] – удалить строку;
- [Ctrl]+[Z] – отменить последнюю команду;
- [Ctrl]+[C] – скопировать в буфер обмена данными;
- [Ctrl]+[X] – вырезать и поместить в буфер обмена данными;

- [Ctrl]+[V] – вставить из буфера обмена данными.

Имеются также комбинации клавиш для выполнения следующих операций:

- [Ctrl]+[O] – открыть файл;
- [Ctrl]+[S] – сохранить файл;
- [Ctrl]+[P] – распечатать файл.

2.2 Практическая часть

Запустите программу GPSS World и выберите меню «Help». Откроется общее меню справки и поддержки для программы GPSS World. В случае когда нужно выбрать справку по контекстному меню, следует выделить нужное меню, нажать и удерживать левую клавишу мыши и одновременно нажать кнопку [F1]. Данная процедура вызовет ту часть справки, которая относится к выделенному пункту контекстного меню.

Далее изучим использование назначаемых функциональных клавиш. Для этого откройте модель «Sample10.GPS» («File/Open») из папки «Sample Models». Выделите в открытом окне слово «GENERATE». Нажмите клавишу [F1]. Обратите внимание: открылось окно справки с перечислением всех справочных статей, где упоминается слово «GENERATE». Аналогичным образом можно получить информацию о любом операторе, блоке или определении модели.

Задание 2.1: рассмотреть возможности настроек модели.

Откройте свойства модели («Edit/Settings»). В журнале настроек перейдите на вкладку «Function Keys». На данной вкладке можно ввести в поле напротив каждой горячей клавиши любую команду. Каждая модель может быть настроена на использование собственных функциональных клавиш. Закрепите за клавишей [F9] команду «SHOW Q\$Barber» (рисунок 2.1). Закройте журнал, нажав «OK».

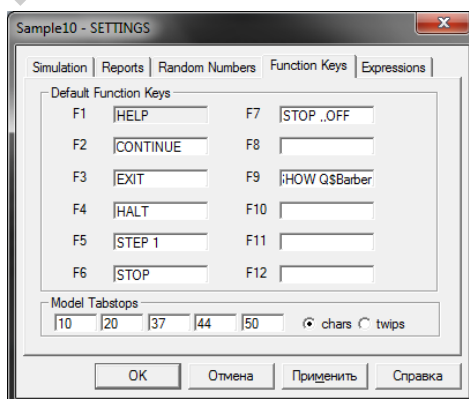


Рисунок 2.1 – Страница «Function Keys» журнала настроек модели

Так как рассматривается пример модели очереди к парикмахеру с небольшой производительностью, то очередь оказывается чрезмерно длинной.

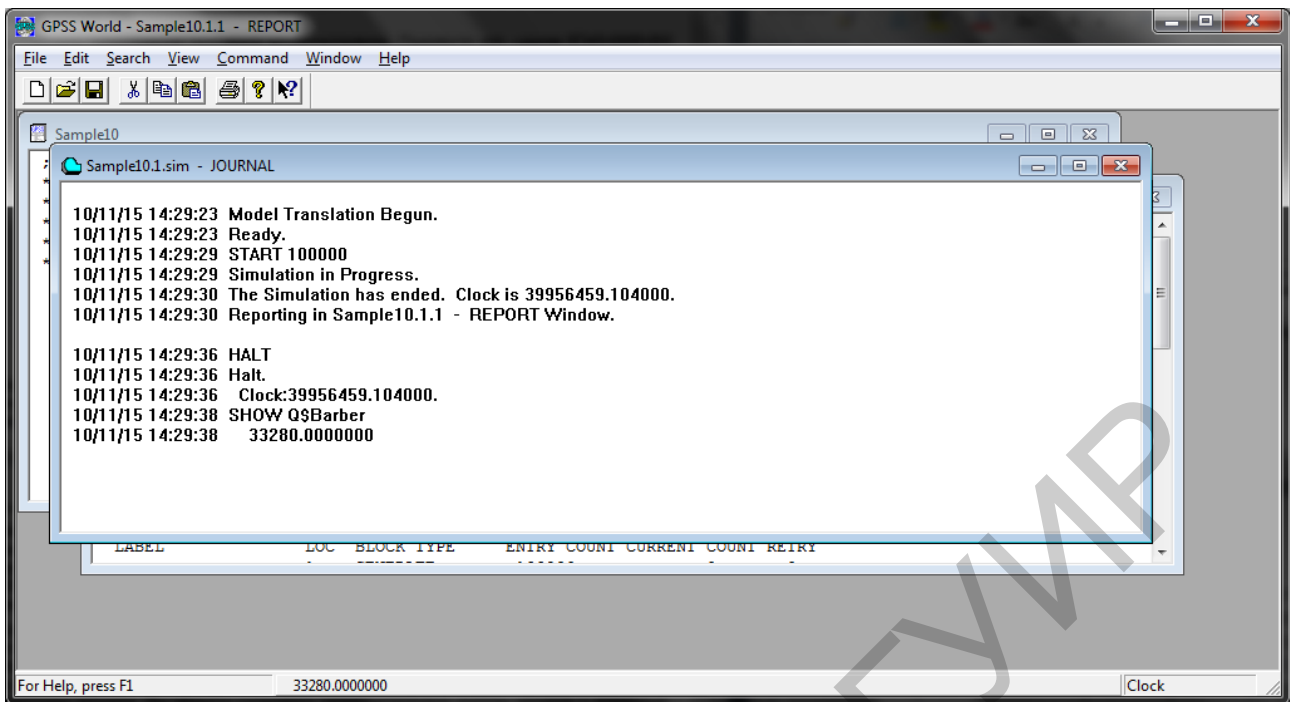


Рисунок 2.2 – Окно журнала с выведенной командой «SHOW»

Файл, содержащий последовательность операторов модели, называется файлом модели. Оператором модели является блок, команда или определение процедуры. Файлы модели, открытые в окне модели, называются главными файлами модели, любые другие файлы модели, включая командные файлы, называются дополнительными файлами модели. Создавать дополнительные файлы модели можно с помощью окна модели GPSS World или с помощью любого текстового редактора, сохраняя их как текстовые файлы (.txt). С помощью команды «INCLUDE» можно считать целый командный файл. Команду «INCLUDE» также можно закреплять за «горячей» клавишей.

Каждая из моделей может использовать уникальный набор функциональных клавиш. Следующие функциональные клавиши установлены заранее, однако их значения можно переопределить:

- [F1] – справка;
- [F2] – команда «CONTINUE»;
- [F3] – выход в Windows;
- [F4] – команда «HALT»;
- [F5] – команда «STEP 1»;
- [F6] – команда «STOP»;
- [F8] – команда «STOP ..OFF».

Задание 2.2: применить команду «SHOW» для вычисления значения выражения.

Выберите команду меню «Command/SHOW», затем в поле ввода после команды наберите выражение «1234+5678/345#(64+94)». Следует обратить внимание, что знак «#» используется для представления операции умножения,

так как знак «*» зарезервирован для косвенной адресации в СЧА. Однако в программе GPSS World зарезервированные знаки могут быть изменены в настройках модели. После ввода команды в журнале отобразится результат вычисления выражения. Вычисления производятся согласно общепринятой иерархии арифметических и логических операций.

Задание 2.3: вызвать событие «Останов по ошибке».

В данном случае в окне журнала выведется описание ошибки. Также ошибка будет показана в строке состояния главного окна. Выберите «Command/SHOW» и наберите в качестве параметра 2/0. Обратите внимание на подробное описание ошибки.

Задание 2.4: создать модель, состоящую из 2 строк и содержащую очевидную ошибку.

Закройте все окна текущей модели, не сохраняя изменения. Создайте новую модель («File/New»). При создании нового документа выберите «Model». Откроется окно модели, в которой требуется набрать в первой строке «Henerate 10» (синтаксическая ошибка в названии команды), а во второй строке – «Terminate 1». Далее запустите процесс моделирования. Откроется окно журнала с сообщением, указывающим на неверный элемент (рисунок 2.3).

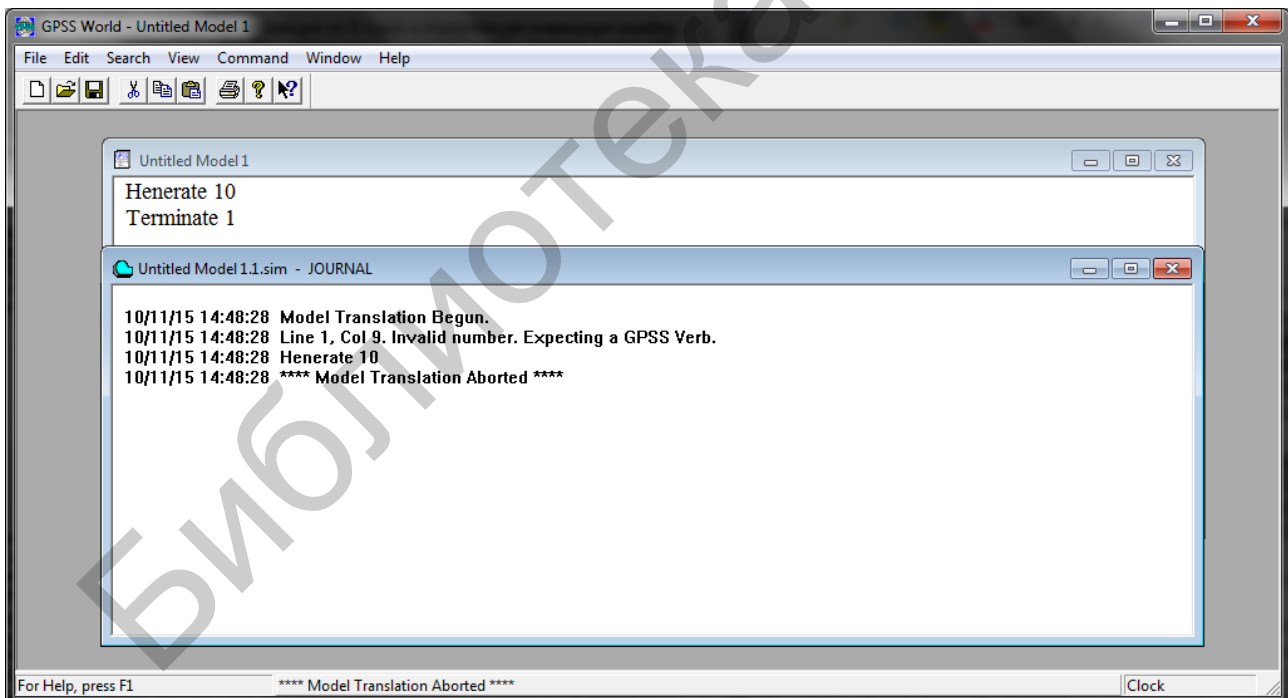


Рисунок 2.3 – Сообщение об ошибке в окне журнала

В меню поиска окна «Model» есть несколько команд, которые перемещают курсор непосредственно к ошибке. Этими командами являются «Search/Go to Line...», «Search/Next Error» и «Search/Previous Error». Данные команды удобны для поиска ошибок в большой модели. При применении этих команд

курсор будет установлен в редакторе модели в место, где произошла ошибка. Таким образом можно быстро локализовать место с ошибкой в тексте модели и быстро ее устранить.

Устранив ошибку, не закрывая окна журнала, выполните команду ретрансляции («Command/Retranslate»). Модель будет проверена еще раз и сообщения об ошибках появляться не будут.

Задание 2.5: запустить процесс моделирования и исследовать результаты работы команды «SHOW».

Закройте все окна текущей модели без сохранения изменений. Откройте файл «Sample1.GPS». Запустите процесс моделирования и откройте новое окно «Facility Entities», чтобы наблюдать за выполнением процесса моделирования. Запустите процесс моделирования с параметром «START 11 111».

Обратите внимание, что устройство в окне «Facility Entities» не появляется, пока не будет выполнено первое обращение к нему в процессе моделирования. Другие типы объектов, требующие специального объявления, создаются при их определении.

Откройте окно «Expression». Включите в выражение переменные состояния процесса моделирования: натуральный логарифм модельного времени плюс квадрат максимальной длины очереди в парикмахерской. Для этого в поле «Label» наберите слово «Сложное», а в поле «Expression» – « $\log(C1)+(QM\$BARBER)^2$ » и нажмите кнопку «View». В поле выражения можно также применять логические выражения, используя операторы «'AND'», «'OR'», «'NOT'». Они выдают в качестве результата операции единицу (истина) или ноль (ложь) и могут сочетаться с арифметическими выражениями. Одинарные кавычки являются частью оператора. Не следует допускать наличия пробелов между оператором и операндом. Добавьте новое выражение: в поле «Label» введите «Менее 6 ожидающих?», в поле «Expression» – « $1+99\#((Q\$BARBER)'GE'6)$ ». Нажмите кнопку «View» и «OK». Данное выражение выдает в качестве результата 1, если в очереди ожидает менее шести клиентов, или 100 – в любом другом случае.

Рассмотрим поток случайных чисел при помощи команды «SHOW». Запустите команду «SHOW RN1000». СЧА «RN» возвращает случайное целое число от 0 до 999. Используйте данную команду несколько раз. Обратите внимание, что результат выполнения будет каждый раз разный. Так как команда «SHOW» будет применяться многократно, для удобства ее можно закрепить за «горячей» клавишей.

Закройте все активные окна без сохранения изменений.

Интегрированный в GPSS World редактор является полнофункциональным текстовым редактором, обладающим большим количеством функций для обработки текста. Кроме того, имеются возможности поиска ошибок трансляции модели.

Задание 2.6: открыть модель и сделать в ней изменения при помощи текстового редактора.

Откройте модель «Sample2.GPS». Выделите строку, которая начинается с оператора «ADVANCE». Скопируйте блок ([Ctrl]+[C]) и вставьте его в новую строку после оператора «RELEASE» ([Ctrl]+[V]). Выровняйте блоки, добавив несколько пробелов, и задайте операнд для данного блока «A,200». Внесение изменений в модели произведено.

Обратите внимание, что текстовый редактор, как и многие другие, обладает следующими возможностями: найти (команда «Search»), найти и заменить («Find/Replace»), вставить строку («Insert Line») или удалить ее («Delete Line») через меню «Edit».

Закройте все окна модели.

Задание 2.7: установить закладку в тексте модели.

Откройте модель «BOOKMRK.GPS». Поместите курсор в строку 210. Нумерация строк была добавлена специально для примера, в обычной модели нумерация строк не требуется. Выберите «Search/Mark». Повторите операцию для строки 300. Далее поместите курсор на строку 260 и выберите команду «Search/Select to Bookmark» – выделится весь текст, предшествующий закладке от текущего положения курсора. При помощи команды «Next Bookmark» можно осуществлять перемещение по тексту между закладками. При сохранении файла все закладки сохраняются, поэтому при редактировании модели, если были установлены закладки в одном из сеансов, ставить их каждый раз не требуется.

Встроенный текстовый редактор в системе GPSS World позволяет редактировать текст модели с применением стандартных средств. С этой целью были изучены основы ввода и исправления операторов с клавиатуры, использования справочной системы, возможности закрепления команд за «горячими» клавишами.

2.3 Содержание отчета

1. Титульный лист.
2. Цель работы.
3. Краткие теоретические сведения.
4. Реализация решения задачи.
5. Выводы.
6. Список использованных источников.

2.4 Контрольные вопросы

1. Вызов глобальной и контекстной справки в системе GPSS World.
2. Настройка и задание функциональных клавиш для текущей модели.
3. Функциональные «горячие» клавиши, настроенные в системе по умолчанию.

4. Порядок создания новой модели.
5. Команды поиска ошибок в тексте модели.
6. Вызов ретрансляции модели после исправления ошибок.
7. Применение команды «SHOW» в системе GPSS World.
8. Применение выражений в процессе моделирования.
9. Команды и возможности встроенного текстового редактора в системе GPSS World.
10. Помещение закладок в текст модели.

Список использованной литературы

1. Кудрявцев, Е. М. GPSS World. Основы имитационного моделирования различных систем / Е. М. Кудрявцев. – М. : ДМК Пресс, 2004. – 320 с.
2. Шевченко, Д. Н. Имитационное моделирование на GPSS : учеб.-метод. пособие для студентов технических специальностей / Д. Н. Шевченко, И. Н. Кравченя. – Гомель : БелГУТ, 2007. – 97 с.
3. Учебное пособие по GPSS World / пер. с англ. В. В. Девяткова ; под ред. К. В. Кудашова. – Казань : Мастер Лайн, 2002. – 272 с.

Библиотека БГУИР

Практическая работа №3

ТРАНЗАКТЫ И БЛОКИ МОДЕЛИ

Цель работы

Изучить назначение блоков и порядок работы с ними.

3.1 Теоретическая часть

Транзакт GPSS – это объект с набором некоторых атрибутов. Каждый транзакт индивидуален, так как идентифицируется уникальным номером. Транзакты, созданные в процессе моделирования, последовательно нумеруются, начиная с первого. Имеется возможность изменять атрибуты, связанные с каждым транзактом. Существуют следующие атрибуты (в порядке приоритета): запланированное время, семейство и «параметры транзактов», которым присваиваются задаваемые пользователем имена или числа. Параметры несут о транзакте информацию, которая является уникальной и касается только этого транзакта. Чтобы посмотреть информацию, содержащуюся в этих параметрах, можно воспользоваться окнами кадров. Транзакты в процессе моделирования переходят от одного блока к другому, воздействуя на них и подвергаясь воздействию со стороны других объектов GSPP.

В отличие от универсальных языков программирования, каждый транзакт в GPSS содержит маркер положения. Если в компьютерной программе существует только одно место, которое необходимо помнить, то в процессе моделирования GPSS могут использоваться тысячи маркеров, связанных с тысячами транзактов, расположенных во множестве различных блоков. Транзакты необязательно запускаются с начала программы. Процесс моделирования GPSS может иметь множество операторов «GENERATE», каждый из которых создает множество транзактов с собственным маркером положения.

Транзакты могут создаваться группами (пакетами) или отеляться друг от друга временными интервалами. Для этого необходимо использовать блок «GENERATE». Все процессы моделирования должны иметь по крайней мере один блок «GENERATE». Рассмотрим случай, в котором новый транзакт создается через каждые 60 единиц времени. Пусть единица времени выражает одну секунду. Тогда соответствующий оператор «GENERATE», который будет создавать новый транзакт раз в минуту, имеет вид

GENERATE 60 ;Создание нового транзакта

Так как GPSS World представляет время с помощью чисел с плавающей точкой, мы можем также представить создание транзакта в минутных интервалах, то есть

GENERATE 1 ;Создание нового транзакта

Затем секундный интервал можно представить как $1/60$ часть минуты, или 0,1666667. Следует выбрать представление, которое является наиболее выразительным для исследуемой модели. Если выражение используется в этом операторе, оно должно быть заключено в скобки.

Блок «GENERATE» в общем случае имеет следующий формат:

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

Здесь 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, то есть самый низкий приоритет); F, ..., I – количество и формат параметров транзактов (по умолчанию 12 параметров формата «полуслово»).

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.

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 TG1), значение которого задается командой «START».

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

Здесь A – имя устройства, занимаемого (освобождаемого) транзактом.

3.2 Практическая часть

Задание 3.1: создать транзакт.

Создайте новую модель. Наберите строку:

GENERATE 60 ;Создание нового транзакта

Используйте в качестве пробелов знаки табуляции. Сохраните модель, назвав ее «MyModel». Вызовите создание процесса моделирования, затем отобразите окно «Блоки». Отобразится окно блоков, где можно увидеть созданный только что блок «GENERATE». Обратите внимание, что блоки всегда нумеруются, начиная с первого.

Блоки «TERMINATE» удаляют транзакты из процесса моделирования. Транзакт уничтожается, когда входит в блок «TERMINATE». Также в операторе «TERMINATE» имеется необязательный операнд, на величину которого уменьшается значение специального числа, называемого счетчиком завершения. Счетчик завершения выбирается параметром при вводе команды «START». Этот счетчик определяет, как долго должен выполняться процесс моделирования. Если в операнде блока «TERMINATE» будет записано число 1, то каждый раз при уничтожении транзакта этим блоком счетчик завершения будет уменьшаться на 1. Когда счетчик завершения становится равным нулю или отрицательным, процесс моделирования останавливается. После этого автоматически выдается стандартный отчет, если это не было запрещено операндом «NP» команды «START».

Задание 3.2: добавить в модель блок «TERMINATE».

В окне модели перейдите на новую строку и запишите:

TERMINATE 1 ;Уничтожение транзакта

Ретранслируйте модель («Command/Retranslate»). В окне блоков появится второй созданный блок.

Два перечисленных блока представляют собой простейшую модель. Они могут моделировать ситуацию, при которой клиент приходит в магазин точно один раз в минуту. Воспользуемся этой простейшей моделью, чтобы изучить, что происходит в процессе моделирования.

Каждый транзакт имеет атрибут, который определяет, когда он должен сделать попытку войти в следующий блок. Если текущее модельное время еще не достигло нужного значения, транзакт должен быть отложен. Это то, для чего используется список будущих событий. Список будущих событий (СБС) – это область промежуточного хранения транзактов, которыми запланировано произвести некоторое действие в будущем. Каждый раз, когда транзакт входит в блок «GENERATE», создается еще один транзакт. Новый транзакт входит в блок «GENERATE» не сразу. Сначала он помещается в список будущих событий,

согласно интервалу времени прибытия, который был задан в блоке «GENERATE».

Список текущих событий (СТС) – это множество транзактов, имеющих блоки, в которые они должны войти в текущий момент начального времени. Транзакты выбираются по одному из начала списка текущих событий, чтобы войти в как можно большее количество блоков. Когда список текущих событий пуст, GPSS World увеличивает значение модельного времени.

Задание 3.3: изучить СТС и СБС простейшей модели.

Определите условие остановки, которое заставит процесс моделирования останавливаться при каждой предпринятой попытке входа в блок. Выберите окно блоков. Выделите блок «GENERATE». Нажмите кнопку «Place» на панели инструментов. Это действие поместит в блок «GENERATE» условие остановки. Процесс моделирования остановится при попытке транзакта войти в блок «GENERATE». Для просмотра списка текущих и будущих событий выберите «Window/Simulation «Snapshot»/FEC «Snapshot» и «Window/Simulation «Snapshot»/CEC «Snapshot». Список СТС пуст, в списке СБС присутствует один транзакт, так как блок «GENERATE» был заполнен, то есть он создавал первый транзакт, когда была выполнена команда «STOP». Как только будет выполнена команда «START», транзакт 1 будет перемещен в СТС и время будет установлено в 60. Закройте окна списков событий.

Кадры процесса моделирования являются статическими отображениями текущего состояния и не изменяются во время выполнения процесса моделирования. Они могут быть обновлены только повторным вызовом при остановке процесса моделирования.

Выполните команду «STEP 1» (клавиша [F5]) и снова откройте окна СБС и СТС (рисунок 3.1).

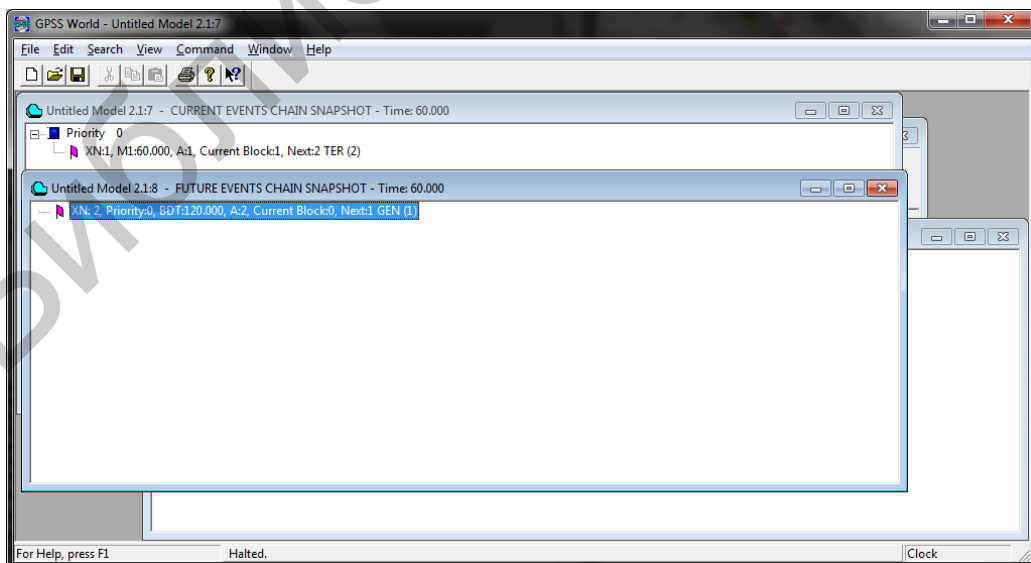


Рисунок 3.1 – Окна СТС и СБС процесса моделирования

Теперь в окне СБС можно видеть всю информацию о транзактах, ожидающих разрешения на вход в процесс моделирования. Время в СБС – это «время выхода из блока», то есть запланированное значение модельного времени, при котором транзакты из СБС должны возвратиться в процесс моделирования. Следующая колонка блока показывает номер блока, в который будет входить транзакт в следующий раз. Блок с номером 1 в нашем случае это блок «GENERATE», а блок с номером 2 – блок «TERMINATE». Транзакт 2 ожидает входа в блок «GENERATE», находится в СБС и выполнит вход, когда модельное время достигнет 120 с. Закройте окна СБС и СТС.

Задача 3.4: посмотреть текущее значение модельного времени и счетчика завершения.

Для этого выберите команду «SHOW AC1». «AC1» – это СЧА, который возвращает значение абсолютного модельного времени. Для просмотра счетчика завершения используется СЧА «TG1». Выполните команду «SHOW TG1».

Следует обратить внимание, что когда транзакты, намеченные для повторного входа в процесс моделирования в одинаковое время, перемещаются из СБС в СТС, их порядок следования перемешивается случайным образом. Это предотвращает появление непреднамеренных синхронных последовательностей.

Следует запомнить, что модельное время обновляется каждый раз, когда транзакт должен быть выбран из списка будущих событий. Таким образом, время в процессе компьютерного моделирования не протекает равномерно. Процесс моделирования обычно состоит из многократных входов в блоки в заданный момент времени, после которых следует мгновенный скачок к следующему значению времени.

При интерпретации процесса моделирования все входы в блоки GSPP, происходящие в данный момент времени, совершаются одновременно. Быстрые переходы к новым моментам времени позволяют процессам моделирования выполняться намного быстрее, чем при моделировании процесса в режиме реального времени.

Задание 3.5: изучить блок «ADVANCE».

После уничтожения транзакт 1 больше не используется в процессе моделирования. Все его входы в блоки произошли в отдельный момент времени, а именно, когда модельное время было равно 60 с. В случае моделирования прихода покупателей будет пониматься последовательность прихода покупателей, но не длительность времени, проведенного ими в магазине. Нужно будет отправить каждый транзакт обратно в список будущих событий, чтобы смоделировать некую продолжительность времени, пока транзакт все еще находится в процессе моделирования. Затем, когда модельное время снова будет увеличено, транзакт будет выбран из списка будущих событий. Для этого понадобится блок «ADVANCE». Первый операнд блока «ADVANCE» – это приращение времени, после которого транзакт должен выйти из списка будущих событий.

Установим его для транзакта 2 процесса моделирования. Для начала следует добавить блок «ADVANCE» после блока «GENERATE». Добавьте новую строку:

ADVANCE 61 ;Покупатели ходят по магазину 61 секунду

Выполните ретрансляцию модели. Нажимая клавишу [F5], добейтесь того, чтобы транзакт 1 вошел в блок «ADVANCE». Откройте СБС и СТС и проследите, как изменяются события, вызывая команду пошагового выполнения [F5] и обновляя СБС и СТС. Видно, что каждый транзакт уничтожается через 61 с после генерации. Это та величина, которую мы задали при помощи блока «ADVANCE».

Задание 3.6: смоделировать ситуацию, когда каждый покупатель обслуживается продавцом отдельно.

Допустим, что каждый покупатель должен разговаривать с продавцом 61 с. Для этой модели нельзя позволить транзакту сразу же войти в блок «ADVANCE», так как при этом начнется отсчет 61-секундного интервала, даже если продавец обслуживал другого покупателя. Если продавец уже общался с покупателем, следует помешать входу последующих транзактов в блок «ADVANCE». Для этого можно использовать объект GPSS, называемый устройством, и сопутствующие ему блоки «SEIZE» и «RELEASE».

У устройства имеется несколько атрибутов, самый важный из которых – право на его использование. Когда транзакт занимает устройство, оно считается занятым. Иначе устройство свободно. Транзакт может занимать только одно устройство. Это достигается при помощи блока «SEIZE», в первом операнде которого задается устройство. Попытки транзактов занять устройство путем входа в блок «SEIZE» не допускаются, пока устройство не станет свободным. Это препятствует продвижению транзактов, которые ждут права занять устройство. Транзакт, претендующий на вход в блок «SEIZE», останавливает движение в процессе моделирования, пока не наступит его очередь занять устройство. Эти транзакты перемещаются из СТС в список задержки устройства до тех пор, пока они не займут устройство.

Занимающий транзакт продолжает двигаться в процессе моделирования и занимает устройство до тех пор, пока не войдет в блок «RELEASE», в первом операнде которого задано устройство. Таким образом, блоки «SEIZE» и «RELEASE» нужны для задержки входа «транзактов-покупателей» в блок «ADVANCE» до того, как они займут «продавца». Только затем транзакт удаляется из СБС и входит в блок «RELEASE». Следующий моделируемый «покупатель» подходит к «продавцу». Выполним моделирование этой ситуации.

Откройте модель «Seizemod.GPS». Запустите процесс моделирования. Задайте условие остановки в блоке «GENERATE» в окне «Blocks» (кнопка «Place»). Запустите моделирование командой «START 1000».

Продвиньте транзакт на один шаг в блок «GENERATE» (клавиша [F5]). В окне журнала событий появится сообщение, что транзакт 1 войдет в блок «SEIZE». Это действие произойдет, так как продавец не занят. Откройте СБС и СТС. Транзакт 1 в СТС готов войти в блок «SEIZE». Транзакт 2 в СБС запланирован для входа в процесс в момент времени 40.

Перейдите в окно блоков и, нажав клавишу [F5], позвольте транзакту 1 занять устройство. После этого транзакт 1 останется активным транзактом, то есть первым в СТС. Сейчас он занимает устройство, представляющее продавца. Откройте окно устройств «Facilities». Обратите внимание на столбец «Owner XN» («Занимающий транзакт»): занимающий его транзакт – это транзакт 1, который готов начать 41-секундный интервал, во время которого покупатель будет разговаривать с продавцом с помощью входа в блок «ADVANCE». Выполните команду «SHOW AC1» (клавиша [F8]). Блок «ADVANCE» поместил транзакт 1 в список будущих событий. Покупатель, которого представляет транзакт, перестанет обслуживаться продавцом в момент времени 61. Обратите внимание, что текущее время равно 40 (отображено в строке состояния и заголовке СБС).

Перейдите в окно блоков и выполните еще 1 шаг (клавишей [F5]). Транзакт 2 попытается войти в блок «SEIZE» в момент времени 40, но он не сможет этого сделать, поскольку продавца занимает транзакт 1. Транзакт 1 закончит разговаривать с продавцом в момент времени 61 и будет готов войти в блок «RELEASE», а транзакт 2 войдет в блок «SEIZE».

Таким образом, осуществляется задержка транзакта в блоке «SEIZE», не позволяя ему войти в занятый блок «ADVANCE».

В практической работе были рассмотрены блоки создания («GENERATE»), уничтожения («TERMINATE»), задержки («ADVANCE») и устройства («SEIZE» и «RELEASE»).

3.3 Содержание отчета

1. Титульный лист.
2. Цель работы.
3. Краткие теоретические сведения.
4. Реализация решения задачи.
5. Выводы.
6. Список использованных источников.

3.4 Контрольные вопросы

1. Определение транзакта модели в GSPP World.
2. Определение блока модели. Создание и удаление блока в структуре модели.
3. Функциональное назначение и применение блока «GENERATE».
4. Функциональное назначение и применение блока «TERMINATE».

5. Назначение и просмотр модельного времени и счетчика завершения.
6. Функциональное назначение и применение блока «ADVANCE».
7. Назначение и использование списка текущих событий в GPSS World.
8. Назначение и использование списка будущих событий в GPSS World.
9. Функциональное назначение и применение блока «SEIZE».
10. Функциональное назначение и применение блока «RELEASE».

Список использованной литературы

1. Жернов, Ю. В. Создание моделей систем обслуживания в среде GPSS World / Ю. В. Жернов. – Саарбрюккен : Palmarium Academic Publishing, 2014. – 208 с.
2. Учебное пособие по GPSS World / пер. с англ. В. В. Девяткова ; под ред. К. В. Кудашова. – Казань : Мастер Лайн, 2002. – 272 с.
3. Шевченко, Д. Н. Имитационное моделирование на GPSS : учеб.-метод. пособие для студентов технических специальностей / Д. Н. Шевченко, И. Н. Кравченя. – Гомель : БелГУТ, 2007. – 97 с.

Библиотека БГУИР

Практическая работа №4

ОЧЕРЕДИ И Q-ТАБЛИЦЫ: БЛОКИ «QUEUE» И «DEPART»

Цель работы

Изучить назначение и порядок использования блоков создания очередей.

4.1 Теоретическая часть

Система моделирования GPSS обеспечивает возможность сбора статистики с помощью такого средства, как регистратор очереди.

При использовании регистратора очереди в тех точках модели, где число ресурсов ограничено, интерпретатор автоматически начинает собирать различную информацию об ожидании с помощью СЧА, а именно:

- число входов транзактов в очередь;
- количество транзактов, которые фактически присоединились к очереди и сразу ее покинули, то есть имели время ожидания, равное нулю;
- максимальная длина очереди;
- среднее число ожидавших транзактов;
- среднее время ожидания тех транзактов, которым пришлось ждать.

При входе транзакта в блок «QUEUE» выполняются четыре действия:

- 1) счетчик входов для данной очереди увеличивается на В;
- 2) длина очереди (счетчик текущего содержимого) для данной очереди увеличивается на В;
- 3) значение текущей длины очереди хранится в стандартном числовом атрибуте Q\$<имя очереди>;
- 4) транзакт присоединяется к очереди с запоминая ее имени и значения текущего модельного времени.

Транзакт перестает быть элементом очереди только после того, как он переходит в блок «DEPART» соответствующей очереди. Когда это происходит, интерпретатор выполняет следующие операции:

- 1) уменьшает длину соответствующей очереди на В;
- 2) используя привязку к значению времени, определяет, является ли время, проведенное транзактом в очереди, нулевым; если да, то такой транзакт по определению является транзактом с нулевым пребыванием в очереди и тогда изменяет счетчик нулевых вхождений;
- 3) ликвидирует «привязку» транзакта к очереди.

Если в модели используются объекты типа «очередь», то в файле стандартной статистики будет представлена информация об этих объектах. В конце моделирования интерпретатор автоматически выдает статистические данные: значение счетчика входов, максимальное значение длины очереди, среднее значение длины очереди, текущее значение длины очереди в конце периода моделирования, среднее значение времени нахождения в очереди и т. д.

Блоки «QUEUE» и «DEPART» в отличие от уже изученных блоков служат только для сбора статистики, хранящейся в объекте GPSS, называемом очередью. Очередь так же, как и устройство, создается автоматически при необходимости.

Важным атрибутом очереди является число, называемое «содержимое очереди» (длина очереди). Когда транзакт входит в блок «QUEUE», содержимое очереди уменьшается. GPSS World будет автоматически сохранять статистику очереди. Эта статистика выдается в стандартном отчете и доступна в процессе моделирования в качестве нескольких системных числовых атрибутов. Кроме того, можно определить Q-таблицу, которая отображается в окне «Table» в виде диаграммы. Также можно открыть окно «Queues» для динамического просмотра текущего состояния очередей процесса моделирования.

Блоки очередей могут быть установлены в любой точке программы, в которой предполагается возникновение очереди, и имеют следующий формат:

QUEUE A,[B]

Здесь А – имя очереди, в которую заносится транзакт; В – 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 А и В определяются аналогично блоку «QUEUE».

4.2 Практическая часть

Откройте модель «Sampque.GPS». Обратите внимание, что блоки «QUEUE» и «DEPART» определяют продолжительность времени при сборе автоматической статистики очереди, связанной с очередью Barber. В этом случае блоки «QUEUE» и «DEPART» заключают между собой блок «SEIZE». Эта пара блоков автоматически регистрирует время пребывания в очереди для каждого транзакта, входящего в блок «DEPART». Для транзактов, которые могут немедленно занять устройства, будет зарегистрировано нулевое время ожидания.

Блоки «QUEUE» и «DEPART» также помещены для учета общего времени, затраченного на стрижку, в том числе и на ожидание. Таким образом, получим статистику для времени ожидания и для общего времени стрижки, включая ожидание.

Задание 4.1: просмотреть статистику очереди во время выполнения процесса моделирования.

Запустите процесс моделирования. Откройте окна модели и очереди. Очереди начнут отображаться сразу же после запуска процесса моделирования,

когда в них войдет первый транзакт. Запустите моделирование командой «START 1111». Окно очереди примет вид, показанный на рисунке 4.1.

Остановите процесс моделирования (клавиша [F4]).

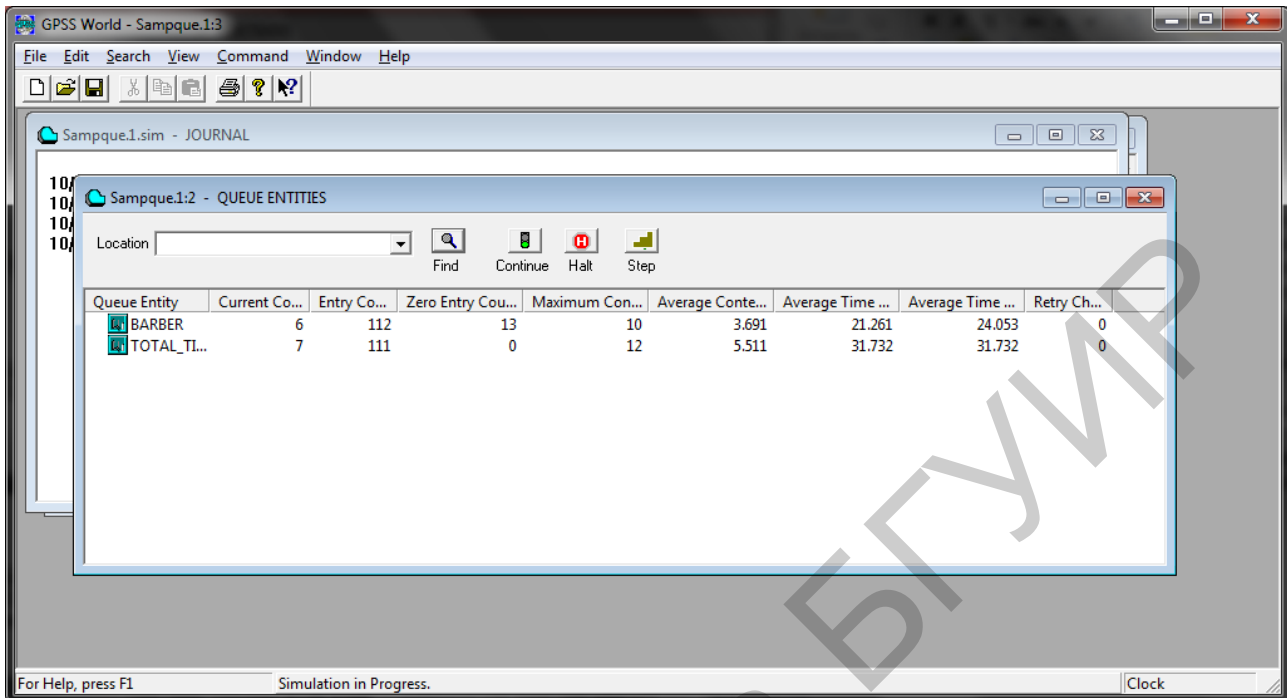


Рисунок 4.1 – Окно состояния очередей

Задание 4.2: создать диаграмму времени пребывания в очереди.

Диаграммы в GPSS World создаются с помощью определения таблиц и Q-таблиц. Добавим определение Q-таблицы к модели. Выберите окно модели и наберите следующую строку в последней строке модели:

Waittime QTABLE Barber,2,3,20 ;Таблица времени ожидания

Эта команда определит диаграмму (Q-таблицу). В отличие от устройства или очереди Q-таблицу необходимо объявить с помощью команды «QTABLE». Ретранслируйте модель, запустите моделирование с параметром «START 1111» и остановите через некоторое время клавишей [F4]. Откройте окно таблицы, выполнив команду меню «Window/Simulation Window/Table Window». В диалоговом окне потребуется выбрать таблицу, однако так как таблица в данной модели единственная, то автоматически выберется «Waittime». Откроется окно диаграммы. При выполнении процесса моделирования, продолжить который можно клавишей [F2], окно диаграммы будет динамически изменяться. На вертикальной оси откладываются транзакты, на горизонтальной – интервалы времени ожидания транзактов в очереди (рисунок 4.2).

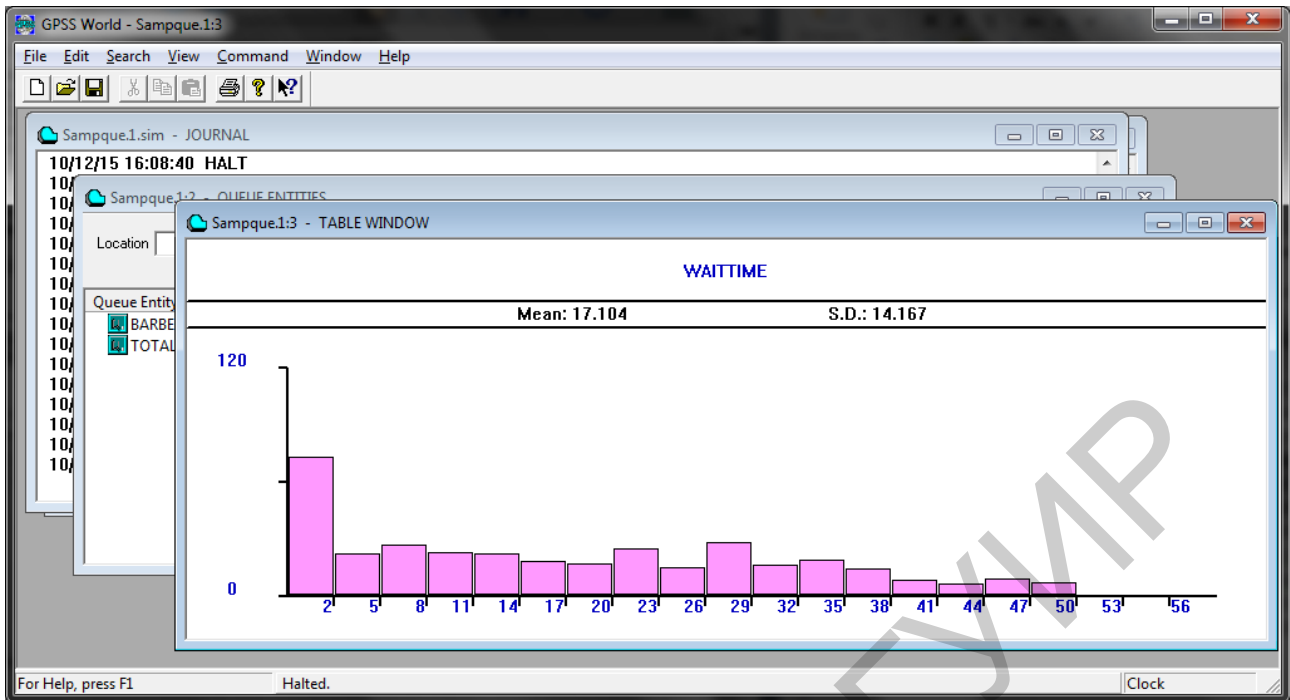


Рисунок 4.2 – Окно диаграммы

Из диаграммы видно, что среднее время ожидания составляет примерно 15 мин, при этом некоторые посетители ждут 56 мин и больше.

Откройте окно блоков. Обратите внимание на содержимое первого блока «TRANSFER». Число транзактов в процессе моделирования меняется в зависимости от того, как посетители (транзакты) ждут парикмахеров. Они ожидают в блоке «TRANSFER», пока не смогут пойти в один из блоков «SEIZE», позволяющих занять одного из парикмахеров.

Откройте окно устройств. Обратите внимание, что коэффициенты загрузки обоих парикмахеров (столбец «Utilization») довольно высоки.

Все транзакты, ожидающие входа в блок «SEIZE», находятся в списках повторных попыток устройств, названных Barber1 и Barber2. Каждый раз, когда состояние парикмахера изменяется от занятого к свободному, GPSS World проверяет все списки, связанные с устройствами, чтобы определить текущего владельца. В этом случае транзакты никогда не достигают списка задержки для каждого парикмахера, а помещаются в списки повторных попыток, связанные с блоком «TRANSFER». Когда оба парикмахера заняты, транзакты помещаются в списки повторных попыток для каждого парикмахера с тем, чтобы снова попытаться занять соответствующее устройство, когда одно из них станет свободным.

Также можно заметить в окне устройств, что продолжительность работы первого парикмахера в среднем составляет 10 мин, но при этом колеблется от 7,5 до 12,5, в то время как у второго – 13 мин в среднем и колеблется от 9 до 17. Данный эффект проявляется, так как в модели у блока «ADVANCE» задан второй операнд, представляющий собой полуинтервал равномерного распределения параметра, заданного в первом операнде.

Задание 4.3: проверить, является ли распределение времени стрижки, производимое блоком «ADVANCE», равномерным.

Поместите блоки «QUEUE» и «DEPART» для сбора информации о времени стрижки. Примените для этого окно создания блока. Установите указатель на первом блоке «DEPART», выберите «Edit/Insert GPSS Block». Появится меню блоков (рисунок 4.3).



Рисунок 4.3 – Окно создания блоков

Выберите оператор «QUEUE». Появится диалоговое окно создания блока. В первом операнде укажите «Haircut», в строке комментария наберите «Длительность стрижки – Парикмахер 1» и нажмите «ОК».

Созданный блок «QUEUE» был помещен в строке, которая была указана как точка ввода. Повторите операции с блоком «DEPART»: для этого установите курсор в любом месте следующего за только что созданным блоком «QUEUE» блока «ADVANCE». Выберите в окне создания блоков блок «DEPART», в качестве первого операнда наберите «Haircut», в строке комментария – «Длительность стрижки – Парикмахер 1» и нажмите «ОК».

Далее требуется определить диаграмму. Добавьте строку после последней строки модели и введите

Cuttimes QTABLE Haircut,8,0.5,10

Ретранслируйте модель и запустите моделирование командой «START 11111». Откройте окно таблицы, в диалоговом окне подтвердите выбор таблицы «Cuttimes» и поставьте моделирование на паузу (клавиша [F4]). Диаграмма

принимает неравномерный вид, однако через длительное время она приобретает прямоугольную форму.

Для создания диаграмм аналогичным образом могут использоваться операторы «TABLE» и «TABULATE». Они позволяют использовать в диаграммах любые выражения, а не только продолжительность модельного времени.

Добавление блоков «QUEUE» и «DEPART» не повлияет на процесс моделирования, так как модельное время не затрачивается и нет воздействия на направление движения транзактов.

4.3 Содержание отчета

1. Титульный лист.
2. Цель работы.
3. Краткие теоретические сведения.
4. Реализация решения задачи.
5. Выводы.
6. Список использованных источников.

4.4 Контрольные вопросы

1. Функциональное назначение и применение блока «QUEUE».
2. Функциональное назначение и применение блока «DEPART».
3. Порядок определения Q-таблицы в процессе моделирования.
4. Вывод Q-таблицы в графическое окно.
5. Анализ информации о процессе моделирования при помощи Q-таблиц.
6. Вызов окна создания блоков и его применение в процессе моделирования.

Список использованной литературы

1. Учебное пособие по GPSS World / пер. с англ. В. В. Девяткова ; под ред. К. В. Кудашова. – Казань : Мастер Лайн, 2002. – 272 с.
2. Шевченко, Д. Н. Имитационное моделирование на GPSS : учеб.-метод. пособие для студентов технических специальностей / Д. Н. Шевченко, И. Н. Кравчяня. – Гомель : БелГУТ, 2007. – 97 с.
3. Томашевский, В. Н. Имитационное моделирование в среде GPSS / В. Н. Томашевский, Е. Г. Жданова ; под ред. Е. Г. Жданова. – М. : Бестселлер, 2003. – 416 с.

Практическая работа №5

КОМАНДЫ GPSS WORLD. РЕЖИМ РУЧНОГО МОДЕЛИРОВАНИЯ

Цель работы

Изучить команды GPSS World и произвести режим ручного моделирования.

5.1 Теоретическая часть

Команды GPSS World применяются для управления созданием и выполнением процесса моделирования, а также для определения некоторых объектов GPSS (таблица 5.1).

Таблица 5.1 – Команды GPSS World

Команда	Функциональное назначение
BVARIABLE	Определяет булеву переменную
CLEAR	Сбрасывает статистику и удаляет из модели все транзакты
CONTINUE	Продолжает процесс моделирования
EQU	Присваивает значение переменной пользователю
EXIT	Завершает работу с GPSS World
FUNCTION	Определяет функцию
FVARIABLE	Определяет переменную с плавающей точкой
HALT	Останавливает процесс моделирования и удаляет все команды из очереди
INCLUDE	Считывает и транслирует дополнительный файл модели
INITIAL	Инициализирует или изменяет значение логического ключа, ячейки или матрицы
INTEGRATE	Автоматически интегрирует переменную пользователя по времени
MATRIX	Определяет матрицу
QTABLE	Определяет таблицу статистики очереди (Q-таблицу)
REPORT	Устанавливает имя файла отчета или запрашивает немедленный отчет
RESET	Сбрасывает статистику процесса моделирования
RMULT	Устанавливает параметры первых семи генераторов случайных чисел
SHOW	Вычисляет и отображает выражение
START	Устанавливает счетчик завершения и запускает процесс моделирования
STEP	Устанавливает ограниченное количество входов транзактов в блоки
STOP	Устанавливает условие остановки, основанное на количестве попыток входа в блок
STORAGE	Определяет память
TABLE	Определяет таблицу
VARIABLE	Определяет переменную

При применении переменных следует обратить внимание на следующую особенность: переменные могут состоять из констант или переменных, состояние которых изменяется в процессе моделирования. Например, переменная «Var1» определяется как

Var1 Variable 3#3/3

Если требуется обработать значение именованной переменной, нужно поставить «V\$» перед именем переменной.

Когда используется имя, система GPSS World назначает этому имени уникальный системный номер. Если имя используется для обозначения сущности, не требуется больше ничего делать. Тем не менее, если нужно, чтобы имени соответствовало определенное значение, следует использовать выражение EQU, для того чтобы присвоить имени значение до создания сущности. В последнем случае система GPSS World присваивает свое собственное значение имени, которое можно увидеть в статусной строке. Если использовать переменную «Var» в выражении «EQU» до того, как определили сущность «Variable», то команда «SHOW» вернет заданное число.

Если изменить имя после определения сущности, то в дальнейшем не получится использовать это имя для того, чтобы обращаться к старой сущности. Изменение имени не приводит к переопределению сущности. Любая старая сущность будет иметь номер, отличный от нового значения имени. Следует помнить, что блоки нумеруются системой GPSS World. Не следует использовать названия блоков для обозначения других сущностей. Если сделать это, то при запуске модели появится сообщение об ошибке.

Любое выражение можно интерактивно отправить в существующий объект «Simulation». В эту категорию попадают также блоки GPSS. Единственным исключением является блок «GENERATE». Блок, который был отправлен в объект «Simulation», выполняется, но не помещается в текст модели. Возможно ввести выражение «Block», используя диалоговое окно «Custom Command», расположенное в меню «Command» главного окна. Когда объект «Simulation» получает выражение «Block», он инициирует попытку входа активным транзактом во временный блок, описываемый этим выражением. Этот метод называется моделированием вручную. Такой блок не помещается в транслированную модель, поэтому сущность «Blocks» не становится постоянным членом модели.

Команды необходимы как для построения программы модели, так и для интерактивного взаимодействия с моделью. Они включают операторы описания данных, операторы управления, которые включены в стандартный GPSS, и команды, которые используются только в языке GPSS World.

5.2 Практическая часть

Задание 5.1: создать модель вручную.

Сначала нужно создать активный транзакт до того, как использовать метод моделирования вручную. Есть несколько способов это сделать. Можно ис-

пользовать команду «STOP» или функциональную клавишу для приостановки процесса моделирования. Также можно подождать до тех пор, пока процесс моделирования не будет завершен. Обычно в тот момент, когда моделирование заканчивается, в модели присутствуют активные транзакты.

Для достижения цели, поставленной в данной практической работе, воспользуемся командой «STOP». Это приведет к регистрации условия «STOP», которое остановит процесс моделирования, когда какой-либо транзакт попытается войти в блок.

Сначала следует открыть модель и создать условие «STOP». Откройте модель «Sample2.GPS». Создайте симуляцию и откройте окно блоков. В окне блоков установите маркер «STOP» на блоке «GENERATE» (кнопка «Place» панели инструментов). Запустите процесс симуляции командой «START 1000». Удалите маркер «STOP» из блока «GENERATE» (кнопка «Remove»). Поскольку у нас появился активный транзакт, мы можем ввести любое выражение Block. Выполните команду меню «Command/Custom» и в открывшемся диалоговом окне введите «Assign Price,19.95», нажмите «OK». Активный транзакт теперь обладает параметром Price, содержащим значение 19,95.

Проверим это, открыв CTC (рисунок 5.1).

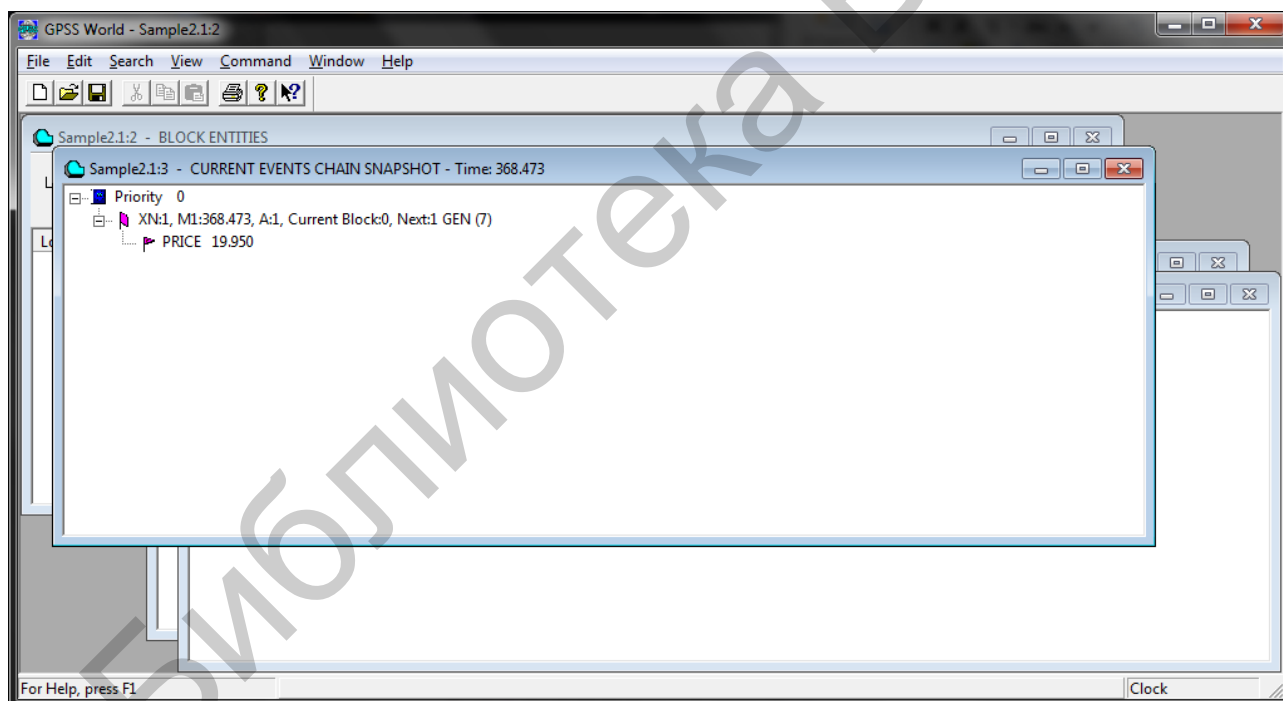


Рисунок 5.1 – Окно CTC при использовании команды «ASSIGN» вручную

Еще одним способом просмотра значений параметров активного транзакта является использование команды «SHOW». Выберите команду меню «Command/SHOW» и введите в диалоговом окне «SHOW P\$Price». Отображенное в журнале значение будет равно 19,95.

Существуют и другие возможности моделирования вручную с использованием команды «TRACE».

Задание 5.2: исследовать моделирование при помощи команды «TRACE».

Введите команду «TRACE» через меню «Command/Custom». Ввод данной команды имеет тот же эффект, что и прохождение активного транзакта через блок «TRACE»: приводит к включению индикатора «TRACE» для активного транзакта. Выполните продолжение моделирования (клавиша [F2]) и проследите за окном журнала. Обратите внимание, что в окне «Journal» появляются результаты входа в каждый блок, поскольку включен индикатор трассировки транзакта. Если изменится модельное время и начнется трассировка нового транзакта, в окне «Journal» появится соответствующее сообщение, содержащее текущее модельное время и номер транзакта. В данном случае мы увидим небольшое количество отладочных сообщений, поскольку трассируется только один транзакт. Окно «Journal/Simulation» отображено на рисунке 5.2.

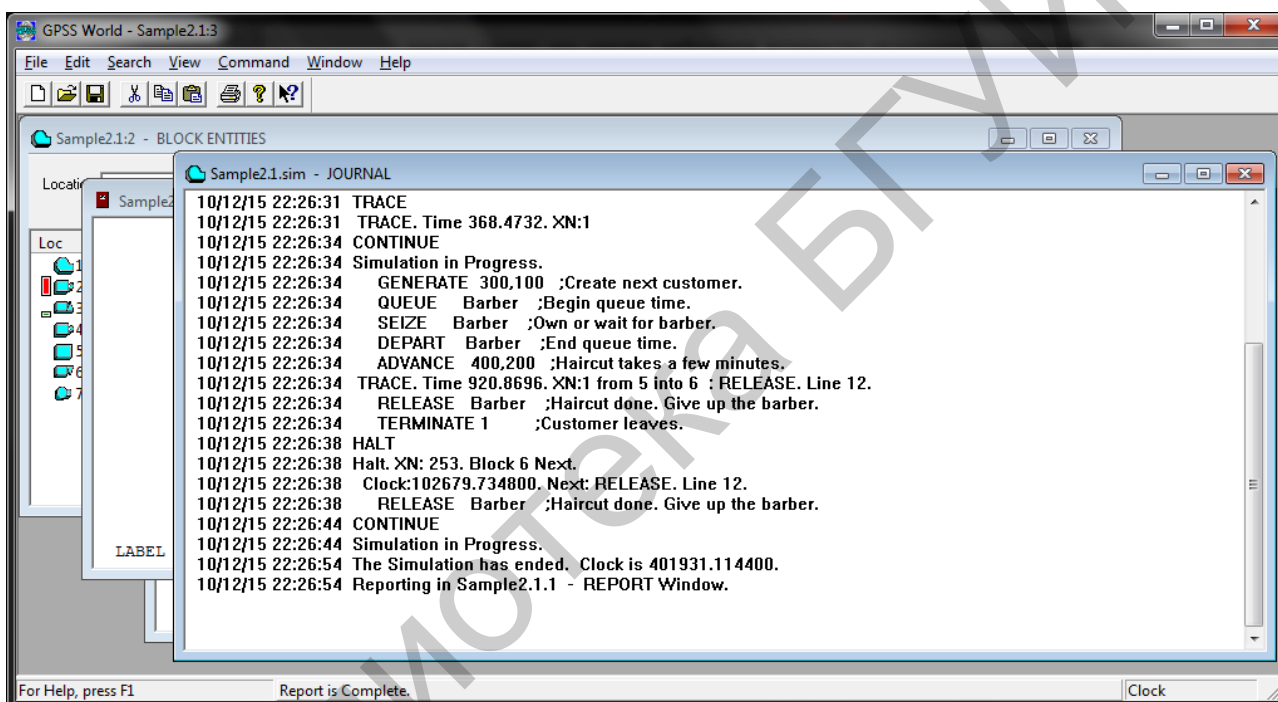


Рисунок 5.2 – Окно журнала с выполненной командой «TRACE»

Если необходима трассировка всех транзактов, то следует поместить блок «TRACE» в текст модели и выполнить ретрансляцию.

Выражение для моделирования вручную, как и любое другое, может быть назначено на функциональную клавишу для дальнейшего использования.

Моделирование вручную открывает следующие возможности:

- отправлять на выполнение в работающую модель любой блок, за исключением блока «GENERATE»;
- создавать транзакты с помощью блока «SPLIT», изменять маршрут транзактов при помощи блока «TRANSFER» или изменять параметры при помощи блока «ASSIGN»;
- проверять принадлежность к группам с помощью блоков «EXAMINE» и «SCAN»;

– вручную изымать ресурсы из оборота при помощи блоков «FUNAVAIL» и «AVAIL»;

– активировать любой блок модели, введя команду «EXECUTE Block» в режиме моделирования вручную.

Однако только активный транзакт предпринимает попытку войти в блок, который описан с помощью выражения, отправленного вручную.

Моделирование вручную является очень мощной и гибкой возможностью, а также удобным инструментом для исправления неправильных условий и проведения экспериментов. Все, что требуется сделать, это передать последовательность блоков активному транзакту при помощи нескольких нажатий «горячей» клавиши.

5.3 Содержание отчета

1. Титульный лист.
2. Цель работы.
3. Краткие теоретические сведения.
4. Реализация решения задачи.
5. Выводы.
6. Список использованных источников.

5.4 Контрольные вопросы

1. Функциональное назначение команд в системе GPSS World.
2. Перечень команд, применяемых в системе GPSS World.
3. Порядок применения и выдачи команд в процессе моделирования.
4. Особенности записи переменных в качестве операнда команд.
5. Порядок процесса ручного моделирования.
6. Применение СТС и СБС в процессе ручного моделирования.
7. Просмотр значений параметров активного транзакта.
8. Команда «TRACE» в исследовании процесса моделирования.
9. Порядок индивидуальной трассировки транзактов.
10. Порядок трассировки всех транзактов модели.

Список использованной литературы

1. Учебное пособие по GPSS World / пер. с англ. В. В. Девяткова ; под ред. К. В. Кудашова. – Казань : Мастер Лайн, 2002. – 272 с.

2. Шевченко, Д. Н. Имитационное моделирование на GPSS : учеб.-метод. пособие для студентов технических специальностей / Д. Н. Шевченко, И. Н. Кравченя. – Гомель : БелГУТ, 2007. – 97 с.

3. Томашевский, В. Н. Имитационное моделирование в среде GPSS / В. Н. Томашевский, Е. Г. Жданова ; под ред. Е. Г. Жданова. – М. : Бестселлер, 2003. – 416 с.

Практическая работа №6

ГРАФИЧЕСКИЕ ОКНА GPSS WORLD

Цель работы

Изучить назначение и функции графических окон в системе GPSS World.

6.1 Теоретическая часть

Система GPSS World позволяет наблюдать и взаимодействовать с моделью посредством 17 графических окон. В дополнение к ним имеются четыре окна для осуществления операций ввода и вывода. Десять окон обновляются динамически в процессе моделирования, в то время как другие являются статическими, позволяя сделать снимок модели в текущий момент моделирования. Почти все окна «Snapshot», за исключением окна «User STOPs Snapshot», недоступны в студенческой версии. В таблице 6.1 приведен список окон системы GPSS World с кратким описанием.

Таблица 6.1 – Графические окна GPSS World

Название графического окна	Функция
Blocks	Обзор блоков модели
Expressions	Обзор выражений, исполняемых в процессе моделирования
Facility Entities	Обзор устройств модели
Logicswitch	Обзор динамики изменения значений логических ключей
Matrix	Обзор динамики изменения элементов матрицы
Plot	Обзор изменения значений СЧА и выражений с помощью графиков
Queues	Обзор динамики изменения очередей
Savevalues	Обзор изменения содержимого ячеек
Storages	Обзор изменения содержимого памяти
Table	Обзор изменения состояния таблицы или Q-таблицы в виде диаграммы

В дополнение к приведенным в таблице 6.1 окнам, обновляемым динамически, в системе GPSS World имеется возможность сделать статические снимки одного транзакта, цепей СТС и СБС, группы транзактов и чисел, а также пользовательских цепей («Userchains»). Все графические окна позволяют взаимодействовать с моделью посредством мыши. Статические окна предназначены для отображения состояния системы в конкретный момент времени и не обновляются динамически. Они могут сообщить информацию об одном транзакте, транзактах в разных цепях и группах в модели.

Все окна открываются с помощью пунктов Simulation Snapshot (Снимок моделирования) и Simulation Window (Окно моделирования) в выпадающем меню пункта Window главного меню. При этом появляются всплывающие меню, в которых можно выбрать те или иные окна моделирования.

Рекомендуется использовать одновременно только несколько динамических окон в процессе моделирования. Но их можно открыть значительно больше, когда моделирование приостановлено.

6.2 Практическая часть

Задание 6.1: изучить назначение графического окна «Block Entities».

Откройте модель «Sample9.GPS». Запустите процесс моделирования. В диалоговом окне введите команду «STOP 5», которая остановит транзакт 5, когда он станет активным.

Откройте окно «Block Entities». Установите упрощенный режим отображения (снимите галочку «View/Entity Details»). Окно блоков примет вид, показанный на рисунке 6.1.

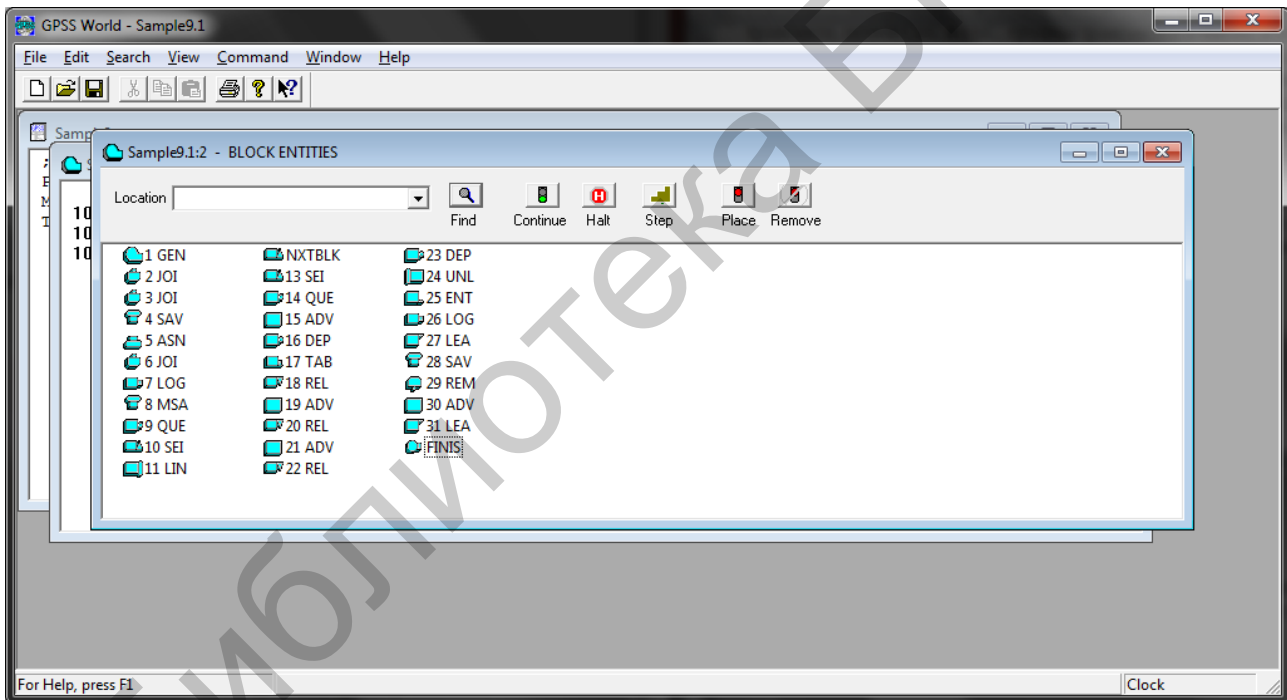


Рисунок 6.1 – Режим упрощенного просмотра окна «Block Entities»

При отладке модели все, что может понадобиться, это быстрый взгляд на поток транзактов. Больше количество подробностей доступно в режиме подробного просмотра. Запустите модель командой «START 111111111,NP». Модель будет остановлена, когда транзакт 5 станет активным. Используя клавишу [F5], можно наблюдать за перемещением транзактов из блока в блок. Когда транзакт 5 войдет в блок «GENERATE», он станет подсвечен. Теперь переместите транзакт 5 непосредственно в блок «TERMINATE», задав команду

вручную. Откройте меню команд «Command/Custom», введите «Transfer ,Finis» и нажмите «ОК». Обратите внимание, что диалоговое окно не закрылось по нажатию клавиши [Enter]. Это связано с тем, что в диалоговом окне «Custom Command» есть возможность введения многострочных команд. Следует нажать кнопку «ОК» в этом диалоговом окне. Теперь убедитесь, что последний блок модели, обозначенный надписью «Finis», находится в пределах видимости, а затем нажмите клавишу [F5]. Транзакт 5 вошел в блок «TERMINATE».

На стадиях проектирования и отладки окно «Block Entities» предоставляет легкий путь определения ошибок в потоке транзактов. Вы можете протестировать иные пути для транзактов и их влияние на поведение модели. Теперь, поскольку условие «STOP» было устранено из модели благодаря ликвидации транзакта 5, запустите модель еще раз. Для начала поместите условие «STOP» в блок «ADVANCE». Нажмите кнопку «Place» и клавишу [F2]. Моделирование будет продолжаться до тех пор, пока транзакт не будет готов войти в блок «ADVANCE».

Задание 6.2: рассмотреть процесс моделирования в окне «Block Entities» пошагово.

Нажмите клавишу [F5] несколько раз и проследите за перемещением транзакта от блока к блоку. Обратите внимание на то, как трассировочные сообщения записываются в окно «Journal» при использовании команды «STEP». Теперь переключитесь на режим детального отображения в окне «Blocks» (установить галочку в меню «View/ Entity Details») и оцените то, что произошло в системе, изучив значения счетчиков входа в блоки. Обратите внимание, что они предоставляют всю информацию, которая отображается в стандартном отчете.

Уберите все условия остановки «STOP». Для этого выберите команду меню «Window/Simulation «Snapshot»/UserStops» и нажмите кнопку «Remove All». Продолжите процесс моделирования клавишей [F2]. Следите за процессом моделирования в окне «Block Entities» в том или ином режиме просмотра. Оба режима могут быть полезны. Иногда нужно посмотреть, в каком месте происходит накопление транзактов, а в других случаях необходимо сделать более точный подсчет количества транзактов в разных блоках.

Задание 6.3: исследовать окно «Facilities».

Загрузив модель «Sample9.GPS», запустите процесс моделирования, откройте окно устройств командой меню «Window/Simulation Window/Facilities Window» и запустите моделирование командой «START 111111111,NP». Отобразится окно устройств, показанное на рисунке 6.2.

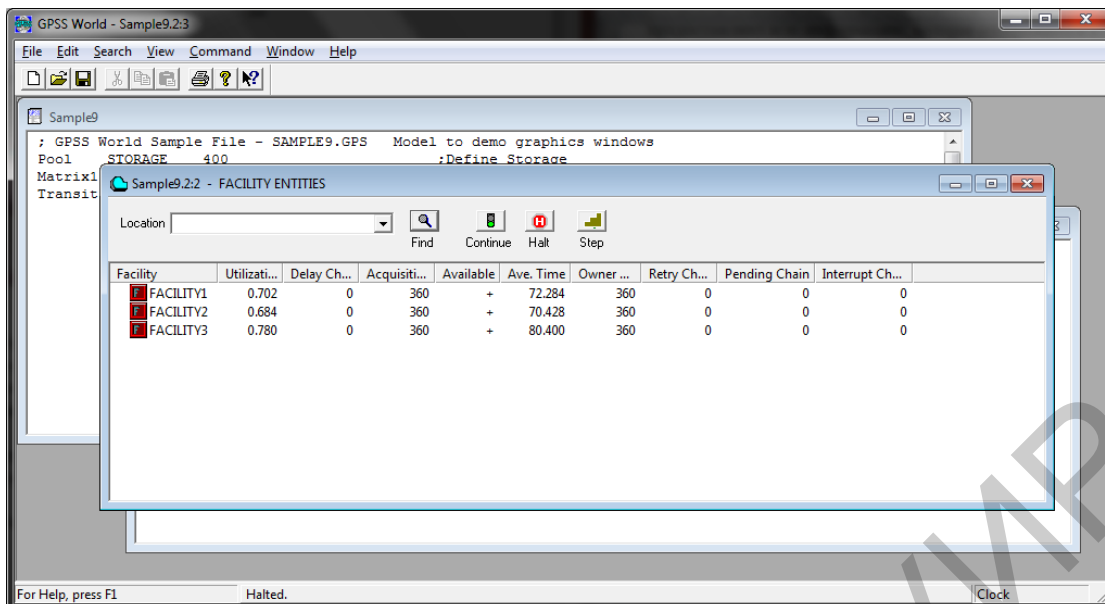


Рисунок 6.2 – Окно устройств в режиме расширенного просмотра

Переключитесь в упрощенный режим просмотра. Обратите внимание, как процесс моделирования влияет на состояние устройств. В этой модели символ устройства изменяется, становясь серым (устройство свободно) либо красным (устройство занято). Маленькая очередь представлена в виде белой полоски на правой стороне значка. Полоска становится большой, когда количество транзактов в очереди превышает значение 10. Можно изменять данное значение в режиме моделирования вручную. Очередь на самом деле представляет собой счетчик транзактов, находящихся в цепи задержек устройства и ожидающих поступления на устройство.

Выведите устройство Facility из строя. Для этого прервите процесс моделирования клавишей [F4]. Введите в командной строке («Command/Custom») команду «FUNAVAIL Facility1» и нажмите «ОК». Значок устройства стал желтого цвета. Переключитесь на расширенный режим просмотра. Введите команду «STEP 150» в командной строке и следите за выстраивающейся в цепи задержек очередью транзактов. Разрешите устройству обработать транзакты, ожидающие обработки. Для этого в командной строке введите команду «FAVAIL Facility1», которая делает устройство доступным. Запустите моделирование клавишей [F2]. Очередь ожидающих обработку транзактов начнет уменьшаться.

Закройте окно «Facilities».

Задание 6.4: использовать метод моделирования вручную для изменения одной из ячеек матрицы.

Откройте окно «Matrix» через меню «Window/Simulation Window/Matrix Window». Появится диалоговое окно, запрашивающее имя матрицы (рисунок 6.3).

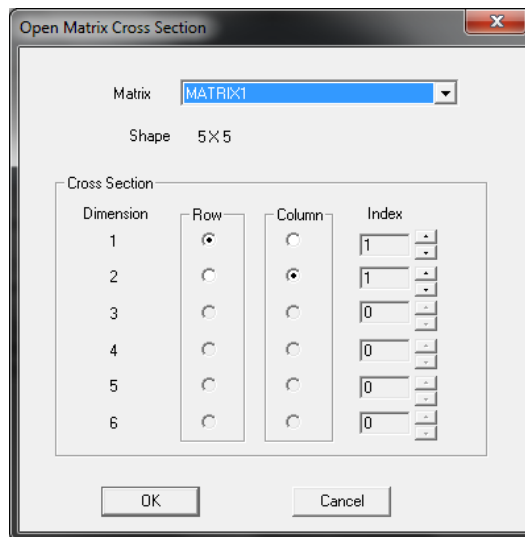


Рисунок 6.3 – Диалоговое окно создания матрицы

Поскольку в данной модели есть только одна матрица, ее имя сразу появится в выпадающем меню. Нажмите «ОК». Запустите процесс моделирования клавишей [F2]. Вы увидите изображение «Matrix1». Следите за изменением ячейки, находящейся на второй строке во втором столбце, в процессе прохождения транзактов через блок «MSAVEVALUE» и сохранения среднего объема очереди в матрице. Динамическое окно матрицы представлено на рисунке 6.4.

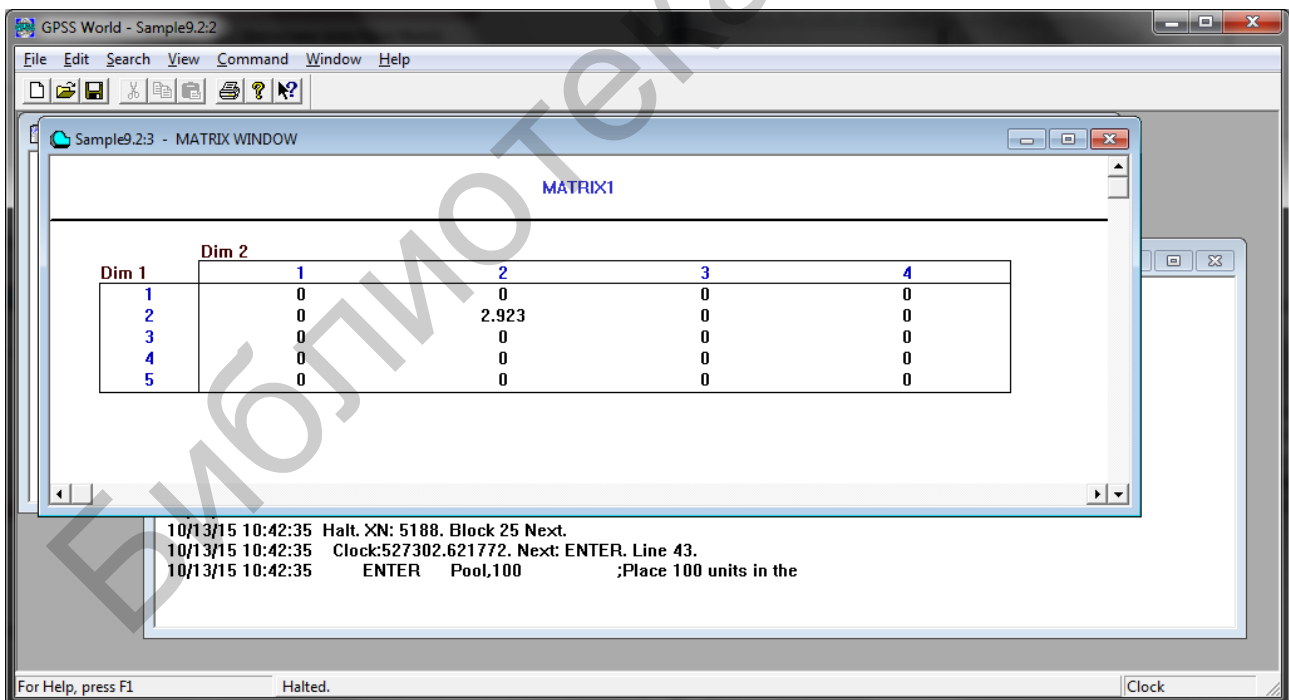


Рисунок 6.4 – Динамическое окно матрицы

Возможно настроить окно «Matrix», если сделать видимыми все элементы поперечного среза. Для увеличения среза предназначены полосы прокрутки.

Приостановите моделирование клавишей [F4]. Наберите команду «MSAVEVALUE Matrix1,3,3,1000» в командной строке. Элемент матрицы, находящийся в третьем столбце третьей строки, приобрел новое значение.

Система GPSS World позволяет объявлять матрицы размером вплоть до шести измерений. Срез для просмотра открывается с помощью диалогового окна меню «View/Cross Section».

Задание 6.5: создать 6-размерную матрицу в окне «Matrix».

В командной строке введите команду

«MATRIX2 Matrix,2,3,4,2,2,2»

Диалоговое окно отображения матрицы показано на рисунке 6.5.

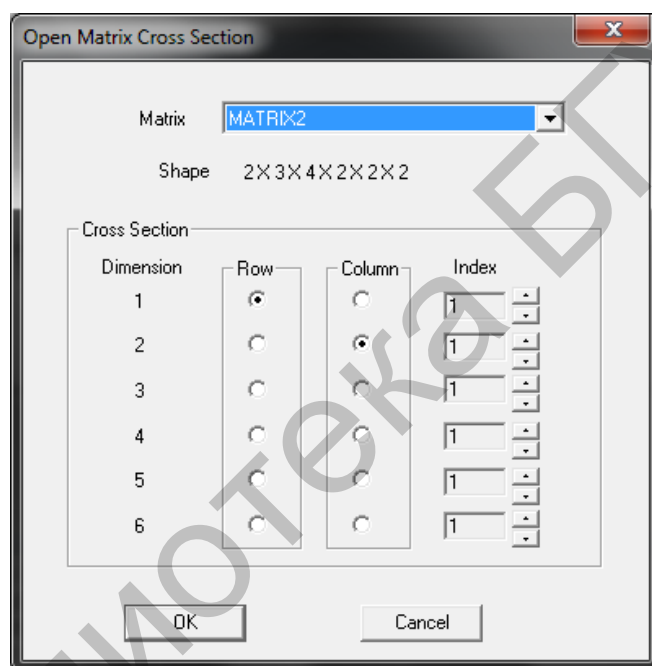


Рисунок 6.5 – Диалоговое окно создания 6-размерной матрицы

Размеры всех данных приведены под названием матрицы. Обратите внимание, что по умолчанию первая цифра используется для обозначения строк, вторая – для обозначения столбцов. Другие данные устанавливаются с первого элемента в каждом направлении.

Если требуется изменить видимый срез матрицы, то необходимо выполнить следующие действия:

- 1) определить, какие измерения будут использоваться в качестве строк и столбцов;
- 2) внутри группы «Cross Section» диалогового окна щелкнуть на колонке «Row», чтобы выбрать размер строки матрицы;
- 3) затем щелкнуть на колонке «Column», чтобы выбрать размер столбца матрицы;

4) для оставшихся размеров выберите номер индекса, на котором срез переходит в соответствующее измерение, заполните значения в крайнем правом столбце и нажмите «ОК».

Это действие выведет на экран выбранный срез матрицы. Также возможно открытие нескольких срезов одной матрицы при открытии нового окна «Matrix» для каждого среза.

Задание 6.6: изучить функции окна хранилищ «Storage Entities».

Откройте модель «SAMPLE9.GPS» и запустите процесс симуляции. Откройте окно «Storages» командой меню «Window/Simulation Window/Storages Window». Вид окна «Storage Entities» показан на рисунке 6.6.

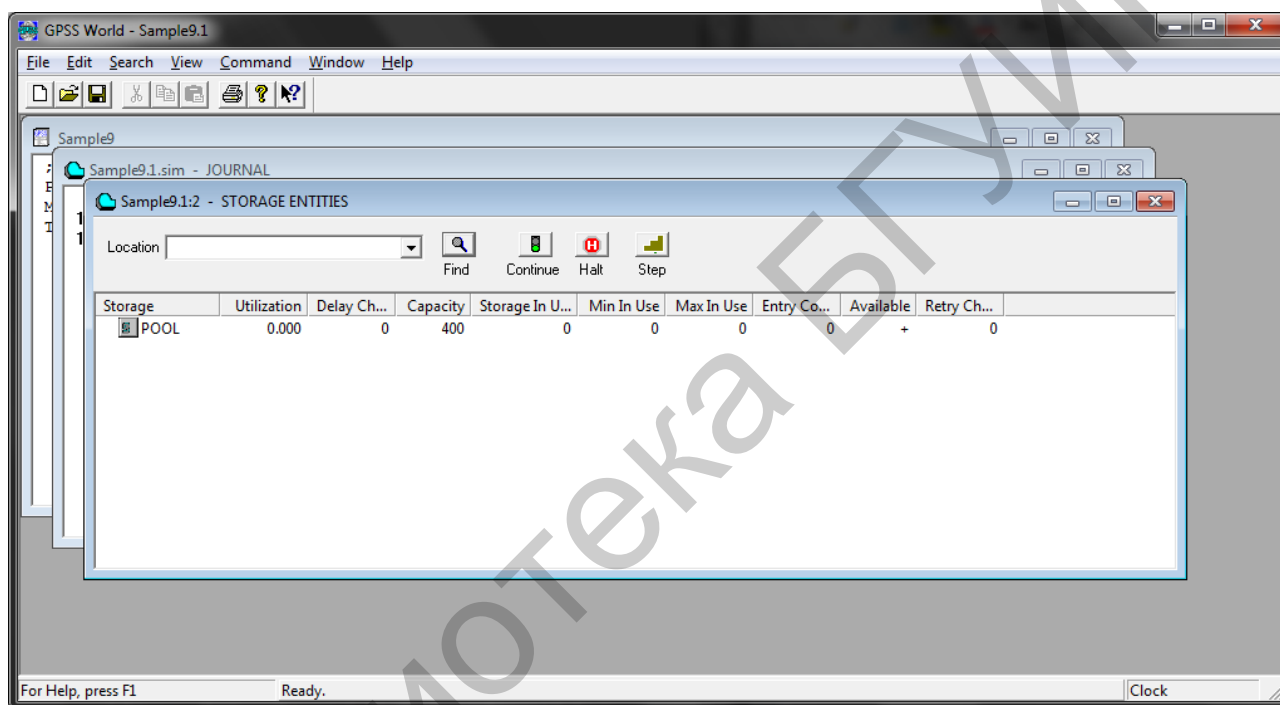


Рисунок 6.6 – Окно «Storage Entities»

Запустите процесс моделирования. Обратите внимание, что процесс моделирования привел к изменению состояния сущности «Storage». Возможно изменить ее состояние вручную. Выведем хранилище Pool из строя. Для этого нужно воспользоваться командой «SUNAVAIL Pool» в командной строке. Значок хранилища Pool окрасился в желтый цвет. Запомните содержимое цепи задержек хранилища и введите команду «STEP 150» в командной строке. Обратите внимание, как очередь из транзактов выстраивается в цепи задержек хранилища. Запустите обработку прибывающих транзактов командой «SAVAIL Pool». Данная команда делает хранилище доступным. Чтобы предотвратить накопление транзактов в очереди, система GPSS World немедленно выдает максимально возможному числу транзактов в цепи задержек запрашиваемое ими место для хранения, но не двигает их дальше по модели до тех пор, пока до них не дойдет очередь. Поэтому можно увидеть, как четыре транзакта входят в блок

«ENTER»), а цепь задержек уменьшается на 4. Снова запустите модель. Обратите внимание, что очередь уменьшается. Этот режим может быть использован для манипуляции сущностями в графическом окне.

Задание 6.7: изучить функции окна таблиц «Table».

Запустите модель «Sample9.GPS». Создайте процесс моделирования и откройте окно таблиц командой меню «Window/Simulation Window/Table Window». В диалоговом окне выбора таблицы выберите таблицу «Transit». Эта таблица является единственной для данной модели, но если бы в модели имелось несколько таблиц, можно было бы выбрать одну из них посредством выпадающего меню. Нажмите «ОК». Появится изображение «TABLE»1, на котором видны оси X и Y, но отсутствует гистограмма. Запустите модель, и будем следить за ходом моделирования (рисунок 6.7).

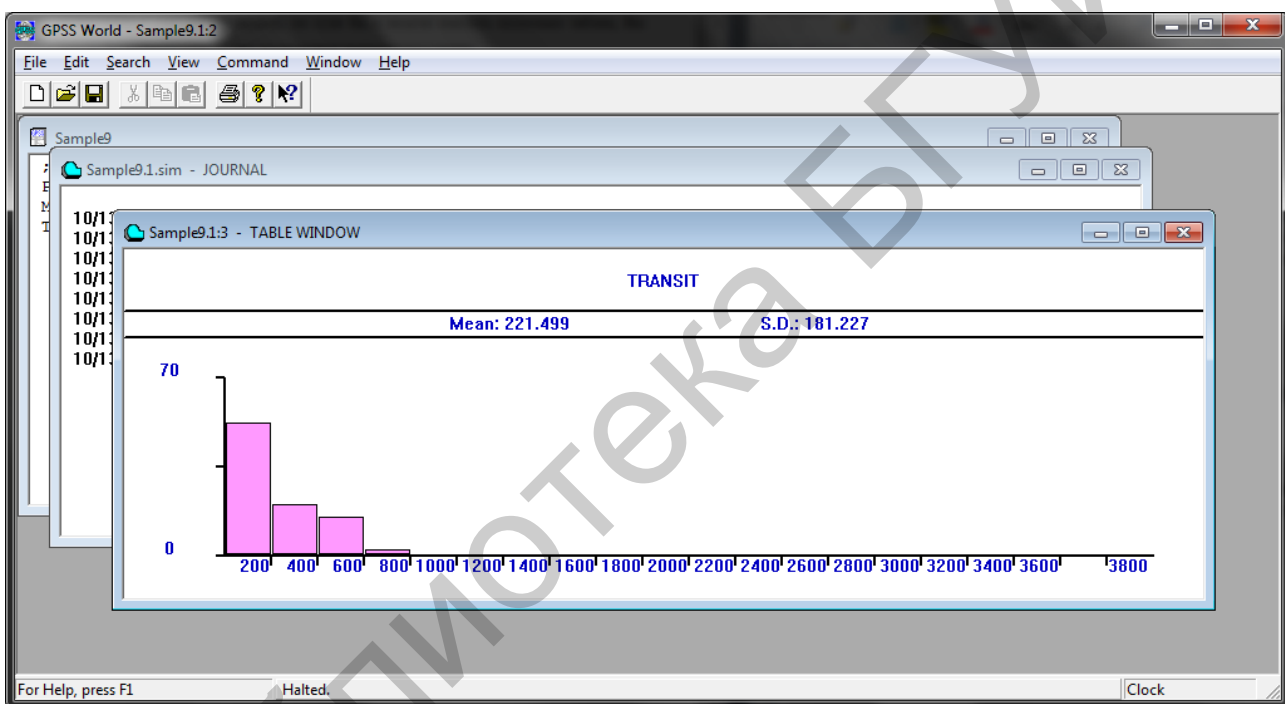


Рисунок 6.7 – Окно «Table»

Обратите внимание, как состояние модели влияет на изменение гистограммы. С ростом счетчиков в каждом классе частот происходит изменение среднего и стандартного отклонений. Когда гистограмма становится слишком высокой, система GPSS World автоматически масштабирует ее. Чтобы создать гистограмму для окна «TABLE», вам необходимо определить ее с помощью команды «TABLE» или «QTABLE». Система GPSS World автоматически отобразит ее состояние.

Задание 6.8: изучить функции окна «Plots».

В модели «Sample9.GPS» создайте симуляцию, запустите выполнение модели и остановите его клавишей [F4]. Приостановив моделирование, выполните

построение графика для нескольких переменных. Система GPSS World позволяет строить до восьми графиков в одном окне «Plots». Обратите внимание, что останавливать модель, для того чтобы построить график, необязательно.

Откройте окно графика командой меню «Window/Simulation Window/Plot Window». Появится окно «Edit Plot», показанное на рисунке 6.8.

Окно «Edit Plot» имеет два назначения. Можно использовать его только лишь для установки осей и определения основных значений, либо можно также определять дополнительные выражения, которые необходимо построить на графике.

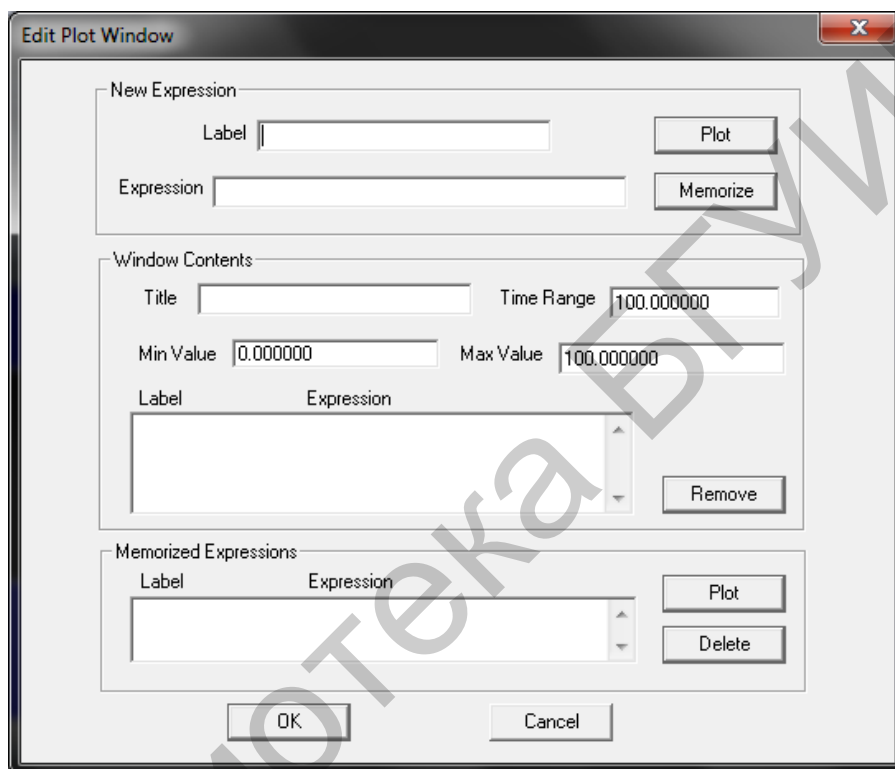


Рисунок 6.8 – Окно «Edit Plot»

Поле «Label» должно содержать название, используемое в легенде, расположенной в нижней части графика. Введите в него строку «Storage in Use». В поле «Expression», определяющем первое значение на графике, введите «\$\$Pool». В поле «Plot Title» вводится имя, описывающее оба параметра, которые требуется построить на графике. Этими параметрами являются размер очереди «Total_Time» и среднее время пребывания в очереди, представляющие собой общее время пребывания в модели с момента создания до момента уничтожения. Введите в него строку «Storage in Use & Process Time».

Для каждого графика необходимо добавить интервал времени, на котором будет построен график. Следует сделать этот интервал достаточно длинным, чтобы график не «пролетел» мгновенно в ходе работы модели, но не настолько длинным, чтобы он не начал «заедать». Возможно, придется провести несколько экспериментов, для того чтобы определить корректное значение.

Следует знать, что ограниченное количество точек графика сохраняется для перерисовки окна «Plots». По умолчанию для каждого выражения сохраняется не более 10 000 точек. Если были изменены размеры окна «Plot», а крайняя левая часть графика не была перерисована, то следует увеличить количество сохраняемых точек в настройках модели. Если обновить окно, поменять размеры окна либо сделать что-то, что приведет к автоматическому обновлению (к примеру, перекрытие окна «Plot» другим окном), то перерисованы будут только лишь те точки, которые были сохранены. Вполне очевидно, что при увеличении количества сохраняемых точек будет увеличиваться количество расходуемой на эту операцию памяти. Поскольку система GPSS World выделяет большое количество виртуальной памяти, это не должно быть проблемой.

Добавьте интервал времени для оси X. В поле Time Range введите «10000». Значения по оси Y по умолчанию находятся в диапазоне от 0 до 100. Измените максимальное значение на 200. Нажмите кнопку «Plot». Эти данные останутся доступными, если мы сохраним объект «Simulation», для того чтобы позднее еще раз построить график кнопкой «Memorize».

Добавьте второй набор значений для того же графика: в поле «Label» введите «Process Time», в поле «Expression» – «QT\$Process_Time» и нажмите кнопки «Plot» и «Memorize». Начните процесс моделирования (клавиша [F2]). Окно «Plots» в процессе моделирования примет вид, показанный на рисунке 6.9.

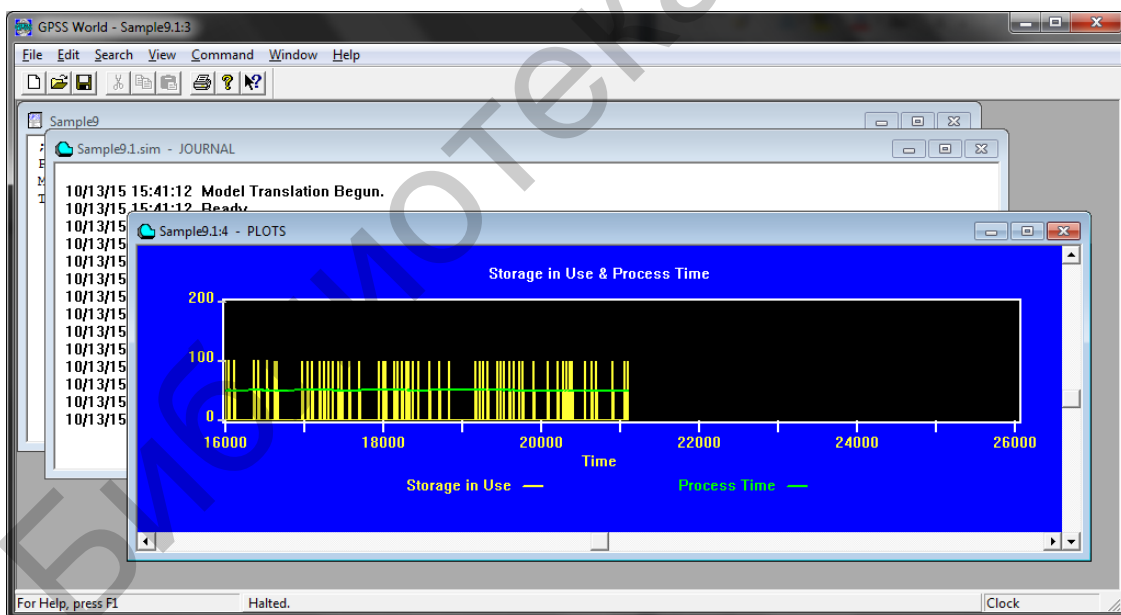


Рисунок 6.9 – Окно «Plots» в процессе моделирования

Таким же образом можно построить до восьми значений на одном графике, продолжая добавлять название и выражение в соответствующие поля, а затем нажимая кнопки «Plot» и «Memorize».

Задание 6.9: изучить функции окон «Logicswitches» и «Savevalues».

Запустите моделирование в модели «Sample9.GPS». Остановите процесс моделирования (клавиша [F4]).

Откройте окна одновременно командами меню «Window/Simulation Window/Logicswitches Window» и «Window/Simulation Window/Savevalues Window». Осуществите несколько шагов моделирования, следя за изменением значений. Нажмите клавишу [F5] несколько раз и проследите за тем, как изменяются значения. Переключитесь на упрощенный режим просмотра для каждого окна. Заметно, что окно «Logicswitches» ничем кардинально не отличается от своего детализированного аналога (рисунок 6.10).

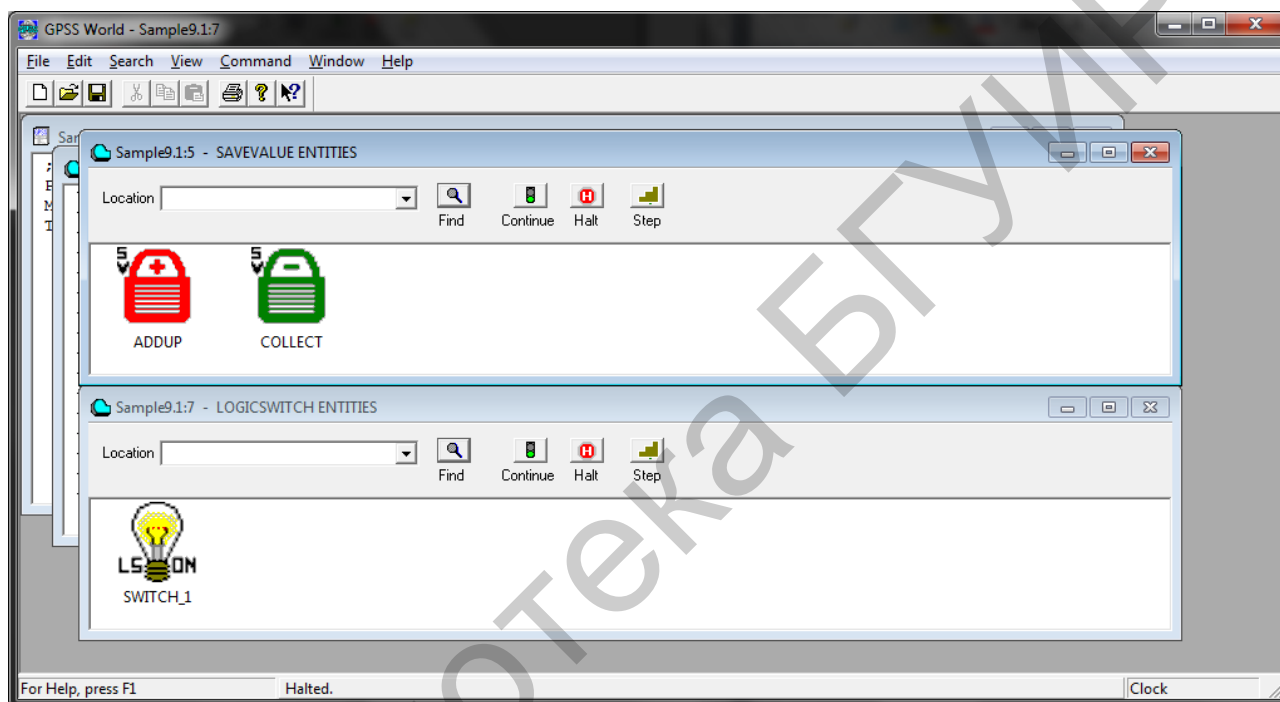


Рисунок 6.10 – Окна «Logicswitches» и «Savevalues»

В окне «Savevalues» в упрощенном режиме просмотра можно видеть только лишь значение сущности Savevalue (положительное, отрицательное или ноль). После того как моделирование началось, в модели не будет ни одной сущности Savevalue с нулевым значением.

Задание 6.10: изучить функции окна очередей «Queue».

Запустите модель «Sample9.GPS», создайте симуляцию и запустите процесс моделирования.

Откройте окно очередей командой меню «Window/Simulation Window/Queues Window».

Отобразится подробная информация об объекте «QUEUE». В этой модели объект «QUEUE» использовался несколько иным способом. Обычно мы представляем очередь как линию из требований, ожидающих доступ к ресурсу. В данной модели мы используем очередь «QUEUE» не только для сбора значений

времени ожидания в очереди, но также и для сбора значений времени на обработку. Каждый индивидуальный ресурс («Facility» или «Storage») обладает собственной цепью, в которой накапливаются транзакты, ожидающие доступ к этому ресурсу. Блоки «QUEUE» и «DEPART» при совместном использовании с этими ресурсами собирают информацию об отрезках времени и длине цепей. Перейдите в пошаговый режим (остановите моделирование клавишей [F4] и нажимайте клавишу [F5]). Обратите внимание на изменение содержимого окна очередей при пошаговом исполнении программы.

Окно «QUEUE» в упрощенном режиме просмотра показывает размеры очереди. Если очередь заполнена, то количество транзактов в очереди (маленькое, среднее или большое) определяется белой полоской на правой стороне значка.

Задание 6.11: изучить функции графического окна «Expression».

Запустите моделирование модели «Sample9.GPS». Откройте окно выражений командой меню «Window/Simulation Window/Expressions Window». Появится окно «Expression», требующее ввести значения в поля «Label» и «Expression».

В поле «Label» введите «Clock», в поле «Expression» – «AC1». По аналогии с окном «Plot» используйте кнопки «View» и «Memorize», для того чтобы отобразить указанные выражения и запомнить их для использования в будущем. Добавьте еще два значения таким же образом: метка «Active Transaction» и выражение «XN1», а также метка «Wait+Process» и выражение «Q\$Tot_Process». Окно редактирования выражений примет вид, показанный на рисунке 6.11.

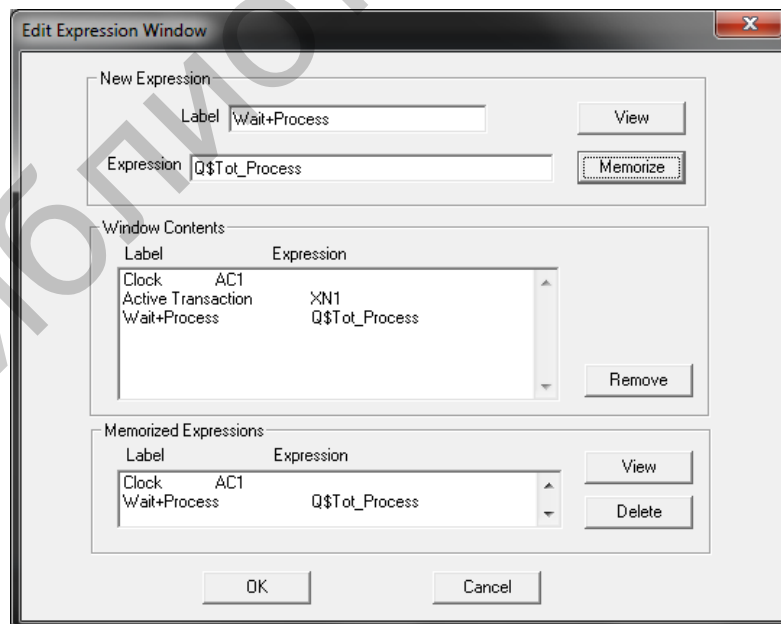


Рисунок 6.11 – Окно редактирования выражений

Нажмите кнопку «ОК». Теперь можно наблюдать за значением системных часов, количеством активных транзактов и размером очереди «Tot_Process» в процессе работы модели (рисунок 6.12).

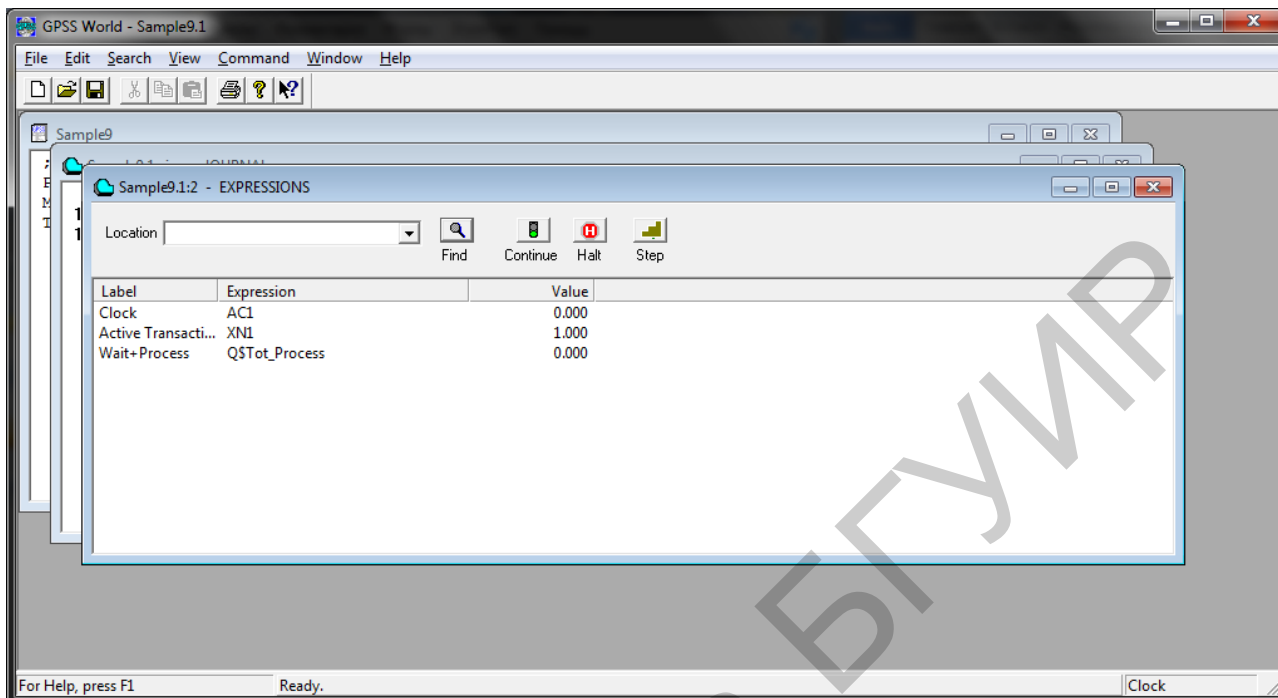


Рисунок 6.12 – Окно выражений

Задание 6.12: запустить процесс моделирования и проследить за изменением значений.

Если при создании выражения, оно было сохранено (кнопка «Memorize»), то при последующем открытии окна редактора выражений его можно выбрать из списка «Memorized Expressions» (см. рисунок 6.11).

Стоит отметить, что вы можете задавать выражения для просмотра в настройках модели на закладке «Expressions». Каждый раз, когда вы создаете объект «Simulation», эти значения будут доступны для использования в окнах «Expressions» или «Plot».

Задание 6.13: изучить функции окон «Snapshot».

Окна «Snapshot» действуют подобно фотоаппарату. Они запечатлевают один момент из жизни модели и в отличие от динамических окон не обновляются в процессе моделирования. После этого мы выберем один из транзактов по его номеру, для того чтобы изучить, как работает окно «Transaction Snapshot». Мы можем использовать один и тот же транзакт для двух разных снимков, поскольку модель остановлена.

Запустите процесс моделирования и остановите его через некоторое время. Откройте окно СТС (команда меню «Window/Simulation «Snapshot»/CEC «Snapshot»»). Транзакты в окне сгруппированы по приоритетам. Если присутствует знак «+», то всегда можно щелкнуть по нему, чтобы отобразить допол-

нительную информацию. Нажатие на первый знак «+» отобразит список транзактов, находящихся в цепи, в форме флагов. Нажатие на второй знак «+» отобразит список параметров соответствующего транзакта. Запомните номер транзакта, находящегося в данный момент в СТС. Откройте окно «Transaction «Snapshot»» командой меню « Window/Simulation «Snapshot»/Transaction Snapshot», и в диалоговом окне введите номер выбранного транзакта. Откроется окно, содержащее информацию о выбранном транзакте. Строка информации о транзакте содержит следующие параметры: текущее время, приоритет, текущий блок, номер следующего блока, тип и номер линии, номер «Assembly Set».

Самостоятельно изучите окна «FEC «Snapshot», «Transaction Groups» и «User Chains», поскольку они имеют один и тот же формат.

Окно Numeric Groups выглядит несколько иначе. Откройте его командой меню «Window/Simulation «Snapshot»/Numeric Group Snapshot». Вид окна представлен на рисунке 6.13.

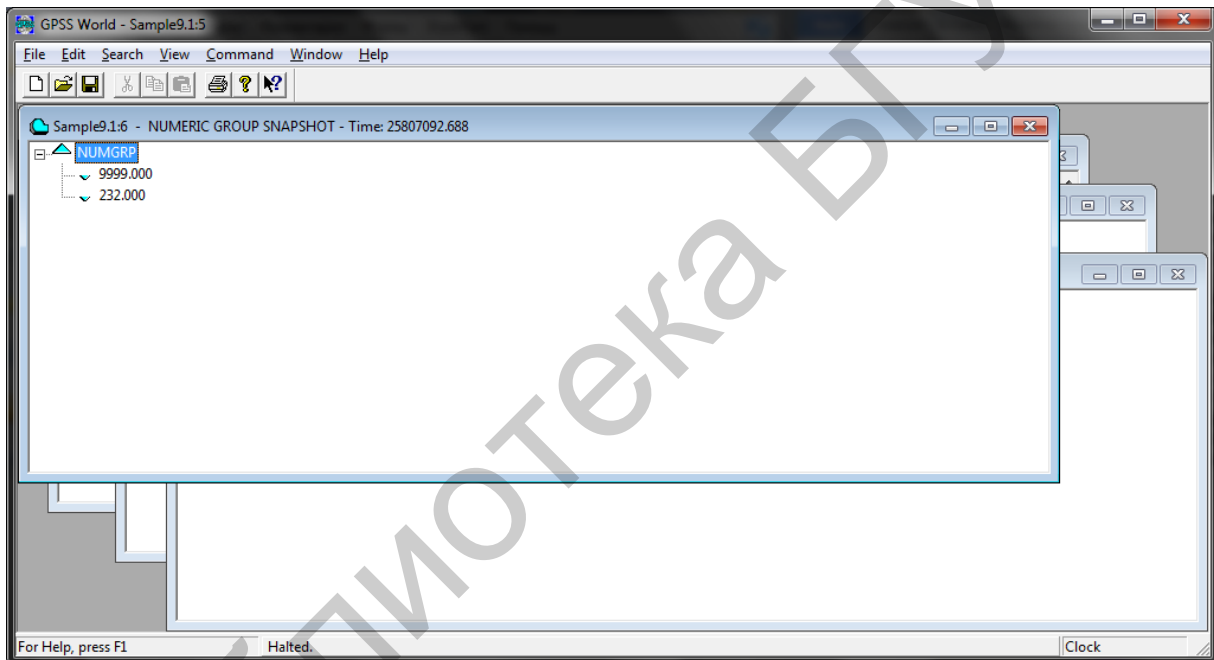


Рисунок 6.13 – Окно «Numeric Groups Snapshot»

Вы увидите два значения, которые добавляются к группе чисел в модели. Обычно при моделировании необходимо добавлять значения параметров транзактов или устанавливать значения в начале прогона для использования в различных тестовых ситуациях. В этой модели каждый транзакт добавляет два одинаковых значения к группе.

6.3 Содержание отчета

1. Титульный лист.
2. Цель работы.
3. Краткие теоретические сведения.

4. Реализация решения задачи.
5. Выводы.
6. Список использованных источников.

6.4 Контрольные вопросы

1. Функциональные возможности графического окна «Blocks».
2. Функциональные возможности графического окна «Facilities».
3. Функциональные возможности графического окна «Matrix».
4. Функциональные возможности графического окна «Storages».
5. Функциональные возможности графического окна «Table».
6. Функциональные возможности графического окна «Plot».
7. Функциональные возможности графических окон «Logicswitches» и «Savevalues».
8. Функциональные возможности графического окна «Queues».
9. Функциональные возможности графического окна «Expressions».
10. Функциональные возможности графических окон «Snapshot».

Список использованной литературы

1. Кудрявцев, Е. М. GPSS World. Основы имитационного моделирования различных систем / Е. М. Кудрявцев. – М. : ДМК Пресс, 2004. – 320 с.
2. Шевченко, Д. Н. Имитационное моделирование на GPSS : учеб.-метод. пособие для студентов технических специальностей / Д. Н. Шевченко, И. Н. Кравчяня. – Гомель : БелГУТ, 2007. – 97 с.
3. Учебное пособие по GPSS World / пер. с англ. В. В. Девяткова ; под ред. К. В. Кудашова. – Казань : Мастер Лайн, 2002. – 272 с.

Практическая работа №7

ЗАПИСЬ И ЧТЕНИЕ ВНЕШНИХ ФАЙЛОВ

Цель работы

Изучить возможности записи и чтения информации из внешних подключаемых файлов.

7.1 Теоретическая часть

В системе GPSS World есть блоки, предназначенные для записи и чтения из внешних ASCII файлов. Эти блоки могут быть очень полезны, если требуется считать данные из внешнего файла, составить свой собственный отчет, собрать один или два набора данных, полученных в разные интервалы модельного времени, в один файл.

Для подключения внешних файлов применяется команда «INCLUDE», являющаяся немедленной командой. При получении этой команды транслятор начинает транслировать подключаемый файл GPSS-модели «example_1.txt» так, как будто подключаемая модель заменила строку с командой «INCLUDE». Допускается подключение не более пяти файлов.

Система GPSS World работает только с файлами, имеющими расширение «.txt». В этих файлах может находиться либо неоттранслированная модель GPSS-программы, либо последовательность команд. При попытке подключения с помощью этой команды оттранслированной модели с расширением «.GPS» транслятор выдает сообщение о том, что подключаемый файл должен быть только с расширением «.txt».

7.2 Практическая часть

Задание 7.1: применяя несколько простых моделей, изучить порядок использования блоков для работы с потоками данных.

Каждая модель сопровождается текстовым объектом, с которым осуществляется взаимодействие. Текстовые объекты представляют собой обычные текстовые файлы с расширением «.txt». Так, существует файл «TSTTEMPL.TXT», который можно использовать для восстановления файлов, применяемых для записи данных.

Другим способом восстановления тестовых файлов является их предварительное копирование в безопасную директорию. Эти файлы расположены в папке «Sample Models». После выполнения данной практической работы восстановите файлы «TST*.TXT» из резервной копии или же скопируйте содержимое файла «TSTTEMPL.TXT» в следующие файлы: «TSTSKINW.TXT», «TSTAPPW.TXT», «TSTSKRW.TXT», «TSTSQRW.TXT», «TSTSQR2W.TXT». Затем удалите два файла: «TSTCATW.TXT» и «TSTSTW.TXT».

Файлы «TSTRD.TXT» и «TSTSTRD.TXT» используются только для считывания данных. Содержимое этих файлов не будет изменяться, поэтому нет необходимости осуществлять их инициализацию перед началом выполнения практической работы. Имена тестовых файлов созданы посредством специальных кодов. Ниже приведен список сокращений и соответствующих процедур ввода-вывода.

- CAT = Catenate (объединить).
- SK = «SEEK» (искать).
- IN = Insert (вставить).
- SQ = Sequential (последовательный).
- R = «Replace» (заменить).
- ST = String (строка).
- RD = Read (считать).
- W = «WRITE» (записать).

В блоках «OPEN» данной модели не указан путь к файлам. Это означает, что система GPSS World будет использовать папку «Sample» для сохранения файлов.

Откройте модель «TSTRD.GPS» и файл «TSTRD.TXT». Запустите процесс моделирования, откройте окно матрицы («Window/Simulation Window/Matrix Window»), чтобы наблюдать за тем, как модель считывает данные из файла.

В окне «OPEN Matrix» в выпадающем меню будет выбрана матрица «Total», поскольку она является единственной матрицей в данной модели.

Запустите моделирование командой «START 1,NP». Окно матрицы примет вид, показанный на рисунке 7.1.

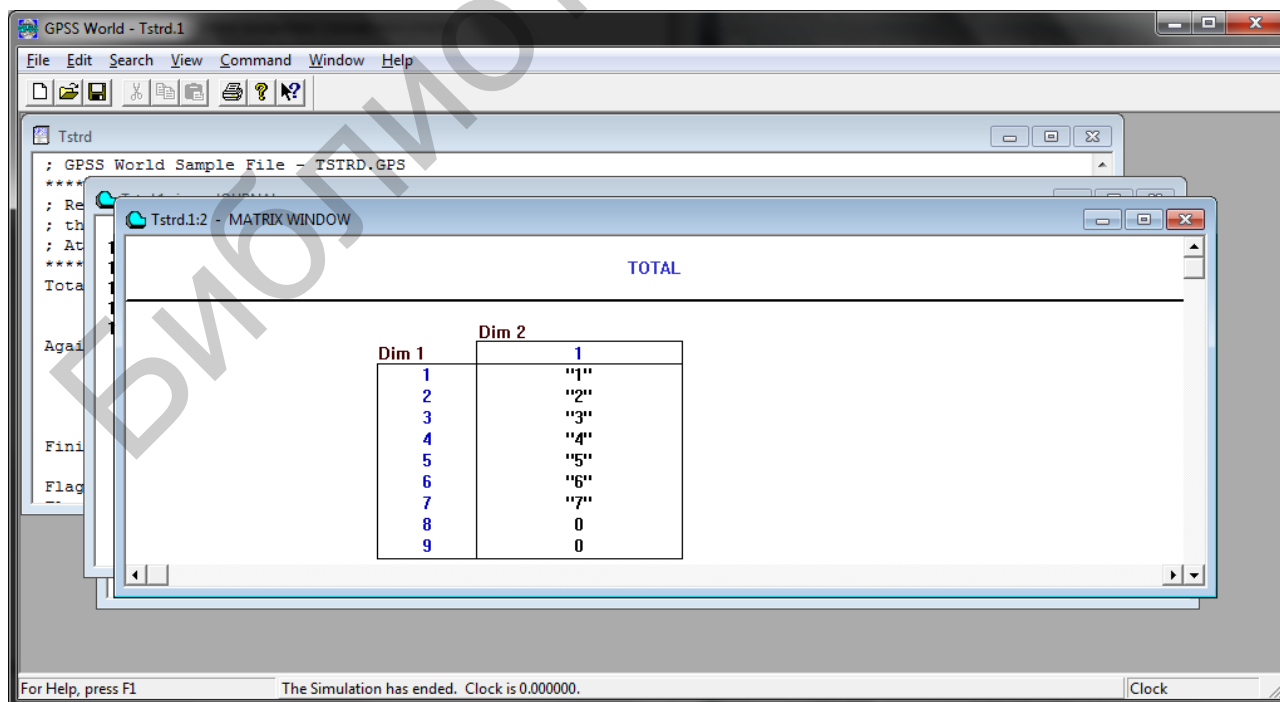


Рисунок 7.1 – Матрица «Total»

Прогон завершился очень быстро, записав в матрицу значения от 1 до 7 числами, найденными в файле.

Блок «OPEN» помещается в отдельный сегмент модели, поскольку файл необходимо открыть только один раз. Первым действием модели является открытие транзактом с высоким приоритетом файла для считывания данных.

Затем транзакт, осуществляющий считывание данных, начинает работу, циклически проходя через блок «READ» и последующие четыре блока. Каждое считанное значение помещается в качестве параметра «Numero» транзакта. Следующие три блока обновляют значение «SAVEVALUE», которое затем становится параметром и используется в качестве номера строки в матрице. Затем в блоке «MSAVEVALUE» значение, считанное в «P\$Numero», помещается в соответствующую строку матрицы «Total». По достижении конца файла транзакт отправляется в блок «CLOSE», помеченный надписью «Finis», в котором осуществляется закрытие файла данных. Затем транзакт входит в блок «TERMINATE», завершая процесс моделирования.

Если при попытке входа в блоки «OPEN» или «CLOSE» произойдет ошибка, транзакт отправится в блок, помеченный флагом «Flag» или «Flag1» соответственно. Подобную ошибку легко распознать, поскольку по умолчанию значение количества входов в эти блоки должно равняться нулю.

Если осуществлять множественные попытки считывания или записи данных, то мы можем использовать транзакт с высоким приоритетом для открытия файла с данными. После считывания данных файл будет закрыт в блоке «CLOSE».

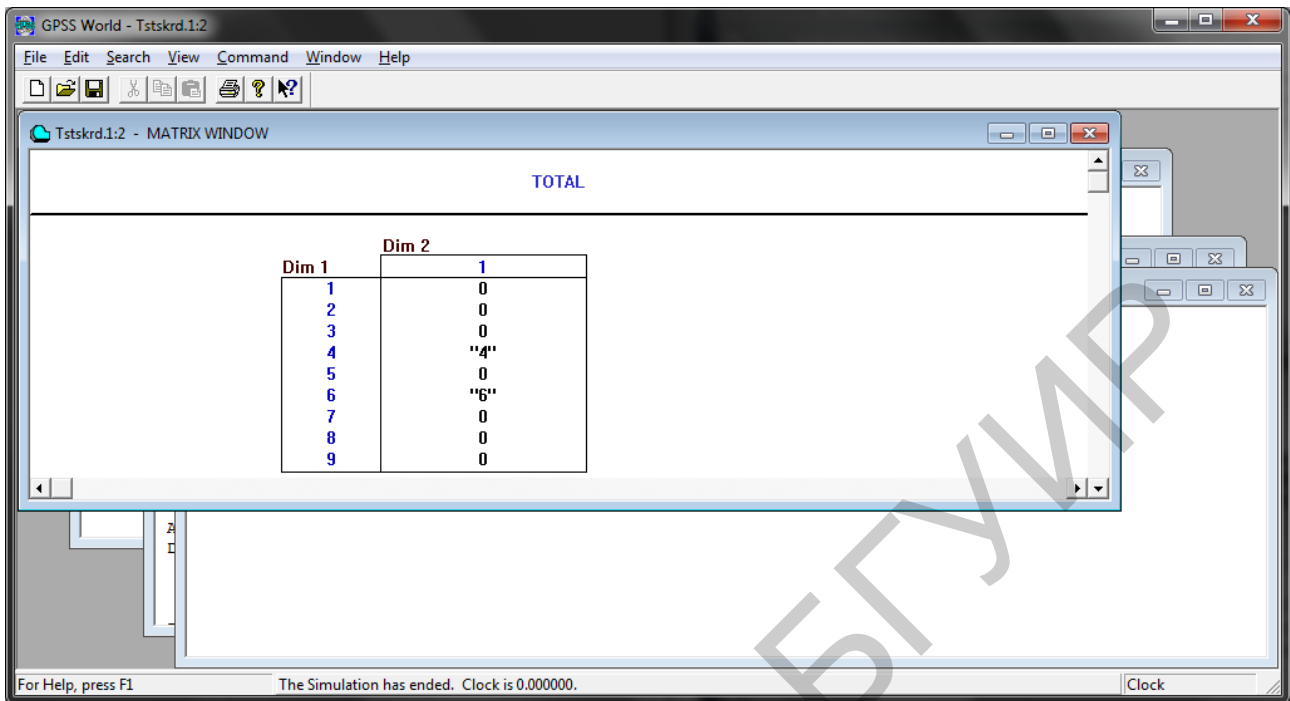
Задание 7.2: произвести поиск данных на определенных участках файла и помещение этих данных в указанные участки матрицы.

Откройте модель «TSTSKRD.GPS» и файл данных «TSTRD.TXT». Запустите процесс моделирования и откройте окно матрицы. Запустите моделирование командой «START 1,NP». Матрица «Total» примет вид, показанный на рисунке 7.2.

Процесс моделирования закончится очень быстро, но в этот раз в матрицу будут добавлены только лишь цифры 4 и 6. Механизм работы блоков ввода-вывода в данной модели следующий. Блок «OPEN» расположен в отдельном сегменте модели, поскольку требуется только одно открытие файла. Сначала транзакт с высоким приоритетом открывает файл для чтения. Затем транзакт, отвечающий за считывание данных из файла, входит в блок «SEEK», переходя на четвертую строку файла данных. Вход в блок «READ» приводит к считыванию четвертой строки. Значение помещается в матрицу, затем осуществляется тест, определяющий, есть ли необходимость в еще одной операции считывания. Происходит еще одно считывание, второе число изымается из файла и помещается в матрицу.

Если при попытке входа в блоки «OPEN» или «CLOSE» произойдет ошибка, транзакт отправится в блок, помеченный флагом «Flag» или «Flag1»

соответственно. Подобную ошибку легко распознать, поскольку по умолчанию значение количества входов в эти блоки должно равняться нулю.



The screenshot shows a window titled 'Tstskrd.1.2 - MATRIX WINDOW' with a table titled 'TOTAL'. The table has two columns, 'Dim 1' and 'Dim 2', and nine rows. The values in the 'Dim 2' column are 0, 0, 0, '4', 0, '6', 0, 0, 0.

Dim 1	Dim 2
1	0
2	0
3	0
4	"4"
5	0
6	"6"
7	0
8	0
9	0

Рисунок 7.2 – Матрица «Total»

Задание 7.3: изучить запись строки в файл, используя метод случайного доступа.

Откройте модель «TSTSKINW.GPS» и запустите процесс моделирования командой «START 1,NP». Файл данных «TSTSKINW.TXT» модифицируется. Откройте модифицированный файл с данными.

Обратите внимание, что строка «123456» была вставлена непосредственно перед третьей строкой файла. Если бы не использовался блок «SEEK», данные были бы помещены перед первой строкой. В данном примере блок «OPEN» не изолировался, поскольку была сделана только одна запись, а затем закрыт файл и прекращен процесс моделирования. Блок «SEEK» используется для того, чтобы поместить указатель текущей строки на третью позицию, в которую записывается информация при помощи блока «WRITE».

В случае возникновения ошибок ввода-вывода для блоков «OPEN», «WRITE» и «CLOSE» существуют альтернативные выходы. При возникновении ошибки произойдет не только перемещение транзакта в указанный блок, но еще и извлечение кода ошибки блоком «CLOSE» и помещение его в параметр транзакта.

Задание 7.4: изучить запись данных в конец файла.

Откройте модель «TSTAPPW.GPS». Запустите процесс моделирования командой «START 1,NP». Откройте модифицированный файл данных «TSTAPPW.TXT».

Этот пример отличается от предыдущего лишь тем, что программа записывает данные в конец файла. Вы увидите, что строка «123456» была добавлена в конец файла. У этого примера есть два главных отличия: в блоке «SEEK» указано значение 10000, а блок «WRITE» работает в режиме «Insert». В режиме «Insert» данные будут дописываться в конец файла.

Если режим «Insert» выключен, то блок «WRITE» работает в режиме «Replace». В этом случае система GPSS World заполняет все пространство вплоть до номера в блоке «SEEK» пустыми строками, а затем добавляет данные в указанную позицию.

Задание 7.5: изучить использование блока «WRITE» в режиме «Replace».

Откройте модель «TSTSKRW.GPS» и запустите процесс симуляции командой «START 1,NP». Модель внесет модификации в файл «TSTSKRW.TXT». Откройте этот файл. Обратите внимание, что вторая строка файла данных была заменена значением 123456. При этом нет необходимости изолировать блоки «CLOSE» и «OPEN», поскольку осуществляется только один прогон модели. Отличие от предыдущего примера заключается в том, что блок «WRITE» работает в режиме «Replace». Необходимо нажать **OFF**, чтобы включить режим «Replace». Также мы использовали блок «SEEK» для указания строки, которую необходимо заменить.

Задание 7.6: изучить работу режима «Replace» в случае, если в блоке «SEEK» не указан номер строки.

Откройте модель «TSTSQRW.GPS» и запустите процесс моделирования командой «START 1,NP». Откройте модифицированный файл с данными «TSTSQRW.TXT». Эта модель работает так же, как и предыдущая, за исключением того, что указатель «Current Line Position» не был переопределен, поэтому он имеет значение 1. Заменяются две строки файла с использованием текстовой строки и результата расчета простого выражения. Можно поместить в эти строки любые значения.

Задание 7.7: изучить последовательную запись нескольких строк в режиме «Replace».

Откройте модель «TSTSQR2W.GPS» и запустите процесс моделирования командой «START 1,NP». Откройте модифицированный файл с данными «TSTSQR2W.TXT». Единственное отличие данного примера заключается в том, что после каждого входа в блок «WRITE» показатель «Current Line Position» увеличивается на единицу. Следовательно, строки 1 и 2 были заменены. Также был добавлен флаг ошибки для каждого блока «WRITE», для того чтобы знать, в каком блоке произошла ошибка ввода-вывода.

Задание 7.8: изучить работу со строками.

Процедуры обработки строк позволяют совершать разнообразное количество манипуляций над данными, считанными из файла или предназначенными

для записи в файл. Вы можете форматировать заголовки отчета, добавлять данные, а также считывать и вырезать порции данных из файла.

Откройте модель «TSTCATW.GPS» и запустите процесс моделирования командой «START 1,NP». Откройте созданный в процессе моделирования файл «TSTCATW.TXT». В блоке «WRITE» данной модели была скомбинирована порция текстовых данных с данными, полученными в ходе работы модели, и с помощью процедуры «Catenate», предназначенной для обработки строк. Эта операция исключительно полезна для форматирования отчетов или вывода информации, полученной в указанные промежутки работы модели. Часть модели может быть предназначена исключительно для генерации транзактов, осуществляющих запись статистики моделирования. Значения показателей моделирования могут быть снабжены подписями для удобства понимания после того, как процесс моделирования завершен.

Задание 7.9: изучить управление данными, полученными из текстового файла.

Откройте модель «TSTSTRDW.GPS» и запустите процесс моделирования командой «START 1,NP». Откройте созданный в процессе моделирования файл «TSTSTW.TXT».

В последней модели осуществляется считывание данных из файла «TSTRDST.TXT», обработка путем разбиения на небольшие порции и последующее помещение порции считанных данных в указанное место строки данных.

Восстановите текстовые файлы из резервных копий и удалите два созданных в процессе моделирования файла.

7.3 Содержание отчета

1. Титульный лист.
2. Цель работы.
3. Краткие теоретические сведения.
4. Реализация решения задачи.
5. Выводы.
6. Список использованных источников.

7.4 Контрольные вопросы

1. Процедуры ввода-вывода, используемые в процессе моделирования.
2. Операция чтения информации во внешнем текстовом файле.
3. Операция поиска информации во внешнем текстовом файле.
4. Операция замены информации во внешнем текстовом файле.
5. Операция записи информации во внешний текстовый файл.
6. Операция создания нового внешнего текстового файла.
7. Последовательная запись нескольких строк в режиме «Replace».
8. Управление данными из внешнего текстового файла.

Список использованной литературы

1. Учебное пособие по GPSS World / пер. с англ. В. В. Девяткова ; под ред. К. В. Кудашова. – Казань : Мастер Лайн, 2002. – 272 с.
2. Шевченко, Д. Н. Имитационное моделирование на GPSS : учеб.-метод. пособие для студентов технических специальностей / Д. Н. Шевченко, И. Н. Кравченя. – Гомель : БелГУТ, 2007. – 97 с.
3. Томашевский, В. Н. Имитационное моделирование в среде GPSS / В. Н. Томашевский, Е. Г. Жданова ; под ред. Е. Г. Жданова. – М. : Бестселлер, 2003. – 416 с.

Библиотека БГУИР

Практическая работа №8

БИБЛИОТЕКА ПРОЦЕДУР ANOVA

Цель работы

Изучить возможности и применение библиотеки процедур ANOVA.

8.1 Теоретическая часть

Чтобы определить, является фактор значимым или нет, используется дисперсионный анализ ANOVA (analysis of variance), который применим только к количественным факторам. С помощью него определяются количественные отклонения наблюдений от среднего значения. Если какой-либо фактор не оказывает влияние на отклик, то он является незначимым. С другой стороны, если фактор влияет на отклик, то количественное значение сравнивают с оценкой изменчивости наблюдения, то есть со стандартной ошибкой.

Это делается для исключения эффектов, которые являются не более чем случайной флуктуацией.

Неявно в ANOVA используется аддитивная математическая модель, которая определяет компоненты изменения в наблюдениях. Ее называют статистической моделью. Самая простая статистическая модель

$$y_{ig} = \mu + e_{ig}, \quad (8.1)$$

то есть каждое i -е наблюдение представляет собой общее среднее по всем опытам μ и случайную ошибку e_{ig} . В этой модели общее среднее не изменяется от опыта к опыту в отличие от ошибки.

Статистическая модель для анализа данных экспериментов с одним фактором A имеет следующий вид:

$$y_{ig} = \mu + \alpha_i^A + e_{ig}, \quad (8.2)$$

где α_i^A – главный эффект фактора A на уровне i .

Все наблюдения на данном уровне обработки анализируются с использованием того же самого значения для α^A . Так как в этом эксперименте имеется только один фактор, число комбинаций обработки определяется числом уровней I этого фактора.

Для двух факторов общая модель факторного плана такова:

$$y_{ijg} = \mu + \alpha_i^A + \alpha_j^B + \alpha_{ij}^{AB} + e_{ig}, \quad (8.3)$$

где α_j^B – главный эффект фактора B на уровне j ; α_{ij}^{AB} – взаимодействие фактора A на уровне i и фактора B на уровне j .

Сумма эффектов двух факторов не равна сумме их отдельных эффектов из-за взаимодействия между ними. Главный эффект фактора определяет долю участия фактора в значении функции отклика во время перехода его с нижнего уровня к верхнему.

Дисперсионный анализ, основанный на статистической модели (8.3), заканчивается построением таблицы ANOVA, в которой анализируется влияние факторов *A*, *B*, взаимодействие между ними и случайные помехи наблюдения.

С помощью ANOVA проверяется гипотеза об отсутствии влияния фактора. Если справедлива гипотеза об отсутствии влияния фактора, то считается, что все наблюдения получены из одной генеральной совокупности. Для проверки гипотезы используется *F*-распределение Фишера. Критерий Фишера определяет отношение двух выборочных дисперсий. Если фактор существенно влияет на отклик, то значения *F*-распределения становятся большими, а *F*-статистика – значимой. Таким образом, большие значения *F* приводят к отбрасыванию гипотезы об отсутствии влияния фактора, то есть фактор является значимым.

Библиотечная процедура ANOVA системы GPSS World анализирует эксперименты от первого до шестого факторов, включая взаимодействия второго и третьего порядка между факторами.

На рисунке 8.1 представлена таблица ANOVA, полученная в GPSS World.

ANOVA					
Source of Variance	Sum of Squares	Degrees of Freedom	Mean Square	F	Critical Value of F (p=.05)
A	28.000	2	14.000	5.600	3.89
B	21.000	3	7.000	2.800	3.49
AB	11.000	6	1.833	0.733	3.00
Error	30.000	12	2.500		
Total	90.000	23			

Treatment Level A B	Count	Mean	Minimum	Maximum	95% C.I. (SE)
1 1	2	3.000	2.000	4.000	{ 0.564, 5.436 }
1 2	2	1.000	1.000	1.000	{ -1.436, 3.436 }

01/27/01 13:26:44 1.5811388

Рисунок 8.1 – Таблица ANOVA

Сначала рассмотрим среднюю часть таблицы. Полная сумма квадратов (Total) отделена от компонентов, связанных с эффектами факторов и их взаи-

модействиями (A , B , AB). В строке Error (ошибка) приведена остаточная сумма квадратов. Средняя сумма квадратов (Mean Square) остаточного члена используется для оценки стандартной ошибки эксперимента (в данном случае это величина 2,5).

Каждая сумма квадратов делится на число степеней свободы для уровней. Из статистических соображений степени свободы – это делитель, который должен использоваться для получения несмещенной оценки стандартной ошибки. Для наших целей достаточно представлять степени свободы как соответствующий делитель, связанный с суммой квадратов в таблице ANOVA. Система GPSS World всегда вычисляет степени свободы.

Каждый фактор и взаимодействие в статистической модели представлены отдельной строкой в верхней части таблицы ANOVA. В каждой строке указана сумма квадратов и число степеней свободы, связанные с оценкой факторов и их взаимодействий. Это основы, из которых получены другие числа. Частное от деления определяет средний квадрат ошибки, и в предпоследнем столбце таблицы выдается F -статистика для этого эффекта.

Сделаем некоторые заключения. Необходимо решить, достаточно ли большое значение F -критерия получено для объявления эффекта значимым. Пороговое значение, которое используется для сравнения, называется «критическим значением F » и помещено справа от F -статистики в той же самой строке. Если полученное значение F превышает критическое значение, то делаем заключение, что имеем дело со значимым эффектом фактора; если нет, то считаем, что эффект фактора незначимый и игнорируем любое связанное с ним изменение в наблюдениях, считая, что оно вызвано случайными помехами. Представленная таблица ANOVA показывает, что эффект фактора A значимый, а эффект фактора B и взаимодействие AB незначимы.

Иногда при выполнении эксперимента невозможно обнаружить эффект даже в том случае, если он фактически существует. Одна из задач эксперимента заключается в том, чтобы сделать это маловероятным. Из таблицы ANOVA видно, что для получения лучших результатов необходимо иметь или большую F -статистику или меньшее значение F -критерия. Желательно удалить часть суммы квадратов ошибки из-за какого-либо важного эффекта, не включенного в анализ. Если это можно сделать, то F -статистика будет больше. Для этого определяют дополнительные факторы, которые должны быть включены в эксперимент.

Для увеличения степени свободы остаточного члена можно использовать два подхода. Первый просто увеличивает число повторений в эксперименте, он обычно более дорогой, но может быть весьма эффективным. Второй касается плана эксперимента и статистической модели дисперсионного анализа. Средний квадрат ошибки – это фактически остаточный член, оставшийся после удаления других квадратов. Если можно найти приемлемый способ, который позволит большему количеству данных оставаться после удаления эффектов, то получим оценку стандартной ошибки с большим числом степеней свободы. Окончательное значение F -критерия будет уменьшаться при увеличении мощ-

ности анализа. При этом фактически игнорируются некоторые из взаимодействий.

В многофакторных экспериментах для упрощения статистической модели и уменьшения количества экспериментов игнорируют взаимодействия самого высокого порядка. Например, статистическая модель с двумя факторами без учета их взаимодействия имеет следующий вид:

$$y_{ijg} = \mu + \alpha_i^A + \alpha_j^B + e_{ig}. \quad (8.4)$$

Если действительно эффектами этих взаимодействий можно пренебречь, то это позволит использовать дополнительные степени свободы для получения лучшей оценки F -статистики. Кроме того, большее число степеней свободы означает также и меньшее значение F -критерия. Однако необходимо понимать, что удаление членов из статистической модели предполагает, что не существует взаимодействий более высоких порядков.

В GPSS World от учета взаимодействий высоких порядков можно отказаться, используя третий параметр процедуры ANOVA. Иногда целесообразней учитывать только одну случайность, когда для улучшения статистических оценок просто добавляются повторения наблюдений. Число повторений задается во втором параметре процедуры ANOVA.

Для работы процедуры ANOVA ей необходимо передать имя GPSS-матрицы с сохраненными результатами всех прогонов модели. Можно иметь несколько матриц результатов и для каждой из них выполнить эту процедуру. Любые матрицы GPSS World могут иметь максимальную размерность 6, эта величина ограничивает максимальное число анализируемых факторов.

Прежде чем начать эксперимент, необходимо инициализировать элементы матрицы в НЕОПРЕДЕЛЕННОЕ (UNSPECIFIED) состояние с помощью оператора

INITIAL MyResultMatrix,UNSPECIFIED

Этот оператор сообщает программе ANOVA, что эксперимент не был закончен.

Позиция в матрице результатов для каждого результата эксперимента определена комбинацией уровней обработки. Например, если в эксперименте рассматривается три вида рабочих и два значения их численности (минимальное и максимальное) для каждого вида, то результат моделирования для третьего вида с максимальной численностью помещается в матрицу результатов в позицию [3, 2]. Так как число повторений указывается в третьем индексе матрицы, то результат первого прогона этой комбинации уровней обработки помещается в элемент [3, 2, 1] матрицы результатов. Каждое измерение в матрице результатов анализируется библиотечной процедурой ANOVA как фактор. Размерность матрицы результатов должна быть, по крайней мере, такой же, как и число уровней обработок этого фактора или максимальное рассчитанное число

повторений. Нет никакого предела для числа уровней обработок факторов (вернее, предел накладывается только виртуальной памятью используемого компьютера).

8.2 Практическая часть

Изменение начальных значений для генераторов случайных величин может привести к большим различиям в выходных данных. Если используется моделирование для решения реальных проблем, определить, какие эффекты являются случайными, а какие – настоящими, помогают команды библиотеки ANOVA.

Это практическое занятие является кратким введением в использование статистического метода, известного как дисперсионный анализ. При использовании этого метода рассматривается библиотека процедуры ANOVA PLUS как инструмент, облегчающий первую стадию статистического анализа. Тем не менее необходимо также иметь понятие об элементарных статистических определениях и методах планирования экспериментов.

При использовании статистических методов для определения разницы между двумя альтернативами следует избегать появления посторонних неконтролируемых событий в процессе моделирования.

Будет некорректно осуществлять один прогон с дополнительными действиями, происходящими в неизвестной точке модели, а другие прогоны оставить без этих действий. Более того, не следует использовать интерактивные методы моделирования в процессе осуществления прогона, результаты которого отправятся в итоговый отчет.

Вернемся к примеру с парикмахером, который не справляется с обслуживанием потока клиентов. Управляющему парикмахерской необходимо определить, как скажется на ситуации появление парикмахера, работающего быстрее. Нужно проверить, может ли парикмахер, осуществляющий одну стрижку в среднем за 5 мин, вызвать значительное уменьшение размера очереди клиентов.

Фактором, требующим проверки, является использование первого или второго парикмахера. Парикмахеры отличаются друг от друга только лишь средним временем обработки. Таким образом, эксперимент будет состоять из двух уровней фактора – 5 и 6,8 мин, в среднем затрачиваемых на одну стрижку. Для каждого уровня фактора будет произведено три прогона, хотя обычно придется делать большее количество прогонов.

Поскольку результаты должны быть представлены в виде арифметического среднего значения показателей, мы можем доверить статистический анализ командам библиотеки ANOVA. Можно проверить, удовлетворяют ли полученные значения заданному критерию, используя среднее время ожидания клиентов в очереди.

Задание 8.1: построить матрицу «Results» специального формата, используемую процедурой ANOVA.

Поскольку есть один фактор с двумя уровнями, понадобится два измерения в матрице «Results». Последнее измерение в матрице «Results» всегда отвечает за отклики, по три отклика на каждый уровень фактора. Поэтому будем использовать матрицу 2×3.

Откройте модель «ANOVA.GPS». Обратите внимание, что основной экспериментальный параметр, расположенный в блоке «ADVANCE», представлен пользовательской переменной «Cut_Time». Это позволит изменять время, затрачиваемое на стрижку. Для генератора случайных чисел будет использоваться значение, состоящее из трех цифр. Средняя цифра будет указывать на то, какой прогон (1, 2 или 3) осуществляется в данный момент. Во втором наборе экспериментов с парикмахером, работающим быстрее, используем такие же начальные значения для генераторов случайных чисел, изменив только лишь скорость работы парикмахера.

Перед началом моделирования обратите внимание на подключаемый файл, в котором содержатся инструкции для автоматического осуществления нескольких прогонов.

```
RESULTS «Matrix» ,2,3 ; Set up for 3 replicates of two levels Cut_Time EQU 6.8
Treatment EQU 1
RMULT 411
Start 100,NP
MSAVEVALUE «Results»,1,1,QT$Barber
Clear Off
RMULT 421
Start 100,NP
MSAVEVALUE «Results»,1,2,QT$Barber
Clear Off
RMULT 431
Start 100,NP
MSAVEVALUE «Results»,1,3,QT$Barber
Clear Off
Cut_Time EQU 5
Treatment EQU 2
RMULT 411
Start 100,NP
MSAVEVALUE «Results»,2,1,QT$Barber
Clear Off
RMULT 421
Start 100,NP
MSAVEVALUE «Results»,2,2,QT$Barber
Clear Off
RMULT 431
Start 100,NP
MSAVEVALUE «Results»,2,3,QT$Barber
```


Разберемся, что произойдет после трансляции этого файла. Сначала происходит определение глобальной матрицы под названием «Results», которая будет содержать значения, предназначенные для дисперсионного анализа. Матрица состоит из двух измерений, поддерживающих два уровня факторов, с тремя значениями откликов для каждого уровня.

Команда «EQU» используется для того, чтобы установить среднее время обработки равным 6,8 мин. Затем, после осуществления трех прогонов и сбора статистики, время обработки выставляется на 5 мин для следующих трех прогонов. Также можно увидеть, что задается значение фактора перед каждой серией прогонов. Это позволяет отличить прогоны со средним временем на обработку 6,8 (первый уровень фактора) и со временем 5 (второй уровень фактора).

Обратите внимание на использование команды «CLEAR OFF» для очистки статистических показателей во избежание корреляции между прогонами. Параметр «OFF» необходим, чтобы избежать обнуления матрицы «Results». В данном примере не происходит обнуление парикмахерской в начале каждого прогона. Можно убедиться, что начальное состояние не влияет на работу модели. Этого можно добиться, периодически используя команду «RESET» в процессе моделирования. Команда «Plot» системы GPSS World полезна для определения переходных процессов, которые следует исключить из рассмотрения. Затем можно собрать показатели ожидания в очереди для каждого прогона, основываясь на том, что в начале моделирования в очереди присутствуют клиенты.

Выполним еще два дополнительных действия. Мы запускаем модель с опцией NP в качестве операнда 2 команды «START». Это приведет к подавлению генерации стандартного отчета. Управлять генерацией отчета можно в настройках объекта «Simulation». Помимо этого, мы устанавливаем начальные значения для генераторов случайных чисел. Те же самые значения используются для второй серии экспериментов, с разницей лишь во времени, которое уходит на одну стрижку. Данные действия снизят дисперсию полученных результатов.

Создайте процесс моделирования. Затем используйте подключаемый внешний файл для осуществления экспериментов. Помните, что в нашем случае система GPSS World будет искать внешний файл в той же директории, где находится файл модели. Если объект «Simulation» был сохранен в другую директорию, поиск будет осуществляться там, где располагается объект «Simulation». Подключим команды исполняемого файла. В командной строке введите «INCLUDE “ctlanova.txt”» и нажмите «ОК». Процесс моделирования начался автоматически. Можно увидеть, что 6 прогонов осуществляются всего за несколько секунд. Осмотрите матрицу «Results», которая была построена для дисперсионного анализа (команда меню «Window/Simulation Window/Matrix Window»).

В открывшемся окне (рисунок 8.2) будут отображены результаты каждого прогона.

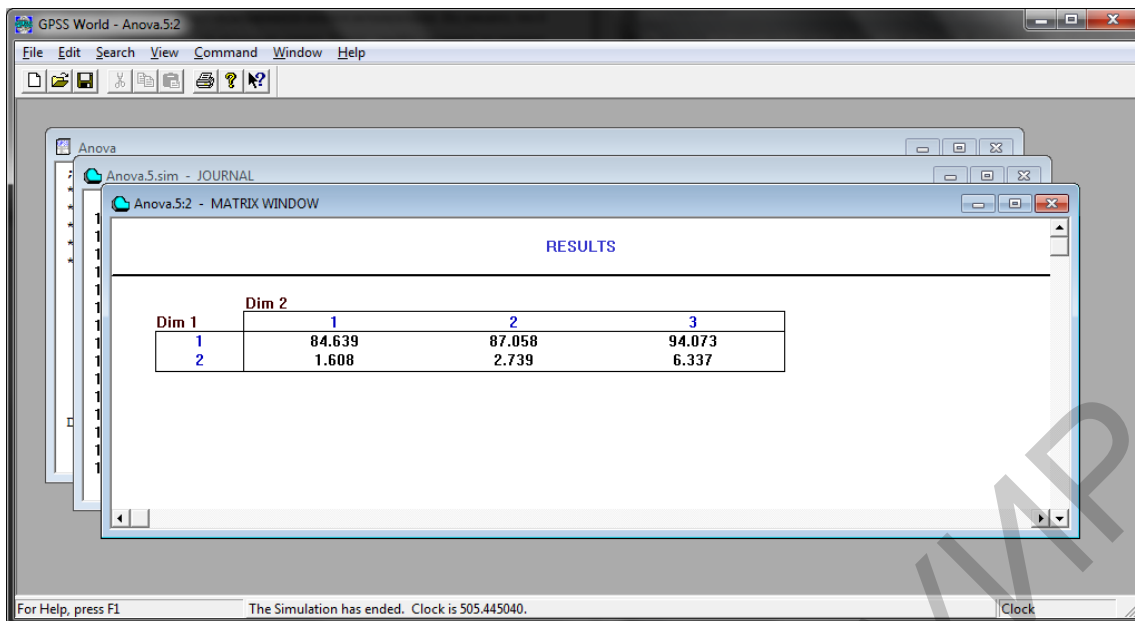


Рисунок 8.2 – Окно матрицы «Results»

Задание 8.2: провести анализ результатов.

Выполните команду, указанную в командной строке «SHOW ANOVA(Results,2,1)». Данная процедура запишет таблицу в окно «Journal» и вернет значение F , полученное в ходе анализа (рисунок 8.3).

Source of Variance	Sum of Squares	Degrees of Freedom	Mean Square	F	Critical Value of F (p=.05)
A	10844.768	1	10844.768	720.459	7.71
Error	60.210	4	15.053		
Total	10904.978	5			

Treatment Level	Count	Mean	Minimum	Maximum	95% C.I. (SE)
A					
1	3	88.590	84.640	94.073	(82.372, 94.809)
2	3	3.562	1.609	6.337	(-2.656, 9.780)

3.8797650

Рисунок 8.3 – Таблица ANOVA в окне «Journal»

Проведем интерпретацию результатов. Основной вопрос заключается в том, является ли выше уровня случайного шума значение показателя, полученного для парикмахера, работающего быстрее.

Обратите внимание на строку для фактора A в таблице ANOVA, изображенной в окне «Journal». Мы обнаружим, что критическое значение F для данного эксперимента на уровне 95 % составляет 7,71. Поскольку рассчитанное значение F значительно больше, мы можем сделать вывод, что полученные показатели для парикмахера 2 являются статистически важными.

Иногда удается выявить значимые эффекты путем увеличения длительности прогонов и количества полученных значений откликов. В данном эксперименте это необязательно, так как статистическая важность первого парикмахера доказана.

Теперь обратите внимание на последнюю строку журнала. На ней представлено значение 3,8797650, полученное в результате выполнения процедуры из библиотеки ANOVA. Это стандартная погрешность анализа, пригодная для использования программными процедурами ANOVA.

Функции ANOVA упрощают первую стадию статистического анализа. Но, возможно, потребуется более подробно исследовать полученные результаты. В этом случае полезным будет создание файла с результатами для дальнейшего применения в других программных пакетах, предназначенных для статистического анализа. Сделать это достаточно просто, используя блоки «OPEN», «CLOSE», READ, «WRITE» и «SEEK», а также процедуры обработки строк библиотеки PLUS.

8.3 Содержание отчета

1. Титульный лист.
2. Цель работы.
3. Краткие теоретические сведения.
4. Реализация решения задачи.
5. Выводы.
6. Список использованных источников.

8.4 Контрольные вопросы

1. Функциональное назначение библиотеки ANOVA.
2. Перечень команд библиотеки ANOVA.
3. Порядок построения матрицы результатов с применением процедур ANOVA.
4. Просмотр результатов выполнения процедур ANOVA.

Список использованной литературы

1. Учебное пособие по GPSS World / пер. с англ. В. В. Девяткова ; под ред. К. В. Кудашова. – Казань : Мастер Лайн, 2002. – 272 с.
2. Томашевский, В. Н. Имитационное моделирование в среде GPSS / В. Н. Томашевский, Е. Г. Жданова ; под ред. Е. Г. Жданова. – М. : Бестселлер, 2003. – 416 с.
3. Боев, В. Д. Моделирование систем. Инструментальные средства GPSS World / В. Д. Боев. – СПб. : БХВ-Петербург, 2012. – 368 с.

Практическая работа №9

ПЕРЕМЕННЫЕ. ОТЧЕТЫ. ВЕДЕНИЕ ЖУРНАЛА СЕССИЙ

Цель работы

Изучить порядок введения в модель переменных, возможности системы управления отчетами, а также порядок ведения журналов сессий системы.

9.1 Теоретическая часть

При построении модели системы иногда возникает необходимость задать сложные математические или логические соотношения между атрибутами системы. Для этой цели в программе используются переменные.

В GPSS имеется три типа переменных:

- 1) арифметические переменные;
- 2) арифметические переменные с «плавающей точкой»;
- 3) булевы переменные.

Значение арифметической переменной может использоваться как:

1) операнд блока, в этом случае значение арифметической переменной может представлять собой:

- номер объекта (устройства, МКУ, очереди и т. п.);
- номер параметра транзакта;
- значение стандартного числового атрибута;

2) операнд №1 функции;

3) операнд №1 таблицы;

4) операнд выражения другой переменной.

В выражениях арифметические переменные используют следующие арифметические операции: «+» – алгебраическое сложение; «-» – алгебраическое вычитание; «#» – алгебраическое умножение; «/» – алгебраическое деление (результатом операции является целая часть частного); «@» – деление по модулю; «^» – возведение в степень; «\» – деление без остатка (перед делением у обоих операндов отбрасываются дробные части, результатом операции является целая часть частного).

Арифметические переменные аналогичны арифметическим выражениям в алгоритмических языках. Переменная задается оператором «VARIABLE», называемым оператором описания переменной, который содержит арифметическое выражение. Формат оператора описания переменной следующий: при обращении к переменной используется обозначение V<номер переменной> или V\$<имя переменной>, то есть V – это СЧА переменной. Например, оператор описания VARIABLE определяет арифметическую переменную RSL:

RSL VARIABLE QT\$WAITL+3-FN\$DSTRB#P7

При любом обращении к переменной RSL (употребляется обозначение V\$RSL) ее значение вычисляется как текущая длина очереди «WAITL» («QT\$WAITL» – СЧА регистратора очереди) плюс константа 3 и минус произведение значения функции «DSTRB» на значение параметра 7 транзакта, обрабатываемого в данный момент. В приведенном выражении «FN» – СЧА для обращения к функции, а «P» – СЧА транзакта.

Перед выполнением любой арифметической операции определяется значение каждого элемента и выделяется его целая часть. Постоянные без знака считаются положительными числами.

В выражении арифметической переменной могут быть использованы любые СЧА, функции и другие арифметические переменные. Запрещается использование самой вычисляемой переменной, а также переменных со знаком, так как знаки в данном случае рассматриваются как арифметические операции.

Система моделирования GPSS допускает использование скобок в выражениях арифметических переменных (для группировки членов или обозначения операции умножения).

В GPSS World выражения, записанные в круглых скобках, обрабатываются вычислительной процедурой встроенного алгоритмического языка PLUS. Поэтому их можно использовать в качестве операндов блоков и операторов языка GPSS.

Арифметические переменные с плавающей точкой аналогичны рассмотренным арифметическим переменным, за исключением того, что все операции над операндами выражений переменных с плавающей точкой выполняются без преобразования операндов и промежуточных результатов в целые значения. Лишь окончательный результат вычисления преобразуется в целое число.

Формат операторов описания арифметических переменных с плавающей точкой идентичен рассмотренному ранее формату операндов описания арифметических переменных за исключением того, что в поле операции записывается слово «FVARIABLE». Правила написания операторов те же, что и для арифметических переменных. Арифметическая переменная и переменная с плавающей точкой не могут иметь одинаковые номера. Если они имеют одинаковые номера, то при вычислении используется более позднее из двух описаний.

Различие результатов, полученных при вычислении с плавающей и фиксированной точкой, можно увидеть из такого примера:

FLOAT	FVARIABLE	10#(11/3)
FIXED	VARIABLE	10#(11/3)

Значение переменной «FLOAT» равно 36, так как константа 10 умножается на 3,67 и от результата 36,7 взята целая часть. Переменная «FIXED» равна 30, так как результат промежуточной операции деления будет округлен до 3.

Для переменных с плавающей точкой не допускается операция деления по модулю.

Использование дробных констант допускается только при описании переменных с плавающей точкой.

Стандартный числовой атрибут «V\$<имя переменной>» используется для обращения к значениям как арифметических переменных, так и переменных с плавающей точкой. Способ вычисления переменной определяется оператором описания этой переменной. Принимать решения в зависимости от значений СЧА и состояния объектов GPSS с использованием только одного выражения позволяют булевы переменные.

Булевы переменные – это логические выражения, состоящие из различных СЧА и (или) других булевых переменных. В булевой переменной проверяется одно или несколько логических условий. Результатом проверки является единица (истина), если условие выполняется, и ноль (ложь) – в противном случае.

При описании булевых переменных используются три типа операторов: логические, булевы и операторы отношений.

Логические операторы связаны с такими ресурсами, как устройства, многоканальные устройства (МКУ) и логические ключи. Они используются для определения состояния данных объектов. Логические операторы, используемые в GPSS, представлены в таблице 9.1

Таблица 9.1 – Логические операторы, используемые в GPSS World

Логические операторы	Значение оператора, отражающее состояние ресурса
FVj или Fj	Равно 1, если устройство j занято или обслуживает прерывание, в противном случае – 0
FNVj	Равно 1, если устройство j не занято и не обслуживает прерывание, в противном случае – 0
Ij	Равно 1, если устройство j обслуживает прерывание, в противном случае – 0
NIj	Равно 1, если устройство j не обслуживает прерывание, иначе – 0
NUj	Равно 1, если устройство j не используется, в противном случае – 0
UJ	Равно 1, если устройство j используется, в противном случае – 0
SFJ	Равно 1, если МКУ j заполнено, иначе – 0
SNFj	Равно 1, если МКУ j не заполнено, иначе – 0
SEj	Равно 1, если МКУ j пусто, иначе – 0
SNEj	Равно 1, если МКУ j не пусто, иначе – 0
SVj	Равно 1, если МКУ j находится в состоянии использования, в противном случае – 0
SNVJ	Равно 1, если МКУ j не используется, в противном случае – 0
LRj	Равно 1, если логический ключ j выключен, иначе – 0
LSj	Равно 1, если логический ключ j включен, иначе – 0

Операторы отношения выполняют алгебраическое сравнение операндов. Операндами могут быть константы или стандартные числовые атрибуты. Все операторы отношений записываются в кавычках:

"G" (Greater) – больше;

"L" (Less) – меньше;

"E" (Equal) – равно;

"NE" (Not Equal) – не равно;

"LE" (Less than or Equal) – меньше или равно;

"GE" (Greater than or Equal) – больше или равно.

Есть два булевых оператора: "OR" – оператор «или» – и "AND" – оператор «и». Оператор «или» проверяет, выполняется ли хотя бы одно из проверяемых условий. Оператор «и» требует выполнения обоих условий.

Система управления отчетами обладает определенным количеством уникальных возможностей. Прежде всего необходимо знать, что отчеты никогда не удаляются и всегда создаются по окончании моделирования, если не используется оператор подавления генерации отчетов «NP» команды «START». Каждому отчету присваивается имя, указывающее, для какого прогона он был сгенерирован (к примеру, «SAMPLE2.1.1», «SAMPLE2.1.2», и т. д.). По умолчанию имя отчета состоит из названия модели, последовательно приведенных номеров объекта «Simulation», созданного на основе данной модели, и номера отчета для данной версии модели. Эти отчеты будут доступны даже после того, как завершён сеанс работы с системой GPSS World.

Если просто пользоваться командой «REPORT», то невозможно будет задавать имена своим отчетам. Поскольку отчеты являются отформатированными файлами, следует скопировать текст отчета в буфер обмена и вставить в текстовый файл, если требуется присвоить отчету иное имя. Файлы отчетов имеют расширение «.GPR». Система нумерации предназначена для сортировки отчетов. Обнулить номер прогона можно в настройках.

9.2 Практическая часть

Задание 9.1: получить отчеты по результату выполнения моделирования.

Откройте модель «Sample9.GPS». Перейдите в настройки модели командой меню «Edit/Settings». Настройка «Child Object Sequence Number» устанавливает номер файла для объекта «Simulation». Если вы сохраняли объекты «Simulation» для данной модели, этот номер будет больше единицы. Можно изменить его на любое значение, которое будет использоваться при именовании отчета.

При запуске нового сеанса работы с системой GPSS World отчеты, генерируемые во время этой сессии с использованием стандартного имени (имя объекта «Simulation» + номер прогона + номер отчета), будут нумероваться, начиная с первого доступного числа, следующего за наибольшим числом.

Команда «REPORT» вызывается из диалогового окна «Custom Command» для интерактивной генерации отчета в любой момент времени. Перейдите на закладку «Reports» окна настроек модели. Отобразится информация, которая будет по умолчанию включена в отчет, генерируемый путем запуска модели. Закройте окно настроек и запустите процесс моделирования командой «START 1». По завершении моделирования появится окно «Report». Если желаете, можете увеличить размеры окна и более внимательно изучить отчеты.

Файлы отчета можно сохранять и открывать для просмотра в любое время. Следует отметить также, что создание отчета возможно в любой момент выполнения процесса моделирования. Для этого в командной строке нужно ввести команду «REPORT».

Журнал системы GPSS World представляет собой хронологическую последовательность событий, протоколируемую в окне «Journal/Simulation». Когда сохраняется объект «Simulation», вместе с ним сохраняются все имеющиеся в окне «Journal/Simulation» сообщения. Журнал сессии собирает трассировочные сообщения, сообщения об ошибках и все интерактивные команды

Задание 9.2: изучить журнал сессии системы GPSS World.

Откройте модель «Sample5.GPS» и запустите симуляцию. Поместите универсальное условие «STOP» в модель при помощи командной строки для того, чтобы следить за ходом моделирования при помощи команды «STEP». Сразу после этого запустите моделирование командой «START 1». Моделирование сразу же после запуска будет приостановлено. Используйте команду «STEP» в командной строке несколько раз. Обратите внимание на окно «Journal» (рисунок 9.1). Данное окно является полезным при отслеживании ошибок.

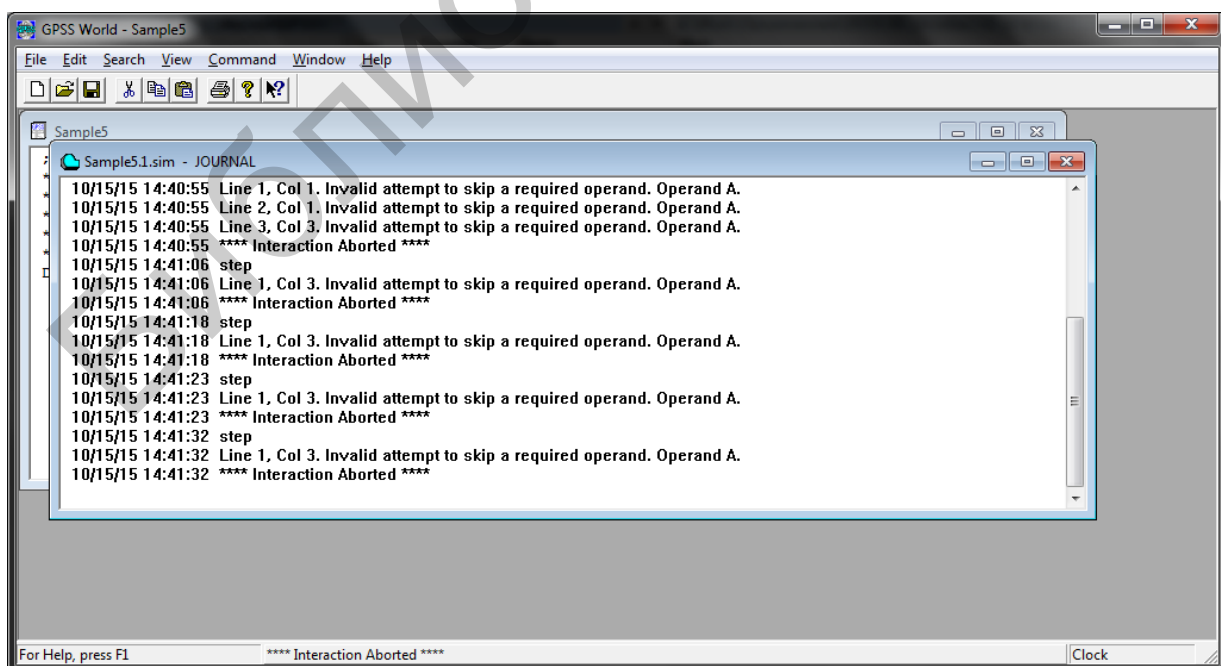


Рисунок 9.1 – Окно «Journal» с выведенными сообщениями об ошибках

Чтобы быстро закрыть программу и при этом избежать появления окон подтверждения или отрицания сохранения изменений, можно использовать в командной строке команду «EXIT (1)».

9.3 Содержание отчета

1. Титульный лист.
2. Цель работы.
3. Краткие теоретические сведения.
4. Реализация решения задачи.
5. Выводы.
6. Список использованных источников.

9.4 Контрольные вопросы

1. Типы переменных в GPSS World.
2. Арифметические операции, используемые в арифметических переменных.
3. Порядок обращения к арифметическим переменным.
4. Особенности арифметических переменных с плавающей точкой.
5. Логические операторы, используемые в логических переменных.
6. Функции и назначение системы управления отчетами в GPSS World.
7. Порядок получения отчета по результату выполнения моделирования.
8. Просмотр и анализ журнала сессии системы GPSS World.

Список использованной литературы

1. Кудрявцев, Е. М. GPSS World. Основы имитационного моделирования различных систем / Е. М. Кудрявцев. – М. : ДМК Пресс, 2004. – 320 с.
2. Шевченко, Д. Н. Имитационное моделирование на GPSS : учеб.-метод. пособие для студентов технических специальностей / Д. Н. Шевченко, И. Н. Кравченя. – Гомель : БелГУТ, 2007. – 97 с.
3. Учебное пособие по GPSS World / пер. с англ. В. В. Девяткова ; под ред. К. В. Кудашова. – Казань : Мастер Лайн, 2002. – 272 с.

Практическая работа №10

НЕПРЕРЫВНОЕ МОДЕЛИРОВАНИЕ

Цель работы

Изучить принципы и порядок непрерывного моделирования в системе GPSS World.

10.1 Теоретическая часть

Если есть возможность описать изменения переменных в терминах их производных во времени, то система GPSS World будет постоянно обновлять их значение. Используя команду «INTEGRATE», вы можете внедрить систему обыкновенных дифференциальных уравнений различного порядка в вашу модель. Процесс непрерывного моделирования является более медленным по сравнению с другими методами, но он достаточно распространен и легко справляется с нелинейными системами. Основными инструментами для непрерывного моделирования являются команда «INTEGRATE», позволяющая задавать дифференциальные уравнения и пороги, и блок «INTEGRATION», который может включать и выключать интегрирование.

Процесс задания непрерывной переменной достаточно прост. Все, что необходимо сделать, это инициализировать пользовательскую переменную и утвердить команду «INTEGRATE». Это дает системе GPSS World стартовое значение и всю необходимую информацию для того, чтобы поддерживать состояние этой переменной.

Команда INTEGRATE устанавливает интегрирование и пороги непрерывной переменной. Формат команды

NAME INTEGRATE A,[B],C,[D],[E]

Здесь NAME – имя переменной пользователя. Значение операндов:

A – производная: имя, число, строка, выражение в скобках или СЧА.

B – первое пороговое значение (порог): имя, число, строка, выражение в скобках или СЧА.

C – первая метка: имя, положительное целое число, выражение в скобках или СЧА.

D – второе пороговое значение (порог): имя, число, строка, выражение в скобках или СЧА.

E – вторая метка: имя, положительное целое число, выражение в скобках или СЧА.

Примечание – Операнды **B** и **C**, **D** и **E** или используются парами, или не используются.

Когда объект моделирования принимает команду «INTEGRATE», он размещает ее в конце очереди команд.

При выполнении команд «INTEGRATE» переменная пользователя автоматически интегрируется с течением системного времени. Для интегрирования используется модифицированный метод Рунге – Кутты – Фельберга пятого порядка с переменным размером шага.

Всем переменным пользователя, применяемым в интегрировании, до выполнения моделирования должны быть даны начальные значения. Это можно сделать с помощью операторов EQU или операторов присваивания в PLUS-процедуре.

Операнд А команды «INTEGRATE» используется для производной переменной пользователя по времени. Он может быть очень простым или весьма сложным. В последнем случае можно определить процедуру PLUS и разместить команду вызова в вводимом выражении, используемом для операнда А.

Каждая команда «INTEGRATE» может иметь ноль, один или два числовых порога. Операнды В и С могут использоваться для определения первого порога, операнды D и E – для определения второго порога. В любом случае первый операнд пары определяет значение порога, второй – указывает метку блока, принимающего генерируемые транзакты.

Если в течение интегрирования значение интегрируемой переменной достигает значения порога (с любого направления), создается новый транзакт. Этому транзакту устанавливается приоритет 0, и он переходит в блок, связанный с этим порогом в команде «INTEGRATE». Время входа транзакта в модель определяется линейной интерполяцией. Для увеличения точности при приближении к порогу шаг интегрирования уменьшается. Транзакты, генерируемые при пересечении порога, могут использоваться для того, чтобы изменить значение порога.

Оба порога равноправны, нет необходимости определять один как нижний, а второй как верхний. Новый транзакт генерируется при пересечении любого порога с любой стороны. Если направление пересечения важно для функционирования модели, то нужно следить за значением переменной либо проверить направление пересечения.

Интегрирование начинается автоматически в активном или «разрешенном» состоянии. Во время выполнения моделирования можно включать или выключать интегрирование, используя один или несколько блоков «INTEGRATION».

Чтобы настроить автоматическое интегрирование переменной, необходимо:

- 1) объявить командой «INTEGRATE» интегрируемую переменную;
- 2) присвоить начальные значения используемым переменным.

Выражения в командах «INTEGRATE» используются как производные по времени. Начальные значения непрерывных переменных установлены командами EQU.

При использовании интегрирования моделирование выполняется поочередно в непрерывных и дискретных стадиях. В процессе имитации при планировании событий выполняется дискретная стадия моделирования. В этом слу-

чае часы модельного времени продвигаются от события к событию. Между моментами наступления событий выполняется непрерывная стадия моделирования, в течение которой интегрирование осуществляется с маленькими приращениями времени, называемыми «мини-шагами». График переменной интегрирования выводит промежуточные значения в концах «мини-шагов». Когда происходит пересечение порога, генерируется транзакт и моделирование переходит в дискретную стадию. Непрерывная и дискретная стадии могут взаимодействовать. Например, интегрируемым переменным пользователя в дискретной стадии могут быть назначены новые значения. Это можно сделать, используя операторы EQU или вызывая предназначенные для этого PLUS-процедуры.

10.2 Практическая часть

Задание 10.1: инициализировать пользовательскую переменную и утвердить команду «INTEGRATE».

Создайте новую модель. Предположим, что есть производственная система, производящая два устройства в секунду. После того как произведен целый грузовик этих устройств, оператор направляет грузовик, нагруженный устройствами, в один из центров перераспределения. Начнем моделирование с объемом инвентаря, равным нулю. Для того чтобы задать начальное условие, введите строку

Inventory EQU 0

На следующей строке, для того чтобы определить производную, введите

Inventory «INTEGRATE» 2,Full_Load,Get_Truck

Как правило, потребуется процедура PLUS для определения производной. Затем воспользуйтесь вызовом родительской процедуры из операнда 1 команды «INTEGRATE». Обратите внимание, что транзакт будет создан в тот момент, когда будет превышен порог. Используем данное условие для завершения моделирования. Введите следующую строку:

Full_Load EQU 50

Теперь после создания 50 устройств транзакт будет отправлен в блок, помеченный надписью «Get_Truck». Вот и все, что требуется для организации непрерывной части модели. Для организации дискретной части добавьте следующие выражения в модель

GENERATE 100

TERMINATE

Get_truck «TERMINATE» 1

Дискретная часть необходима для того, чтобы начать и закончить процесс моделирования. В данном случае воспользуемся пороговым значением для уменьшения счетчика «Termination Count».

Перед тем как начать моделирование, откройте график пользовательской переменной «Inventory» командой меню «Window/Simulation Window/Plot Window». В окне «Edit Plot» введите «Inventory» во все три поля: «Label», «Expression» и «Title». Остальные значения оставьте без изменений. Нажмите кнопки «Plot» и «Memorize».

Отметим на графике пороговое значение. В окне «Edit Plot» замените значение поля «Label» на «Threshold». В поле «Expression» введите «Full_Load». Остальные значения оставьте без изменений. Нажмите кнопки «Plot» и «Memorize».

Запустите процесс моделирования командой «START 1». Окно «Plots» будет выглядеть, как показано на рисунке 10.1.

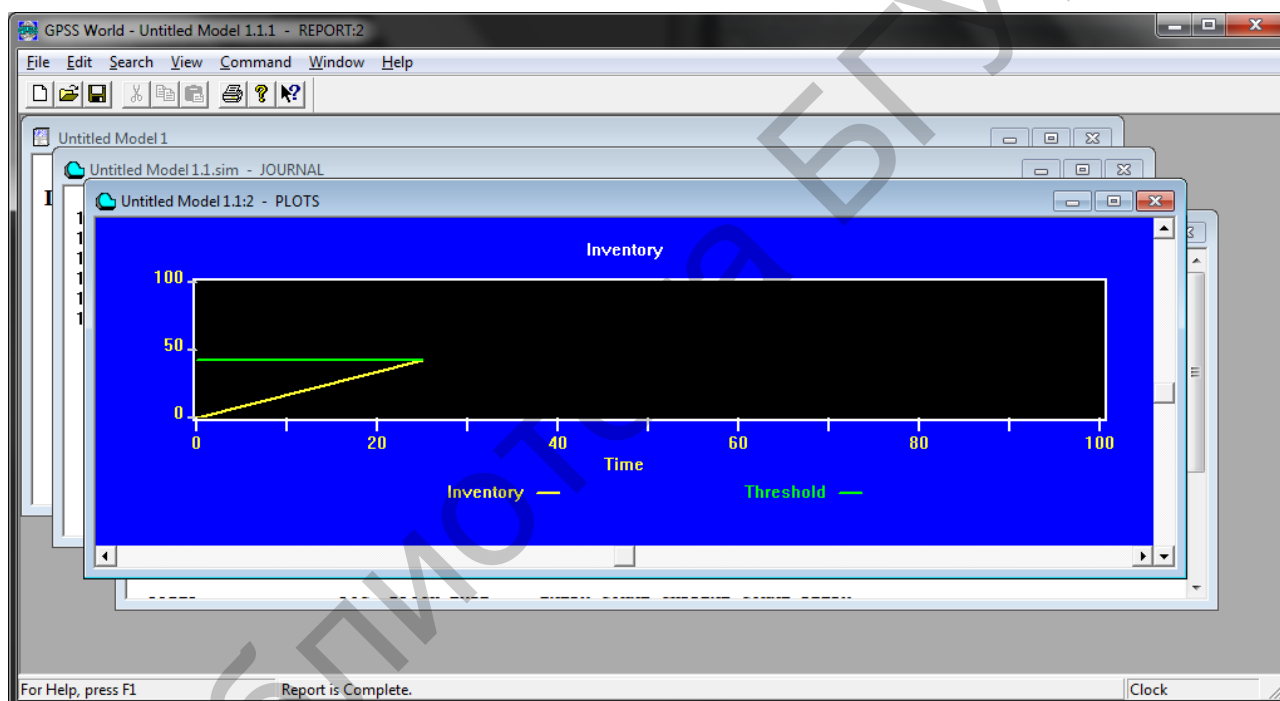


Рисунок 10.1 – Окно «Plots»

Видно, что значение параметра «Inventory» очень быстро достигнет 50, что приведет к завершению моделирования. Судя по стандартному отчету, выполнение работы заняло 25 мин модельного времени. Таким образом, вы можете определить, что в минуту осуществлялось производство двух устройств. Порог был достигнут через 25 мин, транзакт был создан и отправлен в блок, помеченный надписью «Get_Truck». Блок «INTEGRATION» доступен для контроля над уровнем транзактов.

Задание 10.2: рассмотрим систему дифференциальных уравнений (10.1).

$$\begin{aligned}x' &= y, \\ y' &= -x.\end{aligned}\tag{10.1}$$

В начальный момент времени $t = 0$ значения переменных равняются:

$$\begin{aligned}x_0 &= 1, \\ y_0 &= 0.\end{aligned}\tag{10.2}$$

Решение данной системы уравнений следующее:

$$\begin{aligned}x &= \sin(t), \\ y &= -\cos(t).\end{aligned}\tag{10.3}$$

В тексте модели используем знак подчеркивания (« $x_$ » и « $y_$ ») для того, чтобы убедиться, что переменные не будут пересекаться с ключевыми словами системы GPSS World. Рассмотрим, что произойдет при моделировании такой системы.

Откройте модель «Sample13.gps». Запустите симуляцию. Установите график для пользовательских переменных, чтобы следить за динамикой изменения модели. Откройте окно графика (команда меню «Window/Simulation Window/Plot Window»). В окне «Edit Plot» в поле «Label» введите « $x_$ », в поле «Expression» – « $x_$ », в поле «Title» – «Sine/Cosine Sample», в поле «Time Range» – значение 120. В поле значения «Min» по оси Y введите значение -2 , в поле значения «Max» введите 2. Нажмите кнопки «Plot» и «Memorize».

Теперь добавьте еще одну переменную к графику. В окне «Edit Plot», замените значения в полях «Label» и «Expression», введите « $y_$ » в оба поля, другие поля оставьте без изменений. Нажмите кнопку «Plot» и «Memorize». И нажмите «OK».

Запустите моделирование командой «START 1». График примет вид, показанный на рисунке 10.2.

Как видите, из обеих переменных разворачивается синусоидальная волна. Когда моделирование будет завершено, изучите стандартный отчет. Сравните переменные « $x_$ » и « $y_$ » со значением, которое было рассчитано аналитическим методом.

Большинство реальных вычислительных систем являются слишком громоздкими для этих целей. Обычно приходится завершать численное интегрирование при помощи одной или двух команд «INTEGRATE», что занимает много времени.

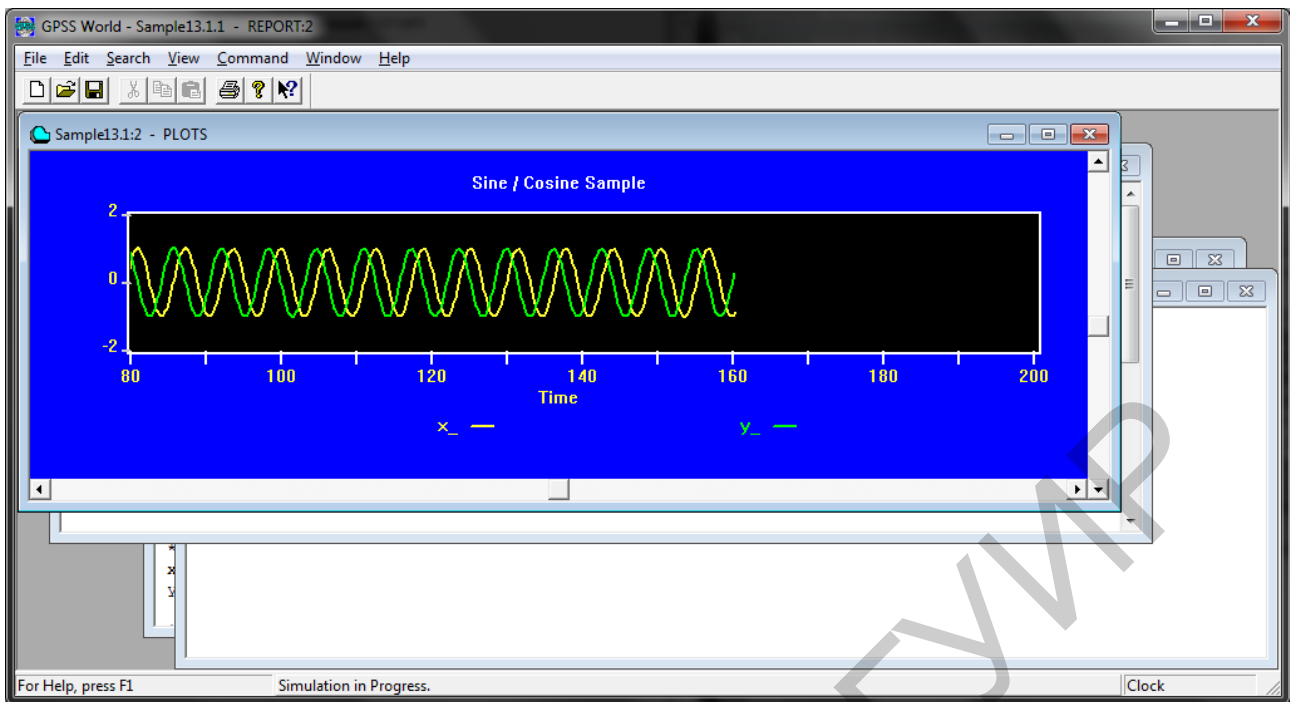


Рисунок 10.2 – График синусоидальной волны

Остановите процесс моделирования и закройте все активные окна.

10.3 Содержание отчета

1. Титульный лист.
2. Цель работы.
3. Краткие теоретические сведения.
4. Реализация решения задачи.
5. Выводы.
6. Список использованных источников.

10.4 Контрольные вопросы

1. Порядок инициализации пользовательской переменной.
2. Использование и порядок утверждения команды «INTEGRATE».
3. Вывод решения задачи командой «INTEGRATE» на график.
4. Порядок решение систем уравнений в GPSS World.
5. Вывод решения системы уравнений на график в окне «Plot».

Список использованной литературы

1. Томашевский, В. Н. Имитационное моделирование в среде GPSS / В. Н. Томашевский, Е. Г. Жданова ; под ред. Е. Г. Жданова. – М. : Бестселлер, 2003. – 416 с.
2. Боев, В. Д. Моделирование систем. Инструментальные средства GPSS World / В. Д. Боев. – СПб. : БХВ-Петербург, 2012. – 368 с.

Практическая работа №11

ЯЗЫК ПРОГРАММИРОВАНИЯ ДЛЯ МОДЕЛИРОВАНИЯ PLUS

Цель работы

Изучить возможности и порядок применения языка программирования для моделирования PLUS.

11.1 Теоретическая часть

Язык GPSS можно отнести к языкам высокого уровня, но с довольно слабыми алгоритмическими возможностями. Для устранения этого недостатка в систему GPSS World добавлен язык низкого уровня PLUS. Выражения, процедуры и эксперименты PLUS можно использовать в GPSS-моделях.

Рассмотрим основные элементы языка PLUS.

Алфавит языка PLUS (GPSS World) содержит алфавитно-цифровые и специальные символы. Для задания имен используются алфавитно-цифровые символы (прописные буквы A–Z, строчные буквы a–z, цифры 0–9 и знак подчеркивания «_»). Для обозначения операторов и пунктуации используются специальные символы («#», «*», «&», «+», «-», «-», «/», «\», «,», «;»). В комментариях допускается использование символов русского алфавита «А–Я».

Имена – это созданные пользователем последовательности символов, используемые для обозначения объектов, переменных и процедур. Имя должно начинаться с буквы, в нем можно использовать от 1 до 250 алфавитно-цифровых символов. При этом имена не должны совпадать с ключевыми словами GPSS и СЧА. Следует отметить, что GPSS World не различает регистр алфавита.

PLUS-выражение – это комбинация одного или нескольких элементов, называемых факторами. Выражения строятся с использованием операторов и вызовов процедур, обрабатывающих факторы, могут использоваться в PLUS-процедурах и операндах операторов GPSS. Если выражение используется в операндах GPSS-блоков, то оно должно записываться в круглых скобках.

Переменные пользователя, элементы матриц, ячейки, параметры транзактов могут иметь значения различных типов данных. В GPSS World используются три основных типа данных:

– целочисленный (Integer) – это 32-битовые двоичные числа; если при выполнении арифметической операции происходит переполнение целочисленного значения, производится автоматическое преобразование его в вещественное;

– вещественный (Real) – это числа с плавающей точкой двойной точности; для представления мантиссы используется 15 десятичных разрядов, для порядка – значения в интервале от –306 до 306;

– строковый (String) – это последовательность символов ASCII; строки не ограничены в размерах (максимальная их длина определяется параметрами настройки «Edit/Setting.../Simulation»).

Выражения могут быть вычисляемыми, численно вычисляемыми, вычисляемыми в виде строки.

Как уже известно, выражения являются блоками, из которых складывается модель. Эти выражения можно интерактивно отправить в работающую модель.

Тем не менее «PROCEDURE» и «EXPERIMENT» – это единственные команды языка PLUS, являющиеся выражениями модели. Все остальные команды языка PLUS могут существовать только внутри тела процедуры. Некоторые выражения PLUS слегка отличаются: они могут существовать как внутри команд PLUS, так и в качестве операндов в выражениях системы GPSS, заключенных в круглые скобки. Существует несколько различных типов выражений PLUS, представленных в таблице 11.1.

Таблица 11.1 – Типы выражений PLUS

Выражение	Описание
EXPERIMENT	Определяет эксперимент PLUS
PROCEDURE	Определяет процедуру PLUS
«TEMPORARY»	Определяет и ограничивает область пользовательской переменной
«TEMPORARY» «Matrix»	Определяет и ограничивает область сущности «Matrix»
BEGIN / END	Составное выражение. Создает блок выражений PLUS
ASSIGNMENT	Задаёт значение именованной переменной или элементу матрицы
Procedure Call	Вызывает библиотечную процедуру
IF / THEN	Проверяет значение выражения и возвращает логический результат
IF / THEN / ELSE	Проверяет значение выражения и возвращает результат
WHILE / DO	Несколько раз выполняет указанные действия
GOTO	Переходит на новую позицию в рамках процедуры
RETURN	Заканчивает обработку и опционально возвращает результат

Процедуры языка PLUS могут записываться в любом месте модели (кроме тела другой процедуры). Остальные операторы PLUS могут появляться только внутри оператора «PROCEDURE».

Различают процедуры пользовательские и встроенные (библиотечные). Пользовательские процедуры обычно используются для изменения значений глобальных переменных и поименованных величин или для вычисления выражения и выдачи результата. В первом случае результат вычисления не требуется, во втором – требуется, а также обязательно наличие оператора возврата «RETURN». Результат, выдаваемый PLUS-процедурой, используется в операндах или других PLUS-выражениях. Если в операторе «RETURN» не задано какое-либо выражение или в процедуре отсутствует оператор возврата «RETURN», то возвращается значение 0.

Для определения PLUS-процедуры необходимо поместить оператор «PROCEDURE» в файл модели и выполнить транслирование оператора вместе с моделью или передать оператор «PROCEDURE» процессу моделирования в интерактивном режиме. После этого уже определенную PLUS-процедуру можно вызывать при вычислении выражений или в PLUS-операторах присваивания.

Если процедура используется более чем в одной модели, то ее можно сохранить в исходном файле, называемом библиотекой процедур пользователя. Командой «INCLUDE» можно включать эту библиотеку в каждую модель, в которой будет использоваться данная процедура

11.2 Практическая часть

В этом практическом задании полный набор команд языка PLUS использоваться не будет. Рассмотрим лишь определение и интерактивное переопределение процедур, а также некоторые особенности интерактивной отладки процедур PLUS.

Задание 11.1: выполнить определение процедуры установки значения пользовательской переменной.

Создайте новую модель. Введите в модели следующую строку:

```
PROCEDURE SetPop(level) Foxes = level ;
```

Несмотря на то что обычно процедуры PLUS являются более сложным, это все, что нужно для определения данной процедуры. В ней даже не определяется возвращаемое значение, поэтому по умолчанию процедура будет возвращать значение 0. Возвращаемые значения полезны, когда процедура PLUS передает информацию вызывающей процедуре. В этом примере не будем пользоваться подобными возможностями языка. Сохраните модель, задав в качестве имени модели «Fox». Запустите процесс моделирования.

Мы включили процедуру «SetPop» в модель, а затем создали объект «Simulation». Любые интерактивные вызовы процедуры «SetPop» будут использовать это определение. Откройте команду меню «Command/SHOW» и введите «SHOW SetPop(100)». Обратите внимание, что процедура вернула значение 0.

Выведите в журнал объем популяции. Откройте команду меню «Command/SHOW» и введите «SHOW Foxes».

Пользовательская переменная Foxes была обновлена после того, как была выполнена команда «SHOW» первый раз. Теперь интерактивно переопределим процедуру «SetPop». Для этого в командной строке введите «Procedure SetPop(level) Foxes = level# 10». Теперь при вызове процедуры «SetPop» значение переменной популяции будет приведено к значению аргумента, умноженного на 10. Введите команду «SHOW SetPop (100)» в меню «Command/SHOW» еще раз.

Обратите внимание, что процедура «SetPop» по-прежнему возвращает значение 0. Мы могли бы вставить команду «INSERT» в процедуру. Более того, процедура «SetPop» может представлять собой не что иное, как единственное выражение «INSERT». Просмотрите объем популяции командой «SHOW Foxes». Как видите, несмотря на то что модель уже была транслирована в объект «Simulation», можно интерактивно переопределить процедуру «SetPop». Это может быть полезно при отладке собственных процедур PLUS.

Задание 11.2: выполнить отладку процедуры PLUS.

Во время отладки процедур PLUS первым делом следует включить опцию PLUS «TRACE» на закладке «Reports» окна настроек. Это приведет к тому, что каждый вызов процедуры будет сопровождаться записью в окне «Journal». Эти отладочные сообщения предназначены для того, чтобы продемонстрировать глубину процесса обработки. Помимо этого, они выводят список обрабатываемых при вызове процедуры аргументов. Организация трассировки внутри процедур также не представляет никаких сложностей. Для этого достаточно лишь создать пустую процедуру PLUS, которая абсолютно ничего не делает. Затем можно дополнить вашу основную процедуру обращениями к этой отладочной процедуре. Теперь при включенной опции PLUS «TRACE» будут отображаться трассировочные сообщения, отображающие состояние переменных.

Создайте новую модель. Добавьте следующие процедуры PLUS:

```
PROCEDURE RabbitRate() BEGIN  
«TEMPORARY» BirthRate,DeathRate,TotRate;  
BirthRate = 100;  
Dummy(BirthRate);  
DeathRate = 80 ;  
RETURN TotRate;  
END;  
PROCEDURE Dummy( argument ) RETURN;
```

Здесь была определена пустая процедура, предназначенная для трассировки значения нашей локальной переменной. Опция «PLUS TRACE» должна быть включена, чтобы эта функция заработала. Включите ее. Откройте настройки модели («Edit/Settings»), выберите закладку «Reports» и установите флажок «PLUS TRACE».

Сохраните модель и укажите в качестве имени модели «Rates». Запустите процесс моделирования. Если была совершена опечатка, воспользуйтесь пунктом меню «Search/Next Error» окна «Model» для ее поиска и исправления. Вызовите процедуру «SHOW RabbitRate ()».

Получим сообщения об ошибке, рассмотрим их каждую строку. Поместите курсор на строку, где указана ошибка. Убедитесь, что окно «Model» является активным. Выберите «Search/Go To Line» и введите 5. В этой строке происходит попытка использовать временную пользовательскую переменную, которая не была проинициализирована. Этой переменной является «TotRate», так как она не была объявлена в строке, предшествующей той, в которой произошла ошибка. Введите

TotRate = BirthRate – DeathRate ;

Dummy(TotRate);

Произведите ретрансляцию модели. Переопределите процедуру, оставив текущую имитационную среду без изменений. Осуществите это при помощи интерактивной команды «INCLUDE». Возможно, данный подход выглядит несколько громоздким, но это делается для того, чтобы продемонстрировать возможности команды «INCLUDE».

Скопируйте текст из окна «Model» в буфер обмена Windows. Затем создайте текстовый файл. Для этого выберите «File/New» и выделите тип создаваемого объекта «Text File». Затем скопируйте содержимое буфера обмена в новый текстовый файл и сохраните его как «RateDef.txt».

Эта последовательность действий создает текстовый файл, который может быть использован как внешний файл для подключения. Все файлы для подключения должны состоять из простого текста. Теперь переопределите процедуру «RabbitRate()» в существующем объекте «Simulation». Это действие подразумевает использование команды «INCLUDE», а не «Retranslate». В диалоговом окне командной строки введите команду «INCLUDE "RateDef.txt"». Теперь осуществите вызов процедуры «SHOW RabbitRate()».

Выполненные действия исправили возникшую проблему. Интерактивное переопределение процедуры оказалось достаточно несложным. Удобно пользоваться интерактивным переопределением при внесении мелких изменений в процедуры PLUS в процессе их отладки. Сначала следует «закомментировать» выражения «TEMPORARY», если ни одна пользовательская переменная или матрица не пересекается глобальными сущностями модели. Возникновение подобного конфликта может произойти при изменении глобальных переменных.

Наилучшим вариантом будет поддержание уникальности имен сущностей «TEMPORARY». Поместите «/*» в начале каждого выражения «TEMPORARY» или «TEMPORARY Matrix» и «*/» в конце. Это приведет к тому, что промежуточные результаты будут доступны после выполнения процедуры. Можно использовать дополнительные команды «ASSIGNMENT» и «RETURN» для остановки процедуры в желаемой точке.

Всегда имеется возможность добавить обращение к пустой процедуре для отслеживания изменения состояния переменных. При этом опция «PLUS TRACE» должна быть включена в настройках модели, если требуется осуществлять протоколирование трассировочных сообщений в окне «Journal».

11.3 Содержание отчета

1. Титульный лист.
2. Цель работы.
3. Краткие теоретические сведения.
4. Реализация решения задачи.
5. Выводы.
6. Список использованных источников.

11.4 Контрольные вопросы

1. Типы выражений PLUS, применяемые в GPSS World.
2. Применение процедур PLUS в процессе моделирования.
3. Вывод результатов работы процедур PLUS в журнал событий системы.
4. Порядок проведения отладки процедур PLUS.

Список использованной литературы

1. Томашевский, В. Н. Имитационное моделирование в среде GPSS / В. Н. Томашевский, Е. Г. Жданова ; под ред. Е. Г. Жданова. – М. : Бестселлер, 2003. – 416 с.
2. Боев, В. Д. Моделирование систем. Инструментальные средства GPSS World / В. Д. Боев. – СПб. : БХВ-Петербург, 2012. – 368 с.
3. Учебное пособие по GPSS World / пер. с англ. В. В. Девяткова ; под ред. К. В. Кудашова. – Казань : Мастер Лайн, 2002. – 272 с.

Практическая работа №12

ИСПОЛЬЗОВАНИЕ ЯЗЫКА PLUS

Цель работы

Ознакомиться с примерами и освоить приемы применения языка PLUS для моделирования технических систем.

12.1 Теоретическая часть

Язык PLUS представляет собой мощный язык программирования, возможности которого можно использовать практически в любой точке модели GPSS World. Большинство операндов в выражениях GPSS могут использовать команды языка PLUS, заключенные в круглые скобки. Конечно, выражения PLUS могут представлять собой набор атрибутов SNA и численных значений. Но они также могут содержать объявленные процедуры. Возможно, достаточно лишь внедрить специальные расчеты в обработку выражения, но целесообразнее воспользоваться процедурами PLUS в связи с их возможностью изменять постоянные значения.

Процедуры PLUS должны быть определены до их использования. Определение происходит внутри команды «PROCEDURE», а использование – путем помещения вызовов процедуры в выражения.

Каждая процедура PLUS начинается с имени процедуры и формального списка аргументов. Под списком подразумевается ноль или большее количество имен, определяющих, как будут использоваться аргументы, передаваемые при вызове процедуры.

Хотя каждый аргумент, передаваемый при вызове процедуры, называется «актуальным параметром», на самом деле он может представлять собой сложное выражение, обрабатываемое в момент вызова процедуры. В таком случае в качестве «формального аргумента», определенного при объявлении процедуры, используется результат обработки данного выражения. Итоговое значение каждого аргумента определяется его позицией в списке параметров.

12.2 Практическая часть

Во всех примерах зарезервированные имена PLUS представлены в верхнем регистре, что является необязательным.

Задание 12.1: изучить применение процедур PLUS на примере модели «PLUSANS.GPS».

Откройте модель «PLUSANS.GPS». В данном случае вызов процедуры PLUS происходит при обработке операнда 2 блока «SAVEVALUE». Эта же процедура используется в двух различных точках модели. Действия, выполняемые процедурой, определяются в зависимости от значения аргумента, иденти-

фицирующего тип операции. Когда тип, передаваемый в качестве третьего аргумента, равен 1, выполняется первая часть условия «IF...THEN...ELSE», если тип равен 2, тогда выполняется ветвь «ELSE». В обоих случаях внутри процедуры создается переменная «TEMPORARY», в которую записываются результаты расчетов. Расчет можно осуществлять непосредственно внутри команды «RETURN», тогда наличие переменной «TEMPORARY» будет необязательным. В данном случае она используется для большей ясности. Переменная «TEMPORARY» сохраняется до тех пор, пока процедура не возвратит значение. При вызове процедуры в качестве параметров 1 и 2 передаются значения 500 и 1234, используемые для расчета переменной «SAVEVALUE Ans». Третьим аргументом передается параметр тип. Поскольку $Arg3 = 1$, исполняются две команды внутри блока «IF...THEN». Выражение «RETURN» использует функцию для обработки строк «CATENATE» с целью объединения двух значений в одну строку, которая затем помещается в «SAVEVALUE Ans». После этого выполнение процедуры завершается. Команды «BEGIN» и «END», окружающие команды «IF» и «ELSE», необходимы для того, чтобы убедиться в выполнении обоих действий.

Обратите внимание, что все определения процедуры заключены в блоки «BEGIN...END». Каждое выражение завершается точкой с запятой. Даже выражения «IF...THEN...ELSE» должны завершаться с помощью точки с запятой.

Чтобы запустить модель «PLUSANS.GPS» после трансляции, используйте команду «START 2». После того как работа модели будет завершена, можно просмотреть значения внутри переменных «Ans» и «Ans1» с помощью клавиш [F8] и [F9]. На эти функциональные клавиши на закладке «Function Keys» окна настроек модели были назначены команды «SHOW».

Задание 12.2: изучить применение процедур языка PLUS в модели «MIN.GPS».

Откройте модель «MIN.GPS». Эта модель может быть запущена при помощи команды «START 1». После того как четырем переменным «SAVEVALUE» присваиваются значения при помощи дискретной функции-рандомизатора под названием Myfunc, переменная с минимальным значением помещается в «SAVEVALUE Low» при помощи вызова процедуры «Minimum». Эта процедура сравнивает значения при помощи серии выражений IF. Временная переменная «Lowval» используется для хранения временного значения и итогового минимального значения. Это значение возвращается в операнд 2 блока «SAVEVALUE». Когда модель закончит работать, нажмите кнопки [F5], [F6], [F7] и [F8], для того чтобы увидеть в статусной строке главного окна числа, которые сравнивались в процессе работы процедуры. Клавиша [F9] выведет наименьшее число из четырех.

Задание 12.3: изучить применение процедур языка PLUS в модели «FACTORL.GPS».

Откройте модель «FACTORL.GPS». В этой процедуре языка PLUS используется цикл «WHILE». Процедура принимает аргумент, предоставляемый в результате выполнения «SAVEVALUE», и находит факториал этого значения, циклически умножая значение на временную переменную «Result» (по умолчанию ее значение равно 1). В конце каждой итерации значение аргумента 1 уменьшается на единицу до тех пор, пока оно достигнет 1. В этот момент происходит завершение цикла и выполнение команды «RETURN», которое передаст итоговое значение в операнд 2 блока «SAVEVALUE Answer». Вы можете просмотреть значение переменной «SAVEVALUE Answer», нажав клавишу [F8], функционал которой был установлен в окне настроек модели. Используйте команду «START 1» для запуска модели.

Можно вызывать процедуры PLUS интерактивно, после трансляции модели, используя команду «SHOW Factorial(10)» в командной строке. Это может быть полезным при проверке и отладке модели. Переопределение процедуры тоже может быть осуществлено интерактивно.

Задание 12.4: изучить применение процедур языка PLUS в модели «FACTORL1.GPS».

Откройте модель «FACTORL1.GPS». Данный подход к расчету факториала использует рекурсивный вызов процедуры. Процедура повторяет вызов самой себя до тех пор, пока значение аргумента 1 не станет равным единице. Язык PLUS также позволяет вызвать другие процедуры из тела процедуры. Аргументы, используемые для таких вызовов, хранятся в специальном стеке «Procedure Call Stack» до тех пор, пока в них есть необходимость. Максимальное значение стека может быть изменено в настройках модели.

Далее приведено содержимое входных и выходных файлов, ассоциированных с моделью «MATINIT.GPS», текст которой приведен ниже.

Input File - Inpt.txt

1	2	6
2	4	3
33	3	1
6	6	4
5	2343	1
6	45	4
4	6	7

Output File - Outpt.txt

Initial MX\$Values(1,1),1
Initial MX\$Values(1,2),2
Initial MX\$Values(1,3),6

Initial MX\$Values(2,1),2
Initial MX\$Values(2,2),4
Initial MX\$Values(2,3),3
Initial MX\$Values(3,1),33
Initial MX\$Values(3,2),3
Initial MX\$Values(3,3),1
Initial MX\$Values(4,1),6
Initial MX\$Values(4,2),6
Initial MX\$Values(4,3),4
Initial MX\$Values(5,1),5
Initial MX\$Values(5,2),2343
Initial MX\$Values(5,3),1
Initial MX\$Values(6,1),6
Initial MX\$Values(6,2),45
Initial MX\$Values(6,3),4
Initial MX\$Values(7,1),4
Initial MX\$Values(7,2),6
Initial MX\$Values(7,3),7

Задание 12.5: изучить применение процедур языка PLUS в модели «MATINIT.GPS».

Откройте модель «MATINIT.GPS». Эта модель считывает данные из внешнего ASCII-файла в матрицу системы GPSS World. Затем она записывает выражения, которые могут быть использованы для инициализации матрицы во время последующих прогонов модели с использованием команды «INCLUDE».

В данном примере использования языка PLUS сложный и длинный блок «WRITE» разбит на несколько доступных для понимания частей. Преимущество процедур PLUS здесь заключается в том, что выражения могут быть разбиты на несколько строк. Помимо этого, возможно определение переменных «TEMPORARY» для разбиения выражения на удобные для использования части. Все это позволяет оформить программу так, чтобы она сама себя документировала. Срок жизни переменных «TEMPORARY» длится до тех пор, пока длится выполнение процедуры PLUS. Можно оформить всю логику работы с файлами внутри процедуры. В этом случае потребуется использовать библиотечные процедуры «Open()», «Close()», «Read()» и «Write()».

Передача аргументов в процедуру «WriteInitial» не происходит. Все необходимые данные, существующие внутри объекта «Simualtion», доступны языку PLUS. Порой ценность передачи аргументов в процедуру заключается в том, что это делает программу более «читабельной», а также позволяет отслеживать значения переменных.

Команда «RETURN» обрабатывает комплексную строку и возвращает ее в блок «WRITE» для записи в файл «Outpt.txt». Блок «WRITE» понимает, в какой файл следует записать данные путем обработки операнда 2, представляю-

шего собой поток данных. Файл «Outpt.txt» был открыт при помощи потока данных 2. Команда «START 1» запускает модель.

12.3 Содержание отчета

1. Титульный лист.
2. Цель работы.
3. Краткие теоретические сведения.
4. Реализация решения задачи.
5. Выводы.
6. Список использованных источников.

12.4 Контрольные вопросы

1. Определение процедур языка PLUS.
2. Применение команд и процедур языка PLUS в процессе моделирования в системе GPSS World.
3. Применение циклов в процедурах PLUS.
4. Библиотечные процедуры «OPEN()», «Close()», «Read()» и «Write()».

Список использованной литературы

1. Томашевский, В. Н. Имитационное моделирование в среде GPSS / В. Н. Томашевский, Е. Г. Жданова ; под ред. Е. Г. Жданова. – М. : Бестселлер, 2003. – 416 с.
2. Учебное пособие по GPSS World / пер. с англ. В. В. Девяткова ; под ред. К. В. Кудашова. – Казань : Мастер Лайн, 2002. – 272 с.

Практическая работа №13

ОТЛАДКА МОДЕЛЕЙ

Цель работы

Изучить методы и приемы отладки сложных моделей в системе GPSS World.

13.1 Теоретическая часть

Во время отладки модели можно использовать многочисленные средства визуализации, имеющиеся в системе GPSS World. Среди 10 динамических окон есть «PLOT» и «EXPRESSIONS», которые позволяют визуализировать оценку любого выражения PLUS, поскольку оно изменяется динамически. Окно «TABLE» дает возможность визуализировать сходимость гистограмм. Кроме того, имеется 6 дополнительных окон для представления снимков. Они предназначены для профессионалов моделирования, нуждающихся в детальной информации о состоянии моделирования изнутри. Система GPSS World имеет большой набор стандартных сообщений. Они описывают конечное состояние и/или результаты моделирования. Можно изменять содержание стандартных сообщений, редактируя установки во время моделирования объекта. Промежуточные сообщения часто используются во время отладки модели.

Новый высокоэффективный транслятор системы работает как минимум на два порядка быстрее, чем его предшественник. Когда транслятор обнаруживает одну или большее число ошибок в модели, то создается круговой список сообщений об ошибках, к которым можно обращаться из моделируемой системы. Для этого используйте команды «Next Error» (Следующая ошибка) или «Previous Error» (Предыдущая ошибка) в меню, выпадающем при щелчке по пункту «Search» главного меню. После каждой остановки в списке на той или иной ошибке выдается сообщение о ней в строке состояния в нижней части главного окна системы. При этом курсор размещается перед синтаксическим элементом, который вызвал ошибку. Это значительно облегчает устранение ошибок. Для устранения ошибок используется текстовый редактор.

Если ошибка произошла в текстовом объекте, вводимом оператором «INCLUDE», можно вставить исправленные операторы в новый временный модельный объект и снова его оттранслировать.

13.2 Практическая часть

Интерактивные возможности системы GPSS World значительно упрощают отладку и тестирование различных вариантов модели по сравнению со старыми версиями GPSS. В предыдущей практической работе было показано, что отладка процедур PLUS проходит значительно быстрее в интерактивной среде.

Задание 13.1: изучить методы отладки на примере модели «BARBER.GPS».

Откройте модель «BARBER.GPS». Перед тем как продолжать, назначьте команды «SHOW» на функциональные клавиши. Откройте окно настроек модели, перейдите на закладку «Function Keys» и рядом с [F8] и [F9] введите следующие две команды: «SHOW P\$Custnum» и «SHOW X\$Custnum». Теперь в процессе отладки можно интерактивно изучать значение переменных посредством одного нажатия клавиши.

Создайте процесс моделирования и откройте окно блоков. Поместите условие «STOP» в транзакт 5. Для этого в командной строке введите команду «STOP 5». Выстройте окна так, чтобы была возможность одновременного наблюдения за окнами «Blocks» и «Journal».

Запустите моделирование командой «START 100». В журнале появится сообщение, подтверждающее, что транзакт 5 был остановлен при попытке войти в первый блок модели. В данном случае это блок «GENERATE».

Теперь приступим к пошаговому моделированию, используя функциональную клавишу [F5], на которую назначена команда «STEP 1». Для начала удалите условие «STOP» в окне «Window/Simulation «Snapshot»/User «STOP»s».

Нажмите клавишу [F5]. Транзакт 5 войдет в блок «GENERATE». Появится соответствующее трассировочное сообщение в окне «Journal». Теперь осуществите еще один шаг, чтобы транзакт попал в блок TEST. Когда транзакт попадет в блок «TEST», он обнаружит, что размер очереди в парикмахерскую превышает 1. Этот клиент решит не стоять в очереди и покинет парикмахерскую. Вы увидите, что следующим запланированным блоком для транзакта 5 станет блок «TERMINATE».

Введите в командной строке команду «SHOW Q\$Barber». Вы увидите, что размер очереди 2 был отображен в статусной строке главного окна и в окне «Journal». Теперь воспользуемся командами «SHOW», которые мы назначили на функциональные клавиши. Нажмите [F9]. В статусной строке будет отображено число 4, обозначающее количество клиентов, оставшихся в парикмахерской. Нажмите [F5], для того чтобы переместить транзакт 5 в блок «TERMINATE». Теперь проверим количество клиентов, оставшихся в парикмахерской. Нажмите [F9]. Возвращенное значение по-прежнему равно 4. Транзакт 5 не стал дожидаться своей очереди, поэтому не был учтен при подсчете количества клиентов. Для продолжения мы используем команду «CONTINUE» (клавиша [F2]).

Если окно «Blocks» открыто, модель будет работать медленнее. Закройте окно «Blocks» и сверните окно «Journal» для использования в дальнейшем. Вскоре моделирование будет завершено, в результате чего будет составлен отчет.

Откройте окно «Table» (команда меню «Window/Simulation Window/Table Window»). В выпадающем меню уже указана таблица «Waittime», являющаяся единственной для данной модели. На этой гистограмме представлены различ-

ные значения времени ожидания в очереди. Среднее время ожидания составляет 10,709 мин при стандартном отклонении в 2,703 мин. Окно «Table» является динамическим, его состояние обновляется в процессе моделирования.

Далее проследите за появляющимися в процессе работы с условиями «Stop» сообщениями. Снова откройте окно блоков. Теперь в окне «Blocks» выберите блок «RELEASE» (номер 9) данной модели и нажмите кнопку «Place» на панели «Debug». Условие «Stop» было установлено в блок 9. Этого эффекта можно достичь при помощи меню «Command» главного окна или команды «Custom», но использование панели инструментов «Debug» значительно проще. Запустите модель командой «START 100». Моделирование будет приостановлено, когда следующий транзакт будет готов войти в блок «RELEASE».

Измените структуру модели. Нажмите клавишу [F8]. Значение количества клиентов, отображенное в статусной строке главного окна, равно 54. Уберите условие «STOP» из блока «RELEASE»: в окне «Window/Simulation «Snapshot»/User «STOP»s» нажмите кнопку «Remove All».

Предположим, что нужно убрать из каждого транзакта параметр, отвечающий за количество клиентов в очереди. Удалите строку с блоком «ASSIGN». Теперь ретранслируйте модель и запустите моделирование командой «START 11111». Подождите 1–2 с и нажмите клавишу [F4], чтобы прервать моделирование, и нажмите [F8]. Наиболее вероятным является появление сообщения об ошибке, информирующего, что параметр не существует, поскольку больше не происходит его назначение.

Задание 13.2: изучить возможности внесения интерактивных изменений в процессе моделирования без ретрансляции модели.

Для исследования этих возможностей загрузите новую модель «SAMPLE7.GPS». Эта модель предназначена для имитации пересечения двух улиц с односторонним движением. Изменяя продолжительность горения зеленого света на светофоре, можно экспериментировать с потоком движения. Создадим график с двумя переменными, отвечающими за трафик по направлениям «север – юг» и «восток – запад» соответственно.

Запустите моделирование и откройте окно графика командой меню «Window/Simulation Window/Plot Window». Поле «Label» содержит надпись «East West Traffic», используемую в легенде, расположенной в нижней части графика, поле «Expression» определяет переменную «Q\$EastWest», значения которой будут отображены на графике. В поле Title наберите имя, определяющее оба параметра, которые мы хотим изобразить на графике, в данном случае это «Traffic «QUEUE»s» – потоки машин в обоих направлениях. Пределы по оси y имеют значения 0 и 150 соответственно. Временной промежуток установите на 8000. Следует сделать этот интервал достаточно длинным, чтобы он не «пролетел» мгновенно в ходе работы модели, но не настолько длинным, чтобы график не начал «заедать». Возможно, придется провести несколько экспериментов для того, чтобы определить подходящее значение. Следует знать, что ограниченное количество точек графика сохраняется для перерисовки окна

«Plot». По умолчанию для каждого выражения сохраняется не более 10 000 точек.

Нажмите кнопки «Plot» и «Memorize». Добавьте на график вторую переменную. В поле «Label» введите «North South Traffic», в поле «Expression» – «Q\$NorthSouth» и нажмите кнопки «Plot» и «Memorize». Запустите модель командой «START 4000». Обратите внимание на окно «Plot». Происходит скопление транспорта по направлению «восток – запад». Прервите процесс моделирования клавишей [F4]. Введите команду «Greentime EQU 1000» в командной строке и нажмите [F2]. Это приведет к выполнению команды «CONTINUE». Обратите внимание на скопление транспорта по направлению «север – юг». Снова прервите процесс моделирования клавишей [F4]. Введите команду «Greentime EQU 400» в командной строке и нажмите [F2]. Теперь поток трафика является умеренным в обоих направлениях. Закройте модель.

Система GPSS World позволяет вызывать командные файлы, содержащие INITIAL-выражения или другие команды. Сделать это можно, используя команду «INCLUDE» или помещая соответствующее выражение в текст модели.

Загрузите модель парикмахерской. Создайте процесс моделирования. Далее следует пример вызова командного ASCII-файла, содержащего начальные выражения. Первая команда продемонстрирует интерактивное подключение командного файла, а вторая является частью программы. Введите «INCLUDE «Init.txt»» в командной строке.

Помните, что необходимо указать путь в команде «INCLUDE», если местоположение командных файлов было изменено. Содержимое данного файла приведено ниже:

INITIAL X\$One,45

INITIAL X\$Two,765

Обратите внимание на значения переменных «Savevalue». Объекты «SAVEVALUE» создаются в тот момент, когда им присваиваются начальные значения. Выполните команду «SHOW X\$One».

В окне «Journal» и статусной строке главного окна должно появиться значение 45. Повторите данную последовательность команд, для того чтобы увидеть значение второй переменной «Savevalue». Введите команду «SHOW X\$Two» и проверьте значение, появившееся в статусной строке.

Теперь добавьте следующую строку в конец текста модели, отображенного в окне «Model».

INCLUDE «Init.txt»

Ретранслируйте модель. Будучи ретранслированной, модель автоматически считает внешний файл и создаст переменные «Savevalue».

Не забывайте, что если нужно отладить сложную систему или процедуру PLUS, можно включить опцию PLUS «TRACE» на закладке «Reports» в окне настроек. После этого каждый вызов процедуры будет сопровождаться появлением трассировочной записи в окне «Journal». Эти сообщения предназначены для отображения глубины обработки и списка, обрабатываемого в процессе вызова процедуры. Несложной является также и внутренняя отладка процедур PLUS. Для этого создайте одну или несколько пустых процедур PLUS. Затем можно дополнить свою процедуру вызовами отладочных процедур. При вызове функция PLUS «TRACE», будучи включенной, отобразит состояние аргументов, переданных отладочной процедуре, в окне «Journal».

13.3 Содержание отчета

1. Титульный лист.
2. Цель работы.
3. Краткие теоретические сведения.
4. Реализация решения задачи.
5. Выводы.
6. Список использованных источников.

13.4 Контрольные вопросы

1. Управление пользовательскими паузами процесса моделирования в процессе отладки модели.
2. Порядок внесения интерактивных изменений в процессе моделирования.
3. Вывод информации о внесенных интерактивных изменениях в процессе моделирования.
4. Присоединение команд, содержащихся во внешнем файле, к модели.

Список использованной литературы

1. Боев, В. Д. Моделирование систем. Инструментальные средства GPSS World / В. Д. Боев. – СПб. : БХВ-Петербург, 2012. – 368 с.
2. Учебное пособие по GPSS World / пер. с англ. В. В. Девяткова ; под ред. К. В. Кудашова. – Казань : Мастер Лайн, 2002. – 272 с.
3. Кудрявцев, Е. М. GPSS World. Основы имитационного моделирования различных систем / Е. М. Кудрявцев. – М. : ДМК Пресс, 2004. – 320 с.

Практическая работа №14

ПОЛЬЗОВАТЕЛЬСКИЕ ЭКСПЕРИМЕНТЫ

Цель работы

Исследовать использование языка PLUS для создания и проведения собственных экспериментов.

14.1 Теоретическая часть

С точки зрения представления поведения моделируемой системы имитационные модели относятся к классу описательных. Если в модели учитываются случайные факторы, то в процессе имитации обычно осуществляется большое число прогонов модели как с разными входными данными, так и с разными значениями последовательностей случайных чисел. Для детерминированной модели (без учета случайностей) достаточно одного прогона модели для каждой комбинации входных данных, однако на практике такие модели встречаются крайне редко.

В результате экспериментирования с моделью получают большое количество выходных данных, которые должны быть структурированы и интерпретированы так, чтобы их можно было использовать для принятия решений по результатам моделирования. Для правильной интерпретации полученных от модели выходных данных необходимо организовать эксперименты с моделью.

Организация эксперимента – это разработка плана проведения эксперимента, который дает возможность за минимальное число прогонов модели и при минимальной стоимости работ сделать статистически значимые выводы или найти наилучшее решение. При организации эксперимента обычно определяют:

- входные данные для каждого эксперимента;
- количество прогонов имитационной модели;
- длительность одного прогона модели;
- длительность переходного процесса моделирования, после которого необходимо собирать выходные данные;
- стратегию сбора данных для каждого прогона модели;
- методы оценки точности выходных данных с построением доверительных интервалов;
- чувствительность модели к входным данным, различным видам распределений, сценариям поведения моделируемой системы;
- условия эксперимента и сценарии;
- условия генерации потоков случайных чисел внутри системы моделирования и для вероятностных входных данных;
- стратегию достижения цели эксперимента (например, сравнение альтернативных вариантов или оптимизация целевой функции).

Конечная цель проведения экспериментов – это получение достаточной статистической информации для принятия решений по результатам моделирования. Как правило, моделирование проводится с целью нахождения некоторых экстремальных значений характеристик моделируемой системы (оптимизирующий эксперимент) или для выявления важных факторов, влияющих на моделируемую систему (отсеивающий эксперимент). Оба эти эксперимента используют факторные планы и аппроксимируют поверхность отклика полиномами разного порядка, а для поиска экстремальных значений применяются численные методы оптимизации. Для этих экспериментов необходима некоторая функциональная зависимость значений выходной переменной (отклика) от входных переменных или факторов, которая, как правило, отражает критерий эффективности моделируемой системы. Таким образом, поиск наилучшего решения выражается численной характеристикой этого критерия, а для нахождения экстремальных значений необходимо исследовать поверхности отклика (проводить эксперименты) в разных точках. От выбора начальной точки в факторном пространстве во многом зависит эффективность экспериментов.

Другой вид экспериментов, проводимых с моделью, это структурная оптимизация, под которой будем понимать поиск наилучшей структуры моделируемой системы. В этом случае эксперименты проводятся с разными моделями, а не с одной, как в предыдущем случае, причем модели могут отличаться структурой, параметрами и принятыми алгоритмами поведения. Для таких экспериментов нет единого числового критерия оптимизации, что затрудняет использование классических методов. Однако количество рассматриваемых вариантов, как правило, невелико, поэтому для структурной оптимизации можно использовать метод выдвижения гипотез с перебором вариантов. Оптимизация каждого варианта моделируемой системы обычно осуществляется с помощью поиска узких мест и их устранения, то есть балансировки моделируемой системы. Узкие места определяют пропускную способность всей системы. Поиск наилучшего решения осуществляется сравнением рассмотренных вариантов.

Перечислим основные проблемы, возникающие при экспериментировании с имитационными моделями.

1. *Задание начальных условий эксперимента.* Обычно эксперимент начинают из состояния «пусто и свободно», то есть когда в модели нет транзактов. Если рассматривать достаточно длительный период моделирования, то можно наблюдать так называемый период «разогрева» модели, или *переходной период*, после которого модель может перейти в стационарный режим работы. Учет данных переходного периода для выходных переменных модели будет вносить смещение в статистические оценки. Чтобы уменьшить влияние данных переходного процесса на конечные результаты, можно поступать следующим образом:

- запускать модель с модальных (наиболее вероятных) значений установившегося режима;
- запускать модель со средних значений установившегося режима.

При оценивании выходной величины рекомендуется не учитывать данные переходного процесса, так как они могут давать существенное смещение искомых оценок. Это достигается путем удаления данных переходного процесса (с помощью команды «RESET»). Лучший способ определения установившегося процесса – это использование графиков для наблюдения за изменением переходного процесса во времени.

2. *Правильность определения правила останова*, определяющего длительность имитационного прогона. От продолжительности прогона зависит точность результатов моделирования.

3. *Состояния модели в момент прекращения прогона*. Часто при моделировании возникает вопрос: «Что делать с оставшимися компонентами (транзактами) модели в момент окончания ее работы?» Учет оставшихся компонентов может привести к смещению оценки в большую сторону. Например, при моделировании работы некоторого цеха использовалось правило, что наиболее короткие работы запускаются раньше. На момент окончания в модели останутся незавершенные работы с длительными временами выполнения. Если их не учитывать, то оценка средней длительности работ в цеху будет занижена.

4. *Определение длительности прогона модели при наличии в модели процессов с различными скоростями протекания*. Оценку точности результатов моделирования обычно выполняют для самого медленного процесса в модели. В этом случае оценки для быстрых процессов будут заведомо намного лучше, чем для медленного процесса, то есть доверительные интервалы для них будут меньше. При разработке имитационной модели обычно выбирают степень детализации модели так, чтобы скорости протекающих в ней процессов не различались более чем на два порядка. В случаях моделирования редких событий (медленные процессы), например, отказов оборудования, необходимо укрупнять состояния для быстрых процессов. Для этого обычно используют аналитико-имитационные модели.

14.2 Практическая часть

После того как модель была протестирована при различных условиях, наверняка возникает задача ее исследования. В этот момент возникает необходимость в экспериментировании. Для того чтобы определить отклики факторов, находящихся под контролем, необходимо смоделировать каждую из возможностей и определить, превышает ли результат ожидаемое значение только из-за случайных изменений. Модель должна быть протестирована при наиболее напряженных условиях. Она должна быть снабжена средствами, предоставляющими все необходимые для анализа измерения.

В задании эксперимент будет состоять из трех прогонов для каждого из четырех уровней факторов. Прогоны осуществляются при одинаковых условиях. Изменяется только лишь начальное значение для генератора последовательностей псевдослучайных чисел. Это делается с целью создания базы для вариативности результатов.

В системе GPSS World определение эксперимента происходит в момент трансляции специальной процедуры PLUS, называемой «Experiment». Отличие процедуры от эксперимента заключается лишь в том, что ключевое слово «PROCEDURE» заменяется ключевым словом «Experiment». Еще одним отличием эксперимента от процедуры является возможность использования процедур библиотеки «DoCommand». В итоге эксперименты наследуют всю мощь языка PLUS, что делает их полностью программируемыми и автоматизируемыми.

Для запуска ранее определенного эксперимента может быть интерактивно использована команда «CONDUCT», если она не была определена в коде модели. Для запуска эксперимента PLUS достаточно использовать текст вызова процедуры в качестве единственного операнда команды «CONDUCT». После запуска эксперимента при помощи команды «CONDUCT» возможности по взаимодействию с моделью будут ограничены. Всегда можно вывести значения модельного времени («View / Clock»), но для изучения модели придется приостановить эксперимент.

Задание 14.1: изучить эксперимент, определенный для модели «ONEWAY.GP».

Откройте модель «ONEWAY.GPS». Прежде всего обратите внимание на то, что эксперимент PLUS под названием «BestLines» вызывает вторую процедуру PLUS, названную «DoTheRun». Задачей данной процедуры является создание начальных условий для каждого прогона. При проведении пользовательского эксперимента мы должны написать эти условия вручную.

После трансляции модели процедуры будут преобразованы и зарегистрированы в объекте «Simulation» наравне с командами GPSS. Они в любой момент будут доступны для вызова. Можно отправить эти процедуры в существующую симуляцию, используя команду «Custom» или команду «INCLUDE».

При взаимодействии с моделью мы можем не обращать внимания на эксперимент BestLines, потому что он ничего не делает до тех пор, пока в модель не отправлена команда «CONDUCT». Обратите внимание на выражения эксперимента.

Эксперимент «BestLines» спроектирован таким образом, чтобы заполнить глобальную матрицу результатами и передать ее процедурной библиотеке ANOVA для дальнейшего статистического анализа и протоколирования. Мы инициализируем элементы матрицы как «UNSPECIFIED» для того, чтобы процедура ANOVA знала, какие прогоны были завершены. Генерируемые эксперименты проверяют матрицу результатов на предмет наличия элементов «UNSPECIFIED», для того чтобы не осуществлять уже выполненные прогоны.

В дополнение к матрице Result была определена сущность «TABLE», для того чтобы получить разности, обнаруженные подпрограммой ANOVA в ходе дисперсионного анализа. Все что нам необходимо сделать, это определить сущность «TABLE» с таким же именем, что и у матрицы результатов, за тем лишь исключением, что к имени таблицы добавляется «_Residuals».

Обратите внимание на наличие внешнего и внутреннего блоков «WHILE/DO», осуществляющих все необходимое для выполнения эксперимента моделирование. В нашем эксперименте задействован единичный фактор – количество телефонных линий, имеющий четыре разных уровня. Для каждого уровня фактора мы осуществляем три прогона, изменяя начальные параметры для генератора случайных величин. Использование разных значений для генератора улучшает значение стандартной погрешности эксперимента.

В теле внутреннего цикла эксперимента «BestLines» первое выражение вызывает процедуру «DoTheRun». Процедура «DoTheRun» использует набор команд для инициализации и запуска прогонов. Поскольку процедура «DoTheRun» вызывается экспериментом, она может пользоваться процедурами библиотеки «DoCommand». Таким образом, она посылает модели команд «CLEAR OFF», «RESET» и «START». Каждый раз по завершении выполнения процедуры «DoTheRun» эксперимент «BestLines» сохраняет полученный результат в соответствующую ячейку глобальной матрицы «MainResult», которая определена в листинге сразу после «BestLines()». В эксперименте может присутствовать разное количество результирующих матриц, каждая из которых должна иметь по одному измерению для каждого из факторов (вплоть до пяти) и еще одно измерение для индексирования прогонов. Поскольку осуществляется эксперимент типа «One Way ANOVA», первый индекс матрицы результатов определяет уровень фактора, представляющий количество телефонных линий, а второй индекс позволяет осуществлять несколько экспериментов для каждого уровня фактора. Таким образом, матрица результатов имеет два измерения. После завершения экспериментов процедура «BestLines» вызывает библиотеку ANOVA, которая осуществляет вывод статистического отчета в окно «Simulation «Journal»».

Вот несколько приемов, связанных с использованием библиотеки «DoCommand». Если требуется включить строку в строковое выражение, используйте четыре пары двойных кавычек. Каждая внутренняя строка должна быть заключена в пару двойных кавычек. Не передавайте процедуре «DoCommand» строку, содержащую название переменной «TEMPORARY». Строка аргументов транслируется на глобальный уровень, из которого переменные «TEMPORARY» становятся недостижимыми. В случае с аргументами и временными переменными сначала сформируйте строку, а затем передайте ее процедуре «DoCommand».

Обычно система GPSS World кэширует все команды (за исключением «HALT» и «SHOW») в очередь с низким приоритетом «Command QUEUE», а затем работает с ними поочередно до тех пор, пока очередь не опустеет. «DoCommand» работает несколько иначе. Она не возвращается в вызывающую процедуру до тех пор, пока командная очередь не окажется пустой. Это означает, что после запуска команды «START» работа процедуры «DoCommand» не будет завершена до тех пор, пока не будет завершено моделирование. Только после этого осуществляется возврат в вызывающее выражение. Таким образом,

эксперимент может получить доступ к результатам прогона сразу после того, как он использует процедуру «DoCommand» для запуска.

Обратите внимание, что процедура «DoTheRun» использует команду «CLEAR» с аргументом «OFF». В противном случае матрица «MainResult» будет обнулена и все результаты будут потеряны. После завершения экспериментов мы сможем открыть окно, содержащее результаты моделирования.

Обратите внимание на то, как используются команды «SHOW» для вывода сообщений в журнал моделирования. Это довольно удобный способ отслеживания текущего состояния экспериментов. Все, что для этого требуется, это один взгляд в журнал.

Эксперимент «BestLines» также создает отдельный файл результатов в форме текстового объекта с использованием режима ручного моделирования, а также делает вызовы процедур «Open()», «Seek()», «Write()» и «Close()» для создания файла с результатами, «Result.txt». Затем он может быть прочитан с помощью пакетов для анализа данных и статистики либо просто использован как запись о проделанном эксперименте.

Запустите процесс моделирования. Был создан объект «Simulation» типа «One Way». Мы хотим запустить эксперимент с помощью команды «CONDUCT BestLines(1,4,1)». Аргументы команды «CONDUCT» используются для присвоения значений переменным «LeastLinesToRun», «MostLinesToRun» и «IncrementLeast» соответственно. Нужная функция уже назначена на клавишу [F10] – нажмите ее. Эксперимент запустился автоматически. Есть также возможность запустить несколько экспериментов одновременно, создав еще объекты «Simulation». Обратите внимание на модельное время работающей модели (меню «View/Simulation Clock»). В нижнем левом углу главного окна появится отображение работающего системного таймера. Работающие часы замедляют проведение эксперимента, поэтому выключите их (меню «View/Show Clock»). Повторный вызов этой команды убирает с экрана отображение системных часов. Дождитесь окончания эксперимента, это займет несколько минут. Откройте матрицу результатов командой меню «Window/Simulation Window/Matrix Window». В окне «Matrix» отобразятся все результаты. Возможно, придется расширить окно для того, чтобы увидеть результаты трех прогонов для каждого уровня фактора. Теперь откройте и просмотрите файл с результатами «Result.txt». Результаты каждого эксперимента были помещены в текстовый файл путем вызова процедуры «WRITE ()».

Теперь обратите внимание на таблицу ANOVA в журнале модели. Из таблицы ANOVA мы видим, что значение статистического показателя F настолько велико, что не остается сомнений в том, что количество линий не может быть определяющим фактором. Система GPSS World помещает значение «Critical Value of F», которое является нашим стандартом для сравнения со статистическим показателем F . Когда показатель F , рассчитанный для фактора эксперимента, превышает связанное с ним критическое значение, мы делаем вывод, что полученный эффект является значимым (при $p = 0,05$).

В описательных статистических таблицах ANOVA видно, что добавление трех линий к конфигурации приводит лишь к незначительным улучшениям. Как правило, мы обнаруживаем, что затраты на четвертую линию себя не окупают. Поэтому мы делаем вывод, что при данных параметрах загрузки система должна быть оснащена тремя телефонными линиями.

Исследуйте таблицу разностей, построенную подпрограммой ANOVA. Чтобы увидеть таблицу, откройте окно матрицы. Доверительные интервалы посчитаны, исходя из объединенных расчетов стандартной погрешности. Тем не менее присутствует некая неоднородность результатов, поскольку наблюдалось сильное уменьшение значений в рамках диапазона для одного уровня фактора. Но эффективность увеличения количества телефонных линий не подвергается сомнению.

14.3 Содержание отчета

1. Титульный лист.
2. Цель работы.
3. Краткие теоретические сведения.
4. Реализация решения задачи.
5. Выводы.
6. Список использованных источников.

14.4 Контрольные вопросы

1. Правила организации экспериментов в GPSS World.
2. Основные проблемы при организации экспериментов.
3. Порядок запуска эксперимента для текущей модели.
4. Использование библиотеки «DoCommand» в эксперименте.
5. Получение и исследование результата пользовательского эксперимента.

Список использованной литературы

1. Боев, В. Д. Моделирование систем. Инструментальные средства GPSS World / В. Д. Боев. – СПб. : БХВ-Петербург, 2012. – 368 с.
2. Учебное пособие по GPSS World / пер. с англ. В. В. Девяткова ; под ред. К. В. Кудашова. – Казань : Мастер Лайн, 2002. – 272 с.
3. Томашевский, В. Н. Имитационное моделирование в среде GPSS / В. Н. Томашевский, Е. Г. Жданова ; под ред. Е. Г. Жданова. – М. : Бестселлер, 2003. – 416 с.

Практическая работа №15

ГЕНЕРАТОРЫ ЭКСПЕРИМЕНТОВ

Цель работы

Изучить процесс создания отборочного эксперимента PLUS.

15.1 Теоретическая часть

При экспериментировании с моделью различают входные и выходные переменные. Входные переменные называются факторами, выходные – откликами. Каждый фактор в эксперименте может принимать одно или несколько значений, называемых уровнями фактора. Множество уровней факторов определяет одно из возможных состояний моделируемой системы и представляет условия проведения одного из возможных экспериментов.

Факторный эксперимент может быть отсеивающий, когда из всего множества факторов определяются те факторы, которые существенно влияют на отклики модели. Второй вид факторного эксперимента используется для определения экстремальных значений на поверхности отклика. В этом случае серия факторных экспериментов планируется так, чтобы достичь экстремума на поверхности отклика.

Факторный эксперимент представляет собой план, в котором все уровни каждого фактора встречаются в сочетании со всеми уровнями всех других факторов. Различные уровни некоторого фактора могут соответствовать качественным значениям (например, разные дисциплины обслуживания в устройстве) или количественным значениям (например, число устройств обслуживания).

Применение факторного плана вместо классической схемы, согласно которой каждый раз изменяется только один фактор, имеет ряд преимуществ.

1. Становится более полной картина влияния каждого фактора, поскольку они изучаются в самых различных условиях (вследствие одновременного изменения других факторов).

2. Большое число комбинаций факторов, используемых в эксперименте, облегчает предсказание результатов, которые могут быть достигнуты при определенной комбинации условий.

3. Если эффекты, вызываемые каждым фактором, статистически независимы, то о каждом факторе можно получить не меньше информации, чем при изменении в экспериментах только одного фактора при фиксации остальных.

4. Если (как это часто бывает) различные факторы не являются независимыми, а вызывают эффекты, которые в большей или меньшей степени коррелированы, то в этом случае только факторный эксперимент может дать информацию о характере этих взаимодействий. При наличии нескольких взаимосвязанных существенных факторов обойтись без постановки факторного

эксперимента невозможно. Для ряда часто встречающихся специальных задач разработано большое число стандартных факторных планов.

Чтобы определить, является фактор значимым или нет, используется дисперсионный анализ ANOVA (analysis of variance), который применим только к количественным факторам. С помощью него определяются количественные отклонения наблюдений от среднего значения. Если какой-либо фактор не оказывает влияние на отклик, то он является незначимым. С другой стороны, если фактор влияет на отклик, то его (фактора) количественное значение сравнивают с оценкой изменчивости наблюдения, то есть со стандартной ошибкой.

Это делается для исключения эффектов, которые являются не более чем случайной флуктуацией.

С помощью ANOVA проверяется гипотеза об отсутствии влияния фактора. Если справедлива гипотеза об отсутствии влияния фактора, то считается, что все наблюдения получены из одной генеральной совокупности. Для проверки гипотезы используется F -распределение Фишера. Критерий Фишера определяет отношение двух выборочных дисперсий. Если фактор существенно влияет на отклик, то значения F -распределения принимают большие значения и F -статистика становится значимой. Таким образом, большие значения F приводят к отбрасыванию гипотезы об отсутствии влияния фактора, то есть фактор является значимым.

15.2 Практическая часть

Рассмотрим подробно модель «ExperEther.gps», входящую в комплект поставки GPSS World. Она основана на моделировании сети Ethernet. Последовательно изучим все шаги создания эксперимента, предназначенного для определения наиболее значимых факторов.

Откройте модель «EXPERETHER.GPS». Затем откройте генератор отборочных экспериментов командой меню «Edit/Insert Experiment/Screening ...». Выполнение этой команды приведет к появлению диалогового окна, в которое мы внесли всю необходимую для генерации эксперимента информацию (рисунок 15.1).

Объект Model запомнит эксперименты, которые были определены ранее. Данные, которые показаны на рисунке 15.1, были сохранены вместе с объектом ExperEther Model. Поля «Experiment Name» и «'Run PROCEDURE' Name» используются как имена процедур в генерируемом эксперименте. Новый эксперимент PLUS осуществит циклический вызов процедуры «Run».

Далее следуют факторы эксперимента. Название каждого фактора одновременно является названием переменной, и поэтому должно подчиняться соглашениям об именах системы GPSS World. Оно должно начинаться с символа-буквы и не должно пересекаться с ключевым словом, атрибутом или классом «SNA».

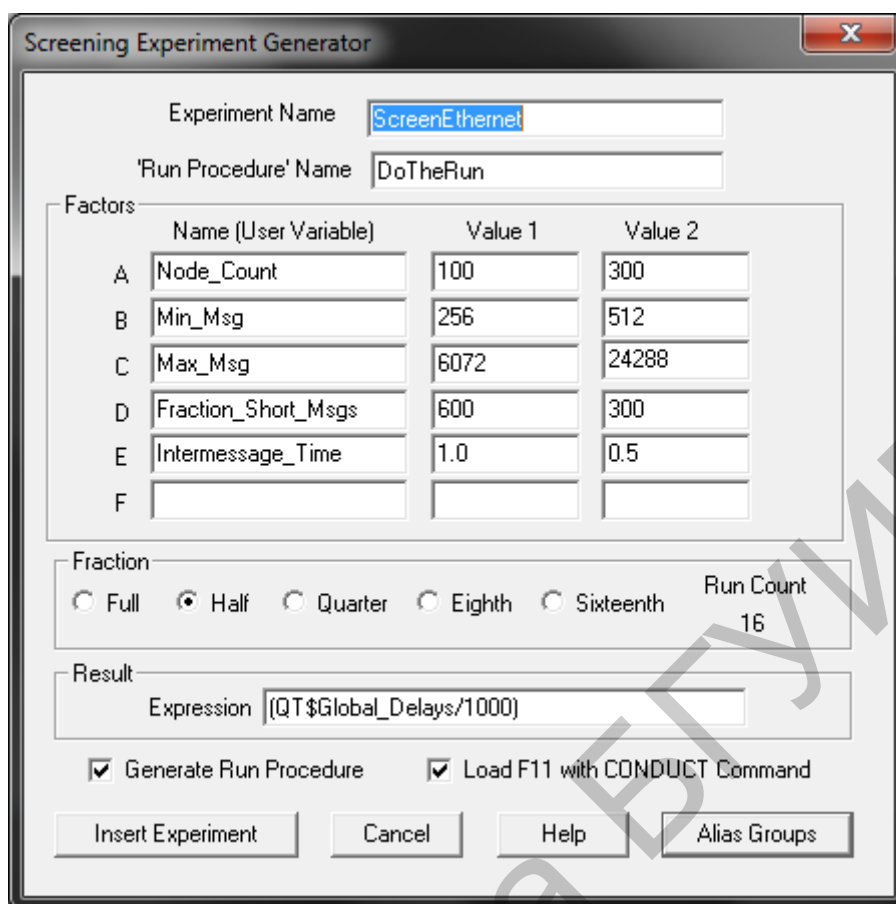


Рисунок 15.1 – Диалоговое окно «Screening Experiment Generator»

Поскольку отборочные эксперименты представляют собой полные или дробные эксперименты типа 2^k , для каждого фактора необходимо указывать два уровня. Укажите название и уровни для каждого из шести факторов. Факторы должны быть определены последовательно, начиная с фактора А. Выбор уровней фактора является критическим моментом. Возможно, следует провести несколько предварительных экспериментов для определения хороших значений. Выберите значения, расположенные достаточно далеко друг от друга, для того чтобы выявить изменения в поведении модели.

Следующей является группа «Fraction». Она позволяет определить, какую фракцию полного 2^k эксперимента следует использовать для прогонов. Итоговое количество прогонов приведено справа. Для уменьшения количества прогонов выбирайте маленькую фракцию.

Группа «Result» имеет поле для ввода, необходимое для заполнения. Здесь нужно определить выражение, представляющее собой основной показатель модели. Система GPSS World автоматически сгенерирует код, определяющий и инициализирующий матрицу результатов, в которую заносятся значения выражения «Result». Когда будет осуществляться эксперимент, результат каждого прогона будет автоматически помещен в соответствующее ему место в матрице результатов.

Далее следуют два флажка, определяющие опции эксперимента. Прежде всего следует дать системе команду на генерацию шаблона процедуры «Run». В дальнейшем можно отредактировать эту процедуру, чтобы она соответствовала требованиям. Также можно назначить на клавишу [F11] подходящую команду «CONDUCT». Если установить этот флажок, то после создания объекта «Simulation» достаточно будет лишь нажать функциональную клавишу [F11] для запуска эксперимента.

Перед генерацией эксперимента следует изучить группы псевдонимов («Alias Groups»). Поэтому переключитесь от половинной фракции к восьмой. Нажмите кнопку «Eight» в группе «Fraction». Затем нажмите кнопку «Alias Groups», чтобы изучить последствия. Появится диалоговое окно «Alias Groups».

Во время исследования групп псевдонимов мы прежде всего ищем недостающие эффекты, а затем важные эффекты, которые являются неразличимыми из-за того, что они находятся в одной группе. Основные эффекты обозначаются одной буквой, например, «E» означает главный эффект пятого фактора. По аналогии «AE» обозначает взаимодействие первого и пятого факторов в рамках эксперимента.

Можно попробовать другие генераторы для групп псевдонимов. Те эффекты, которые используются в качестве генератора, не появятся в качестве группы псевдонимов, и поэтому будут потеряны. По этой причине лучшим выбором обычно является взаимодействие на высоком уровне. В дополнение стоит отметить, что продукты комбинации генератора и продукта также будут потеряны. К примеру, если используются «ABD» и «ADE» в качестве генераторов, «ABDADE = AABDDE = BE», то комбинация «BE» тоже будет потеряна. Каждый выбираемый генератор не должен быть производным от существующих генераторов. Кроме того, нельзя использовать идентичные генераторы.

Тем не менее поскольку существует всего три группы псевдонимов, нет возможности различить пять основных эффектов. Закройте диалоговое окно «Alias Groups».

Вернитесь к генератору эксперимента, выберите 1/4 фракцию с восьмью прогонами. Для этого нажмите кнопку «Quarter» в группе «Fraction». Затем снова откройте диалоговое окно «Alias Groups». Группы в этом случае выглядят значительно лучше. Все главные эффекты находятся в разных группах. Тем не менее обработка информации о взаимодействии двух факторов будет затруднительна. При определенных обстоятельствах это решение является вполне адекватным.

Вернитесь к генератору эксперимента, выберите 1/2 фракцию с 16 прогонами (кнопка «Half» в группе «Fraction»). Затем снова откройте диалоговое окно «Alias Groups». Теперь проект выглядит значительно лучше, но требует 16 прогонов. Все основные факторы изолированы, не пересекаясь с взаимодействиями двух факторов. Более того, основные эффекты частично изолированы от эффектов взаимодействия двух факторов. Приступите к генерации эксперимента (нажимайте кнопку «ОК»).

Задание 15.1: сгенерировать выражения PLUS и добавить их в конец модели.

Добавьте эксперимент («Insert Experiment»). Поскольку мы запросили генерацию процедуры «RUN», далее появится окно, предлагающее нам задать ей свои параметры. Потребуется сделать изменения, предназначенные для конкретно вашей модели, а именно:

- 1) определить иной период технологической подготовки;
- 2) определить длину периода, отводимого на измерения.

Очевидно, можно изменить способ назначения уровней факторов пользовательским переменным или изменить начальные параметры для генераторов случайных чисел. В любом случае можно будет отредактировать сгенерированное выражение непосредственно в окне «Model». Там же можно изменить процедуру «Run». Нажмите «ОК» и все сгенерированные системой GPSS World выражения будут помещены в конец текста модели.

Задание 15.2: запустить эксперимент.

Осуществление происходит с помощью команды «CONDUCT» системы GPSS World. Ее синтаксис схож с вызовом процедуры. Но есть и другой способ. В диалоговом окне «Experiment Generator» мы выбрали опцию, которая приказала системе GPSS World назначить подходящую команду «CONDUCT» на функциональную клавишу [F11] (в меню «Edit/Settings/Function Keys»).

Чтобы осуществить запуск эксперимента, необходимо выполнить два действия. Прежде всего выполнить трансляцию модели, создавая тем самым объект «Simulation». Затем запустить команду «CONDUCT», нажав клавишу [F12]. Эксперимент начнет выполняться, осуществляя вывод и оповещение о текущем статусе в окно «Journal» объекта «Simulation». Отборочный эксперимент будет состоять из шестнадцати прогонов. Каждый отклик, а затем и итоговые результаты будут выведены в окно «Journal». Результаты будут включать в себя эффекты для каждой группы псевдонимов.

Даже на быстром персональном компьютере выполнение эксперимента займет некоторое время. Результаты показывают, что факторы С и Е имеют значимые эффекты, и поэтому заслуживают дополнительного изучения. Оба входили во взаимодействие только четырех факторов, и, вероятно, являются источником рассчитанных эффектов. Ни один другой фактор, включая взаимодействие двух или трех факторов, не имел подобной значимости.

Теперь должно быть ясно, чего мы добились, используя дробный факторный эксперимент. Затратив половину стоимости полного факторного эксперимента, мы смогли отсеять большой набор факторов и их взаимодействий. Теперь вы готовы спроектировать еще один или несколько пользовательских экспериментов, и, наконец, оптимизационный эксперимент, используя другой автоматический генератор экспериментов системы GPSS World.

15.3 Содержание отчета

1. Титульный лист.
2. Цель работы.
3. Краткие теоретические сведения.
4. Реализация решения задачи.
5. Выводы.
6. Список использованных источников.

15.4 Контрольные вопросы

1. Проведение факторного эксперимента в GPSS World.
2. Настройка генератора отборочных экспериментов.
3. Проведение полного эксперимента.
4. Проведение дробного эксперимента.
5. Осуществление запуска пользовательского эксперимента.

Список использованной литературы

1. Томашевский, В. Н. Имитационное моделирование в среде GPSS / В. Н. Томашевский, Е. Г. Жданова ; под ред. Е. Г. Жданова. – М. : Бестселлер, 2003. – 416 с.
2. Учебное пособие по GPSS World / пер. с англ. В. В. Девяткова ; под ред. К. В. Кудашова. – Казань : Мастер Лайн, 2002. – 272 с.

Учебное издание

Алексеев Виктор Федорович
Пискун Геннадий Адамович
Перевощиков Василий Анатольевич

**МОДЕЛИРОВАНИЕ И ОПТИМАЛЬНОЕ ПРОЕКТИРОВАНИЕ
ТЕХНИЧЕСКИХ СИСТЕМ:
ПОСОБИЕ К ПРАКТИЧЕСКИМ ЗАНЯТИЯМ**

ПОСОБИЕ

Редактор *М. А. Зайцева*
Корректор *Е. Н. Батурчик*
Компьютерная правка, оригинал-макет *В. М. Задоя*

Подписано в печать 10.01.2017. Формат 60x84 1/16. Бумага офсетная. Гарнитура «Таймс»
Отпечатано на ризографе. Усл. печ. л. 6,98. Уч.-изд. л. 7,3. Тираж 50 экз. Заказ 249.

Издатель и полиграфическое исполнение: учреждение образования
«Белорусский государственный университет информатики и радиоэлектроники».
Свидетельство о государственной регистрации издателя, изготовителя,
распространителя печатных изданий №1/238 от 24.03.2014,
№2/113 от 07.04.2014, №3/615 от 07.04.2014.
ЛП №02330/264 от 14.04.2014.
220013, Минск, П. Бровки, 6