**Министерство науки и высшего образования Российской Федерации**

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ**

**«МОСКОВСКИЙ АВИАЦИОННЫЙ ИНСТИТУТ**

**(НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ)»**

**МАИ**

ЛАБОРАТОРНАЯ РАБОТА № 1

по дисциплине

«Вычислительные системы»

по теме

«Создание и уничтожение транзактов, работа устройства, сбор статистики»

(Вариант 12)

Выполнил:

студент группы М6О-113М-19

Бурченко К.А.

Проверил:

Доцент кафедры 304

Жигалов В.И.

Москва 2019

**Оглавление**

[Теория 3](#_Toc25670822)

[***Общие сведения о системе GPSS*** 3](#_Toc25670823)

[***Типы объектов в GPSS*** 3](#_Toc25670824)

[***Организация работы модели в системе GPSS*** 4](#_Toc25670825)

[***Синтаксис языка GPSS*** 5](#_Toc25670826)

[***Создание и уничтожение транзактов. Блоки GENERATE и TERMINATE. Блок START.*** 5](#_Toc25670827)

[***Задержка транзактов. Блок ADVANCE.*** 6](#_Toc25670828)

[***Блоки SEIZE и RELEASE.*** 6](#_Toc25670829)

[***Блоки QUEUE и DEPART.*** 6](#_Toc25670830)

[***Статистика (содержимое файла examp.lis).*** 6](#_Toc25670831)

[Задание на работу 8](#_Toc25670832)

[Дано 8](#_Toc25670833)

[N=40 транзактов 8](#_Toc25670834)

[Код программы 8](#_Toc25670835)

[Блок схема модели 9](#_Toc25670836)

[Статистический отчёт 10](#_Toc25670837)

[Обратная задача 15](#_Toc25670838)

[Вывод 17](#_Toc25670839)

[Используемая литература 18](#_Toc25670840)

# Теория

## ***Общие сведения о системе GPSS***

В системе GPSS моделируемая система представляется с помощью набора абстрактных элементов, называемых объектами. Каждый объект принадлежит к одному из типов объектов. Объект каждого типа характеризуется определенным способом поведения и набором атрибутов, определяемыми типом объекта. Например, если рассмотреть работу порта, выполняющего погрузку и разгрузку прибывающих судов, и работу кассира в кинотеатре, выдающего билеты посетителям, то можно заметить большое сходство в их функционировании. В обоих случаях имеются объекты, постоянно присутствующие в системе (порт и кассир), которые обрабатывают поступающие в систему объекты (корабли и посетители кинотеатра). В теории массового обслуживания эти объекты называются приборами и заявками. Когда обработка поступившего объекта (заявки) заканчивается, он покидает систему. Если в момент поступления заявки прибор обслуживания занят, то заявка становится в очередь, где и ждет до тех пор, пока прибор не освободится. Очередь также можно представлять себе как объект, функционирование которого состоит в хранении других объектов.

## ***Типы объектов в GPSS***

В GPSS имеется несколько типов объектов: динамические объекты, объекты типа «оборудование», статистические и операционные объекты.

Динамические объекты (транзакты) служат для задания заявок на обслуживание. Транзакты могут порождаться во время моделирования и уничтожаться (покидать систему). Каждый транзакт может иметь до ста числовых параметров, значения которых можно изменить в процессе моделирования. Обращения к параметрам транзакта выполняется по их номеру. В процессе моделирования транзакты проходят через другие объекты модели, производя в них некоторые действия и испытывая на себе их влияние.

С помощью объектов типа «оборудование» в моделях представляются различные элементы систем, через которые проходят транзакты. Транзакт, поступивший на объект типа "оборудование", изменяет состояние этого объекта. Наиболее часто используемым объектом типа «оборудование» является «устройство». Устройство может находиться в свободном состоянии или может быть занято транзактом, причем только одним в каждый момент времени.

Статистические объекты применяются для сбора и обработки данных, характеризующих работу моделируемой системы. Один из их представителей – объект "очередь", который содержит список транзактов, задержанных в одном или нескольких пунктах системы, и ведет в процессе моделирования подсчет статистических характеристик соответствующей очереди.

К операционным объектам, называемым также блоками, относятся объекты нескольких типов, которые служат для задания логики работы системы путем определения маршрутов движения транзактов и изменений в состоянии системы, обусловленных прохождением транзактов через операционные объекты. Основные типы операционных объектов будут подробно рассмотрены после описания общей схемы работы моделей в системе GPSS. Правила функционирования операционных объектов составляют основную часть семантики GPSS.

## ***Организация работы модели в системе GPSS***

**Модель**

**Ассемблер**

**GPSS**

**Программа ввода**

**Интерпретатор модели**

**Результат моделирования**

**Рис.1 Моделирование в системе GPSS**

Модель, вводится в ЭВМ и поступает на обработку ассемблером GPSS.

Ассемблер проводит синтаксический контроль модели и преобразует ее во внутреннюю форму, в которой она передается с помощью программы ввода интерпретатору модели, который выполняет моделирование. Во внутренней форме все объекты, описанные в модели, получают последовательные номера в порядке поступления. Последовательности номеров выстраиваются отдельно по типам объектов.

Интерпретатор модели является основной частью системы моделирования GPSS. Его главной функцией является создание транзактов и проведение их через блоки модели с одновременным выполнением действий, связанных с каждым блоком. Движение транзактов в модели соответствует движению отображаемых ими объектов в реальной системе.

Всякое изменение состояния модели можно рассматривать как некоторое событие, происходящее в определенный момент условного (системного) времени, задаваемого «часами» системы, работа которых организуется интерпретатором. «Часы» в интерпретаторе GPSS - это целая переменная, значение которой соответствует текущему моменту условного времени модели.

В процессе моделирования интерпретатор автоматически определяет правильную очередность наступления событий. В случае если нужные действия в намеченный момент времени выполнены быть не могут (например, занято устройство, к которому обращается транзакт), интерпретатор временно прекращает обработку "застрявшего" транзакта, но продолжает следить за причиной, которая вызвала блокировку его обработки. Как только эта причина исчезает (например, освобождается занятое устройство), интерпретатор возвращается к обработке задержанного транзакта.

При продвижении транзактов через блоки могут происходить

события следующих 4-х основных типов:

1. создание или уничтожение транзакта;
2. изменение значения атрибута объекта;
3. задержка транзакта на некоторый промежуток системного времени;
4. изменение маршрута движения транзакта по блокам модели.

Программа на GPSS представляет собой последовательность предложений, изображающих объекты того или иного типа, а ее работу нужно представить как движение транзактов через блоки модели. Таким образом, в отличие от традиционных языков программирования, в GPSS по программе идет не управление, а транзакты.

## ***Синтаксис языка GPSS***

Для написания программ на GPSS используются латинские буквы [A-Z] и цифры [0-9], а также специальные символы. Кириллицу допустимо использовать только в комментариях.

Программы записывают в определенном фиксированном формате. Каждый блок записывается в отдельной записи, которая разбивается на ряд полей: поле метки, поле операции и семи полей операндов, обозначаемых буквами А, В, С, ..., G:

Модель должна начинаться управляющим оператором SIMULATE. Заканчивается модель оператором END.

Исходный текст модели система моделирования читает до оператора START, в котором задается значение счетчика для завершения моделирования. Поэтому все блоки модели должны быть размещены до оператора START.

## ***Создание и уничтожение транзактов. Блоки GENERATE и TERMINATE. Блок START.***

Блок **GENERATE** используется в программах в качестве источника транзактов, поступающих на вход следующего за ним блока. Операнды блока определяют режим работы генератора и атрибуты создаваемых транзактов. В полях А и В этого блока задается интервал системного времени между соседними моментами создания транзактов (интерпретация этих полей в блоках ADVANCE и GENERATE аналогична; к примеру, постоянный интервал времени между генерацией транзактов указывается в поле А, при этом поле В оставляется пустым), в поле С - момент выхода первого транзакта, и в поле D - число транзактов, которые должны быть созданы блоком. Если D - пусто, то блок генерирует транзакты до тех пор, пока не будет остановлено моделирование. Поля E, F, G служат для задания атрибутов генерируемых транзактов.

Пример 1:

**A,B,C, D, E, F, G**

**-­­­­­­­­­­­­­­­­­­­­­­­­­­­­­­­­­­­­­**

**GENERATE 10,,150,10000,64,25,H**

Данный блок генерирует транзакты каждые 10 единиц времени, начиная с момента 150. Всего будет создано 10000 транзактов, каждый из которых имеет приоритет 64 и 25 параметров формата в полуслово.

Замечания:

1. После создания транзакта все параметры имеют нулевое значение. Обозначение параметров P1, P2, Р3 и т.д.
2. Каждый транзакт имеет стандартный числовой атрибут - "отметка времени", в которой блок GENERATE заносит время входа транзакта в систему.
3. Момент появления следующего транзакта вычисляется после выхода предыдущего из блока GENERATE. Выходом транзакта считается его вход в следующий блок. Если он по какой-либо причине невозможен, то транзакт не считается вышедшим из блока GENERATE, что может приводить к искажению временных характеристик потока транзактов.

Для обеспечения выхода транзакта из блока GENERATE за ним помещают блок ADVANCE 0 (пример 1). В этом случае все транзакты при невозможности перемещаться далее по модели скапливаются в этом блоке. Кроме блока ADVANCE 0 возможно использование некоторых других блоков.

Блок **TERMINATE** служит для удаления из модели попадающих на него транзактов. Если в поле А этого блока указано какое-либо число, то интерпретатор уменьшает на это число значение глобального "счетчика завершений", служащего для определения конца моделирования (пример 2). Моделирование прекращается, когда его значение станет равным 0. Начальное значение этого счетчика определяется операндом А блока START.

Моделирование будет завершено по истечении 4000 тактов модельного времени. Блоком GENERATE в 4000-м такте сгенерируется транзакт, который далее поступит на блок TERMINATE с операндом А равным 1. В результате транзакт будет выведен из модели, а из счетчика завершений будет вычтена единица. Начальное значение счетчика завершений равно 1, т.к. операнд А блока START равен 1. После вычитания из него значения операнда А блока TERMINATE счетчик завершений примет нулевое значение и моделирование завершится.

В модели может быть любое количество блоков TERMINATE

## ***Задержка транзактов. Блок ADVANCE.***

Удобным средством моделирования различного рода обслуживания, длящегося некоторый промежуток времени, является задержка транзакта, которая выполняется в GPSS с помощью блока ADVANCE.

Поля А и В блока ADVANCE служат для задания времени, на которое этот блок задерживает входящий в него транзакт. Постоянное время задержки указывается в поле А, при этом поле В оставляется пустым.

## ***Блоки SEIZE и RELEASE.***

В результате входа транзакта в блок SEIZE указанное в нем устройство становится занятым данным транзактом и остается в этом состоянии до тех пор, пока этот же самый транзакт не пройдет через блок RELEASE, указывающий на это устройство. Один транзакт может занять любое число устройств, но каждое устройство в любой момент времени может быть занято лишь одним транзактом.

## ***Блоки QUEUE и DEPART.***

Одним из важных показателей качества функционирования систем массового обслуживания являются возникающие в процессе работы очереди заявок к тем или иным устройствам. Основными характеристиками очереди являются ее средняя и максимальная длина, а также среднее время пребывания заявки в очереди.

Для сбора статистических данных об очередях в языке GPSS предусмотрены блоки QUEUE и DEPART. Блок QUEUE фиксирует вход транзакта в очередь, а блок DEPART - выход из очереди. Один транзакт может входить в несколько блоков QUEUE, присутствуя, таким образом, одновременно в нескольких очередях. При входе транзакта в блок QUEUE интерпретатор увеличивает длину очереди, указанной в поле А этого блока, на число, записанное в поле В. Если это число опущено, длина очереди увеличивается на 1.

## ***Статистика (содержимое файла examp.lis).***

Большая часть необходимой статистики собирается в GPSS автоматически и не требует поэтому дополнительных усилий разработчика модели.

Для всех выполняемых блоков, составляющих модель, выводится номер блока, количество транзактов, задержанных в блоке на момент завершения моделирования, и общее количество транзактов, прошедших через блок. Эти данные помогают собрать часть статистики и проанализировать работу модели, что особенно полезно при отладке модели.

# Задание на работу

ЦП через время Т1 генерирует заявки к ОП. ОП обрабатывает каждую заявку в течение времени Т2. Промоделировать работу системы пока не будет обработано N транзактов; собрать статистику по очереди заявок на обработку в памяти и по функционированию ОП; определить, как долго продолжалось моделирование.

# Дано

Т1=5+/-1 тактов,

Т2=7 тактов,

N=40 транзактов

Работа центрального процессора (ЦП) и оперативной памяти (ОП)

# Код программы

SIMULATE

GENERATE 5,1

QUEUE AAA

SEIZE MEM

DEPART AAA

ADVANCE 7

RELEASE MEM

TERMINATE 1

START 40

END

# Блок схема модели

GENERATE 5,1

QUEUE

AAA

SEIZE

DEPART

ADVANCE 7

RELEASE

TERMINATE

1

AAA

MEM

MEM

# Статистический отчёт

**Результат моделирования с пронумерованными блоками:**

Student GPSS/H Release 3.70 (PR221) 12 Nov 2019 07:45:32 File: Z:\VB\GPSSH\lr1.gps

Line# Stmt# If Do Block# \*Loc Operation A,B,C,D,E,F,G Comments

1 1 SIMULATE

2 2 1 GENERATE 5,1

3 3 2 QUEUE AAA

4 4 3 SEIZE MEM

5 5 4 DEPART AAA

6 6 5 ADVANCE 7

7 7 6 RELEASE MEM

8 8 7 TERMINATE 1

9 9

10 10 START 40

11 11 END

**Словарь сущностей – все сущности, встречающиеся в модели и их номера:**

Entity Dictionary (in ascending order by entity number; "\*" => value conflict.)

Facilities: 1=MEM

Queues: 1=AAA

**MEM – имя устройства модели ; AAA – имя очереди системы. Многоканальные устройства отсутствуют.**

**Сведения о сущностях модели (Имя блока, назначение сущности, номера строк, в которых она встречается):**

Symbol Value EQU Defns Context References by Statement Number

MEM 1 Facility 4 7

AAA 1 Queue 3 5

**Symbol – имя сущности.**

**Value – присвоенное системой числовое значение.**

**Context – назначение (Тип) сущности.**

**References by Statement Number - номера строк, в которых она встречается**

**В данной модели используется 2 сущности – сущность АAA типа «очередь» (Queue) и сущность MEM типа «устройство» (Facility), которые встречаются по 2 раза в программе.**

**Требования по памяти в байтах:**

Storage Requirements (Bytes)

Compiled Code: 206

Compiled Data: 40

Miscellaneous: 0

Entities: 368

Common: 10000

-----------------------

Total: 10614

**Всего на обработку модели было затрачено 10614 байт**

**Размер модели:**

GPSS/H Model Size:

Control Statements 3

Blocks 7

**В данной модели у нас присутствует 7 блоков и 3 управляющих оператора (SIMULATE, START, END).**

**Начало моделирования:**

Simulation begins.

Relative Clock: 284.5406 Absolute Clock: 284.5406

**Начальное время моделирования принимаем нулевым, поскольку операнд С блока GENERATE не задан.**

|  |  |  |
| --- | --- | --- |
| Номер(имя)блока | Количество транзактов в блоке на текущий момент | Всего прошло транзактов |

Block Current Total

1 56

2 16 56

3 40

4 40

5 40

6 40

7 40

**Все 40 требуемых транзактов прошли через 7 блоков. На момент окончания моделирования в очереди (блок Queue) осталось 16 блоков, которые были созданы моделью, но уже не успели быть обработаны устройствами, поскольку моделирование было завершено после обработки 40-го транзакта.**

**Статистика по устройствам:**

--Avg-Util-During--

Facility Total Avail Unavl Entries Average Current Percent Seizing Preempting

Time Time Time Time/Xact Status Avail Xact Xact

MEM 0.984 40 7.000 AVAIL

**Имя или номер устройства (Facility) – MEM.**

**Производительность в долях к общему времени (Total Time) – 0.984**

**Количество транзактов, прошедших через устройство\число входов (Entries) –40.**

**Среднее время на одно занятие устройства\Среднее время обработки одного транзакта (Average Time\Unit) – 7.000**

**По условию стоит точная величина времени обработки транзакта - 7 тактов, соответственно, среднее время обработки будет постоянным для каждого транзакта и имеет целую величину. Из показателя параметра Entries можно сделать вывод, что через устройство B прошли все требуемые 40 транзактов. На момент завершения моделирования. в устройстве отсутствуют транзакты.**

**Статистика по очередям:**

Queue Maximum Average Total Zero Percent Average $Average Qtable Current

Contents Contents Entries Entries Zeros Time/Unit Time/Unit Number Contents

AAA 16 7.561 56 1 1.8 38.416 39.114 16

**Имя или номер очереди (Queue) – AAA.**

**Максимальная длина очереди (Maximum contents) – 16 транзактов.**

**Общее количество транзактов, вошешдших в очередь (Total Entries) – 56.**

**Количество «нулевых» входов (Zero Entries) – 1.**

**Средняя длина очереди (Average Contents) ~ 7.561 транзактов.**

**Среднее время ожидания в очереди с учётом всех транзактов (Average Time\Unit) – 38.416**

**Среднее время ожидания в очереди без учёта «нулевых» входов ($Average Time\Unit) – 39.114**

**Текущее содержимое очереди – 16 транзактов**

**В очереди побыли все транзакты, которые были созданы в процессе моделирования. Максимальная длина очереди – 16 транзактов, которые были созданы моделью, но уже не успели быть обработаны устройствами, поскольку моделирование было завершено после обработки и удаления из модели 40-го транзакта. В другие моменты моделирования длина очереди была другой. Очередь образовывалась в результате того, что текущий транзакт еще не обработался, а следующий созданный уже мог поступить на обработку и встал в очередь для избежания конфликта и возникновения ошибок. Первый транзакт прошёл на обработку без очереди, из-за чего появился один так называемый «нулевой вход».**

Random Antithetic Initial Current Sample Chi-Square

Stream Variates Position Position Count Uniformity

1 OFF 100000 100057 57 N/A

**Информация о затраченной и свободной памяти:**

Status of Common Storage

7688 bytes available

2312 in use

2424 used (max)

**Указан объем памяти, остававшейся свободной при прогоне модели – 7688 байт, а также максимальный объем, использовавшийся при прогоне модели – 2424 байт.**

**Окончание моделирования. Указано время, затраченное на моделирование – это модальное время конца прогона. Всего на обработку всех требуемых транзактов и моделирование системы было затрачено время, примерно равное 285 тактам.**

Simulation complete. Absolute Clock: 284.5406

**Количество выполненных блоков, количество блоков в секунду и количество микросекунд за один блок:**

Total Block Executions: 312

Blocks / second: 18461538

Microseconds / Block: 0.05

**Потраченное на моделирование время:**

Elapsed Time Used (Sec)

Pass1: 0.00

Sym/Xref 0.00

Pass2: 0.00

Load/Ctrl: 0.00

Execution: 0.00

Output: 0.00

---------------------

Total: 0.00

**Как видно из таблицы, время, затраченное на моделирование всей системы, слишком мало, поскольку модель довольна проста.**

# Обратная задача

В процессе выполнения лабораторной работы было предложено промоделировать обратную задачу на основе данных, полученных из отчёта изначальной модели. Это поможет убедиться в правильности работы исходной модели.

**Задание на работу:**

ЦП через время Т1 генерирует заявки к ОП. ОП обрабатывает каждую заявку в течение времени Т2. Промоделировать работу системы в течение времени; собрать статистику по очереди заявок на обработку в памяти и по функционированию ОП; определить, сколько транзактов было.

**Дано:**

Т1=5+/-1 тактов,

Т2=7 тактов,

T3=285 тактов.

**Код программы:**

SIMULATE

GENERATE 5,1

QUEUE AAA

SEIZE MEM

DEPART AAA

ADVANCE 7

RELEASE MEM

TERMINATE

GENERATE 285

TERMINATE 1

START 1

END

**Результат работы:**

Simulation begins.

Relative Clock: 285.0000 Absolute Clock: 285.0000

Block Current Total

1 56

2 15 56

3 41

4 41

5 1 41

6 40

7 40

8 1

9 1

Всего за 285 тактов системой было создано 56 транзактов, все они успели побыть в очереди Queue AAA. 41 из транзактов попали на обработку в устройство B, 40 закончили обработку и были удалены из модели. 1 транзакт на момент завершения моделирования находился на обработке. Сравнивая данные из исходной и обратной задачи, можно сделать вывод, что количество обработанных транзактов совпадает и равно 40-ти. Исходная модель работает правильно.

# Вывод

В результате выполнения лабораторной работы была промоделирована система взаимодействия ЦП И ОП. В систему было введено 40 транзактов согласно условию варианта; была собрана статистика по очереди заявок на обработку в памяти и по функционированию ОП.

В результате анализа статистического отчёта было выявлено:

* Все 40 транзактов были обработаны моделью примерно за 285 тактов.
* Максимальная длина очереди составила 16 транзактов (на момент завершения моделирования и удаления из модели 40-го транзакта)
* Среднее время нахождения транзакта в очереди составила 38.416 единиц
* Среднее время обработки одного транзакта составляет 7 единиц, оно одинаково для каждого транзакта и определяется блоком Advance.
* Средняя длина очереди ~ 8 транзактов.

Решив обратную задачу с временем моделирования 285 тактов, мы получили совпадающее количество обработанных и удалённых транзактов – 40. Из этого можно сделать вывод, что исходная модель работает верно.

# Используемая литература

1. О.М. Брехов, Г. А. Звонарева, А.В. Корнеенкова; Имитационное моделирование. Учебное пособие. МАИ. Москва 2015

2. О.М.Брехов, Г.А.Звонарева, А.В.Корнеенкова, Лабораторные работы по курсу «Моделирование ЭВМ» Под редакцией д.т.н., профессора О.М.Брехова. Москва 2007