Министерство образования Республики Беларусь
Учреждение образования «Витебский государственный университет
имени П.М. Машерова»

Компьютерное моделирование систем

*Учебно-методический комплекс
«Компьютерное моделирование систем»*

Витебск
2011

СОДЕРЖАНИЕ

ВВЕДЕНИЕ

Курс «Компьютерное моделирование систем» предназначен для ознакомления студентов с основными положениями теории моделирования, видами математических моделей, базовыми концепциями и основными принципами компьютерного моделирования, планированием имитационных экспериментов, математическими методами анализа результатов моделирования. Изучение данного курса позволит сформировать понятие оптимального решения, основанное на анализе математических задач выбора в заданном множестве допустимых решений, удовлетворяющего тем или иным критериям оптимальности; а также получить представление об основных программных средствах, используемых при моделировании.

В результате изучения дисциплины студент должен

знать:

- основные понятия теории моделирования, принципы построения моделей;
- требования, предъявляемые к разработке математических моделей, способы оценки их качества;
- методологии и технологии машинного моделирования систем;
- методы решения линейных оптимизационных задач;
- приемы имитационного моделирования систем;
- инструментальные средства моделирования систем.

уметь:

- использовать основные математические схемы для моделирования систем;
- решать различные классы оптимизационных задач с использованием методов исследования операций;
- планировать имитационный эксперимент, анализировать результаты моделирования;
- применять статистические методы обработки экспериментальных данных, полученных по данным пассивных и активных экспериментов;
- использовать инструментальные системы и языки моделирования систем.

В соответствии с учебным планом программа предусматривает для изучения дисциплины всего – 114 часов; аудиторных – 52 часа; из них: лекции – 26 часов; лабораторные занятия – 26 часов.

Тема 1. Основные понятия моделирования

Модель – это идеальный образ или материальный прообраз системы (оригинала данной модели), подобный ей в конечном числе отношений, который в определенных условиях может заменять объект-оригинал, воспроизводя интересующие нас свойства и характеристики оригинала. При этом модель имеет существенные преимущества, облегчающие изучение соответствующего ей объекта, (наглядность, обозримость, доступность испытаний и пр.).

Замещение одного объекта другим с целью получения информации о важнейших свойствах объекта-оригинала с помощью объекта-модели называется моделированием. Модель, с помощью которой успешно достигается цель моделирования, называется адекватной этой цели.

Виды моделей

Всякая деятельность человека направлена на достижение какой-либо цели: *теоретической* и *практической*. Соответственно типам целей модели делятся на *познавательные* и *прагматические*.

Познавательные модели являются формой средством получения новых знаний. Если такая модель расходится с действительностью, то необходимо изменить модель в сторону приближения к реальности. Прагматические модели являются средством управления, способом представления образцово правильных действий или их результата.

В зависимости от способа реализации модели или от способа представления системы, познавательные модели подразделяются на: а) физические; б) математические.

Физические модели – это "материальные" модели, эквивалентные или подобные в той или иной степени оригиналу. В общем случае физические модели — это модели, процесс функционирования которых такой же, как у оригинала, имеет ту же или подобную физическую природу. Виды физических моделей: натуральные; квазинатуральные; масштабные; аналоговые;

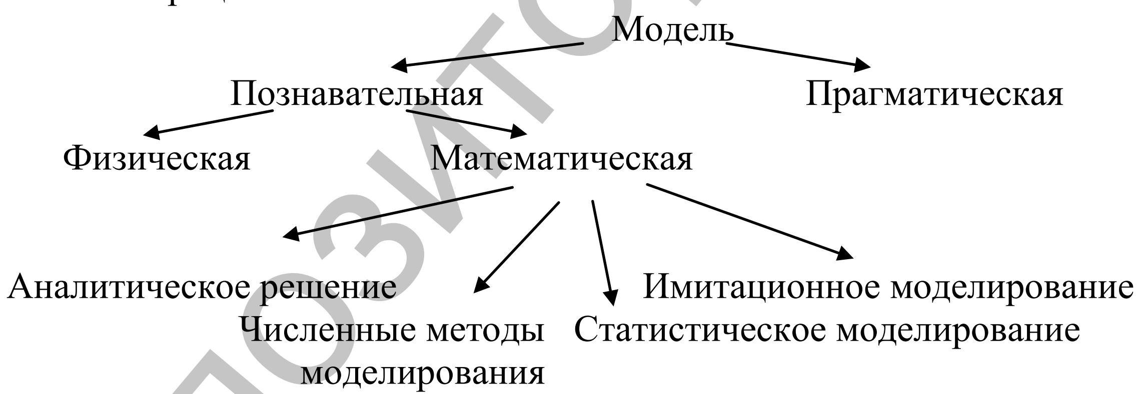
Натуральные модели — это реальные исследуемые системы (макеты, опытные образцы). Имеют полную адекватность (соответствия) с системой оригиналом, но дороги.

Квазинатуральные модели — совокупность натуральных и математических моделей. Этот вид используется тогда, когда модель части системы не может быть математической из-за сложности её описания (модель человека оператора) или когда часть системы должна быть исследована во взаимодействии с другими частями, но их ещё не существует или их включение очень дорого (вычислительные полигоны, АСУ).

Масштабная модель — это система той же физической природы, что и оригинал, но отличается от него масштабами. Методологической основой масштабного моделирования является теория подобия. При проектировании ВС масштабные модели могут использоваться для анализа вариантов компоновочных решений.

Аналоговыми моделями называют системы, имеющие физическую природу, отличающуюся от оригинала, но сходные с оригиналом процессы функционирования. Для создания аналоговой модели требуется наличие математического описания изучаемой системы. В качестве аналоговых моделей используются механические, гидравлические, пневматические и электрические системы. Аналоговое моделирование использует при исследовании средства ВТ на уровне логических элементов и электрических цепей, а так же на системном уровне, когда функционирование системы описывается, например, дифференциальными или алгебраическими уравнениями.

Математические модели – это "абстрактные" модели, представляющие собой формализованное описание изучаемой системы с помощью абстрактного языка, в частности, с помощью математических соотношений, отображающих процесс функционирования системы. Для составления математических моделей можно использовать любые математические средства — алгебраическое, дифференциальное, интегральное исчисления, теорию множеств, теорию алгоритмов и т.д. По существу вся математика создана для составления и исследования моделей объектов и процессов.



Аналитической моделью называется такое формализованное описание системы, которое позволяет получить решение уравнения в явном виде, используя известный математический аппарат. Для *аналитического моделирования* характерно то, что процессы функционирования элементов системы записываются в виде некоторых математических соотношений (алгебраических, дифференциальных) или логических условий. Однако математическую модель можно построить только для сравнительно простых систем. Аналитические модели более грубы, учитывают меньшее число факторов, всегда требуют каких-то допущений и упрощений. Зато результаты расчета по ним легче обозримы, отчетливее отображают присущие явлению основные закономерности.

Численные методы моделирования – преобразование модели к уравнениям, решение которых возможно методами вычислительной математики когда не удается получить в явном виде решения.. Однако численные методы не дают точных решений, но позволяют задать точность решения.

Статистические модели, по сравнению с аналитическими, более точны и подробны, не требуют столь грубых допущений, позволяют учесть большее число факторов. Однако существенный недостаток статистического моделирования — это трудоемкость построения модели и частный характер получаемых результатов.

Имитационная модель – это совокупность описания системы и внешних воздействий, алгоритмов функционирования системы или правил изменения состояния системы под влиянием внешних и внутренних возмущений. *Имитационное моделирование* – это вид моделирования, для которого характерно воспроизведение на ЭВМ (имитация) процесса функционирования исследуемой сложной системы. При этом имитируются элементарные явления, составляющие процесс, с сохранением их логической структуры и последовательности протекания во времени. Это позволяет получить информацию о состоянии системы в заданные моменты времени. Методы имитационного моделирования базируются на теории статистических испытаний.

Теория моделирования является основной составляющей общей теории систем – системологии, где в качестве главного принципа постулируются осуществимые модели: система представима конечным множеством моделей, каждая из которых отражает определённую грань её сущности. Познание любой системы сводится по существу к созданию её модели. Обычно процесс разработки сложной системы осуществляется итерационно с использованием моделирования проектных решений. Если характеристики не удовлетворяют предъявленным требованиям, то по результатам анализа производят корректировку проекта, затем снова проводят моделирование. При анализе действующих систем с помощью моделирования определяют границы работоспособности системы, выполняют имитацию экспериментальных условий, которые могут возникнуть в процессе функционирования системы. Искусственное создание таких условий на действительной системе затруднено и может привести к катастрофическим последствиям.

<u>Компьютерное моделирование</u> используется во всех областях математического моделирования систем. А в некоторых областях, например, в задачах исследования операций, в задачах оптимизации, прогнозирования, планирования и управления компьютерное моделирование становится основным инструментом исследования.

Тема 2. Исследование операций. Линейные оптимизационные модели

2.1. Основные понятия исследования операций

Первые публикации по исследованию операций относятся к 1939-1940 гг, в которых методы применены для решения военных задач, в частности для анализа и исследования военных операций. Отсюда и пошло название дисциплины. Позднее принципы и методы исследования операций стали применяться в сфере промышленно-финансового управления.

Операцией называется управляемое целенаправленное действие.

Исследование операций – наука, занимающаяся разработкой и практическим применением методов наиболее оптимального управления организационными системами.

Предмет исследования операций – системы или организации, которые состоят из большого числа взаимодействующих между собой подразделений. Причем интересы подразделений не всегда согласуются между собой и могут быть противоположными.

Решение – это выбор из ряда возможностей. Решение можно принимать интуитивно, на основе опыта и здравого смысла. Однако, чем сложнее и масштабнее планируемое мероприятие, тем менее допустимы в нем «волевые» решения и тем важнее становятся научные методы, позволяющие заранее оценить последствия каждого решения, заранее отбросить недопустимые варианты и рекомендовать наиболее удачные.

Цель исследования операций – количественное обоснование принимаемых решений по управлению организацией (системой).

Решение, наиболее выгодное всей организации называется *оптимальным.* Решение, выгодное одному или нескольким подразделениям – *субоптимальное.* Надо заметить, оптимальное решение для всей организации необязательно оптимально для входящих в нее подразделений.

Те параметры, совокупность которых образует решение, называются элементами решения. В качестве элементов могут фигурировать переменные, функции, векторы и т.д.

Кроме элементов решения, которые мы можем изменять, в любой задаче имеются еще условия, которые фиксированы с самого начала и не могут быть нарушены. В своей совокупности они формируют ограничения, которые 1) устанавливают зависимости между переменными и 2) граничные условия, которые показывают, в каких пределах могут быть значения искомых переменных в оптимальном решении.

Чтобы сравнить между собой по эффективности разные решения, нужно иметь какой-то качественный критерий, показатель эффективности или целевую функцию. Этот показатель выбирается так, чтобы он отражал целевую направленность операции. Лучшим будет считаться то решение, которое в максимальной степени способствует достижению поставленной цели. При этом возможны три вида назначения целевой функции: максимизация, минимизация и назначение заданного значения.

Таким образом, говоря более строго, **исследование операций –** это область математики, разрабатывающая теорию и численные методы решения многомерных экстремальных задач с ограничениями.

Основные этапы операционного исследования

Любое операционное исследование, при всем возможном многообразии конкретных работ по исследованию операций, проходит последовательно следующие этапы:

1) постановка задачи;
2) формализация задачи;
3) нахождение метода решения;
4) проверка и корректировка модели;
5) реализация найденного решения на практике.

Типичные классы задач

По содержательной постановке наиболее часто возникают следующие типичные классы задач:

- *Задачи управления запасами*. Такие задачи обладают следующей особенностью: с увеличением запасов увеличиваются расходы на хранение, но уменьшаются потери из-за возможной их нехватки. (Логистика)
- *Задачи распределения ресурсов*. Такие задачи возникают, когда существует определенный набор работ, которые необходимо выполнить, а наличных ресурсов для выполнения работы должным образом не хватает.
- *Задачи ремонта и оборудования*.
- *Задачи массового обслуживания* рассматривают вопросы образования и функционирования очередей, с которыми приходится сталкиваться в повседневной практике, при управлении технологическими процессами, в линиях связи и компьютерных сетях.
- *Задачи календарного планирования или составления расписания*.
- *Задачи сетевого планирования и управления*. Здесь рассматриваются соотношения между сроком окончания крупного комплекса операций и моментами начала всех операций комплекса.
- *Задачи выбора маршрута или сетевые задачи*. Чаще всего встречаются при исследовании разнообразных процессов на транспорте и в системах связи (компьютерные сети).

Часто задачи оказываются комбинированными.

Прямые и обратные задачи исследования операций

Задачи ИСО делятся на две категории: 1) прямые и 2) обратные.

Прямые задачи отвечают на вопрос: что будет, если в заданных условиях мы примем решение $x \in X$? Например, будет равен при данном решении x выбранный показатель f? В этом случае строится математическая модель, позволяющая выразить один или несколько показателей эффективности через заданные условия и элементы решения.

Обратные задачи отвечают на вопрос: как выбрать решение x для того, чтобы обратить показатель эффективности (критерий оптимальности) – целевую функцию f в максимум? Одним из методов решения обратных задач является линейное программирование.

2.2. Линейное программирование

Линейное программирование – область математики, разрабатывающая методы решения задач, нахождения экстремума (min и max) линейной функции многих переменных при наличии линейных ограничений, т.е. равенств или неравенств, связывающих эти переменные.

Основной (стандартной) задачей линейного программирования называется задача нахождения минимума линейной целевой функции (линейной формы) вида:

$$f(x) = \sum_{j=1}^{n} c_j x_j = c_1 x_1 + c_2 x_2 + \ldots + c_n x_n$$

при условиях

$$\sum_{j=1}^{n} a_{ij} x_j \geqslant b_i \quad (i = 1,\ 2,\ \ldots,\ m),$$

$$x_i \geqslant 0 \quad (i = 1,\ 2,\ \ldots,\ m).$$

Задача линейного программирования будет иметь канонический вид, если в основной задаче вместо первой системы неравенств имеет место система уравнений:

$$\sum_{j=1}^{n} a_{ij} x_j = b_i \quad (i = 1,\ 2,\ \ldots,\ m),$$

Основную задачу можно свести к канонической форме.

Задачи линейного программирования наиболее общего вида (задачи со смешанными ограничениями: равенствами и неравенствами, наличием переменных, свободных от ограничений) могут быть приведены к эквивалентным (имеющим то же множество решений) заменами переменных и заменой равенств на пару неравенств.

Легко заметить, что задачу нахождения максимума можно можно заменить задачей нахождения минимума, взяв коэффициенты cс обратным знаком.

Для решения задачи надо составить её экономико-математическую модель. Общая схема формирования модели:

1) выбор переменных;

2) выражение взаимосвязей переменных модели в виде уравнений и неравенств; эти соотношения образуют ограничений задачи;

3) построение целевой функции (критерий оптимальности);

4) математическая формулировка задачи как задачи отыскания экстремума целевой функции при условии выполнения ограничений, накладываемых на переменные.

2.3. Геометрический метод решения задачи ЛП

Геометрический метод решения задач линейного программирования целесообразно использовать для решения задач с двумя переменными, записанными в общей форме, а также для задач со многими переменными при условии, что в их канонической записи содержатся не более двух свободных переменных, т.е. число неизвестных n и число линейно-независимых уравнений m связаны соотношением n-m≤2.

Рассмотрим задачу ЛП в стандартной форме записи:

$$\max f(x_1, x_2, \ldots, x_n) = \sum_{j=1}^{n} c_j x_j$$

$$\begin{cases} \sum\limits_{j=1}^{n} a_{ij} x_j \le b_i \ , & i = 1, 2, \ldots m, \\ x_j \ge 0, & j = 1, 2, \ldots, n. \end{cases}$$

Положим $n=2$, т.е. рассмотрим эту задачу на плоскости. Пусть система неравенств совместна (имеет хотя бы одно решение).

Каждое неравенство этой системы геометрически определяет полуплоскость с граничной прямой $a_{i1}\,x_1 + a_{i2}\,x_2 = b_i$, $i=1,2,\ldots m$. Условия неотрицательности определяют полуплоскости, соответственно, с граничными прямыми x1=0, x2=0. Система совместна, поэтому полуплоскости, как выпуклые множества, пересекаясь, образуют общую часть, которая является выпуклым множеством и представляет собой совокупность точек, координаты каждой из которых являются решением данной системы. Совокупность этих точек называют многоугольником решений. Он может быть точкой, отрезком, лучом, многоугольником, неограниченной многоугольной областью.

Таким образом, геометрически задача линейного программирования (ЗЛП) представляет собой отыскание такой точки многоугольника решений, координаты которой доставляют линейной функции цели максимальное (минимальное) значение, причем допустимыми решениями являются все точки многоугольника решений.

Для нахождения экстремального значения целевой функции при графическом решении задач ЛП используют вектор–градиент, координаты которого являются частными производными целевой функции, т.е.

$$\nabla = (\frac{\partial f}{\partial x_1} = c_1, \frac{\partial f}{\partial x_2} = c_2) \cdot$$

Этот вектор показывает направление наискорейшего изменения целевой функции. Прямая $c_1 x_1 + c_2 x_2 = f(\bar{x}_0)$, перпендикулярная вектору–градиенту, является линией уровня целевой функции. В любой точке линии уровня целевая функция принимает одно и то же значение. Приравняем целевую функцию постоянной величине *“а”*. Меняя значение *“а”*, получим семейство параллельных прямых, каждая из которых является линией уровня.

Важное свойство линии уровня линейной функции состоит в том, что при параллельном смещении линии в одну сторону уровень только возрастает, а при смещении в другую сторону – убывает.

С геометрической точки зрения в задаче линейного программирования ищется такая угловая точка или набор точек из допустимого множества решений, на котором достигается самая верхняя (нижняя) линия уровня, расположенная дальше (ближе) остальных в направлении наискорейшего роста.

Графический метод решения ЗЛП состоит из следующих этапов.

1. Строится многоугольная область допустимых решений ЗЛП – ОДР,

2. Строится вектор-градиент ЦФ в какой-нибудь точке X_0 принадлежащей ОДР – $\nabla = (C_1, C_2)$.

3. Линия уровня $C_1x_1+C_2x_2 = a$ (*a*–постоянная величина) - прямая, перпендикулярная вектору-градиенту ∇ – передвигается в направлении этого вектора в случае максимизации $f(x_1,x_2)$ до тех пор, пока не покинет пределов ОДР. Предельная точка (или точки) области при этом движении и является точкой максимума $f(x_1,x_2)$.

4. Для нахождения ее координат достаточно решить два уравнения прямых, получаемых из соответствующих ограничений и дающих в пересечении точку максимума. Значение $f(x_1,x_2)$, найденное в получаемой точке, является максимальным.

При минимизации $f(x_1,x_2)$ линия уровня перемещается в направлении, противоположном вектору-градиенту. Если прямая при своем движении не покидает ОДР, то соответствующий максимум или минимум $f(x_1,x_2)$ не существует.

Если линия уровня параллельна какому-либо функциональному ограничению задачи, то оптимальное значение ЦФ будет достигаться в любой точке этого ограничения, лежащей между двумя оптимальными угловыми точками, и, соответственно, любая из этих точек является оптимальным решением ЗЛП.

2.4. Контрольные вопросы по теме

1. Что называется операцией?
2. Какова цель исследования операций?
3. Что называется решением?
4. Какое решение называют оптимальным?
5. Какие категории задач исследования операций Вы знаете?
6. На какой вопрос отвечают обратные задачи исследования операций?
7. Какие формы записи задачи линейного программирования вы знаете?
8. В чем состоит основная задача линейного программирования?
9. Что представляет собой система ограничений задачи ЛП?
10. Что представляет собой целевая функция задачи ЛП?
11. В каких случаях можно использовать графический метод решения задачи ЛП?
12. Перечислите этапы графического метода решения задачи ЛП.

Лабораторная работа № 1. Тема: решение задачи линейного программирования графическим методом

Задание. Решить задачу линейного программирования графическим методом. Проверить решение с помощью надстройки «Поиск решения» Excel.

Пример 1. Предприятие выпускает два вида продукции A_1, A_2. Для их производства требуется затратить такие производственные факторы, как сырье, физический труд и управленческий труд. Затраты ресурсов на единицу продукции каждого вида, ежедневный объем имеющихся ресурсов, а также прибыль на единицу продукции приведены в таблице. Составить план ежедневного выпуска продукции, при котором получаемая прибыль будет максимальной.

Тип ресурсов	Затраты на ед.прод.		Объем ресурсов
	A_1	A_2	
Сырье (кг)	8	25	800
Физический труд (чел.-ч.)	8	5	640
Управленческий труд (чел.-ч.)	1	5	145
Прибыль на единицу продукции (т.р.)	80	70	

Прежде всего, составим математическую модель задачи.

Обозначим через x_1 и x_2 количество единиц продукции вида A_1 и A_2 соответственно, которое предприятие планирует выпустить за сутки. Тогда $80x_1$ – прибыль от реализации продукции вида A_1 и $70x_2$ – вида A_2, $f = 80x_1 + 70x_2$ – прибыль от реализации всей продукции. Отразим в модели ограничения на факторы. Имеем:

$8x_1$ – количество сырья, необходимое для изготовления продукции вида A_1,

$25x_2$ – количество сырья, необходимое для изготовления продукции вида A_2,

$8x_1 + 25x_2$ – всей продукции.

Так как в наличии имеется 800 кг сырья, то получаем:

$8x_1 + 25x_2 \le 800$.

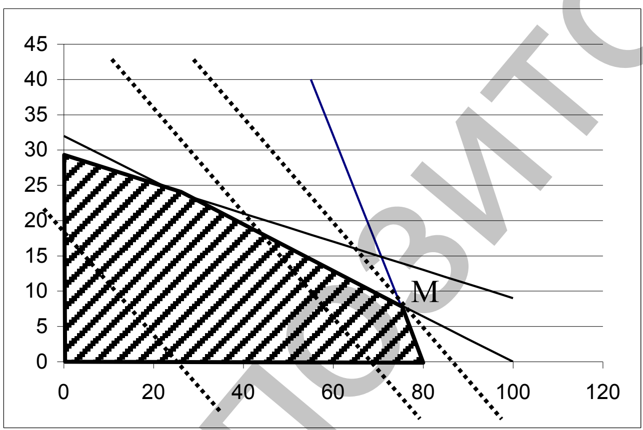
Затраты физического труда: $8x_1$ – количество человеко-часов, необходимое для производства продукции вида A_1, $5x_2$ – вида A_2, $8x_1 + 5x_2$ – всей продукции. Т.к. в течение суток можно использовать 640 чел.-ч. физического труда, то получим: $8x_1 + 5x_2 \le 640$. Аналогично имеем для затрат управленческого труда: $x_1 + 5x_2 \le 145$. Из физического смысла переменных следует, что они неотрицательны: продукция либо выпускается ($x_j > 0$), либо нет ($x_j = 0$). Производственная модель свелась к решению следующей задачи:

$f = 80x_1 + 70x_2 \to \max$,

$$\begin{cases} 8x_1 + 25x_2 \le 800, \\ 8x_1 + 5x_2 \le 640, \\ x_1 + 5x_2 \le 145, \\ x_1 \ge 0;\ x_2 \ge 0. \end{cases}$$

Решение графическим методом

Каждое из приведенных неравенств определяет в пространстве переменных x_1 и x_2 некоторое полупространство. На рисунке изображено множество планов задачи. Пунктирной линией изображено семейство прямых, отвечающих целевой функции. Найдем точку М, в которой линия уровня целевой функции в последний раз касается множества планов задачи. Из рисунка видно, что такой точкой является точка с координатами (75, 8). При этом максимальное значение целевой функции будет равно f_{max}=6560. Также значение x_1 и x_2 можно найти, решив систему уравнений



$$\begin{cases} 8x_1 + 25x_2 \le 800, \\ 8x_1 + 5x_2 \le 640. \end{cases}$$

Пример 2. Предприятие выпускает 2 вида продукции. Цена единицы 1 вида продукции – 25 000, 2 вида продукции – 50 000. Для изготовления продукции используются три вида сырья, запасы которого 37; 57,6 и 7 условных единиц. Нормы затрат каждого сырья на единицу продукции представлены в следующей таблице:

Продукция		Запасы сырья
1-й вид продукции	2-й вид продукции	
1,2	1,9	37
2,3	1,8	57,6
0,1	0,7	7

Требуется определить плановое количество выпускаемой продукции таким образом, чтобы стоимость произведенной продукции была максимальной (в индивидуальном задании искать max и min).

Составим математическую модель задачи.

Пусть продукция производится в количестве: 1-й вид – x_1 единиц, 2-й вид – x_2 единиц. Переменные x_1, x_2 должны быть неотрицательными и целыми: $x_1 \geq 0$, $x_2 \geq 0$. Тогда стоимость произведенной продукции выражается целевой функцией: $f(x_1,x_2)=25000\ x_1+50000x_2$, для которой необходимо найти максимум. При этом следует учесть ограничения по запасам сырья:

$1{,}2\ x_1+1{,}9\ x_2 \leq 37;$

$2{,}3\ x_1+1{,}8\ x_2 \leq 57{,}6;$

$0{,}1\ x_1+0{,}7\ x_2 \leq 7.$

Решение в Excel

Введем в рабочую таблицу исходные данные:

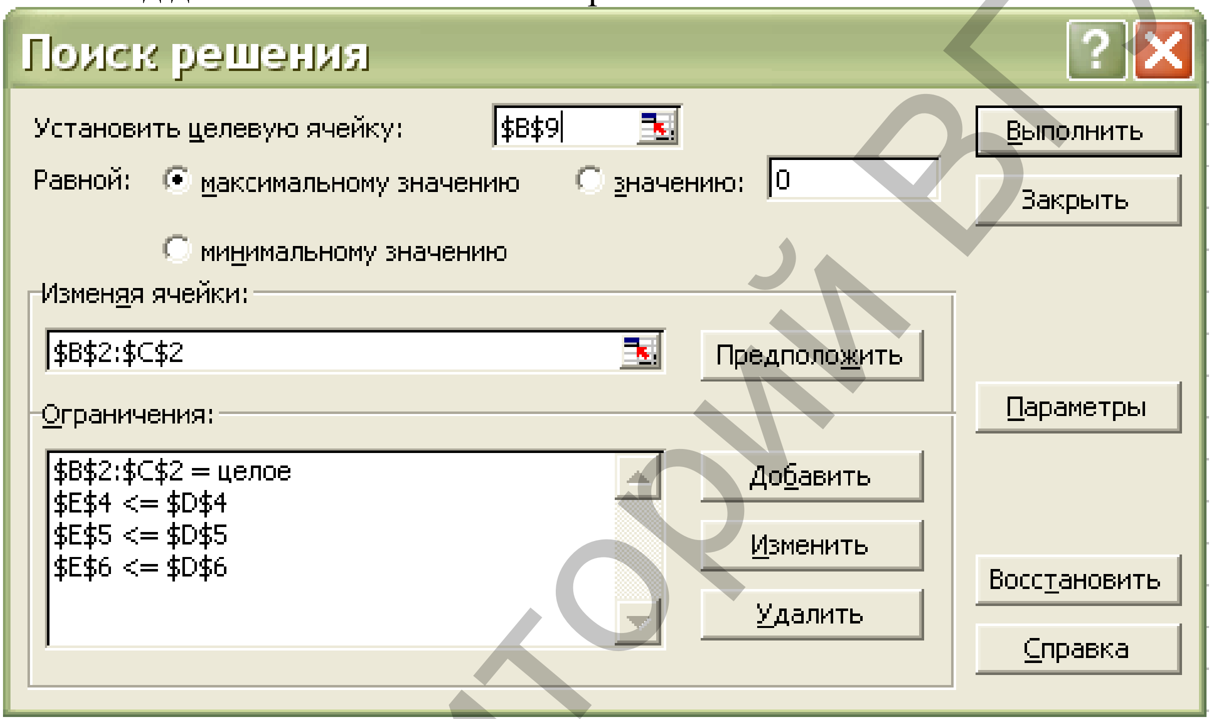
	A	B	C	D	E
1		**Продукт 1**	**Продукт 2**		
2	Искомые значения	0	0		
3	***Используемые ресурсы:***			***Имеющиеся запасы ресурсов***	***Расчет расхода ресурсов***
4	Сырье вида 1	1,2	1,9	37	
5	Сырье вида 2	2,3	1,8	57,6	
6	Сырье вида 3	0,1	0,7	7	
7	Цена одного изделия	25000	50000		
8					
9	***Целевая функция:***				

Поиск значений переменных x_1, x_2 будем вести соответственно в ячейках B2:C2. В ячейку B9 введем формулу для вычисления значения целевой функции (суммарной стоимости продукции): = СУММПРОИЗВ(B2:C2;B7:C7), а в ячейки E4:E6 – формулы для вычисления реального расхода сырья:

Я	Формула

чейка	
E 4	= СУММПРОИЗВ(B$2C$2;B4:C4)
E 5	= СУММПРОИЗВ(B$2C$2;B5:C5)
E 6	= СУММПРОИЗВ(B$2C$2;B6:C6)

Вид диалогового окна «Поиск решения»:



Результат решения задачи: *$x_1=19$, $x_2=7$, общая стоимость (значение целевой функции) равна 825000.*

Варианты заданий для самостоятельного выполнения

1. $$\begin{aligned} &-3x_1 + 14x_2 \le 78, \\ &5x_1 - 6x_2 \le 26, \\ &x_1 + 4x_2 \ge 26; \\ &f = 5x_1 + 7x_2 \to extr \end{aligned}$$

2. $$\begin{aligned} &11x_1 - 3x_2 \ge 24, \\ &9x_1 + 4x_2 \le 110, \\ &-2x_1 + 7x_2 \ge 15; \\ &f = 9x_1 + 2x_2 \to extr \end{aligned}$$

3. $$\begin{aligned} &-4x_1 + 5x_2 \le 29, \\ &3x_1 - x_2 \le 14, \\ &5x_1 + 2x_2 \ge 38; \\ &f = 3x_1 + 2x_2 \to extr \end{aligned}$$

4. $$\begin{aligned} &2x_1 - x_2 \ge 4, \\ &x_1 + 3x_2 \le 37, \\ &-4x_1 + 9x_2 \ge 20; \\ &f = 4x_1 + 3x_2 \to extr \end{aligned}$$

5. $$\begin{aligned} &10x_1 - x_2 \ge 57, \\ &2x_1 + 3x_2 \le 53, \\ &6x_1 - 7x_2 \le 15; \\ &f = 5x_1 + x_2 \to extr \end{aligned}$$

6. $$\begin{aligned} &4x_1 - x_2 \ge 6, \\ &9x_1 + 8x_2 \le 157, \\ &-3x_1 + 11x_2 \ge 16; \\ &f = x_1 + x_2 \to extr \end{aligned}$$

7. $$\begin{cases} -x_1 + x_2 \le 3, \\ 5x_1 + 3x_2 \le 97, \\ x_1 + 7x_2 \ge 77; \end{cases}$$
$$f = 7x_1 + 2x_2 \to extr$$

8. $$\begin{cases} 3x_1 - x_2 \ge 9, \\ 2x_1 + 3x_2 \le 50, \\ -x_1 + 4x_2 \ge 19; \end{cases}$$
$$f = 6x_1 + x_2 \to extr$$

9. $$\begin{cases} x_1 + 4x_2 \le 53, \\ x_1 - x_2 \le 3, \\ 7x_1 + 3x_2 \ge 71; \end{cases}$$
$$f = x_1 + 7x_2 \to extr$$

10. $$\begin{cases} 6x_1 - 5x_2 \ge 17, \\ x_1 + 2x_2 \le 34, \\ -4x_1 + 9x_2 \ge 17; \end{cases}$$
$$f = x_1 + 9x_2 \to extr$$

11. $$\begin{cases} -3x_1 + 14x_2 \le 78, \\ 5x_1 - 6x_2 \le 26, \\ x_1 + 4x_2 \ge 26; \end{cases}$$
$$f = x_1 + 8x_2 \to extr$$

12. $$\begin{cases} 11x_1 - 3x_2 \ge 24, \\ 9x_1 + 4x_2 \le 110, \\ -2x_1 + 7x_2 \ge 15; \end{cases}$$
$$f = 7x_1 + x_2 \to extr$$

13. $$\begin{cases} -4x_1 + 5x_2 \le 29, \\ 3x_1 - x_2 \le 14, \\ 5x_1 + 2x_2 \ge 38; \end{cases}$$
$$f = 3x_1 + x_2 \to extr$$

14. $$\begin{cases} 2x_1 - x_2 \ge 4, \\ x_1 + 3x_2 \le 37, \\ -4x_1 + 9x_2 \ge 20; \end{cases}$$
$$f = x_1 + 3x_2 \to extr$$

15. $$\begin{cases} 10x_1 - x_2 \ge 57, \\ 2x_1 + 3x_2 \le 53, \\ 6x_1 - 7x_2 \le 15; \end{cases}$$
$$f = 2x_1 + 3x_2 \to extr$$

16. $$\begin{cases} 4x_1 - x_2 \ge 6, \\ 9x_1 + 8x_2 \le 157, \\ -3x_1 + 11x_2 \ge 16; \end{cases}$$
$$f = 8x_1 + 5x_2 \to extr$$

2.5. Симплекс-метод решения задачи ЛП

Одним из универсальных методов решения задач ЛП (линейного программирования) является симплекс-метод, называемый также методом последовательного улучшения плана, так как он основан на последовательном улучшении первоначального плана при помощи ряда последовательных преобразований (итераций). Он состоит из двух этапов:

1) отыскания какого-нибудь начального допустимого решения (опорного плана);

2) перехода от этого решения к новому допустимому решению, для которого целевая функция изменяется в нужном направлении, т.е. убывает в задаче на минимум или возрастает в задаче на максимум.

В результате оптимальное решение находят за конечное число шагов. Алгоритм симплекс-метода позволяет также установить является ли задача ЛП разрешимой.

Выбор конкретной вычислительной процедуры осуществляется после приведения исходной ЗЛП к каноническому виду (КЗЛП):

$$\max f(x_1, x_2, \ldots, x_n) = \sum c_j x_j$$

$$\sum a_{ij} x_j = b_i, i = 1,2,\ldots,m$$

$$x_j \geq 0, j = 1,2,\ldots,n; b_i \geq 0, i = 1,2,\ldots,m$$

В теории линейного программирования показано, что оптимальное решение ЗЛП связано с угловыми (крайними) точками многогранника решений, которым отвечают опорные планы (неотрицательные базисные решения системы уравнений КЗЛП). Приведя модель к предпочтительному виду, составляют симплексную таблицу.

Признак оптимальности плана

Пусть задача решается на max. Если для некоторого плана все оценки индексной строки неотрицательны, то такой план оптимален. Если задача решается на min и оценки индексной строки неположительны (не считая значения ЦФ), то план – оптимален.

Переход к не худшему опорному плану

1. Среди отрицательных оценок индексной строки находят максимальную по абсолютной величине. Столбец j_0 называется разрешающим. Вычисляют симплексные отношения $\theta = \dfrac{b_i}{a_{ij_0}};\ a_{ij_0} > 0$, $i = \overline{1,m}$. Среди них находят наименьшее. Соответствующая строка i_0 называется разрешающей. На пересечении разрешающей строки и разрешающего столбца находится разрешающий элемент.

2. Для получения нового опорного плана составляют новую таблицу по правилам:

- элементы строки i_0 новой таблицы равны соответствующим элементам старой таблицы, деленной на разрешающий элемент. Оставшиеся элементы столбца j_0 равны 0;
- для получения всех остальных элементов таблицы, включая индексную строку, надо из соответствующего элемента прежней таблицы вычесть произведение элемента разрешающей строки на элемент разрешающего столбца, разделенное на разрешающий элемент.

Процесс решения продолжают либо до получения оптимального плана, либо до установления неограниченности ЦФ. Если среди симплекс-разностей (оценок) оптимального плана нулевые только оценки, соответствующие базисным векторам, то это говорит о единственности оптимального плана. Если же нулевая оценка соответствует вектору, не входящему, то в общем случае это означает, что оптимальный план не единственный.

Примечание. Для использования приведенной процедуры к минимизации линейной функции f $(x_1,x_2,\ldots, x_n)$ следует искать максимум

- f (x_1,x_2,…, x_n), затем полученный максимум взять с противоположным знаком. Оптимальное решение то же.

Признак неразрешимости или признак неограниченности целевой функции

Если в разрешающем столбце нет ни одного строго положительного элемента, то задача не имеет оптимального решения, а целевая функция на множестве допустимых планов неограничена сверху (в задаче на максимум) или неограничена снизу (в задаче на минимум).

Признак альтернативного оптимума (бесконечного множества оптимальных планов

Если в индексной строке последней (оптимальной) таблицы оценка какой-либо небазисной переменной равна нулю, то задача, имеет бесконечное множество оптимальных решений.

2.6. Двойственная задача ЛП

Двойственная задача – это вспомогательная задача ЛП, формулируемая с помощью определенных правил непосредственно из условий исходной, или прямой задачи.

Пара симметричных задач имеет вид:

$$
\begin{array}{l|l}
\max f = \sum_{j=1}^{n} c_j x_j & \min f = \sum_{i=1}^{m} b_i y_i \\
\sum_{j=1}^{n} a_{ij} x_j \le b_i \ (i = \overline{1,m}) & \sum_{i=1}^{m} a_{ij} y_i \ge cj \ (j = \overline{1,n}) \\
x_j \ge 0 \ (j = \overline{1,n}) & y_i \ge 0 \ (i = \overline{1,m})
\end{array}
$$

Двойственная задача получается путем симметричного преобразования условий прямой задачи в соответствии со следующими правилами:

1) если прямая задача на максимум, то двойственная к ней – на минимум, и наоборот;
2) каждому ограничению прямой задачи соответствует переменная двойственной задачи;
3) каждой переменной прямой задачи соответствует ограничение двойственной задачи;
4) коэффициенты целевой функции прямой задачи являются свободными членами ограничений двойственной задачи;
5) свободные члены ограничений прямой задачи являются коэффициентами целевой функции двойственной;
6) матрицы ограничений прямой и двойственной задач являются транспонированными друг к другу;
7) все переменных в обеих задачах неотрицательные.

Из указанных правил построения двойственных задач следует, что она имеет m переменных и n ограничений.

Экономическая интерпретация:

- прямая задача – сколько и какой продукции x_j надо произвести, чтобы при заданных стоимостях единицы продукции c_j, объемах имеющихся ресурсов b_i и нормах расходов a_{ij} максимизировать выпуск продукции в стоимостном выражении?
- двойственная задача – какова должна быть оценка единицы каждого из ресурсов y_i, чтобы при заданных b_i, c_j и a_{ij} минимизировать общую оценку затрат на все ресурсы?

Теоремы двойственности и их экономическое содержание

Основное неравенство теории двойственности: для любых допустимых планов x и y пары взаимно двойственных задач справедливо неравенство $f(x)<f(y)$. Его экономическое содержание состоит в том, что для любого допустимого плана производства x и любого допустимого о вектора оценок ресурсов y общая созданная стоимость не превосходит суммарной оценки ресурсов.

Критерий оптимальности Канторовича (достаточный признак оптимальности): если для некоторых допустимых планов x и y пары двойственных задач выполняется равенство $f(x)=f(y)$,. то x и y являются оптимальными планами соответствующих задач. Экономический смысл критерия следующий: план производства x и вектор оценок ресурсов y являются оптимальными, если цена всей произведенной продукции и суммарная оценка ресурсов совпадают.

Первая теорема двойственности: если одна из двойственных задач имеет оптимальное решение, то и другая имеет оптимальное решение, причем экстремальные значения целевых функций совпадают: $f(x)=f(y)$. Если одна из двойственных задач неразрешима вследствие неограниченности целевой функции на множестве допустимых решений, то система ограничений другой задачи противоречива.

Решая симплексным методом одну из задач двойственной пары, автоматически получаем решение другой. Для этого достаточно воспользоваться соответствием переменных прямой и двойственной задач и оценок в последней симплексной таблице:

x_1	x_2	…	x_j	…	x_n	x_{n+1}	x_{n+2}	…	x_{n+i}	…	x_{n+m}
⇳	⇳		⇳		⇳	⇳	⇳		⇳		⇳
y_{m+1}	y_{m+2}	…	y_{m+j}	…	y_{m+n}	y_1	y_2	…	y_j	…	y_m
Δ_1	Δ_2	…	Δ_j	…	Δ_n	Δ_{n+1}	Δ_{n+2}	…	Δ_{n+i}	…	Δ_{n+m}

Вторая теорема двойственности (о дополняющей нежесткости): для того, чтобы планы пары двойственных задач были оптимальными, необходимо и достаточно выполнение условий:

$$x_j\left(\sum_{i=1}^{m} a_{ij}y_i - c_j\right) = 0, \quad y_i\left(\sum_{j=1}^{n} a_{ij}x_j - b_i\right) = 0$$

Эти условия называются условиями дополняющей нежесткости. Из них следует, что если какое-либо неравенство системы ограничений одной из задач не обращается в строгое равенство оптимальным планом этой задачи, то соответствующая компонента оптимального плана двойственной задачи должна равняться нулю; если же какая-либо компонента оптимального плана одной из задач положительна, то соответствующее ограничение в двойственной задаче ее оптимальным планом должно обращаться в строгое равенство:

$$\sum_{j=1}^{n} a_{ij}x_j < b_i, \text{ то } y_i = 0, \text{ если } y_i > 0, \text{ то } \sum_{j=1}^{n} a_{ij}x_j = b_i \quad (i = \overline{1,m});$$

$$\sum_{i=1}^{m} a_{ij}y_i > c_j, \text{ то } x_j = 0, \text{ если } x_j > 0, \text{ то } \sum_{i=1}^{m} a_{ij}y_i = c_j \quad (j = \overline{1,n})$$

Если в некотором оптимальном плане оценок ресурсов его i-я компонента строго больше нуля, то в оптимальном плане производства расход соответствующего ресурса равен его запасу.

Вывод – двойственные оценки могут служить мерой дефицитности ресурсов. Дефицитный ресурс (полностью используемый по оптимальному плану производства) имеет положительную оценку, а избыточный ресурс (используемый не полностью) имеет нулевую оценку.

2.7. Контрольные вопросы по теме

1. Из каких этапов состоит симплекс-метод решения задачи ЛП?
2. Что является признаком оптимальности плана?
3. Как найти начальное опорное решение?
4. Как перейти к нехудшему опорному плану?
5. Что является признаком неограниченности целевой функции?
6. Что является признаком альтернативного оптимума?
7. Что представляет собой двойственная задача ЛП?
8. Как формулируется двойственная задача ЛП?
9. Как, зная решение прямой задачи, получить решение двойственной?
10. Какова экономическая интерпретация двойственной задачи?
11. Что представляет собой первая теорема двойственности?
12. Что представляют собой условия дополняющей нежесткости?

Лабораторная работа № 2. Тема: решение задачи линейного программирования симплекс-методом

Задание

1. Сформулировать заданную задачу как задачу линейного программирования. Составить математическую модель.

2. Получить решение задачи с помощью надстройки «Поиск решения» в ЭТ Excel.

3. Решить задачу симплексным методом.

4. Сформулировать двойственную задачу и получить ее решение. Проверить условия о дополняющей нежесткости. Провести анализ чувствительности полученного решения.

Дополнительно в каждой задаче требуется:

- определить дефицитные и недефицитные ресурсы;
- для недефицитных ресурсов найти величину избыточного ресурса;
- определить двойственные оценки каждого ресурса;
- определить суммарную стоимостную оценку ресурсов, используемых при производстве единицы каждого изделия;
- определить, имеется ли продукция, выпуск которой нерентабелен;
- изменить условие задачи в соответствии с дополнительными вопросами, указанными в каждом варианте, и найти ответы на данные вопросы.

Пример 1. Решение задачи симплекс-методом

Предприятие выпускает 2 вида продукции А1 и А2. Для их производства требуется затратить сырье С1, С2 и С3. Затраты сырья на единицу продукции и прибыль приведены в таблице.

Ресурс	Затраты на ед. продукции		Объем ресурса
	А1	А2	
С1	8	25	800
С2	8	5	640
С3	1	5	145
Прибыль	80	70	

Обозначим через x_1 – количество единиц продукции А1, которое производит предприятие, через x_2 – количество единиц продукции А2.

Целевая функция – максимизация прибыли.

Построим модель.

Ограничения: $\begin{cases} 8x_1 + 25x_2 \le 800, \\ 8x_1 + 5x_2 \le 640, \\ x_1 + 5x_2 \le 145, \end{cases}$

$x_1 \ge 0; \;\; x_2 \ge 0.$

Приведем задачу к каноническому виду: $\begin{cases} 8x_1 + 25x_2 + x_3 = 800, \\ 8x_1 + 5x_2 + x_4 = 640, \\ x_1 + 5x_2 + x_5 = 145, \end{cases}$

$x_j \ge 0; \; j = \overline{1{,}5}.$

Целевая функция: f = $80x_1 + 70x_2 + 0\cdot x_3 + 0\cdot x_4 + 0\cdot x_5$. Здесь x_1 и x_2 – свободные переменные, а x_3, x_4, x_5 – базисные.

Начальный опорный план X_0 = (0, 0, 800, 640, 145).
Занесем полученные данные в симплекс-таблицу и решим задачу:

БП	**$C_Б$**	**A_0**	**X_1**	**X_2**	**X_3**	**X_4**	**X_5**	θ
			80	**70**	**0**	**0**	**0**	
X_3	0	800	8	25	1	0	0	100
X_4	0	640	$\boxed{8}$	5	0	1	0	80
X_5	0	145	1	5	0	0	1	145
Индексная строка		***0***	-80	-70	0	0	0	
X_3	0	160	0	$\boxed{20}$	1	-1	0	8
X_1	0	80	1	5/8	0	1/8	0	128
X_5	0	65	0	35/8	0	-1/8	1	104/7
Индексная строка		***6400***	0	-20	0	10	0	
X_2	0	8	0	1	1/20	-1/20	0	
X_1	0	75	1	0	-1/32	5/32	0	
X_5	0	30	0	0	-7/32	3/32	1	
Индексная строка		***6560***	0	0	1	9	0	

В результате двух итераций получим оптимальный план: X = (75, 8, 0, 0, 30) при максимальном значении целевой функции, равном 6560; при этом x_5 = 30 имеет смысл неиспользованного ресурса.

Пример 2. Решение задачи с помощью надстройки «Поиск решения»

Фабрика может выпускать ковры четырех видов и имеет в своем распоряжении следующие ресурсы: рабочая сила, сырье и оборудование, которые имеются в количестве соответственно 80 (чел/дней), 480 (кг), 130 (станко/часов). Информация о количестве единиц каждого ресурса, необходимых для производства одного ковра каждого вида и доходах, получаемых предприятием от единицы каждого вида товаров, приведена в таблице:

Ресурсы	Нормы расхода ресурсов на единицу изделия				Наличие ресурсов
	Ковер A	Ковер B	Ковер C	Ковер D	
Труд	7	2	2	6	80
Сырье	5	8	4	3	480
Оборудование	2	4	1	8	130
Цена (ден.ед.)	3	4	3	1	

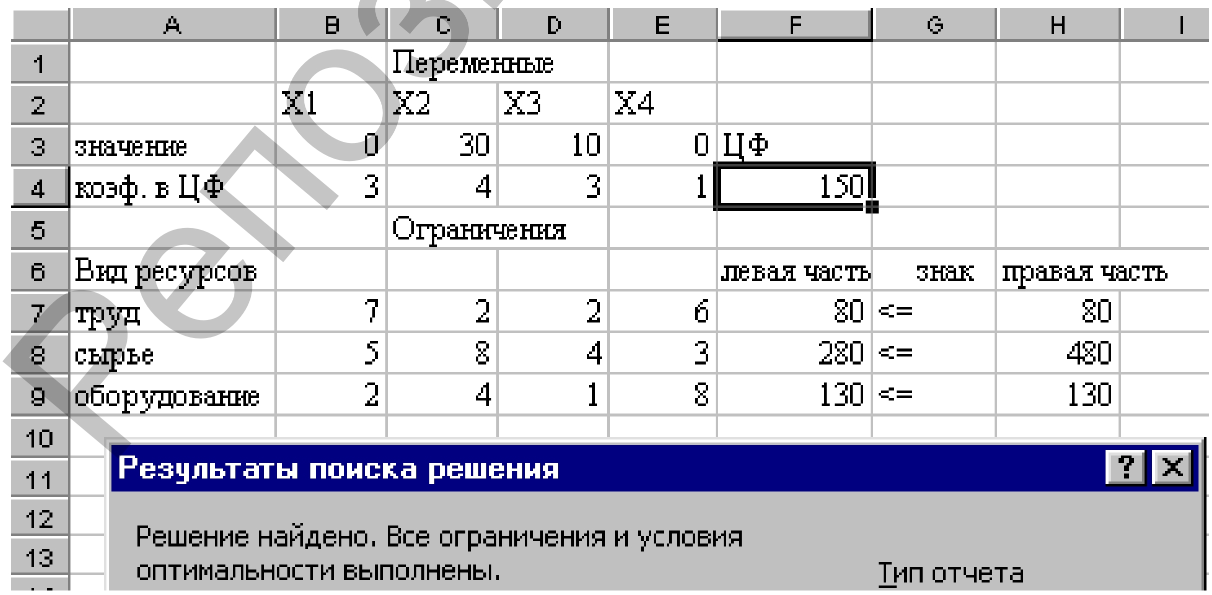
Требуется найти такой план выпуска продукции, при котором общая стоимость продукции будет максимальной.

Решение. Обозначим через X_1, X_2, X_3, X_4 количество ковров каждого типа.

Целевая функция – Max $f(x) = 3X_1 + 4X_2 + 3X_3 + X_4$. Ограничения:

$$\begin{cases} 7X_1 + 2X_2 + 2X_3 + 6X_4 \le 80 \\ 5X_1 + 8X_2 + 4X_3 + 3X_4 \le 480 \\ 2X_1 + 4X_2 + X_3 + 8X_4 \le 130 \\ X_1, X_2, X_3, X_4 \ge 0 \end{cases}$$

Ниже представлен результирующий вид таблицы для решения данной задачи. Оптимальные значения вектора $X = (X_1, X_2, X_3, X_4)$ помещены в ячейках B3:E3, оптимальное значение целевой функции в ячейке F4.



	A	B	C	D	E	F	G	H	I
1			Переменные						
2		X1	X2	X3	X4				
3	значение	0	30	10	0	ЦФ			
4	коэф. в ЦФ	3	4	3	1	150			
5			Ограничения						
6	Вид ресурсов					левая часть	знак	правая часть	
7	труд	7	2	2	6	80	<=	80	
8	сырье	5	8	4	3	280	<=	480	
9	оборудование	2	4	1	8	130	<=	130	
10									
11									
12									
13									

Полученное решение означает, что максимальный доход 150 ден.ед. фабрика может получить при выпуске 30 ковров второго вида и 10 ковров третьего вида. При этом ресурсы труд и оборудование (дефицитные ресурсы) будут использованы полностью, а из 480 кг пряжи (недефицитный ресурс) будет использовано 280 кг, величина избыточного ресурса равна 200 кг.

Создание отчета по результатам поиска решения

Excel позволяет представить результаты поиска решения в форме отчёта. Существует три типа таких отчетов:

- *Результаты*. В отчет включаются исходные и конечные значения целевой и влияющих ячеек, дополнительные сведения об ограничениях
- *Устойчивость*. Отчет, содержащий сведения о чувствительности решения к малым изменениям в изменяемых ячейках иди в формулах ограничений.
- *Пределы*. Помимо исходных и конечных значений изменяемых и целевой ячеек в отчет включаются верхние и нижние границы значений, которые могут принимать влияющие ячейки при соблюдении ограничений.

Отчет по результатам

Отчет по результатам

Целевая ячейка (Максимум)

Ячейка	Имя	Исходно	Результат
$F$4	коэф. в ЦФ ЦФ	0	150

Изменяемые ячейки

Ячейка	Имя	Исходно	Результат
$B$3	значение X1	0	0
$C$3	значение X2	0	30
$D$3	значение X3	0	10
$E$3	значение X4	0	0

Ограничения

Ячейка	Имя	Значение	Формула
$F$7	труд левая часть	80	$F$7<=$H$7
$F$8	сырье левая часть	280	$F$8<=$H$8
$F$9	оборудование левая часть	130	$F$9<=$H$9

В отчете по результатам содержатся оптимальные значения переменных X_1, X_2, X_3, X_4, которые соответственно равны 0,10, 30, 0; значение целевой функции – 150, а также левые части ограничений.

Сформулируем математическую модель *двойственной* задачи к задаче о коврах:

Y_1 – двойственная оценка ресурса труд, или «цена» труда;

Y_2 – двойственная оценка ресурса сырье, или «цена» сырья;

Y_3 – двойственная оценка ресурса оборудование, или «цена» оборудования.

Целевая функция двойственной задачи формулируется на минимум.

$g(\bar{y}) = 80Y_1 + 480Y_2 + 130Y_3 \rightarrow min$

Необходимо найти такие оценки ресурсов Y_i, (теневые цены) чтобы общая стоимость используемых ресурсов была минимальной. Ограничения:

$$\begin{cases} 7Y1 + 5Y2 + 2Y3 \geq 3 \\ 2Y1 + 8Y2 + 4Y3 \geq 4 \\ 2Y1 + 4Y2 + 1Y3 \geq 3 \\ 6Y1 + 3Y2 + 8Y3 \geq 1 \\ Y1\ ,Y2\ ,Y3 \geq 0 \end{cases}$$

Решение двойственной задачи можно найти в отчете по устойчивости Поиска решений. Теневые цены ресурсов труд, сырье и оборудование соответственно равны 4/3, 0, 1/3 или в десятичных дробях 1.3333, 0, 0.3333.

Отчет по устойчивости

Изменяемые ячейки

Ячейка	Имя	Результ. Значение	Нормир. Стоимость	Целевой Коэффициент	Допустимое Увеличение	Допустимое Уменьшение
$B$3	Значение X1	0	-7	3	7	1E+30
$C$3	Значение X2	30	0	4	8	1
$D$3	Значение X3	10	0	3	1	1.75
$E$3	Значение X4	0	-9.667	1	9.667	1E+30

Ограничения

Ячейка	Имя	Результ. Значение	Теневая Цена	Ограничение Правая часть	Допустимое Увеличение	Допустимое Уменьшение
$F$7	труд левая часть	80	1.333	80	150	15
$F$8	сырье левая часть	280	0	480	1E+30	200
$F$9	Оборудование левая часть	130	0.333	130	30	90

Проведем анализ полученного оптимального решения исходной задачи с помощью анализа условий дополняющей нежесткости.

В рассматриваемой задаче ресурсы труд и оборудование имеют отличные от нуля оценки 4/3 (1.333) и 1/3 (0.333) – эти ресурсы полностью используются в оптимальном плане, являются дефицитными, сдерживающими рост целевой функции. Правые части этих ограничений равны левым частям.

$7X1 + 2X2 + 2X3 + 6X4 \leq 80$	$7\times0 + 2\times30 + 2\times10 + 6\times0 = 80 = 80$
$2X1 + 4X2 + X3 + 8X4 \leq 130$	$2\times0 + 4\times30 + 1\times10 + 8\times0 = 130 = 130$

Ресурс сырье используется не полностью (280<480), поэтому имеет нулевую двойственную оценку (Y_2=0).

$5X_1 +8X_2 +4X_3 +3X_4 \le 480$ $\qquad$ $5\times 0 +8\times 30 +4\times 10 +3\times 0=280<480$

Этот ресурс не влияет на план выпуска продукции.

Общая стоимость используемых ресурсов при выпуске 30 ковров второго вида и 10 ковров третьего вида составит 150 ден.ед.

g = 80 ×Y1 + 480×Y2 + 130×Y3 =80 ×4/3 +480×0+130×1/3 =150 ден.ед.

Заметим, что оценки (теневые цены) различных видов ресурсов нельзя отождествлять с действительными ценами, по которым осуществляется его закупка. В данном случае речь идет о некоторой мере, имеющей экономическую природу, которая характеризует ценность ресурса только относительно полученного оптимального решения.

Анализ эффективности отдельных вариантов плана также выполняется на основе соотношений второй теоремы двойственности.

Если изделие вошло в оптимальный план (Xj > 0), то в двойственных оценках оно не убыточно, то есть, стоимость ресурсов, затраченных на производство единицы изделия равна его цене. В нашей задаче это ковры второго и третьего видов. Если стоимость ресурсов, затраченных на производство одного изделия больше его цены, то это изделие не войдет в оптимальный план из-за его убыточности. В нашей задаче в план выпуска не вошли ковры первого и четвертого видов, потому что затраты по ним превышают цену на 7 ден.ед. и 9.666 ден.ед. соответственно. Это можно подтвердить, подставив в ограничения двойственной задачи оптимальные значения вектора Y.

7 ×4/3 + 5×0+ 2×1/3 = 30/3 =10 >3

2 ×4/3 + 8×0+ 4×1/3 = 12/3 = 4 = 4

2 ×4/3 + 4×0+ 1×1/3 = 9/3 = 3 = 3

6×4/3 + 3×0+ 8×1/3 = 32/3 = 10.666 > 1

Разницу между правыми и левыми частями ограничений двойственной задачи можно найти в *Отчете по устойчивости* в столбце *Нормируемая стоимость*.

Также необходимо знать интервалы изменения каждого из свободных членов системы ограничений исходной ЗЛП, или интервалы устойчивости двойственных оценок, в которых оптимальный план двойственной задачи не менялся бы. Информацию для анализа влияния изменения правых частей ограничений на значения целевой функции (чувствительность решения к изменению запасов сырья) можно получить из *Отчета по устойчивости.* В нашей задаче в приведенном фрагменте отчета видно, что запасы дефицитных ресурсов труд и оборудование могут быть, как уменьшены, так и увеличены, увеличение запаса ресурса сырье не повлияет на план выпуска продукции.

Ограничение правая часть	Допустимое Увеличение	Допустимое уменьшение
80	150	15
480	1E+30	200
130	30	90

Варианты заданий для самостоятельного выполнения

1. Предприятие располагает ресурсами сырья, рабочей силой и оборудованием, необходимыми для производства любого из 4 видов производимых товаров. Затраты ресурсов на изготовление единицы данного вида товара, прибыль, получаемая предприятием, а также запасы ресурсов указаны в таблице.

Вид ресурса	Товары				Объем ресурсов
	1	2	3	4	
Сырье, кг	3	5	2	4	60
Рабочая сила, часы	22	14	18	30	400
Оборудование, станко-часы	10	14	8	16	128
Прибыль на единицу товара	300	250	560	480	

- Определить, какой ассортимент товара надо выпускать, чтобы прибыль была максимальной?
- Определить оптимальный ассортимент при дополнительном условии: 1-го товара выпустить не более 5 ед., 2-го - не менее 8 ед.

2. Цех мебельного комбината выпускает комоды, столы и тумбочки под телевизоры. Норма расхода материала в расчете на одно изделие, плановая себестоимость, оптовая цена предприятия, плановый ассортимент и трудоемкость единицы продукции приведены в таблице. Запас древесно-стружечных плит, досок еловых и березовых 90, 30 и 14 м^3 соответственно. Плановый фонд рабочего времени 16 800 человеко-часов.

Показатель	Изделия		
	комод	*стол*	*тумбочка*
древесно-стружечные плиты, м^3	0,032	0,031	0,038
доски еловые, м^3	0,020	0,020	0,008
доски березовые, м^3	0,005	0,005	0,006
трудоемкость, чел.-ч	10,2	7,5	5,8
плановая себестоимость, ден.ед.	88,81	63,98	29,60
оптовая цена предприятия, ден.ед.	93	67	30

Исходя из необходимости выполнения плана по ассортименту и возможности его перевыполнения, найти такой план выпуска продукции, который обеспечивает:

- максимальную прибыль;

- максимальный объем реализации при плановом ассортименте 350, 290 и 1200 единиц продукции соответственно.

3. Для изготовления трёх видов продукции *A, B* и *C* используют токарное, фрезерное, сварочное и шлифовальное оборудование. Затраты времени на обработку одного изделия для каждого из типов оборудования указаны в таблице. В ней же указаны общий фонд рабочего времени каждого из типов используемого оборудования, а также прибыль от реализации одного изделия данного вида.

Тип оборудования	Затраты времени на обработку одного изделия вида, станко-ч			Общий фонд рабочего времени оборудования, ч
	A	*B*	*C*	
Фрезерное	5	6	8	210
Токарное	3	4	2	320
Сварочное	7	9	4	250
Шлифовальное	2	5	6	160
Прибыль, руб.	8	11	15	

- Требуется определить, сколько изделий и какого вида следует изготовить предприятию, чтобы прибыль от их реализации была максимальной.
- Дополнительно определить изменение в оптимальном плане выпуска продукции, если фонд рабочего времени фрезеровального и токарного оборудования увеличен на 50%, а сварочного и шлифовального – на 30%.

4. При производстве карамели на кондитерской фабрике используются сахарный песок, патока, фруктовое пюре и вкусовые добавки. Нормы расхода сырья каждого вида для производства 1 т карамели "Абрикос"(А), "Вишня"(В) и "Клубника"(К) приведены в таблице.

Нормы расхода сырья на 1 т карамели, т	Вид сырья			Общее количество сырья, т
	А	В	К	
Сахарный песок	0,7	0,6	0,8	900
Патока	0,45	0,5	0,3	700
Фруктовое пюре	0,1	0,2	0,15	250
Вкусовые добавки	0,002	0,005	0,003	16
Прибыль, руб.	1000	1200	1350	

- Требуется определить, план выпуска карамели, чтобы прибыль от её реализации была максимальной.

- Дополнительно заданы производственные издержки в рублях на 1 ед. каждого изделия: 40, 45, 50. Найти оптимальный ассортимент, максимизирующий прибыль, при условии, что суммарные производственные издержки не должны превышать 5000 руб.

5. Для производства столов, стульев и шкафов мебельная фабрика использует два вида древесины. Нормы затрат ресурсов на одно изделие данного вида, прибыль от реализации одного изделия и общее количество имеющихся ресурсов каждого вида приведены в таблице.

Ресурсы	Нормы затрат ресурсов на одно изделие			Общее количество ресурсов
	Стол	Стул	Шкаф	
Древесина I вида, $м^3$	0,3	0,1	0,4	80
Древесина II вида, $м^3$	0,1	0,05	0,5	120
Трудоёмкость, чел.-ч	1,3	0,3	2,5	483,5
Прибыль от реализации одного изделия, руб.	21	25	35	

- Определить сколько изделий мебели каждого вида фабрике следует изготовить, чтобы прибыль от их реализации была максимальной.
- Определить оптимальный ассортимент при дополнительном условии: столов и стульев выпустить не менее чем по 50 ед., шкафов – не более 10 ед.

6. На звероферме могут выращиваться норки, выдры и нутрии. Для обеспечения нормальных условий их выращивания используется три вида кормов. В таблице указаны общее количество корма каждого вида, которое может быть использовано зверофермой, и прибыль от реализации одной шкурки зверька.

Вид корма	Количество единиц корма, которое ежедневно должны получать			Общее количество корма
	норка	выдра	нутрия	
I	4	2	5	190
II	5	3	4	320
III	7	9	5	454
Прибыль от реализации одной шкурки, руб.	150	320	350	

- Определить, сколько зверьков каждого вида следует выращивать на звероферме, чтобы прибыль от реализации их шкурок была максимальной.

• Дополнительно определить изменение в рационе кормления, если корм II должен быть использован полностью.

7. На швейном предприятии для изготовления пяти видов костюмов может быть использована ткань трёх артикулов. Нормы расхода тканей всех артикулов на пошив одного костюма даны в таблице. В таблице также указаны общее количество тканей каждого артикула, имеющееся в данный момент в распоряжении предприятия, и отпускная цена одного костюма каждого вида.

Артикул ткани	Норма расхода ткани на один костюм каждого вида, м					Общее количество ткани, м
	1	2	3	4	5	
I	2	1	1	3	2	190
II	3	2	-	2	1	320
III	-	4	4	1	-	454
Цена одного костюма, руб.	2300	4500	6200	6400	8200	

• Определить, сколько костюмов каждого вида должно произвести предприятие, чтобы стоимость изготовленной продукции была максимальной.

• Как изменится оптимальный план выпуска костюмов, если общее количество ткани первого артикула увеличится на 10%, второго артикула - на 25%, а третьего артикула - на 30%?

8. Нефтяная компания «РТ» для улучшения эксплуатационных качеств и снижения точки замораживания дизельного топлива, которое она производит, добавляет в него определенные химикаты. В каждом бензобаке объемом 1000 л должно содержаться не менее 33 мг химической добавки X, не менее 14 мг химической добавки Y и не менее 18 мг химической добавки Z. Необходимые химические добавки в форме готовых смесей поставляют «РТ» две химические компании A и B. В нижеследующей таблице приведено содержание химических добавок в каждом продукте, поставляемом указанными компаниями.

Продукт	*Химические добавки, мг/л*		
	X	*Y*	*Z*
A	4	2	3
B	5	1	1

Стоимость продукта A – 1,5 ф. ст. за 1 л, а продукта B – 1,2 ф. ст. за 1 л.

• Требуется: найти ассортиментный набор продуктов A и B, минимизирующий общую стоимость добавленных в топливо химикатов.

- Дополнительно заданы производственные издержки на 1 ед. каждого продукта: 0,15 для A и 0,05 для B. Решить задачу при условии, что суммарные производственные издержки не должны превышать 1 ф. ст.

9. Фабрика выпускает три вида каш для завтрака «Манго», «Сливки», «Персик». Используемые для производства данных продуктов ингредиенты в основном одинаковы и, как правило, не являются дефицитными. Основным ограничением, накладываемым на объем выпуска, является наличие фонда рабочего времени в каждом из трех цехов фабрики. В приведенной ниже таблице указаны общий фонд рабочего времени и число человеко-часов, требуемое для производства 1 т продукта.

Цех	Необходимый фонд рабочего времени, чел.-ч./т			Общий фонд рабочего времени чел.-ч. в месяц
	«Манго»	«Сливки»	«Персик»	
А. Производство	10	4	7	1000
В. Добавление приправ	3	2	4	360
С. Упаковка	2	5	3	600

Доход от производства 1 т. "Манго" составляет 150 дол., от производства каши "Сливки" - 90 дол., и от производства каши "Персик" - 130 дол.

- Необходимо разработать план производства на месяц, обеспечивающий максимальный доход.
- При определении плана выпуска следует учесть также то, что минимальное количество продукции каждого вида, которое требуется произвести – 30 т.

10. Найти решение задачи оптимизации, состоящей в определении плана изготовления пяти видов хлебобулочных изделий, обеспечивающего максимум стоимости всей изготовленной продукции. Учесть заданные ограничения на использование имеющихся в наличии количеств сырья четырёх видов.

Нормы расхода сырья каждого вида на одно изделие, цена одного изделия соответствующего наименования, а также общее количество сырья данного вида приведены в таблице.

Вид сырья	Изделие					Общее количество сырья
	1	2	3	4	5	
I	16	18	13	14	12	470
II	3	5	4	7	9	186
III	8	4	8	6	4	178

IV	7	6	3	5	5	194
Цена одного изделия, руб	8	6	10	7	9.	

Найти такой план выпуска продукции, который обеспечивает:

- максимальный объем реализации;
- максимальную прибыль.

11. На ткацкой фабрике для изготовления пяти артикулов ткани используются ткацкие станки двух типов, пряжа и красители.

В таблице указаны затраты станков каждого типа, нормы расхода пряжи и красителей на 1 м ткани, цена 1 м ткани данного артикула, также заданы общий фонд рабочего времени станков каждого типа, имеющиеся в распоряжении фабрики фонды пряжи и красителей и ограничения на возможный выпуск тканей данного артикула.

Ресурсы	Нормы затрат на 1 м ткани артикула					Общее количество ресурсов
	1	2	3	4	5	
Затраты на 1 м ткани станков, станко-ч: I типа II типа	 0,16 0,18	 0,08 0,05	 0,03 0,04	 0,04 0,01	 -0,08	 370 530
Пряжа, кг	2,0	1,0	1,3	1,7	1,5	186
Красители, кг	0,08	0,04	0,03	0,035	0,024	264
Цена 1 м ткани, руб.	8	6	10	7	9	

- Составить такой план изготовления тканей данного артикула, согласно которому будет произведено нужное количество тканей данного артикула, а общая стоимость всех тканей будет максимальна.
- Решить ту же задачу при наличии дополнительных ограничений:

Выпуск ткани, м: минимальный максимальный	 800 2200	 1000 8500	 3500 6500	 2500 5500	 1500 4500

12. Фабрика выпускает кожаные брюки, куртки и пальто. В процессе изготовления изделия проходят три производственных участка: дубильный, раскройный и пошивочный. Время обработки изделий на каждом участке, их плановая себестоимость, оптовая цена предприятия приведены в таблице. Ограничения на фонд времени для дубильного, раскройного и пошивочного участков составляют соответственно 3360, 2688 и 5040 ч.

Показатели	Изделия		
	брюки	куртка	пальто
Норма времени на участках, чел-ч:			

дубильном раскройном пошивочном	0,3 0,4 0,5	0,4 0,4 0,4	0,6 0,7 0,8
Полная себестоимость, ден.ед.	15	40,5	97,8
Оптовая цена предприятия, ден.ед.	17,5	42,0	100,0

- Учитывая заданный ассортимент, найти оптимальный план, обеспечивающий максимальную прибыль предприятия.
- Решить ту же задачу при наличии дополнительных ограничений: пусть ассортимент продукции различных видов задан отношением 2:1:3 соответственно.

13. Фирма производит три вида продукции (A, B, C) для впуска каждого из которых требуется определенное время обработки на всех четырех устройствах I, II, III, IV.

Вид продукции	Время обработки (ч.)				Прибыль (долл.)
	I	II	III	IV	
A	1	3	1	2	3
B	6	1	3	3	6
C	3	3	2	4	4

Пусть максимально допустимое время работы на устройствах – соответственно 84, 42, 21 и 42 ч.

- Найти план выпуска продукции, максимизирующий прибыль.
- Как изменится план выпуска, если фирма откажется от обработки всех видов продукции на четвертом устройстве, увеличив время их обработки на первом на 20%?

14. На приобретение оборудования для нового производственного участка выделено 300 тыс. ден. ед. Его предполагается разместить на площади 45 м^2. Участок может быть оснащен оборудованием трех видов (все показатели приводятся на единицу оборудования):

1) машинами стоимостью 6 тыс. ден. ед., размещающимися на площади 9 м^2, производительностью 8 тыс. ед. продукции за смену;

2) машинами стоимостью 3 тыс. ден. ед., занимающими площадь 4 м^2, производительностью 4 тыс. ед. продукции за смену;

3) машинами стоимостью 2 тыс. ден. ед., размещающимися на площади 3 м^2, производительностью 3 тыс. ед. продукции за смену.

- Найти оптимальный план закупки оборудования, обеспечивающий наибольшую производительность всего участка.
- Как изменится план закупки оборудования, если объем финансирования увеличится на 30%?

15. Имеющийся фонд материалов М1, М2, М3 нужно так

распределить между изготовителями, чтобы обеспечить максимальную прибыль от реализации всей продукции, произведенной из данных материалов. Нормы расхода на единицу продукции и прибыль, получаемая от реализованной единицы продукции представлены в таблице.

Материалы	Продукция					Объем материала
	П1	П2	П3	П4	П5	
М1	0,7	0,9	1,5	2,3	1,8	50 000
М2	1,4	0,3	0,7	2,5	2,0	28 000
М3	0,5	2,1	1,8	0,7	2,0	40 000
Прибыль	5	7	6	9	8	

- Найти оптимальный план производства продукции.
- Определить, как повлияет на максимальную прибыль увеличение каждого ресурса на 5%.

16. Из четырех видов сырья необходимо составить смесь, в состав которой должно входить не менее 26 ед. химического вещества А, 30 ед. - вещества В и 24 ед. - вещества С. Количество единиц химического вещества, содержащегося в 1 кг сырья каждого вида, указано в таблице. В ней же приведена цена 1 кг сырья каждого вида.

Вещество	Количество единиц вещества, содержащегося в 1 кг сырья вида			
	1	2	3	4
А	1	1	-	4
В	2	-	3	5
С	1	2	4	6
Цена 1 кг сырья	5	6	7	4

- Составить смесь, содержащую не менее нужного количества веществ данного вида и имеющую минимальную стоимость.
- Как повлияет на состав смеси увеличение цены 1 кг сырья первого и второго видов на 10 % и уменьшение цены 1 кг сырья третьего и четвертого видов на 5 %?

2.8. Транспортная задача ЛП

Постановка транспортной задачи. Однородный груз, имеющийся в m пунктах отправления A_1, A_2, ..., A_m соответственно в количествах a_1, a_2, ..., a_m единиц, требуется доставить в каждый из n пунктов назначения (потребления) B_1, B_2, ..., B_n соответственно в количествах b_1, b_2, ..., b_n единиц. Стоимость перевозки (тариф) единицы продукта из A_i в B_j известна для всех маршрутов A_iB_j и равна C_{ij} ($i=\overline{1,m}$; $j=\overline{1,n}$).

Требуется составить такой план перевозок, при котором весь груз из пунктов отправления вывозиться и запросы всех пунктов потребления удовлетворяются, а суммарные транспортные расходы минимальны. Условия задачи удобно располагать в таблицу.

В левом верхнем углу произвольной клетки стоит коэффициент, равный стоимости перевозки от поставщика, номер которого указан в этой строке, к потребителю, номер которого указан в столбце. Таблица такого вида называется таблицей поставок.

Поставщики	Потребители				Запасы поставщиков, a_i
	B_1	B_2	...	B_n	
A_1	C_{11} x_{11}	C_{12} x_{12}	…	C_{1n} x_{14}	a_1
…	…	…	…	…	…
A_m	C_{m1} x_{m1}	C_{m2} x_{m2}	…	C_{mn} x_{mn}	a_m
Спрос потребителей b_j	b_1	b_2	…	b_n	$\sum a_i = \sum b_j$

Построим экономико-математическую модель данной задачи, обозначив через x_{ij} объем поставляемого товара от i-го поставщика к j-му потребителю. Чтобы спрос каждого из потребителей был удовлетворён, должны быть справедливы уравнения баланса для каждого столбца таблицы поставок, то есть $\sum_{i=1}^{m} x_{ij} = b_j,\ j = 1,2,\ldots,n,$

Чтобы запасы каждого поставщика были полностью реализованы, должны быть справедливы уравнения баланса для каждой строки таблицы поставок, т. е. выполняться равенства $\sum_{j=1}^{n} x_{ij} = a_i, i = 1,2,\ldots,m.$

Поскольку объём перевозимого груза величина неотрицательная, то должны выполняться ограничения на переменные x_{ij} : $x_{ij} \geq 0$, i =1, ,m; j =1, ,n.

Суммарные затраты f на перевозку определяются указанными в таблице поставок тарифами перевозок и размерами поставок:

$$f = \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} x_{ij} \to \min$$

Решить транспортную задачу – значит *на множестве неотрицательных решений системы ограничений найти такое решение, при котором целевая функция принимает минимальное значение*.

Транспортная задача называется *закрытой*, если сумма запасов всех *n* поставщиков равна сумме потребностей всех *m* потребителей:

$$\sum_{i=1}^{m} a_i = \sum_{j=1}^{n} b_j$$

В противном случае транспортная задача называется *открытой*. Решение открытой транспортной задачи сводят к решению закрытой транспортной задачи введением фиктивных потребителей, когда сумма запасов превышает сумму потребностей, или фиктивных поставщиков, когда сумма потребностей превышает сумму запасов. При этом тарифы перевозок для фиктивных поставщиков и потребителей принимаются равными нулю.

Для нахождения опорного плана рассмотрим два метода: "северо-западного угла" и наименьшей стоимости.

Рассмотрим *метод "северо-западного угла"*.

Сущность этого метода заключается в том, что на каждом шаге заполняется левая верхняя клетка (северо-западная) оставшейся части таблицы, причем максимально возможным числом: либо полностью вывозиться груз из A_i, либо полностью удовлетворяется потребность B_j. Т.е. $x_{ij} = \min(a_i;b_j)$. Процедура продолжается до тех пор, пока на каком-то шаге не исчерпаются запасы a_i и не удовлетворяются потребности b_j.

Если распределение выполняется без вычислительных ошибок, в последнюю заполняемую клетку запишется число, получаемое автоматически и равное остатку у последнего поставщика или количеству спроса последнего потребителя Число заполненных клеток в полученном распределении должно быть равным числу базисных (основных) переменных.

Другим методом определения опорного плана поставок является *метод наименьшей стоимости* или *метод наименьшего тарифа*.

Сущность метода в том, что на каждом шаге заполняется та клетка оставшейся части таблицы, которая имеет наименьший тариф; в случае наличия нескольких таких равных тарифов заполняется любая из них. В остальном действуют аналогично предыдущему способу.

Критерий оптимальности для транспортной задачи: базисное распределение поставок оптимально тогда и только тогда, когда *оценки* всех свободных (небазисных) клеток неотрицательны.

Для определения оценок свободных клеток используют два метода: потенциалов и распределительный. Рассмотрим один из них, а именно *метод потенциалов.*

Потенциалы ─ числа для нахождения оценок допустимого плана, полученного в ходе распределения запасов поставщиков. Для начала рассмотрим только заполненные клетки. Для этих клеток каждое значение единичной стоимости c_{ij} разделяется на две компоненты — u_i для строк и v_j для столбцов.

Потенциалы для поставщиков u_i и потребителей v_j вычисляются по тарифам c_{ij} *занятых* клеток таблицы поставок. Для них справедливы равенства

$u_i + v_j = c_{ij}$, $i = 1,...,m$; $j = 1,...,n$.

Так как занятых клеток на одну меньше, чем число потенциалов, значение одного из потенциалов (все равно какого) назначается произвольно и может быть любым действительным числом (обычно u_1 полагают равным нулю). Разрешая равенства относительно потенциалов, получаем их числовые значения.

Оценки *свободных* клеток таблицы поставок рассчитываются по формулам

$\delta_{ij} = c_{ij} - (u_i + v_j)$, $i = 1,..., m$; $j=1,...,n$.

Алгоритм решения транспортной задачи

1) проверяем тип модели транспортной задачи и в случае открытой модели сводим ее к закрытой;

2) находим опорный план перевозок путем составления 1-й таблицы одним из способов - северо-западного угла или наименьшего тарифа;

3) проверяем план (таблицу) на удовлетворение системе уравнений и на невырожденность; в случае вырождения плана добавляем условно заполненные клетки с помощью «0»;

4) проверяем опорный план на оптимальность, для чего:

а) составляем систему уравнений потенциалов по заполненным клеткам;

б) находим одно из ее решений при $u_1=0$;

в) находим суммы оценки для всех пустых клеток;

г) если оценки всех пустых клеток неотрицательны, то план оптимален (критерий оптимальности). Решение закончено: ответ дается в виде плана перевозок последней таблицы и значения min f.

Если критерий оптимальности не выполняется, то переходим к следующему шагу.

5) Для перехода к следующей таблице (плану):

а) выбираем одну из пустых клеток, где оценка отрицательна;

б) составляем цикл пересчета для этой клетки и расставляем знаки « + », « - » в вершинах цикла путем их чередования, приписывая пустой клетке « + »;

в) находим число перерасчета по циклу: число X=min{Xij}, где Xij – числа в заполненных клетках со знаком « - »;

г) составляем новую таблицу, добавляя X в плюсовые клетки и отнимая X из минусовых клеток цикла

6) См. п. 4 и т.д.

Через конечное число шагов оптимальный план транспортной задачи будет найден, так как он всегда существует.

Особые случаи транспортной задачи

Недопустимые перевозки

Если перевозка из некоторого пункта производства в некоторый пункт назначения по той или иной причине невозможна, то в алгоритме решения задачи данное ограничение можно учесть, присвоив соответствующей клетке достаточно большое значение стоимости. Точное значение в данном случае неважно, однако, оно должно быть больше, чем остальные значения стоимости, указанные в таблице. Таким образом, алгоритм автоматически позволит избежать перевозок через данную клетку.

Вырожденность

Решение называется вырожденным, если число перевозок в транспортной таблице меньше, чем $(m+n-1)$. Данную проблему можно разрешить, проставив в независимые клетки очень маленькие, по сути равные нулю объемы перевозок. Число перевозок увеличивается таким образом до $(m+n-1)$. Выявить клетки, которые следует использовать для этой цели, поможет алгоритм метода потенциалов проверки решения на оптимальность.

Максимизация

Алгоритм решения транспортной задачи предполагает, что ее целевая функция стремится к минимуму. Однако если некоторая проблема требует максимизации целевой функции перед тем, как применять для решения этой задачи стандартный алгоритм, его необходимо немного модифицировать. Например, мы намерены осуществить перевозку товаров таким образом, чтобы максимизировать общий доход. В этом случае нам необходима информация о единичных доходах от транспортировки товаров между всеми пунктами производства и назначения. Модификация заключается в умножении всех значений единичного дохода на (- 1), а затем поступают обычным образом.

2.9. Контрольные вопросы по теме

1. Что требуется определить в транспортной задаче?
2. Что представляет собой открытая транспортная задача? Закрытая транспортная задача?
3. Как преобразовать открытую транспортную задачу к закрытой?
4. Какими методами можно получить начальное решение транспортной задачи?

5. Что является признаком оптимальности плана?
6. Как перейти к нехудшему плану?
7. Каков алгоритм решения транспортной задачи?
8. Как избежать недопустимых перевозок?
9. Какое решение называется вырожденным?

Лабораторная работа № 3. Тема: решение транспортной задачи

Пример

Решить транспортную задачу.

		Потребители			Запасы
		B_1	B_2	B_3	
Поставщики	A_1	9	5	4	25
	A_2	7	8	5	50
	A_3	3	4	6	50
Спрос		25	70	40	

Решение:

Обозначим через x_{ij} $(i = \overline{1,3};\ j = \overline{1,3})$ количество груза, перевозимого от i – го поставщика j – му потребителю. Тогда общая стоимость перевозок равна:

$$f = \sum_{i=1}^{3}\sum_{j=1}^{3} c_{ij}x_{ij} = 9x_{11} + 5x_{12} + 4x_{13} + 7x_{21} + 8x_{22} + 5x_{23} + 3x_{31} + 4x_{12} + 6x_{33}$$

Ограничения для поставщиков:

$$\begin{cases} x_{11} + x_{12} + x_{13} = 25 - \text{для 1-го поставщика} \\ x_{21} + x_{22} + x_{23} = 50 - \text{для 2-го поставщика} \\ x_{31} + x_{32} + x_{33} = 50 - \text{для 3-го поставщика} \end{cases}$$

Ограничения для потребителей:

$$\begin{cases} x_{11} + x_{21} + x_{31} \le 25 - \text{для 1-го потребителя} \\ x_{12} + x_{22} + x_{32} \le 70 - \text{для 2-го поставщика} \\ x_{13} + x_{23} + x_{33} \le 40 - \text{для 3-го поставщика} \end{cases}$$

Объем суммарных поставок любого поставщика к потребителю не может быть отрицательным числом, поэтому справедливы ограничения: $x_{ij} \ge 0$

Стандартная транспортная задача разрешима только в том случае, когда выполняется условие баланса:

$$\sum_{i=1}^{3} a_i = \sum_{j=1}^{3} b_j$$

В нашем случае:

$$\sum_{i=1}^{3} a_i = 125; \quad \sum_{j=1}^{3} b_j = 135$$

Модель транспортной задачи открытая. Вводим фиктивного поставщика, у которого имеется 10 единиц груза.

Заполняем таблицу по правилу минимального элемента. Решать задачу будем методом потенциалов. Число занятых клеток должно быть $m + n - 1$. Потенциал 1-й строки принимаем равным нулю. После этого мы можем вычислить остальные потенциалы (если известны потенциал и тариф занятой клетки, то из соотношения v + u =c легко определить неизвестный потенциал).

	$v_1 = 6$ B_1	$v_2 = 7$ B_2	$v_3 = 4$ B_3	
A_1	9	+ 5	− 4 25	$u_1 = 0$ 25
A_2	7	− 8 35	+ 5 15	$u_2 = 1$ 50
A_3	3 15	4 35	6	$u_3 = -3$ 50
A_4	0 10	0	0	$u_4 = -6$ 10
	25	70	40	

Найдем оценки свободных клеток:

S (1, 1)= 9-(0+ 6)= 3
S (2, 1)= 7-(1+ 6)= 0
S (4, 2)= 0-(-6+ 7)= -1
S (1, 2)= 5-(0+ 7)= -2
S (3, 3)= 6-(-3+ 4)= 5
S (4, 3)= 0-(-6+ 4)= 2

Для клетки (1, 2) строим цикл.

	B_1 ($v_1 = 4$)	B_2 ($v_2 = 5$)	B_3 ($v_3 = 2$)		
A_1	9	5; 25	4	$u_1 = 0$	25
A_2	7	8; 10	5; 40	$u_2 = 3$	50
A_3	+ 3; 15	− 4; 35	6	$u_3 = -1$	50
A_4	− 0; 10	+ 0	0	$u_4 = -4$	10
	25	70	40		

Найдем оценки свободных клеток:

S (1, 1)= 9-(0+ 4)= 5
S (2, 1)= 7-(3+ 4)= 0
S (4, 2)= 0-(-4+ 5)= -1
S (1, 3)= 4-(0+ 2)= 2
S (3, 3)= 6-(-1+ 2)= 5
S (4, 3)= 0-(-4+ 2)= 2

Для клетки (4, 2) строим цикл.

	B_1 ($v_1 = 4$)	B_2 ($v_2 = 5$)	B_3 ($v_3 = 2$)		
A_1	9	5; 25	4	$u_1 = 0$	25
A_2	7	8; 10	5; 40	$u_2 = 3$	50
A_3	3; 25	4; 25	6	$u_3 = -1$	50
A_4	0	0; 10	0	$u_4 = -5$	10
	25	70	40		

Найдем оценки свободных клеток:

S (1, 1)= 9-(0+ 4)= 5
S (2, 1)= 7-(3+ 4)= 0
S (4, 1)= 0-(-5+ 4)= 1
S (1, 3)= 4-(0+ 2)= 2
S (3, 3)= 6-(-1+ 2)= 5
S (4, 3)= 0-(-5+ 2)= 3

Оценки свободных клеток не отрицательны, следовательно, полученный план является оптимальным:

$$X^* = \begin{pmatrix} 0 & 25 & 0 \\ 0 & 10 & 40 \\ 25 & 25 & 0 \end{pmatrix}$$

Минимальные транспортные издержки оптимального плана:

$$\min f = 5 \cdot 25 + 8 \cdot 10 + 5 \cdot 40 + 3 \cdot 25 + 4 \cdot 25 = 580$$

При реализации оптимального плана потребитель B_2 останется до конца не загруженным. Нехватка груза составит 10 единиц.

Задание. Найти решение транспортной задачи для заданной таблицы поставок. Решить задачу двумя способами: методом потенциалов и средствами ЭТ Excel («поиск решения»).

Варианты заданий для самостоятельного выполнения

Значение параметра *n* равно последней цифре текущего года.

Вариант 1

Поставщики	Потребители					Запасы поставщиков
	B_1	B_2	B_3	B_4	B_5	
A_1	23	27	26	31	30	22n
A_2	28	22	39	35	29	19n
A_3	41	33	24	36	43	19n
A_4	22	29	21+n	25	47	19n
Спрос потребителей	15n	15n	16n	18n	15n	

Вариант 2

Поставщики	Потребители					Запасы поставщиков
	B_1	B_2	B_3	B_4	B_5	
A_1	14	19	12	13	13	27n
A_2	11	12	11+n	18	16	17n
A_3	17	18	14	15	10	20n
A_4	15	23	16	12	19	16n
Спрос потребителей	14n	14n	24n	18n	10n	

Вариант 3

Поставщики	Потребители					Запасы поставщиков
	B_1	B_2	B_3	B_4	B_5	
A_1	n	4	12	14	10	18n
A_2	11	12	9	5	19	12n
A_3	17	15	4	6	13	19n
A_4	3	5	7	8	2	19n
Спрос потребителей	15n	15n	5n	16n	15n	

Вариант 4

Поставщики	Потребители					Запасы поставщиков
	B_1	B_2	B_3	B_4	B_5	
A_1	3n	10	13	18	19	26n
A_2	2	6	17	9	16	24n
A_3	8	1	14	3	15	21n
A_4	3	12	11	22	9	16n
Спрос потребителей	14n	14n	14n	17n	20n	

Вариант 5

Поставщики	Потребители					Запасы поставщиков
	B_1	B_2	B_3	B_4	B_5	
A_1	23	7n	13	21	14	29n
A_2	18	12	19	3	12	22n
A_3	11	17	24	6	13	21n
A_4	22	19	21	15	7	28n
Спрос потребителей	15n	25n	24n	16n	20n	

Вариант 6

Поставщики	Потребители					Запасы поставщиков
	B_1	B_2	B_3	B_4	B_5	
A_1	42	19	26	13	12	22n
A_2	11	22	10n	18	16	21n
A_3	17	18	14	15	13	28n
A_4	25	29	17	18	22	39n
Спрос потребителей	38n	25n	12n	20n	12n	

Вариант 7

Поставщики	Потребители					Запасы поставщиков
	B_1	B_2	B_3	B_4	B_5	
A_1	18	15	19	24	13	23n
A_2	16	10n	19	15	19	19n
A_3	17	15	24	16	10	37n
A_4	19	12	12	14	17	41n
Спрос потребителей	25n	45n	15n	25n	15n	

Вариант 8

Поставщики	Потребители					Запасы поставщиков
	B_1	B_2	B_3	B_4	B_5	
A_1	3	2n	4	34	19	16n
A_2	26	6	7	9	16	18n
A_3	8	8	7	3	30	25n
A_4	35	25	11	2	9	36n
Спрос потребителей	34n	14n	14n	18n	10n	

Вариант 9

Поставщики	Потребители					Запасы поставщиков
	B_1	B_2	B_3	B_4	B_5	
A_1	23	17	15	21	30	18n
A_2	18	12	19	5n	9	28n
A_3	10	1	24	6	13	48n
A_4	12	19	21	15	27	18n
Спрос потребителей	14n	24n	34n	26n	14n	

Вариант 10

Поставщики	Потребители					Запасы поставщиков
	B_1	B_2	B_3	B_4	B_5	
A_1	14	9	6n	3	12	15n
A_2	21	2	11	18	16	30n
A_3	17	18	4	15	13	25n
A_4	5	19	17	8	6	15n
Спрос потребителей	12n	12n	22n	27n	12n	

Вариант 11

Поставщики	Потребители					Запасы поставщиков
	B_1	B_2	B_3	B_4	B_5	
A_1	21	17	12	24	30	19n
A_2	16	10n	19	15	13	29n
A_3	19	15	24	16	14	19n
A_4	29	22	21	15	11	39n
Спрос потребителей	15n	15n	35n	16n	25n	

Вариант 12

Поставщики	Потребители					Запасы поставщиков
	B_1	B_2	B_3	B_4	B_5	
A_1	33	22	14	34	19	26n
A_2	36	16	37	29	26	47n
A_3	28	18	17	23	30	21n
A_4	35	25	11n	22	39	16n
Спрос потребителей	34n	24n	24n	18n	10n	

Вариант 13

Поставщики	Потребители					Запасы поставщиков
	B_1	B_2	B_3	B_4	B_5	
A_1	13	27	13	22	23	38n
A_2	18	12	7	5n	11	18n
A_3	12	17	24	6	13	28n
A_4	4	19	21	19	17	18n
Спрос потребителей	14n	14n	24n	16n	34n	

Вариант 14

Поставщики	Потребители					Запасы поставщиков
	B_1	B_2	B_3	B_4	B_5	
A_1	14	9	2n	13	12	15n
A_2	12	12	11	18	6	25n
A_3	7	8	4	15	13	25n
A_4	15	10	3	9	16	15n
Спрос потребителей	12n	12n	32n	12n	12n	

Вариант 15

Поставщики	Потребители					Запасы поставщиков
	B_1	B_2	B_3	B_4	B_5	
A_1	3	7	18	21	11	39n
A_2	8	12	9	3n	2	22n
A_3	10	17	14	6	13	21n
A_4	23	14	10	5	7	28n
Спрос потребителей	15n	25n	14n	36n	20n	

Вариант 16

Поставщики	Потребители					Запасы поставщиков
	B_1	B_2	B_3	B_4	B_5	
A_1	14	9n	18	1	12	32n
A_2	11	22	10	18	16	21n
A_3	17	18	14	15	13	38n
A_4	15	19	13	21	6	20n
Спрос потребителей	38n	25n	12n	20n	12n	

2.10. Задача о назначениях

Частным случаем транспортной задачи является задача о назначениях, в которой число пунктов производства равно числу пунктов назначения, то есть транспортная таблица имеет форму квадрата. Кроме того, в каждом пункте назначения объем потребности равен 1, и величина предложения каждого пункта производства равна 1.

Венгерский алгоритм

Алгоритм ищет значения, которые надо вычесть из всех элементов каждой строки и каждого столбца (разные для разных строк и столбцов), такие, что все элементы матрицы останутся неотрицательными, но появится нулевое решение.

Задачей является распределение всех подлежащих назначению единиц в клетки с нулевой стоимостью. Оптимальное значение целевой функции в этом случае равно нулю.

Этап 1:

1. Формализация проблемы в виде транспортной таблицы.
2. В каждой строке таблицы найти наименьший элемент и вычесть его из всех элементов данной строки.
3. Повторить ту же самую процедуру для столбцов. Результат: теперь в каждой строке и в каждом столбце таблицы есть по крайней мере один нулевой элемент.
4. Приведенная транспортная таблица эквивалентна исходной задаче в том смысле, что оптимальное решение для обеих задач будет одним и тем же.

Этап 2.

1. Найти строку, содержащую только одно нулевое значение, в его клетку поместить один элемент (0 обводится квадратиком). Если такие строки отсутствуют, допустимо начать с любого нулевого значения стоимости.
2. Зачеркнуть оставшиеся нулевые значения данного столбца.

• Пункты 1 и 2 повторять до тех пор, пока продолжение описанной процедуры окажется невозможным.

Если на данном этапе окажется, что есть несколько нулей, которым не соответствуют назначения и которые являются незачеркнутыми, то необходимо повторить то же самое для столбцов, то есть:

3. Найти столбец, содержащий только одно нулевое значение, и в соответствующую клетку поместить один элемент.
4. Зачеркнуть оставшиеся нули в данной строке.

• Повторять пункты 3 и 4 до тех пор, пока дальнейшая их реализация окажется невозможной.

• Если окажется, что таблица содержит неучтенные нули, повторять операции этапа 2.

Этап 3.

1. Провести минимальное число прямых через строки и столбцы матрицы (но не по диагоналям) таким образом, чтобы они проходили через все нули, содержащиеся в таблице.
2. Найти наименьший среди элементов, через которые не проходит ни одна из проведенных прямых.
3. Вычесть его из всех элементов, через которые не проходят прямые.
4. Прибавить найденный элемент ко всем элементам таблицы, которые лежат на пересечении проведенных ранее прямых.
5. Все элементы матрицы, через которые проходит только одна прямая, оставить без изменения.

В результате применения данной процедуры в таблице появляется по крайней мере один новый ноль. Необходимо возвратиться к этапу 2 и повторять алгоритм до тех пор, пока не будет получено оптимальное решение.

Лабораторная работа № 4. Тема: решение задачи о назначениях

Пример. Некоторая компания имеет четыре сбытовые базы и четыре заказа, которые необходимо доставить различным потребителям. Складские помещения каждой базы вполне достаточны для того, чтобы вместить один из этих заказов. В таблице содержится информация о расстоянии между каждой базой и каждым потребителем. Как следует распределить заказы по сбытовым базам, чтобы общая дальность транспортировки была минимальной?

Расстояние от сбытовых баз до потребителей

Сбытовая база	Расстояние, миль Потребители			
	I	II	III	IV
A	68	72	75	83
B	56	60	58	63
C	38	40	35	45
D	47	42	40	45

Значения общего спроса и общего предложения для всех строк и столбцов равны единице.

Этап 1 Венгерского метода: приведение транспортной матрицы. В каждой строке находится наименьший элемент.

	Потребители				Наименьший элемент строки
	I	II	III	IV	
A	68	72	75	83	68
B	56	60	58	63	56
C	38	40	35	45	35
D	47	42	40	45	40

Наименьший элемент вычитается из всех элементов соответствующей строки. Далее находится наименьший элемент каждого столбца.

0	4	7	15
0	4	2	7
3	5	0	10
7	2	0	5
0	2	0	5

Наименьший элемент столбца

Найденный наименьший элемент вычитается из всех элементов соответствующего столбца.

0	2	7	10
0	2	2	2
3	3	0	5
7	0	0	0

Этап 2 Венгерского метода: в соответствии с процедурой, описанной в этапе 2, осуществляются назначения. Наличие назначения обозначается через 0, обведенный рамкой.

$\boxed{0}$	2	7	10
~~0~~	2	2	2
3	3	$\boxed{0}$	5
7	$\boxed{0}$	~~0~~	0

Как видим, на данном этапе мы можем осуществить только три нулевых назначения, тогда как требуемое их количество равно четырем. Полученное распределение является недопустимым. Переходим к этапу 3. Проводим наименьшее число прямых, проходящих через все нули таблицы.

0	2	7	10
0	2	2	2
3	3	0	5
7	0	0	0

Наименьшим элементом, через который не проходит ни одна из прямых, является число 2. Скорректируем таблицу так, как это описано выше в соответствии с этапом 3, т.е. вычтем 2 из каждого элемента, через который не проходит ни одна прямая, и добавим 2 ко всем элементам, лежащим на пересечении двух прямых, оставив без изменения все прочие элементы, через которые проходит только одна прямая. Теперь перераспределим соответствующие назначения сбытовых баз и потребителей.

	I	*II*	*III*	*IV*
A	$\boxed{0}$	0	7	8
B	0	$\boxed{0}$	2	0
C	3	1	$\boxed{0}$	3
D	9	0	2	$\boxed{0}$

Теперь требование о размещении четырех назначений в клетки с нулевой стоимостью выполняется, следовательно, полученное решение является оптимальным. Перевозки осуществляются со сбытовой базы A к потребителю I, с базы B — к потребителю II, с базы C—к потребителю III и с базы D — к потребителю IV. Хотя данное решение и является оптимальным, однако оно не единственное. Тем не менее в любом оптимальном решении должен присутствовать маршрут (C,III), поскольку это единственный элемент с нулевой стоимостью в строке C. Два других оптимальных распределения назначений представлены ниже.

Первое альтернативное оптимальное решение

	I	*II*	*III*	*IV*
A	0	0	7	8
B	0	0	2	0
C	3	1	0	3
D	9	0	2	0

Второе альтернативное оптимальное решение

	I	*II*	*III*	*IV*
A	0	0	7	8
B	0	0	2	0
C	3	1	0	3
D	9	0	2	0

Минимальную дальность перевозок для каждого из трех решений можно вычислить из исходной таблицы:

Решение 1: 68 + 60 + 35 + 45 = 208 миль;

Решение 2: 68 + 63 + 35 + 42 = 208 миль;

Решение 3: 72 + 56 +35 + 45 = 208 миль.

Задание.

Найти решение задачи о назначениях. Решить задачу двумя способами: венгерским методом и средствами ЭТ Excel («поиск решения»).

Отдел кадров предприятия устроил конкурсный набор специалистов на две вакантные должности. На эти новые места (НМ) претендуют 3 прежних сотрудника (ПС), уже работающие в других отделах, и 4 новых сотрудника (НС). Номера новых сотрудников, новых и прежних мест выбираются по вариантам из табл.1. Номера прежних мест являются номерами прежних сотрудников.

Отдел кадров оценил по десятибалльной шкале компетентность новых сотрудников (табл.2) и прежних сотрудников (табл.3) для работы и на новых местах, и на прежних местах (ПМ), то есть занимаемых прежними сотрудниками. Необходимо учесть, что руководство предприятия, во-первых, предпочитает, чтобы прежние сотрудники не претендовали на места друг друга, и, во-вторых, не намерено увольнять прежних сотрудников.

Необходимо распределить сотрудников по должностям наилучшим образом.

Особенностью решения данной задачи является моделирование системы предпочтений, сложившейся у руководства предприятия по описанному в условии задачи кадровому вопросу. Увольнение прежнего сотрудника или непринятие на работу нового сотрудника моделируется попаданием единицы в фиктивный столбец матрицы решений задания, поэтому для запрещения таких ситуаций необходимо использовать запрещающие (нулевые) значения компетентностей.

Таблица 1

Номера сотрудников и мест их работы для конкретного варианта

№ варианта	Новые сотрудники (НС)	Места работы прежних сотрудников (ПМ)	Новые места (НМ)
1	3, 4, 7, 8	1, 2, 3	1, 2
2	1, 2, 5, 6	2, 5, 6	2, 3
3	5, 6, 7, 8	1, 2, 5	3, 4
4	3, 4, 5, 6	4, 5, 6	1, 4
5	1, 2, 3, 4	2, 3, 4	2, 4
6	2, 4, 6, 8	3, 4, 6	1, 3
7	1, 3, 5, 7	2, 3, 6	1, 4
8	2, 3, 6, 7	3, 4, 5	2, 3
9	1, 4, 5, 8	2, 3, 5	3, 4
10	2, 3, 4, 5	1, 2, 6	1, 2
11	4, 5, 6, 7	1, 3, 5	2, 4
12	1, 2, 7, 8	2, 4, 6	1, 3

Таблица 2

Компетентность новых сотрудников

	НМ1	НМ2	НМ3	НМ4	ПМ1	ПМ2	ПМ3	ПМ4	ПМ5	ПМ6
НС1	6	5	7	6	5	6	7	6	7	5
НС2	5	5	8	8	7	6	4	5	8	8
НС3	6	7	5	6	4	5	4	5	6	6
НС4	7	8	7	6	5	7	6	8	5	5
НС5	7	6	6	5	5	4	5	5	4	6
НС6	8	8	9	7	6	7	8	7	9	8
НС7	9	8	9	9	8	7	8	9	8	7
НС8	7	7	8	9	7	8	9	6	7	8

Таблица 3

Компетентность прежних сотрудников

	НМ1	НМ2	НМ3	НМ4	Занимаемое место
ПС1	7	6	6	7	7
ПС2	8	9	7	7	8
ПС3	6	5	6	6	6
ПС4	7	9	6	8	8
ПС5	8	7	8	8	7
ПС6	4	5	6	4	5

Тема 3. Статистическое моделирование

3.1. Метод статистического моделирования

В практике моделирования систем наиболее часто приходится иметь дело с объектами, которые в процессе своего функционирования содержат элементы стохастичности или подвергаются стохастическим воздействиям внешней среды. Поэтому основным методом получения результатов с помощью имитационных моделей таких стохастических систем является метод статистического моделирования на ЭВМ, использующий в качестве теоретической базы предельные теоремы теории вероятностей.

Статистическое моделирование представляет собой метод получения с помощью ЭВМ статистических данных о процессах, происходящих в моделируемой системе. Для получения представляющих интерес оценок характеристик моделируемой системы ***S*** с учетом воздействий внешней среды ***E*** статистические данные обрабатываются и классифицируются с использованием методов математической статистики.

Сущность метода статистического моделирования сводится к построению для процесса функционирования исследуемой системы ***S*** некоторого моделирующего алгоритма, имитирующего поведение и взаимодействие элементов системы с учетом случайных входных воздействий и воздействий внешней среды ***E***, и реализации этого алгоритма с использованием программно-технических средств.

Различают две области применения метода:

1) для изучения стохастических систем;

2) для решения детерминированных задач.

Основной идеей, которая используется для решения детерминированных задач методом статистического моделирования, является замена детерминированной задачи эквивалентной схемой некоторой стохастической системы, выходные характеристики последней совпадают с результатом решения детерминированной задачи. Естественно, что при такой замене вместо точного решения задачи получается приближенное решение и погрешность уменьшается с увеличением числа испытаний (реализаций моделирующего алгоритма) ***N***.

В результате статистического моделирования системы ***S*** получается серия частных значений искомых величин или функций, статистическая обработка которых позволяет получить сведения о поведении реального объекта или процесса в произвольные моменты времени. Если количество реализаций ***N*** достаточно велико, то полученные результаты моделирования системы приобретают статистическую устойчивость и с достаточной точностью могут быть приняты в качестве оценок искомых характеристик процесса функционирования системы ***S***.

Теоретической основой метода статистического моделирования систем на ЭВМ являются *предельные теоремы теории вероятностей*.

Множества случайных явлений (событий, величин) подчиняются определенным закономерностям, позволяющим не только прогнозировать их поведение, но и количественно оценить некоторые средние их характеристики, проявляющие определенную устойчивость. Характерные закономерности наблюдаются также в распределениях случайных величин, которые образуются при сложении множества воздействий.

Принципиальное значение предельных теорем состоит в том, что они гарантируют высокое качество статистических оценок при весьма большом числе испытаний (реализаций) ***N***. Практически приемлемые при статистическом моделировании количественные оценки характеристик систем часто могут быть получены уже при сравнительно небольших (при использовании ЭВМ) ***N***. Статистическое моделирование систем на ЭВМ требует формирования значений случайных величин, что реализуется с помощью датчиков (генераторов) случайных чисел.

При использовании статистического моделирования независимо от природы объекта исследования (будет ли он детерминированным или стохастическим) необходимо предварительно построить стохастическую систему, выходные характеристики которой позволяют оценить искомые.

3.2. Генерация случайных величин

Математическое моделирование дискретных систем со стохастическим характером функционирования предполагает использование теории вероятностей. Случайные величины могут быть двух типов:

• дискретные (прерывные), принимающие только отделённые друг от друга значения, которые можно пронумеровать;

• непрерывные (аналоговые), которые могут принимать любое значение из некоторого промежутка.

При статистическом моделировании системы, содержащей <u>случайные компоненты</u>, используются случайные величины с различными законами распределения. В качестве исходной совокупности случайных чисел необходимо выбирать такую совокупность, которая может быть получена с наименьшими затратами машинного времени, и обеспечивает простоту и удобство дальнейших преобразований.

Этим требованиям удовлетворяет <u>стандартная</u> или <u>базовая случайная величина</u>, т.е. непрерывная случайная величина **γ**, равномерно распределенная на интервале [0, 1). *Равномерным распределением называется такое распределение, при котором любое из возможных чисел имеет одинаковую вероятность появления.* С помощью равномерно распределенных случайных чисел можно конструировать случайные величины, обладающие практически любым законом распределения.

Непрерывная случайная величина X распределена равномерно в интервале (a; b), где a<b, если функция F(x) и плотность f(x) распределения соответственно имеют вид:

$$F(x)=\begin{cases}0 & \text{при} \quad x<a;\\ \dfrac{x-a}{b-a} & \text{при} \quad a<x<b;\\ 1 & \text{при} \quad x>b;\end{cases}$$

$$f(x)=\begin{cases}\dfrac{1}{b-a} & \text{при} \quad a<x<b;\\ 0 & \text{при} \quad x>b.\end{cases}$$

Ее <u>функция плотности</u> имеет вид: $f(x)=\begin{cases}1 & \textit{при} \quad 0<x<1\\ 0 & \textit{вне этого интервала}\end{cases}$

<u>Функция распределения</u>: $F(x)=\begin{cases}0, & x\le 0\\ x, & 0<x<1\\ 1, & x\ge 1\end{cases}$

Математическое ожидание: $M(\gamma) = \frac{1}{2}$;

Ср.квадратическое отклонение: $\sigma_\gamma = \frac{1}{2\sqrt{3}}$

На практике используют три способа получения случайных величин: табличный, аппаратный и алгоритмический.

Табличный способ: $\boldsymbol{\gamma = 0,\alpha_1\alpha_2\ldots\alpha_m}$**,** числа берутся из таблицы. Недостатки: любой реальный физический прибор вырабатывает случайные величины с распределением, отличным от равновероятного; запас чисел ограничен; таблицы занимают некоторый объем памяти, обращение к элементу этой таблицы требует времени и замедляет работу алгоритма.

Аппаратный (физический способ):

При физическом способе получения случайных величин используются генераторы случайных чисел. Чаще всего для построения датчика используют «шумящие» электронные приборы. В каждый момент времени датчиком выдается приближенное случайное число $\boldsymbol{\gamma = 0,\alpha_1\alpha_2\ldots\alpha_m}$**,** записанное в форме **m**-разрядной двоичной дроби. Этот метод был предложен А.М.Тьюрингом. Однако датчики имеют свои недостатки: числа, выработанные датчиком, нельзя воспроизвести; приходится содержать и эксплуатировать дополнительное устройство, которое требует и регулярной проверки «качества» вырабатываемых чисел.

Алгоритмический (программный) способ

Пригодность случайных чисел определяется, в конечном счете, не процессом их получения, а тем, удовлетворяют ли они некоторым требованиям. Сформулируем набор требований к идеальной последовательности случайных чисел: она должна состоять из:

1) равномерно распределенных;

2) статистически независимых;

3) воспроизводимых;

4) неповторяющихся чисел. Кроме этого, генератор должен

5) работать быстро, т.е. количество операций, затрачиваемых на формирование одного числа, должно быть небольшим и

6) занимать минимальный объем машинной памяти.

Числа, получаемые с помощью какого-либо алгоритма и имитирующие значения случайной величины γ, называются псевдослучайными числами.

Большинство алгоритмов, используемых для получения псевдослучайных чисел, представляют собой рекуррентные формулы вида: $\boldsymbol{\gamma_{n+1} = F(\gamma_n)}$**,** где начальное число $\boldsymbol{\gamma_0}$ задано. Важная черта алгоритмов рекуррентного вида – они всегда порождают периодические последовательности. Можно записать лишь конечное количество N чисел, заключенных между нулем и единицей, следовательно, рано или поздно значение $\boldsymbol{\gamma_L}$ совпадет с одним из предыдущих значений $\boldsymbol{\gamma_m}$. Пусть $\boldsymbol{L}$ – наименьшее число, удовлетворяющее этому условию **($m<L$);** множество чисел $\boldsymbol{\gamma_0}$, $\boldsymbol{\gamma_1}$, **…** $\boldsymbol{\gamma_{L-1}}$ называется отрезком апериодичности последовательности, число $\boldsymbol{L}$ – длиной отрезка апериодичности, а $\boldsymbol{P = L - m}$ – длиной периода. Отрезок апериодичности состоит из различных чисел. Для экспериментального определения $\boldsymbol{L}$ и $\boldsymbol{P}$ используется следующий алгоритм.

1. Запускается программа генерации последовательности $\boldsymbol{\{\gamma_i\}}$ с начальным значением $\boldsymbol{\gamma_0}$ и генерируется N (10^7-10^9) чисел $\boldsymbol{\gamma_i}$. Число $\boldsymbol{\gamma_N}$ фиксируется. Затем программа запускается повторно с начальным числом $\boldsymbol{\gamma_0}$ и при генерации очередного числа проверяется истинность события $\mathbf{C}\ \boldsymbol{\{\gamma_i = \gamma_N\}}$. Если это событие истинно при $\mathbf{i = i_1}$ и $\mathbf{i = i_2}$ **(**$\mathbf{i < i_1 < i_2 < N}$**),** то вычисляется **длина периода** последовательности $\boldsymbol{P} = \mathbf{i_2 - i_1}$.

2. Для определения **отрезка апериодичности** проводят запуск генератора с начальным числом $\boldsymbol{\gamma_0}$ и вырабатывают им $\boldsymbol{P}$ чисел, где $\boldsymbol{P}$ – длина периода. После этого параллельно с ним генерируют случайные числа вторым генератором, аналогичным первому. Таким образом вырабатывают случайные числа двумя генераторами, продолжая подсчитывать количество чисел, которое сгенерировал первый генератор, до совпадения между собой чисел, выработанных разными генераторами. Подсчитанное количество чисел является длиной отрезка апериодичности. Замечание: если событие **C** оказывается истинным для $\mathbf{i = N}$, то $\mathbf{L > N}$.

Методы получения псевдослучайных чисел

Метод середины квадратов (метод Неймана)

В этом методе число γ_n предполагается **2k**-значным: $\gamma_n = 0, a_1 a_2 \ldots a_{2k}$.

Чтобы получить число γ_{n+1} надо γ_n возвести в квадрат $\gamma_n^{\,2} = 0, b_1 b_2 \ldots b_{4k}$, и затем отобрать среднике 2k цифр этого квадрата: $\gamma_n = 0, b_{k+1} b_{k+2} \ldots b_{3k}$.

Этому методу соответствует функция $F(x) = \{\ 10^{-2k} [10^{3k} \gamma^{\,2}]\ \}$.

Недостаток подобных методов – наличие корреляции между числами последовательности, в последовательности получается больше чем нужно малых значений.

Модификацией метода квадратов является **метод произведений**: два следующих друг за другом случайных числа умножают и из средних разрядов произведения выделяют следующее случайное число.

Конгруэнтные методы

Два целых числа a и b называются конгруэнтными (сравнимыми) по модулю m, где m – целое число, если разность (b a −) делится на m без остатка, а числа a и b дают одинаковые остатки от деления на m. Например, 2568 и 148 (по модулю 10), 1746 и 511 (по модулю 5), 6493 и 2221 (по модулю 2) и т.д.

Пусть X_0, g, c, m – произвольные фиксированные элементы. Принято говорить, что последовательность X_1, X_2, … порождается линейным конгруэнтным генератором с параметрами (X_0, g, c, m), если эта последовательность образована с помощью следующего рекуррентного соотношения: $\mathbf{X_{n+1} = (g \cdot X_n + c) \bmod m}$, $n \geq 0$, где **m** – модуль, m>0; **g** – множитель, $0 \leq$ **g** <m; **c** – приращение (инкремент), $0 \leq$**c**<m;

$\mathbf{X_0}$ – начальное значение (база генератора), $0 \leq \mathbf{X_0} < m$.

Здесь X_{n+1} равно остатку, полученному при делении (gX_n + c) на m. Случайное число из интервала (0, 1) получается путем деления очередного элемента на m: $\boldsymbol{\gamma_n = X_n / m}$.

При **c = 0** формула преобразуется к виду $\mathbf{X_{n+1} = g \cdot X_n \bmod m}$, и определяет мультипликативный конгруэнтный генератор для получения случайных чисел.

Параметры **($\mathbf{X_0}$, g, m)** выбираются из условия максимума периода. Их можно определить аналитическими методами теории чисел или экспериментально. Длина периода равна **m** тогда и т.тогда, когда:

1) НОД(c, m) = 1 (т.е. C и m взаимно просты);
2) g-1 кратно p для всех простых p – делителей m;
3) g-1 кратно 4, если m кратно 4.

Квадратичный конгруэнтный метод

$X_{n+1} = (dX_n^2 + g \cdot X_n + c) \bmod m.$

Для m, которые являются степенью 2, интересный квадратичный метод предложил Ковэю:

$X_{n+1} = (X_n (X_n + 1)) \bmod 2^e$,

где модуль $\mathbf{m=2^e}$, а начальное значение $\mathbf{X_0}$ выбирается так, чтобы $\mathbf{X_0 \bmod 4 = 2}$.

Конгруэнтный метод с переносом

Псевдослучайная нелинейная конгруэнтная последовательность определяется следующим нелинейным рекуррентным соотношением: $X_{n+1} = (g \cdot X_n + c_n) \bmod m$, **n=1,2,…** В отличие от линейного конгруэнтного генератора, приращение **c** изменяется во времени:

$$c_n = \frac{gX_{n-1} + c_{n-1}}{m}.$$

Генераторы Фибоначчи (генераторы с запаздываением)

Генераторы этого типа используют некоторый начальный набор целых чисел x_1, x_2, … x_n, определяющих «стартовое слово» и два числа **a, b,** называемых запаздыванием или лагами. Алгоритм генерации похож на алгоритм генерации чисел Фибоначчи:

$X_n = (X_{n-a} \bullet X_{n-b}) \bmod m$, где $\bullet$ – символ бинарной операции, в качестве которой может использоваться одна из следующих операций: +, -, *, XOR (исключающее **или** для двоичных слов). Примером такого

генератора является аддитивный генератор Митчелла и Мура: $X_n = (X_{n-24} + X_{n-55}) \bmod m$,
где m – четное число, а $X_0 \ldots X_{54}$ – произвольные не все четные числа.

Сложные генераторы

Метод перемешивания

Одним из способов «улучшить» последовательность случайных чисел и увеличить период, является «перемешивание» или «тасование» двух различных генераторов случайных чисел. Рассмотрим метод Макларена-Марсальи (1965г.), который основан на комбинировании двух простейших датчиков.

Пусть $\{a_i\}$, $\{b_i\}$, i=1, 2, … – псевдослучайные последовательности, порождаемые независимо работающими генераторами Ran1 и Ran2 соответственно. V $\{V_0, V_1, \ldots, V_{k-1}\}$ – вспомогательная таблица К чисел. Вначале **V**-таблица заполняется **К** членами последовательности $\{a_i\}$. Через $\{c_i\}$ будем обозначать комбинированную псевдослучайную последовательность. Каждый элемент c_i является результатом последовательности операций: $ind = [b_i * K]$; $c_i = V_{ind}$; $V_{ind} = a_{i+K}$ (i = 1, 2, …).

Отсюда видно, что Ran2 осуществляет случайный выбор индекса, выбранное из V-таблицы число становится очередным элементом результирующей последовательности, а элемент таблицы заменяется новым числом, порожденным генератором Ran1.

Недостаток: методы перемешивания изменяют порядок следования генерируемых чисел, но не сами числа. Не позволяют стартовать с заданного места в периоде.

Комбинирование алгоритмов

Простой вероятностный метод: генерируется 3 случайных числа тремя генераторами γ_1 γ_2 γ_3. Элемент результирующей последовательности равен γ_2, если γ_1>альфа, где альфа – вероятность использования данного генератора и γ_3 в противном случае. При таком методе период главной функции равен или больше суммы периодов второго и третьего генераторов.

Вичманн и Хилл (1982 г.) предложили следующую идею для объединения трех генераторов с целью достижения длинного периода, достижения переносимости и скорости работы: если $\boldsymbol{\gamma_{1i}}$, $\boldsymbol{\gamma_{2i}}$, $\boldsymbol{\gamma_{3i}}$ – i-е случайные числа, произведенные тремя отдельными генераторами, тогда положим $\boldsymbol{\gamma_i}$ будет дробной частью величины $\boldsymbol{\gamma_i = \{\gamma_{1i} + \gamma_{2i} + \gamma_{3i}\}}$.

Комбинирование с помощью псевдослучайного прореживания

Пусть генератор Ran1 порождает «элементарную» двоичную последовательность $\{a_i\}$, а генератор Ran2 – двоичную «селектирующую» $\{b_i\}$. С помощью этих двух последовательностей строится выходная последовательность, включающая те биты, для которых соответствующее значение селектора b_i =1, если b_i=0, то значение a_i отбрасывается.

3.3. Методы тестирование генераторов псевдослучайных чисел

Достоверность и точность результатов имитационного моделирования в значительной степени определяется качеством используемых в моделях программных генераторов псевдослучайных последовательностей. Большинство простых арифметических генераторов хотя и обладают большой скоростью, но страдают от многих серьезных недостатков:

- слишком короткий период;
- последовательные значения не являются независимыми;
- некоторые биты "менее случайны", чем другие;
- неравномерное одномерное распределение.

Поскольку БСВ используются при моделировании всех других случайных элементов, качество датчиков должно проверяться особенно тщательно. Проверка генераторов равномерно распределенных псевдослучайных чисел предполагает формирование представительной выборки случайных чисел и выполнение множества проверочных тестов, позволяющих оценить качество генераторов.

Существуют тесты двух видов: эмпирические – это обычный тип статистических тестов, они основаны на действительных значениях выдаваемых генератором, и теоретические, в которых используются числовые параметры, чтобы оценить генератор глобально без фактического генерирования значений псевдослучайных чисел.

Самыми простыми тестами являются тесты проверки пар и проверки комбинаций.

Тест проверки пар заключается в подсчете количества «1» для каждого разряда случайного числа в двоичном представлении. Теоретическая вероятность появления «1» для равномерно распределенных случайны х чисел равна ½.

Тест проверки комбинаций сводится к подсчету количества «1» в двоич. случайных числах. Теоретическая вероятность появления комбинации с i-м количеством «1» будет

$$P_i = \frac{k!}{i!\,(k-i)!\,2^k}\,.$$ где k – количество разрядов СЧ.

Тест совпадения моментов

Базовая случайная величина имеет математическое ожидание **M=1/2** и дисперсию **D=σ²=1/12**. Пусть в результате N-кратного обращения к датчику БСВ получена выборка. Оценками моментов являются выборочное среднее и дисперсия:

$$M = \frac{1}{N}\sum_{i=1}^{N} X_i; \quad D = \sigma^2 = \frac{1}{N-1}\sum_{i=1}^{N}(X_i - M)^2\,.$$

Проверим гипотезу H_0: базовая случайная величина распределена равномерно на [0, 1).

Для выборочного значения математического ожидания следует проверить, попадает ли оно в доверительный интервал 0,5-d< M < 0,5+d с заданным уровнем значимости α=0,05, где 0,5 – теоретическое значение математического ожидания случайной величины; ***d*** – доверительный интервал.

Для определения границ доверительного интервала можно воспользоваться формулой: $d = t_\alpha \cdot s/\sqrt{n}$, где n – количество элементов в выборке; s – стандартное отклонение (корень из дисперсии); $t_\alpha = \Phi^{-1}\left(1-\frac{\alpha}{2}\right)$ – квантиль ф-и распределения Лапласа.

Критерий равномерности (критерий частот). Критерий хи-квадрат

Пусть имеется выборка случайных чисел $\mathbf{X=\{x_1, x_2, \ldots, x_n\}}$. Необходимо проверить гипотезу H_0: функция распределения случайной величины совпадает с некоторой заданной функцией распределения.

Находим в множестве X минимальное и максимальное значение и разбиваем числовую прямую на **K>1** ячеек с шириной **h = (max (X) – min (X)) / K**.

K рекомендуется выбирать равным [1+3,3 lg n]

Пусть P_i – вероятность того, что случайное число попало в i-ю ячейку, а Y_i – число действительных попаданий X в i-ю ячейку. Образуем статистику:

$$\chi^2 = \sum_{i=1}^{K} \frac{(Y_i - n \cdot P_i)^2}{n \cdot P_i}, \text{ причем } \sum_{i=1}^{K} Y_i = n, \ \sum_{i=1}^{K} P_i = 1.$$

Гипотеза принимается, если $\boldsymbol{\chi^2 < \Delta}$ при заданном уровне значимости $\boldsymbol{\alpha}$ для **K-1** степени свободы. Число степеней свободы на 1 меньше, чем число категорий, т.к. Y_i не являются полностью независимыми: если известны $\mathbf{Y_1, Y_2, \ldots, Y_{K-1}}$, то $\mathbf{Y_K}$ вычисляется.

Лабораторная работа № 5. Тема: получение базовой случайной величины на ЭВМ

Задание

1. Реализовать на ЭВМ конгруэнтный метод получения равномерно распределенных случайных чисел. Значения параметров по вариантам:

Вар.	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
g=	33	51	23	11	11	14	15	35	17	19	22	37	15	11	23	65
m=	96	100	101	100	101	99	98	101	103	99	140	115	121	97	95	89
c=	1	1	3	1	3	1	1	6	5	4	3	5	3	1	3	1
X_0=	3	5	10	10	5	4	3	17	2	2	1	1	4	1	5	2

2. Для каждого набора параметров сгенерировать выборку из 100 псевдослучайных целых чисел. Определить длину периода и отрезок апериодичности для полученной выборки. Для определения длины периода можно воспользоваться функцией =СЧЁТЕСЛИ(), которая позволяет найти заданное значение в диапазоне ячеек. Используя полученные числа, создать выборку из равномерно распределенных в интервале [0,1] псевдослучайных величин.
3. Проверить выборку на равномерность путем построения гистограммы частот (*Сервис, Анализ данных, Гистограмма*) и оценки параметров равномерного распределения: математического ожидания =СРЗНАЧ() и дисперсии =ДИСП(). Для равномерно распределенной в интервале [0,1] случайной величины математическое ожидание должно быть равно M=0,5, а дисперсия D=0,0833.
4. Вычислить значение хи-квадрат, сравнить полученное значение с табличным (для 10 интервалов разбиения табличное значение хи-квадрат равно 16,92). Сделать выводы о качестве полученной выборки.
5. Используя метод середины квадратов, сгенерировать выборку из 100 равномерно распределенных в интервале [0,1] псевдослучайных чисел. Базу генератора задать самостоятельно. Выполнить для данного метода п.2,3.

3.4. Моделирование случайной величины с заданным законом распределения

При решении различных задач приходится моделировать различные случайные величины. Раньше для нахождения каждой случайной величины исследователи пытались строить свою рулетку с неравномерными делениями, пропорциональными p_i. Однако это оказалось совершенно ненужным: значения любой случайной величины можно получить путем преобразования «стандартной» случайной величины γ, равномерно распределенной на (0,1). Процесс нахождения значения случайной величины ξ путем преобразования стандартной случайной величины называют разыгрыванием или моделированием случайной величины ξ.

Метод обратного преобразования (обратной функции)

Предположим, необходимо генерировать случайную величину ξ, являющуюся непрерывной и имеющею функцию распределения $0<F(x)<1$ непрерывную и строго возрастающую. Пусть F^{-1} – это обратная функция F. Тогда алгоритм генерирования СВ с функцией распределения F будет следующим: 1) генерируем $\boldsymbol{\gamma}$; 2) возвращаем $\xi = F^{-1}(\boldsymbol{\gamma})$. Обратите внимание, $F^{-1}(\boldsymbol{\gamma})$ всегда будет определено, поскольку БСВ $0<=\gamma<=1$.

Можно воспользоваться известной функцией плотности вероятности. Допустим, что нужно получать значения случайной величины $\boldsymbol{\xi}$, распределенной в интервале **(a, b)** с плотностью **f(x)>0**. Значение случайной величины **ξ** методом обратной функции определяется в результате решения уравнения:

$$\int_a^{\xi} f(x)dx = \gamma$$

Т.е. выбрав очередное значение **γ**, надо решить данное уравнение и найти очередное значение **ξ**. Достоинства: точность метода. Недостатки:

- метод распространяется только на те функции, для которых известно аналитическое выражение функции распределения или которые позволяют вычислить интеграл от функции плотности аналитически;
- выражение, используемое для вычислений, как правило содержит в себе логарифмы, возведение в степень и т.д., что требует значительных затрат машинного времени.

Рассмотрим дискретную случайную величину ξ с заданным распределением, где pi = P{ξ = xi}:

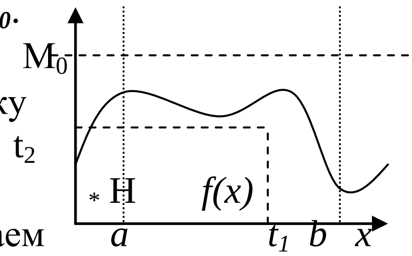
x_1	x_2	x_3	…	x_n
p_1	p_2	p_3	…	p_n

Для того, чтобы разыграть значения этой величины разделим интервал **0<=y<=1** на **n** интервалов, длины которых равны $\mathbf{p_1, p_2, \ldots p_n}$. Координатами точек деления будут $\mathbf{y=p_1}$, $\mathbf{y= p_1+ p_2}$, … $\mathbf{y= p_1+ p_2+\ldots+p_{n-1}}$. Полученные интервалы занумеруем числами 1, 2, …, n. Каждый раз, когда нам надо будет «поставить опыт» и разыграть значение **ξ**, мы будем выбирать значение **γ** и строить точку **y=γ.** Если точка попадет в интервал с номером **i** то будем считать, что $\mathbf{\xi=x_i}$.

Метод отбора-отказа (метод Неймана, 1951)

Может оказаться, что разрешить уравнение обратной функции для ***ξ*** трудно, например, в случае, когда интеграл ***f(x)*** не выражается через элементарные функции или когда плотность ***f(x)*** задана графически.

Предположим, что случайная величина **ξ** определена на конечном интервале ***(a, b)*** и плотность ее ограничена $f(x)<=M_0$.



Разыгрывать **ξ** можно следующим способом:

1) выбираем два значения γ_1 и γ_2 и строим точку $\mathbf{H(t_1,t_2)}$ с координатами:

$\mathbf{t_1} = a+\gamma_1(b-a)$; $\mathbf{t_2} = \gamma_2 M_0$.

2) Если точка лежит под кривой ***f(x),*** то полагаем $\xi=t_1$, если над – то пару (γ_1, γ_2) отбрасываем и выбираем новую пару значений.

3.5. Контрольные вопросы по теме

1. Что представляет собой метод статистического моделирования?
2. Каковы статистические характеристики базовой случайной величины?
3. Какие методы используются для генерации базовой случайной величины?
4. Как можно оценить качество генератора псевдослучайных чисел?
5. Какими методами можно получить последовательность случайных чисел с заданным законом распределения?
6. Что представляет собой метод обратной функции преобразования случайных величин?
7. Как с помощью метода отбора-отказа можно получить случайную величину?

Лабораторная работа № 6. Тема: получение случайных величин с заданным законом распределения

Задание 1 Моделирование дискретной случайной величины

Дискретные случайные величины X и Y заданы законами распределений. С помощью имитационного эксперимента построить законы распределения величин Z1=X+Y и Z2=XY. По выборке вычислить математическое ожидание и дисперсию. Сравнить полученные значения с теоретическими. Показать с помощью критерия хи-квадрат совпадение эмпирического и теоретического законов распределения.

Варианты заданий для самостоятельного выполнения

1.

X	-1	3	**Y**	1	2	3	4	5	6
p(x)	0,5	0,5	**p(y)**	0,6	0,1	0,05	0,1	0,05	0,1

2.

X	2	3	4	**Y**	-2	-1	0	1	2
p(x)	0,3	0,4	0,3	**p(y)**	0,02	0,28	0,5	0,1	0,1

3.

X	-1	0	1	2	**Y**	1	2	3	4
p(x)	0,35	0,2	0,05	0,4	**p(y)**	0,1	0,2	0,3	0,4

4.

X	-2	-1	1	2	3	Y	2	4	6
p(x)	0,5	0,1	0,1	0,1	0,2	p(y)	0,5	0,45	0,05

5.

X	-3	-2	-1	0	1	2	Y	1	3
p(x)	0,2	0,2	0,1	0,3	0,1	0,1	p(y)	0,2	0,8

6.

X	2	5	Y	-1	-2	1	2	3	4
p(x)	0,9	0,1	p(y)	0,25	0,1	0,3	0,15	0,15	0,05

7.

X	1	2	3	Y	1	2	3	4	5
p(x)	0,1	0,6	0,3	p(y)	0,3	0,25	0,1	0,05	0,3

8.

X	-2	0	2	4	Y	1	2	3	4
p(x)	0,2	0,2	0,2	0,4	p(y)	0,2	0,2	0,2	0,4

9.

X	-3	-1	1	2	3	Y	1	2	3
p(x)	0,35	0,2	0,15	0,1	0,2	p(y)	0,1	0,8	0,1

10.

X	-2	-1	0	1	2	3	Y	3	6
p(x)	0,05	0,2	0,2	0,15	0,1	0,3	p(y)	0,6	0,4

11.

X	0	1	Y	0	2	3	4	5	6
p(x)	0,3	0,7	p(y)	0,1	0,1	0,1	0,5	0,1	0,1

12.

X	-1	0	1	Y	2	3	4	5	6
p(x)	0,45	0,3	0,25	p(y)	0,3	0,2	0,1	0,3	0,1

Задание 2. Моделирование непрерывной случайной величины, заданной интегральной функцией распределения.

1. На основе метода обратной функции составить программу моделирования случайной величины с заданной функцией распределения. Получить выборку случайных чисел. Проверить совпадение эмпирического и теоретического закона распределения с помощью критерия хи-квадрат.
2. Найти функцию плотности вероятности. На основе метода отбора-отказа (метод исключения, метод Неймана) составить программу моделирования случайной величины с заданной функцией плотности вероятности. Получить выборку случайных чисел. По выборке построить эмпирическую функцию плотности вероятности.

Варианты заданий для самостоятельного выполнения

1. $$F(x)=\begin{cases}0 \ \ \textit{при} \ \ x\le 2;\\ \dfrac{x^3-8}{19} \ \ \textit{при} \ \ 2<x\le 3;\\ 1 \ \ \textit{при} \ \ x>3.\end{cases}$$

2. $$F(x)=\begin{cases}0 \ \ \textit{при} \ \ x\le 1;\\ a(x-1)^2 \ \ \textit{при} \ \ 1<x\le 3;\\ 1 \ \ \textit{при} \ \ x>3.\end{cases}$$

3. $$F(x)=\begin{cases}0 \ \ \textit{при} \ \ x\le -\dfrac{\pi}{2};\\ 0.5(1+\sin(x)) \ \ \textit{при} \ -\dfrac{\pi}{2}<x\le \dfrac{\pi}{2};\\ 1 \ \ \textit{при} \ \ x>\dfrac{\pi}{2}.\end{cases}$$

4. $F(x)=\begin{cases}0 \ \ при \ \ x \le 0; \\ \dfrac{16}{25}x^2 \ \ при \ 0 < x \le \dfrac{5}{4}; \\ 1 \ \ при \ \ x > \dfrac{5}{4}.\end{cases}$

5. $F(x)=\begin{cases}0 \ \ при \ \ x \le 1; \\ \dfrac{x^2-1}{8} \ \ при \ \ 1 < x \le 3; \\ 1 \ \ при \ \ x > 3.\end{cases}$

6. $F(x)=\begin{cases}0 \ \ при \ \ x \le 0; \\ \sin(2x) \ \ при \ 0 < x \le \dfrac{\pi}{4}; \\ 1 \ \ при \ \ x > \dfrac{\pi}{4}.\end{cases}$

7. $F(x)=\begin{cases}0 \ \ при \ \ x \le 100; \\ 1-\left(\dfrac{100}{x}\right)^3 \ \ при \ \ x > 100.\end{cases}$

8. $F(x)=\begin{cases}0 \ \ при \ \ x \le 1; \\ \dfrac{(x-1)^2}{25} \ \ при \ \ 1 < x \le 6; \\ 1 \ \ при \ \ x > 6.\end{cases}$

9. $F(x)=\begin{cases}0 \ \ при \ \ x \le 3; \\ \dfrac{x^4-81}{175} \ \ при \ \ 3 < x \le 4; \\ 1 \ \ при \ \ x > 4.\end{cases}$

10. $$F(x)=\begin{cases}0\ \ при\ \ x\le -2;\\ \dfrac{x^3+8}{16}\ \ при\ \ -2<x\le 2;\\ 1\ \ при\ \ x>2.\end{cases}$$

11. $$F(x)=\begin{cases}0\ \ при\ \ x\le 0;\\ 1-e^{-2x}\ \ при\ \ x>0.\end{cases}$$

12. $$F(x)=\begin{cases}0\ \ при\ \ x\le -2;\\ \dfrac{(x+2)^3}{216}\ \ при\ \ -2<x\le 4;\\ 1\ \ при\ \ x>4.\end{cases}$$

Тема 4. Имитационное моделирование

4.1. Метод имитационного моделирования

Имитационное моделирование – это вид компьютерного моделирования, для которого характерно воспроизведение на ЭВМ (имитация) процесса функционирования исследуемой сложной системы. При этом имитируются элементарные явления, составляющие процесс, с сохранением их логической структуры, последовательности протекания во времени, что позволяет получить информацию о состоянии системы в заданные моменты времени.

Имитационное моделирование систем можно использовать для решения исследовательских задач в следующих случаях:

1. Если не существует законченной постановки задачи и идет процесс познания объекта моделирования. Имитационная модель служит средством изучения явления.

2. Если аналитические методы имеются, но математические процедуры столь сложны и трудоемки, что имитационное моделирование дает более простой способ решения задачи.

3. Когда имитационное моделирование оказывается единственным способом исследования системы из-за невозможности наблюдения явлений в реальных условиях.

4. Когда необходимо контролировать протекание процессов в системе путем замедления или ускорения явлений в ходе имитации.

5. Когда изучаются новые ситуации в системе, о которых мало что известно или неизвестно ничего. В этом случае имитация служит для предварительной проверки новых стратегий и правил принятия решений перед проведением экспериментов на реальной системе.

В имитационной модели предполагается, что систему можно описать в терминах, понятных ЭВМ. Система и объекты, из которых состоит система, характеризуются набором переменных, каждая комбинация значений которых описывает ее конкретное состояние. Путем изменения значений переменных можно имитировать переход системы из одного состояния в другое.

Когда результаты, полученные при воспроизведении на имитационной модели процесса функционирования системы, являются реализациями случайных величин и функций, т.е. когда параметры или связи являются стохастическими (случайными), тогда для достижения целей моделирования требуется его многократное воспроизведение с последующей статистической обработкой информации.

Если бы на ЭВМ имитировалось поведение только одной компоненты системы, то выполнение активностей можно было бы осуществить строго последовательно и дело свелось бы к пересчету временной координаты t_i после очередного выполнения алгоритма A_{ij}. В действительности сложная система, как правило, состоит из нескольких компонентов, которые могут функционировать одновременно. Однако, в распоряжении исследователя находится, как правило, однопроцессорная вычислительная система, поэтому действительно параллельная реализация всех компонент модели невозможно.

4.2. Управление модельным временем

Необходимость представления параллельных по времени процессов в виде последовательного алгоритма приводит к введению глобальной переменной, называемой (системным) модельным временем.
Существует еще две причины, по которым необходимо введение модельного времени.

1. Процессы в имитационной модели идут со скоростью, отличной от скорости их реализации в реальной системе. Модельное время используется для изменения масштаба времени исследуемой системы.
2. Программа для ЭВМ должна оперировать с конечным множеством данных, и, следовательно, имитировать поведение системы не во все моменты времени, а лишь в некоторые, составляющие конечное множество.

При разработке ИМ и планировании эксперимента следует соотносить между собой три представления времени: реальное, модельное время и машинное время, отражающее затраты времени на ЭВМ для проведения имитации.

Существуют два способа формирования конечного множества моментов модельного времени.

Первый способ – изменение модельного времени с **постоянным шагом**. При имитации первым способом интервал времени моделирования разбивается на интервалы фиксированной длины Δt. Затем на каждом интервале последовательно обрабатываются алгоритмы, имитирующие компоненты моделируемой системы, после чего происходит изменение модельного времени.

Для реализации принципа фиксированных интервалов математическая модель имитируемой системы должна быть предварительно преобразована к виду, допускающему рекуррентные вычисления.

Точность моделирования достигается ценой больших затрат машинного времени, находящихся в обратно пропорциональной зависимости от величины Δt. Обычно принцип фиксированных интервалов используется при построении моделей непрерывных динамических систем, описываемых системами дифференциальных уравнений различных типов. (Метод Эйлера)

В других случаях принцип Δt целесообразно использовать, если:

- события появляются регулярно и распределены равномерно на всем интервале моделирования и исследователь может подобрать интервал изменения Δt, обеспечивающий минимальную погрешность имитации;
- событий очень много и они появляются группами;
- невозможно заранее определить моменты появления событий.

Второй способ – моделирование по особым состояниям. При рассмотрении некоторых сложных систем можно обнаружить существенную неравноправность состояний системы в заданном интервале времени моделирования T. Выделяют два вида состояний: неособые состояния, в которых система находится почти все время, и особые состояния, характерные для системы в некоторые моменты времени, совпадающие с моментами наступления событий в реальной системе (поступление очередной заявки, освобождение канала и т.д.).

При имитации вторым способом функционирование системы рассматривается как совокупность взаимосвязанных процессов, протекающих параллельно во времени, причем каждый из процессов представляет собой некоторый поток событий. Каждому отдельному событию соответствует изменение состояния системы.

При построении модели системного времени методом особых состояний время меняется в моменты, которые соответствуют моментам наступления событий или, что то же самое, в моменты особых состояний.

Для реализации моделирования по особым состояниям требуется разработка специальной процедуры планирования событий (календаря событий).

Если известен закон распределения интервалов между событиями, то такое прогнозирование труда не составляет: достаточно к текущему значению модельного времени добавить величину интервала, полученную с помощью датчика случайных чисел с соответствующим законом распределения.

Если же момент наступления события определяется некоторыми логическими условиями, то необходимо сформулировать эти условия и проверять их истинность для каждого последующего шага моделирования.

Моделирование по особым состояниям экономичнее и точнее способа фиксированного шага, если:

- события распределены неравномерно и появляются через значительные временные интервалы;
- предъявляются повышенные требования к точности определения взаимного положения событий по времени;
- необходимо учитывать наличие одновременных событий.

Моделирующие алгоритмы для таких систем, построенные по принципу фиксированного шага, оказываются не особенно эффективными. В самом деле, при малом Δt будет затрачено очень много машинного времени на бесполезное определение большого количества неособых состояний. Если же Δt сделать недостаточно малым, появится опасность пропуска некоторых особых состояний, что вообще исключает возможность получения правильных результатов при моделировании. Правда, в некоторых случаях для предотвращения пропуска особых состояний могут быть приняты специальные меры, например, разработаны приемы обнаружения пропуска и возврат к предыдущему моменту времени для повторного прохождения данного интервала с малым Δt.

Т.о. выбор величины шага моделирования является очень важным делом. Универсальной методики решения этой задачи не существует, но во многих случаях можно использовать следующие подходы:

- принимать величину шага равной средней интенсивности возникновения событий различных типов;
- выбирать величину шага равной среднему интервалу между наиболее частыми или наиболее важными событиями.

На практике не всегда строго выдерживается один из перечисленных принципов построения моделирующих алгоритмов. Иногда моделирующие алгоритмы строятся на нескольких принципах одновременно. Например, общая структура моделирующего алгоритма базируется на принципе особых состояний, а между особыми состояниями используется принцип Δt.

4.3. Метод Монте-Карло.

Это численный метод исследования математических моделей сложных систем, основанный на моделировании случайных элементов с последующей статистической обработкой результатов. В основе метода Монте-Карло лежат различные предельные соотношения теории вероятностей – законы больших чисел и предельные теоремы.

Следует отметить следующие особенности метода Монте-Карло:

1) простая структура вычислительного алгоритма; метод требует обычно меньших вычислительных затрат, чем традиционные численные методы;
2) погрешность вычисления как правило пропорциональна $\sqrt{D/N}$**,** где **D** – некоторая постоянная, а **N** – число испытаний. Отсюда видно, что чтобы уменьшить погрешность в 10 раз, надо увеличить количество испытаний в 100 раз.

Общая схема метода Монте-Карло

Пусть требуется вычислить скалярную величину **m,** заданную некоторым математическим выражением, сложным для аналитического исследования. Метод М-К реализуется с помощью следующих шагов:

1) подберем такую случайную величину ξ, чтобы матожидание $\mathbf{M(\xi)=m}$;
2) моделируем **N** независимых реализаций случайной величины: $\xi_1, \xi_2, \dots \xi_N$;
3) по случайной выборке $\{\xi_i\}$, **i=1, …, N** строим выборочную оценку $m = \dfrac{1}{N}\sum_{i=1}^{N}\xi_i$.

Оценим погрешность метода. Если **N** достаточно велико, то согласно центральной предельной теореме, распределение суммы $\rho_N=\xi_1+\xi_2+\dots+\xi_N$ будет приблизительно нормальным с параметрами **a=Nm**, σ**=b**$\sqrt{N}$ **.**

Из правила «трех сигм» следует, что

$$P\{Nm - 3b\sqrt{N} < \rho_N < Nm + 3b\sqrt{N}\} \approx 0.997 .$$

Если мы разделим выражение, стоящее в фигурной скобке, на **N**, то получим эквивалентное неравенство, и вероятность его останется такой же:

$$P\{m - 3b/\sqrt{N} < \rho_N / N < m + 3b/\sqrt{N}\} \approx 0.997$$

или в другом виде: $P\{\left|\dfrac{1}{N}\sum_{i=1}^{N}\xi_i - m\right| < 3b/\sqrt{N}\} \approx 0.997$

Очевидно, эта погрешность стремится к нулю с ростом **N.**

На практике очень часто предпочитают ориентироваться не на оценку сверху **3b/**$\sqrt{N}$, а на вероятную ошибку, которая приближенно равна $r_N = 0{,}6745\sqrt{D(\xi)/N}$. Название «вероятная ошибка» вызвано тем, что $P\{|\xi - a| > r\} = 1/2$: отклонение ξ от $\boldsymbol{a}$ большее, чем на $\boldsymbol{r}$, и меньшее, чем на $\boldsymbol{r}$, одинаково вероятны. Поэтому величину $\boldsymbol{r}$ называют вероятной ошибкой.

4.4. Характеристики алгоритмов М-К

Предположим, что для расчета некоторой величины **m** мы придумали случайную величину ξ с заданными характеристиками: матожиданием, дисперсией и плотностью распределения. Однако сам метод М-К не определяет алгоритма расчета, ибо случайную величину ξ можно моделировать различными способами.

Выберем конкретный метод моделирования. Конструктивная размерность равна количеству случайных чисел, затрачиваемых на реализацию одного значения ξ. Даже лучше сказать «максимальному количеству», ибо это количество не обязано быть постоянным во всех опытах. Т.о. с точки зрения уменьшения конструктивной размерности выгоднее всего метод обратных функций, для которого она равна 1. Напротив, если используется метод Неймана (отбора-отказа), то конструктивная размерность = ∞.

Обозначим через $\boldsymbol{t}$ время расчета одного ξ. В качестве единицы измерения $\boldsymbol{t}$ можно выбрать как микросекунды, так и количество затрачиваемых элементарных операций. Полное время расчета методом М-К равно $\boldsymbol{T=Nt}$.

Преобразуем приближенное выражение для вероятной ошибки $r_N = 0{,}6745\sqrt{D(\xi)/N}$,

если ввести сюда вместо N величину $\boldsymbol{T/t}$, то получим выражение $r_N = 0{,}6745\sqrt{tD(\xi)/T}$.

Последняя формула показывает, что если полное время расчета $\boldsymbol{T}$ фиксировано, то вероятная ошибка зависит от произведения $tD(\xi)$.

Это произведение называют трудоемкостью алгоритма М-К. Чем меньше трудоемкость, тем выгоднее алгоритм, так как полученному за то же время $\boldsymbol{T}$ результату соответствует меньшая вероятная ошибка.

Обычно и значение $\mathbf{D(\xi)}$, и значение $\boldsymbol{t}$ оценивают эмпирически по сравнительно небольшому количеству испытаний.

4.5. Примеры использования метода в приближенных вычислениях. Вычисление определенного интеграла и площадей фигур методом Монте-Карло

I способ (выборочного среднего)

Требуется оценить интеграл $I = \int_a^b \varphi(x)dx$. Рассмотрим случайную величину **ξ**, равномерно распределенную на отрезке **[a,b]**. Тогда $I = (b\text{-}a) \cdot M\varphi(\xi)$, где $M\varphi(\xi)$ – математическое ожидание функции **φ(ξ).**

Тогда для вычисления интеграла можно предложить следующую схему: моделируется случайная величина **ξ** на отрезке **[a, b]**, вычисляются значения **$\varphi(x_i)$, i=1, 2, … n,** тогда

$$M\varphi(\xi) = \frac{1}{n}\sum_{i=1}^{N}\varphi(x_i) \;\; \textit{и интеграл будет} \;\; I \approx \frac{b-a}{n}\sum_{i=1}^{N}\varphi(x_i).$$

II способ (площадей)

Предположим, что функция **φ(x)** ограничена 0 ≤ φ(x) ≤c. Воспользуемся тем известным фактом, что величина определенного интеграла – площадь под кривой y=φ(x)**,** ограниченная прямыми x=a и x=b. Рассмотрим двумерную случайную величину R=(x,y), равномерно распределенную в прямоугольнике a≤ x ≤b; o≤ y ≤c. Предположим, всего имеется n точек, являющихся реализациями R(x,y) из них n_1 точек попало под кривую. Вероятность того, что точка оказалась под кривой y=φ(x):

$$\frac{\int_a^b \varphi(x)dx}{c(b-a)} = \frac{n_1}{n}, \;\; I \approx (b-a)c\frac{n_1}{n}$$

Этот же метод можно распространить для вычисления площадей (объемов) фигур, заданных уравнениями линий (поверхностей), их ограничивающих.

Уменьшить влияние статистической ошибки при моделировании методом Монте-Карло можно увеличением числа генерируемых точек (продолжительность прогона модели), а также усреднением результатов, полученных при прогонах, в каждом из которых использовались различные последовательности случайных чисел.

4.6. Контрольные вопросы по теме

1. Что представляет собой метод имитационного моделирования?
2. Каковы достоинства и недостатки имитационного моделирования?
3. В каких случаях применение имитационного моделирования является оправданным?
4. Какими причинами обусловлена необходимость введения модельного времени?
5. Что представляет собой изменение модельного времени с постоянным шагом? По особым состояниям?
6. Каковы особенности метода Монте-Карло?

7. Какова общая схема метода Монте-Карло?
8. Как оценить погрешность метода Монте-Карло?
9. Каковы характеристики метода Монте-Карло?

Лабораторная работа № 7. Тема: вычисление определенного интеграла методом Монте-Карло

Задание. Используя полученные в лабораторной работе № 5 псевдослучайные величины, вычислить интеграл методом выборочного среднего. Сравнить полученное значение с точным, вычисленным с помощью какого-либо из математических пакетов.

Варианты заданий для самостоятельного выполнения

1. $\int\limits_{1}^{10} \frac{x \ln x}{(1+x^2)^3} dx$

2. $\int\limits_{0}^{5} \frac{\sin(x)}{(\cos(x^2)+x)^2} dx$

3. $\int\limits_{2}^{3} (\sqrt{x} + \frac{\sin(x^2)}{\sqrt{\lg(x^2)}}) dx$

4. $\int\limits_{6}^{9} (\sin(x) + \frac{\cos(x)}{\sqrt{x}} dx$

5. $\int\limits_{-\pi/2}^{\pi/4} e^{-x} \sin(x^2) dx$

6. $\int\limits_{0}^{5} \frac{x^2 e^{-x}}{\cos(x)+x} dx$

7. $\int\limits_{0}^{2} \frac{5\sin(x)dx}{\sqrt{(1+x^2)(4+3x^2)}}$

8. $\int\limits_{0}^{3} \frac{\sin(x)dx}{\sqrt{e^x + \cos(x)^2}}$

9. $\int\limits_{0}^{1.5} \frac{tgx}{1+x^3} dx$

10. $\int_0^1 \frac{(xe)^x \sin(3x)}{\sqrt{x+1}} dx$

11. $\int_{2.5}^3 \frac{x^2 \sin(3x)}{\sqrt{e^x + x}} dx$

12. $\int_0^{\pi/3} (\sqrt{\sin x^2 \cos x^2}) dx$

13. $\int_0^{\pi/2} \sin^2 x \cos^2 x dx$

14. $\int_2^3 (\sqrt{x} + \frac{1}{\sqrt{x}}) dx$

15. $\int_1^5 \frac{x \ln x}{(1+x^2)^3} dx$

16. $\int_{\pi/2}^{2\pi} \frac{dx}{(1-\sqrt{x})^3}$

Тема 5. Планирование эксперимента с моделями систем

5.1. Основные понятия теории планирования эксперимента.

Эксперимент – целенаправленное воздействие на объект исследования с целью получения достоверной информации о его свойствах. Эксперимент, в котором исследователь может изменять условия его проведения, называется *активным*, если только регистрировать – *пассивным*.

План эксперимента – совокупность данных, определяющих число, условия и порядок проведения опытов. Планирование эксперимента преследует две основные цели:

1) сокращение общего объема испытаний при соблюдении требований к достоверности и точности результатов;

2)повышение информативности каждого из экспериментов в отдельности.

При планировании эксперимента рассматриваются два типа переменных, которые называются факторами и откликами. Значение фактора будем называть его *уровнем*. Каждый фактор изменяется в *интервале варьирования*, определяемом верхним и нижним уровнем фактора. Множество возможных уровней факторов называют *факторным пространством*.

Точкой *факторного пространства* называют совокупность конкретных уровней всех факторов. Ее можно отобразить с помощью матрицы-столбца.

Составной частью эксперимента является опыт – воспроизведение исследуемого явления в определенных условиях при возможности измерения, регистрации и количественной оценки состояния исследуемой системы. Точка факторного пространства или фиксированный набор уровней определяет условия проведения опыта.

Требования к факторам

- Факторы должны быть управляемыми. Те факторы, которые мы можем только контролировать, в плане эксперимента учитывать нельзя.
- Факторы должны быть определены количественно. Если используются качественные факторы, то каждому их уровню должно быть присвоено число. Учитывая, что факторы могут иметь разные размерности и резко отличаться количественно, используют кодирование факторов. Эта операция заключается в выборе такого нового масштаба, чтобы минимальное значение кодированного фактора соответствовало -1, максимальное +1, а начало координат было перенесено в точку с координатой, соответствующей среднему значению:

$$X_{cp} = \frac{X_{\max} + X_{\min}}{2}; \;\; x' = \frac{X - X_{cp}}{X_{cp} - X_{\min}}$$

- Важно выбирать в качестве факторов только независимые переменные. Исключить можно тот параметр, который труднее измерить или физический смысл которого менее ясен.
- Число уровней следует выбирать минимально возможным и в то же время достаточным для достижения целей эксперимента.
- Надо учитывать совместимость факторов, т.е. любые сочетания факторов в факторном пространстве должны быть реализуемы и не приводить к абсурду.

Требования к откликам

1) необходимо, чтобы отклик был однозначным в статистическом смысле, т.е. заданному набору уровней факторов должно соответствовать одно значение отклика (или вектору отклика);

2) отклик должен быть эффективным. Это значит, что из нескольких функционально связанных откликов выбранный можно измерить с наибольшей точностью.

Изменение уровня отклика под действием монотонно изменяющихся во времени неконтролируемых факторов будем называть ***дрейфом отклика***.

После того, как определены наиболее значимые факторы, приступают к построению плана эксперимента. Существует два основных варианта постановки задачи планирования эксперимента: стратегическое и тактическое планирование эксперимента.

5.2. Стратегическое планирование имитационного эксперимента

Цель стратегического планирования имитационных экспериментов – найти сочетания уровней факторов, при котором может быть получена наиболее полная и достоверная информация о поведении системы. План эксперимента обычно записывается в виде матрицы плана – прямоугольной таблицы, строки которой отвечают отдельным опытам, а столбцы – факторам. Элементами матрицы плана являются уровни факторов. Матрица плана может иметь совпадающие строки (т.е. опыт повторяется).

При стратегическом планировании эксперимента должны быть решены две основные задачи: идентификация факторов – их ранжирование по степени влияния на значение наблюдаемой переменной; выбор уровней факторов, который производится с учетом двух противоречивых требований:

1) уровни фактора должны перекрывать (заполнять) весь возможный диапазон его изменения;
2) общее количество уровней по всем факторам не должно приводить к чрезмерному объему моделирования. Число уровней равно минимальному числу точек, необходимых для восстановления полиномиальной функции.

Способы построения стратегического плана

В случае, когда количество факторов, а также количество уровней невелико, могут быть использованы полные факторные планы. Эксперимент, в котором реализуются все возможные сочетания уровней факторов, называется полным факторным экспериментом (ПФЭ). Общее число различных комбинаций уровней в ПФЭ для k факторов можно вычислить так: $N=p_1 \times p_2 \times p_3 \times \ldots \times p_k$. Здесь p_i – число уровней i-го фактора.

Если число уровней для всех факторов одинаково, то $N=p^k$, p – число уровней. Недостаток ПФЭ – большие временные затраты на подготовку и проведение. Поэтому использование ПФЭ целесообразно только в том случае, если в ходе эксперимента исследуется взаимное влияние всех факторов, влияющих на отклик.

Если такие взаимодействия считают отсутствующими или их эффектом пренебрегают, проводят частичный факторный эксперимент (ЧФЭ).

Рандомизированный план – предполагает выбор сочетания уровней для каждого опыта случайным образом. Чаще всего рандомизированный план используется на стадии «отсеивающего эксперимента» (экранирующая стадия). Целью этих экспериментов является выделение из огромного числа первоначально задуманных факторов лишь важных.

Латинский план (или «латинский квадрат») – используется в том случае, когда проводится эксперимент с одним первичным фактором и несколькими вторичными. Суть такого планирования состоит в следующем. Если первичный фактор A имеет p уровней, то для каждого вторичного фактора также выбирается p уровней. Выбор комбинации уровней факторов выполняется на основе специальной процедуры сдвига.

Факторы B, C	C1	C2	C3	C4
B1	A1	A2	A3	A4
B2	A2	A3	A4	A1
B3	A3	A4	A1	A2
B4	A4	A1	A2	A3

Пусть в эксперименте используется первичный фактор A и два вторичных фактора – B и C, число уровней факторов $p=4$. Соответствующий план можно представить в виде квадратной таблицы размером $p \times p$, относительно уровней фактора A. При этом матрица плана строится таким образом, чтобы в каждой строке и в каждом столбце данный уровень фактора A встречался только один раз:

Эксперимент с изменением факторов по одному. Суть его состоит в том, что один из факторов «пробегает» все p уровней, а остальные k-1 факторов поддерживаются постоянными. Такой план обеспечивает исследование эффектов каждого фактора в отдельности. Он требует проведения всего $N=p_1+p_2+p_3+\ldots+p_n$ опытов (p_i – число уровней i-го фактора).

Построение модели отклика (регрессионная модель)

Пусть интересующее экспериментатора свойство Y объекта зависит от нескольких независимых переменных $X_1, X_2, \ldots, X_n$ и требуется выяснить характер этой зависимости $Y=F(X_1, X_2, \ldots, X_n)$. Показатели $X_1, X_2, \ldots, X_n$ параметров изучаемого объекта – факторы. Величина Y – отклик, а сама зависимость $Y=F(X_1,X_2, \ldots, X_n)$ – функция отклика.

Основные требования к модели отклика – способность предсказывать направление дальнейших опытов и простота. Поэтому наиболее часто используются алгебраические полиномы. Расчет коэффициентов регрессии рассмотрим на примере.

Рассматривается полный факторный эксперимент, где число факторов k=3, число уровней p=2, соответственно, число опытов N=2^3=8, число повторных наблюдений в каждом опыте n=3. Требуется по данным эксперимента построить многофакторную модель отклика.

Будем искать уравнение функции отклика в виде следующего выражения:

$$y = b_0 + b_1x_1 + b_2x_2 + b_3x_3 + b_{12}x_1x_2 + b_{13}x_1x_3 + b_{23}x_2x_3 + b_{123}x_1x_2x_3,$$

где b_0 – значение свободного члена; b_1, b_2, b_3 – линейные коэффициенты; b_{12}, b_{13}, b_{23} – коэффициенты двойного взаимодействия факторов, b_{123} – коэффициент тройного взаимодействия факторов, x_1, x_2, x_3 – кодированные значения факторов.

Сначала строится основная матрица планирования, если предполагается учитывать двойное и тройное взаимодействие факторов, то строится расширенная матрица и вносятся значения отклика в каждом наблюдении (верхний уровень фактора – «1», нижний «-1»).

N	**X_1**	**X_2**	**X_3**	**X_1 X_2**	**X_1 X_3**	**X_2 X_3**	**X_1 X_2 X_3**	**Y**	**$Y_{ср}$**
1	1	1	1	1	1	1	1		9,7
2	-1	1	1	-1	-1	1	-1		10,4
3	1	-1	1	-1	1	-1	-1		11,4
4	-1	-1	1	1	-1	-1	1		4,2
5	1	1	-1	-1	-1	-1	1		4,4
6	-1	1	-1	1	1	-1	-1		4,5
7	1	-1	-1	1	1	1	-1		3,7
8	-1	-1	-1	-1	-1	1	1		3,4

Расчет коэффициентов модели отклика производят следующим образом: сначала вычисляют среднее значение отклика для опыта (в данном случае усредняют данные по трем наблюдениям для каждого опыта).

Значение свободного члена берут как среднее арифметическое всех значений отклика в матрице: i – номер опыта; $\bar{y}_i$ – среднее значение отклика в i-м опыте; N – число опытов в матрице планирования.

$$b_o = \frac{\sum_{i=1}^{N} \bar{y}_i}{N}$$

Линейные коэффициенты рассчитываются:
где x_{ki} – кодированное значение фактора x_k в i-м опыте.

$$b_k = \frac{\sum_{i=1}^{N} x_{ki}\bar{y}_i}{N}$$

Для расчета коэффициентов, характеризующих парные взаимодействия факторов, используется расширенная матрица планирования. Коэффициенты парных взаимодействий рассчитываются:

$$b_{km} = \frac{\sum_{i=1}^{N} x_{ki}x_{mi}\bar{y}_i}{N}$$

$$b_{klm} = \frac{\sum_{i=1}^{N} x_{ki}x_{li}x_{mi}\bar{y}_i}{N}$$

В плане может учитываться тройное взаимодействие факторов. Коэффициент тройного взаимодействия факторов вычисляется по формуле:

Далее рассчитывают доверительный интервал для коэффициентов регрессии: $\Delta b_i = \pm \frac{t \cdot S_B}{\sqrt{N \cdot n}}$, где $t_{(\text{альфа, Nn})}$ – критическая точка распределения Стьюдента для числа опытов N*n. Коэффициент считается статистически значимым, если его абсолютная величина > Δb_i.

Статистическая незначимость коэффициента b_i может быть обусловлена следующими причинами: данная переменная (или произведение переменных) не имеет функциональной связи с выходным параметром Y; велика ошибка эксперимента вследствие наличия неуправляемых и неконтролируемых переменных.

Далее коэффициенты подставляются в уравнение регрессии и рассчитываются теоретические значения отклика для каждой точки плана. Это делается для того, чтобы оценить качество (адекватность) модели.

Для того чтобы оценить качество, нужно вычислить дисперсию воспроизводимости S_B^2 и остаточную дисперсию S_O^2. Модель считается адекватной, если остаточная дисперсия меньше дисперсии воспроизводимости $S_O^2 < S_B^2$.

<u>Дисперсия воспроизводимости</u>:

Если в каждой точке плана проводился один опыт, то для определения дисперсии воспроизводимости в центре плана проводится несколько параллельных опытов, и дисперсия рассчитывается по формуле:

$$S_B^2 = \frac{\sum_{j=1}^{m} (y_{0j} - \bar{y}_0)^2}{m-1}$$

Если в каждой точке проводится несколько наблюдений, то:

$$S_B^2 = \frac{\sum_{i=1}^{N} \sum_{j=1}^{n} (y_{ij} - \bar{y}_i)^2}{N(n-1)} = \frac{\sum_{i=1}^{N} S_i^2}{N}$$

где i – номер опыта; j – номер наблюдения в опыте; y_{ij} – значение отклика в j-м наблюдении i-го опыта; $\bar{y}_i$ – среднее значение отклика в i-м опыте; N – число опытов в матрице планирования; n – число наблюдений в каждом опыте; S_i^2 – дисперсия одного опыта.

<u>Остаточная дисперсия</u>:

1 повтор:

$$S_O^2 == \frac{\sum_{i=1}^{N} (\bar{y}_i - \hat{y}_i)^2}{N-1}$$

несколько:

$$S_O{}^2 = \frac{\sum_{i=1}^{N} \sum_{j=1}^{n} (y_{ij} - \hat{y}_i)^2}{N(n-1)}$$

где k – количество факторов; y_i – теоретическое значение отклика в i-м опыте, вычисленное с помощью полученного уравнения модели отклика. Для получения теоретического значения нужно подставить в уравнение известные для каждого опыта значения фактора x и выполнить вычисления.

Статистическая проверка гипотезы о равенстве дисперсий осуществляется по критерию Фишера: вычисляется отношение $F=(S_O^2/ S_B^2)$ и сравнивается с критическим в зависимости от доверительной вероятности и числа степеней свободы $\mathbf{m_1=N-(k+1)}$ и $\mathbf{m_2=n}$ (условием адекватности модели является выполнение условия $F_{расч}<F_{табл}$).

5.3. Тактическое планирование имитационного эксперимента

Тактическое планирование решает следующие вопросы, связанные с проведением ИЭ:

- определение длительности переходного режима (анализ установившегося состояния)
- задание начальных условий (инициализация);
- установления необходимого объема испытаний (числа прогонов с различными последовательностями чисел);
- определение продолжительности имитационного прогона;
- уменьшение дисперсии отклика.

Во многих имитационных моделях требуется сделать несколько шагов по времени, прежде чем система входит в нормальный режим.

Установившимся или стационарным называется такое состояние модели, когда последовательные наблюдения отклика в ней статистически неразличимы (система моделируется в типичных для нее условиях).

Итак, первоначально результаты моделирования имеют неустойчивый характер (*переходное состояние*), а устойчивость (*стационарность*) обычно достигается при достаточно продолжительном прогоне модели. Однако следует иметь в виду, что существует предел, за которым увеличение продолжительности прогона модели уже не дает существенного повышения точности результата, измеряемой дисперсией.

Правила останова

Имитационный прогон обычно заканчивается в случае, когда происходит некоторое критическое событие. Существует 2 типа критических событий: по состоянию, т.е. прогон заканчивается, когда в модели фиксируется критическое состояние (поломка оборудования, превышение длины очереди); и по времени, когда имитационное время достигает заранее заданной величины (день, год).

Проблема: удлинять прогон или повторять его? Решение принимается исходя из специфики модели, в первую очередь из длины переходного периода.

Количество реализаций

Существует два подхода к определению количества реализаций.

1. С использованием дисперсии $N = \dfrac{t_\alpha{}^2 \sigma^2}{\varepsilon^2}$.

2. С использованием вероятности $\boldsymbol{p}$ наступления события при $\boldsymbol{N}$ реализациях.

$$N = t_\alpha{}^2 \frac{P(1-P)}{\varepsilon^2}$$

Тема 6. Обработка и анализ результатов моделирования систем с помощью пакета STATISTICA

6.1. Планирование эксперимента с помощью пакета STATISTICA

Планирование и обработку результатов эксперимента рассмотрим на примере. Пусть имеются результаты проведения полного факторного эксперимента (ПФЭ) типа 2^3, где число факторов k=3, число уровней p=2, число опытов N=8, число повторных наблюдений в каждом опыте n=5. Требуется по данным эксперимента построить многофакторную модель.

Матрица планирования:

№ опыта	**Уровни факторов**			**Результаты параллельных экспериментов**
	X_1	X_2	X_3	
1	+	+	–	7,8; 8,5; 7,7; 7,6; 8,0
2	–	+	–	1,8; 2,5; 2,0; 1,8; 1,6
3	+	–	–	4,4; 4,8; 5,3; 4,9; 5,2
4	–	–	–	4,3; 4,2; 5,0; 4,9; 4,6
5	+	+	+	9,7; 10,4; 11,4; 10,9; 10,9
6	–	+	+	4,2; 4,4; 4,5; 4,0; 3,8
7	+	–	+	3,7; 3,4; 4,0; 3,6; 4,1
8	–	–	+	4,1; 5,1; 4,8; 5,1; 4,5

Для перехода к процедуре формирования матрицы эксперимента необходимо в меню STATISTICA выбрать раздел ***Статистика*** >

Индустриальная статистика & Сигма шесть > Экспериментальный проект.

В стартовом меню раздела **Design&Analysis of Experiments** (экспериментальный проект) выбирается процедура ***2**(K-p) standard designs***.

В появившемся окне во вкладке ***Design Experiment*** необходимо определить количество ***входных факторов / блоков / опытов*** соответственно условиям эксперимента (рис.2.5). Если эксперимент выполняется в несколько этапов, применяется разделение опытов на серии (блоки) случайным образом.

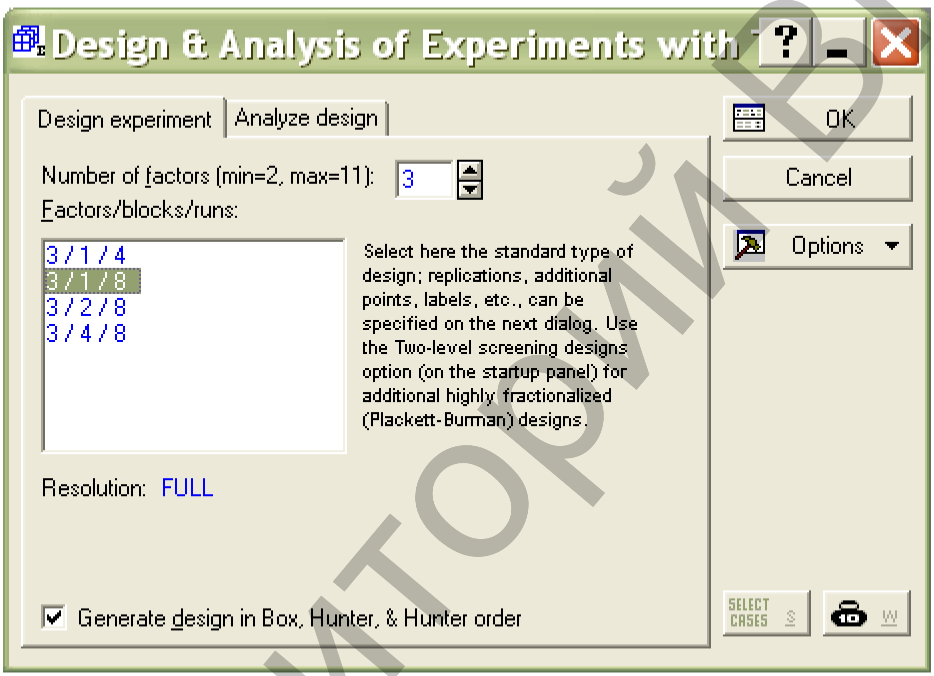


Рис.2.5. Определение условий эксперимента

После нажатия **ОК** будет получено окно ***Design of an Experiment with Two-Level Factors***, в котором необходимо выполнить следующие установки:

- во вкладке ***Display design*** в рамке ***Denote factors*** определяется, как в матрице планирования будут обозначены факторы: числами, буквами или именами; в рамке ***Order of runs*** – порядок размещения опытов в матрице: по порядку, случайным образом или по блокам; в рамке ***Show (in Spreadsheet)*** – как будут обозначены уровни факторов в матрице: числами, мин/макс значением или символами (рис.2.6);

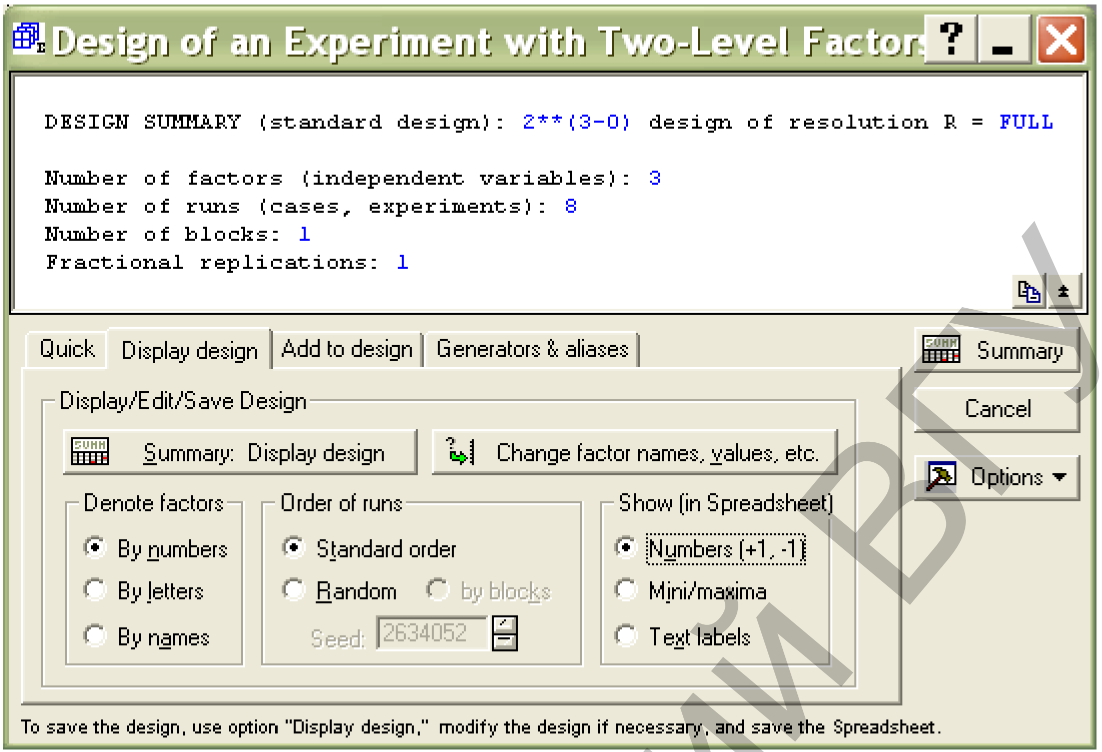


Рис.2.6. Установки вкладки ***Display design***

- во вкладке ***Add to design*** нужно указать количество дополнительных прогонов для каждой точки плана (4) и количество дополнительных пустых столбцов (1), в которые после проведения эксперимента будут внесены значения отклика (рис.2.7).

Для получения матрицы эксперимента необходимо нажать кнопку ***Summary***. Полученная матрица сохраняется в файле в формате *.sta. Для того, чтобы сохранить только матрицу планирования, а не всю рабочую книгу, необходимо в меню ***Инструменты > Опции*** во вкладке ***Менеджер вывода*** установить флажок в позиции ***Поместить все результаты в индивидуальные окна***.

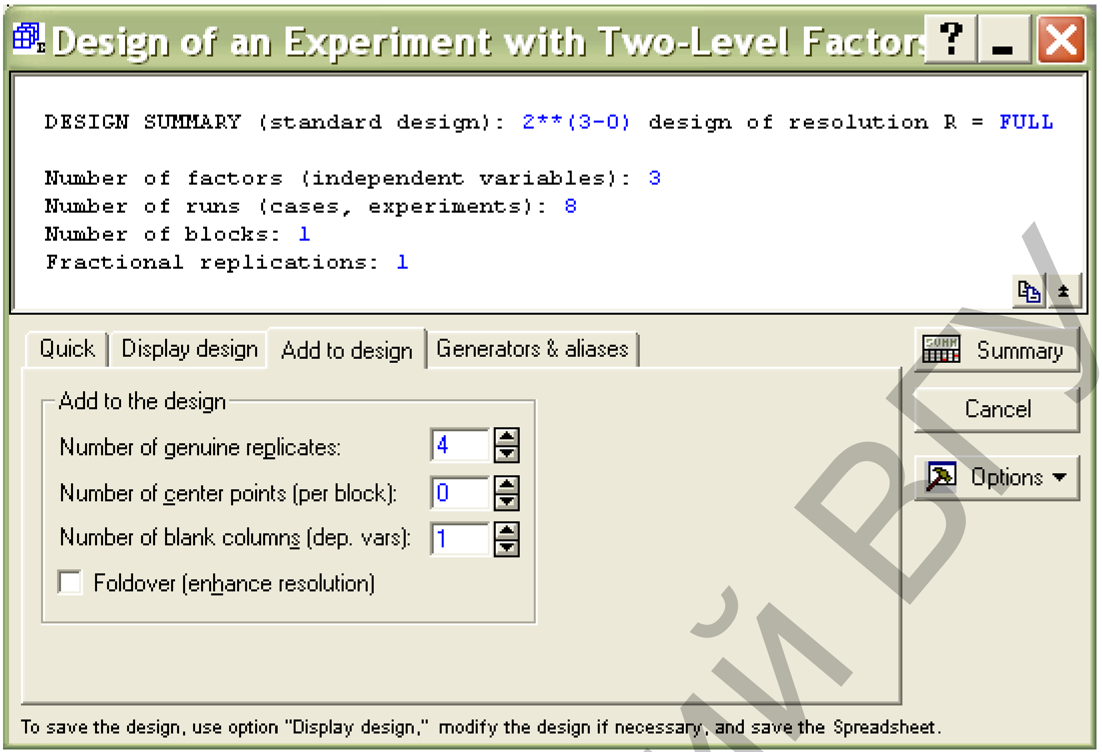


Рис.2.7. Установки вкладки ***Add to design***

После разработки матрицы эксперимента в свободный столбец вводятся значения отклика. Далее можно воспользоваться специальными процедурами обработки данных эксперимента пакета STATISTICA. Для вызова этих процедур нужно перейти во вкладку ***Analyze design***, с помощью кнопки ***Variables*** установить зависимые (отклик) и независимые (факторы) переменные. Далее необходимо нажать **ОК** для получения окна ***Analysis of an Experiment with Two-Level Factors***. Во вкладке ***Model*** полученного окна указывается, надо ли включать взаимодействие двух и более факторов (***3-way interactions***).

6.2. Обработка результатов эксперимента с помощью пакета STATISTICA

Итогами обработки данных эксперимента являются ряд параметров, наиболее интересными из которых являются значения коэффициентов линейной многофакторной модели и результаты дисперсионного анализа.

Для получения значений коэффициентов модели, их ошибок и уровней значимости используется кнопка ***Regression coefficients*** во вкладке ***ANOVA/Effects*** . Результаты обработки данных примера представлены на рис.2.8.

Data: Regr. Coefficients; Var.:DV_1; R-sqr=,98084; Adj:,97...

Regr. Coefficients; Var.:DV_1; R-sqr=,98084; Adj:,97664 (Design1)
2**(3-0) design; MS Residual=,157125
DV: DV_1

Factor	Regressn Coeff.	Std.Err.	t(32)	p	-95,% Cnf.Limt	+95,% Cnf.Limt
Mean/Interc.	5,337500	0,062675	85,16188	0,000000	5,209836	5,465164
(1)F1	1,477500	0,062675	23,57408	0,000000	1,349836	1,605164
(2)F2	0,837500	0,062675	13,36264	0,000000	0,709836	0,965164
(3)F3	0,492500	0,062675	7,85803	0,000000	0,364836	0,620164
1 by 2	1,637500	0,062675	26,12695	0,000000	1,509836	1,765164
1 by 3	-0,097500	0,062675	-1,55565	0,129627	-0,225164	0,030164
2 by 3	0,752500	0,062675	12,00643	0,000000	0,624836	0,880164
1*2*3	0,222500	0,062675	3,55007	0,001216	0,094836	0,350164

Рис.2.8. Значения коэффициентов модели

Коэффициенты с уровнем значимости p>0,5 включать в модель нецелесообразно, поэтому искомая модель будет иметь вид:

$$Y=5{,}3375+1{,}4775x_1+0{,}8375x_2+0{,}4925x_3+1{,}6375x_1x_2+0{,}7525x_2x_3+0{,}2225x_1x_2x_3$$

Коэффициент детерминации R^2 показывает, в какой степени вариация отклика определяется вариацией факторов, включенных в модель. Близкое к единице значение коэффициента детерминации R^2=0,98084 говорит о хорошем качестве полученной модели.

Для получения результатов дисперсионного анализа используется кнопка ***ANOVA tables*** во вкладке ***ANOVA/Effects*** (рис.2.9):

Factor	SS	df	MS	F	p
(1)F1	87,3203	1	87,3203	555,7375	0,000000
(2)F2	28,0562	1	28,0562	178,5601	0,000000
(3)F3	9,7022	1	9,7022	61,7486	0,000000
1 by 2	107,2563	1	107,2563	682,6173	0,000000
1 by 3	0,3802	1	0,3802	2,4200	0,129627
2 by 3	22,6503	1	22,6503	144,1543	0,000000
1*2*3	1,9802	1	1,9802	12,6030	0,001216
Error	5,0280	32	0,1571		
Total SS	262,3737	39			

Рис.2.9. Результаты дисперсионного анализа

Полученные данные позволяют осуществить оценку степени влияния контролируемых факторов x_1, x_2, x_3 и их взаимодействий на отклик. Дисперсия факторов задана в столбце ***SS***, остаточная дисперсия, описывающая влияние неконтролируемых факторов, задана в строке ***Error***, общая дисперсия – в строке ***Total SS***. Выразив в процентах долю каждой дисперсии в общей дисперсии, можно получить значения, оценивающие степень влияния факторов на отклик модели:

Фактор	x_1	x_2	x_3	x_1x_2	x_1x_3	x_2x_3	$x_1x_2x_3$	неконтр.ф.
Доля дисперсии (%)	33,3	10,7	3,7	40,9	0,1	8,6	0,8	1,9

Проведем анализ остатков. Остатками называют разности между фактическими значениями зависимой переменной и предсказанными (вычисленными по модели). Если модель адекватна, остатки представляют собой случайную величину с нормальным распределением и нулевым математическим ожиданием. Для визуальной оценки остатков используется гистограмма остатков и график остатков на нормальной вероятностной бумаге, которые можно получить, нажав кнопки ***Histogram of residuals*** и ***Normal plot*** на вкладке ***Residual plots*** (рис.2.10).

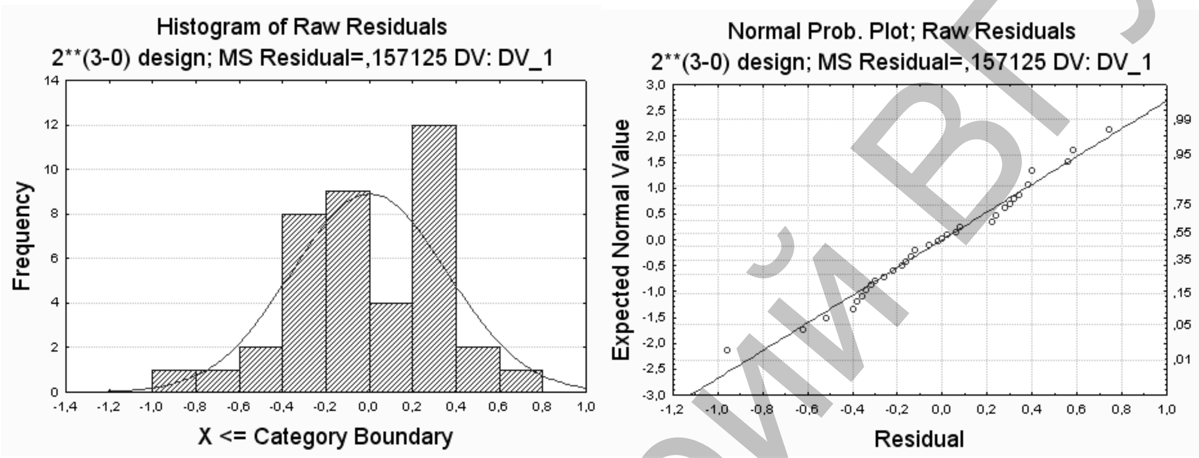


Рис.2.10. Гистограмма и график остатков на вероятностной бумаге

Вкладка ***Prediction&profiling*** позволяет визуализировать результаты эксперимента. Нажав ***Surface plot*** можно получить изображение поверхности отклика для исследуемой зависимости. Если количество входных факторов более двух, то STATISTICA предложит определить только те два фактора, которые необходимо использовать при построении графика. Для получения поверхности отклика в виде изолиний, т.е. линий одинакового уровня выходного параметра, необходимо нажать ***Contour plot*** (рис.2.11).

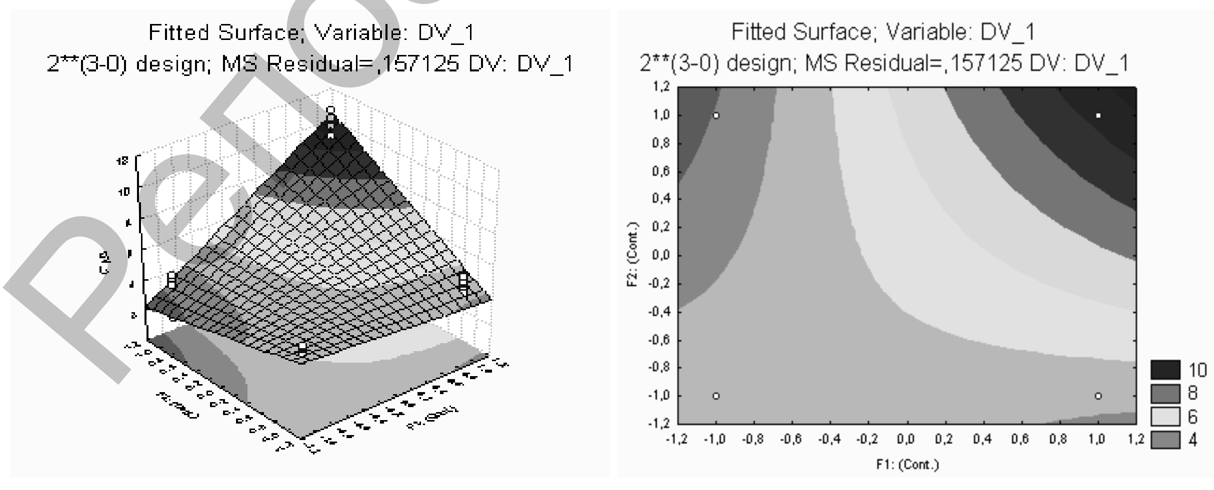


Рис.2.11. Графическое отображение результатов эксперимента

Если данная модель неадекватна, то, чтобы получить модель второго порядка, к плану проведенных опытов, который называется «ядром» эксперимента, добавляется несколько специальным образом расположенных точек. Функция отклика формируется в виде полинома второй степени. Для корректной оценки всех коэффициентов полинома второй степени необходимо, чтобы в плане эксперимента каждый фактор имел не менее трех уровней варьирования. Для разработки таких планов используют процедуру центрального композиционного планирования ***Central composite designs***.

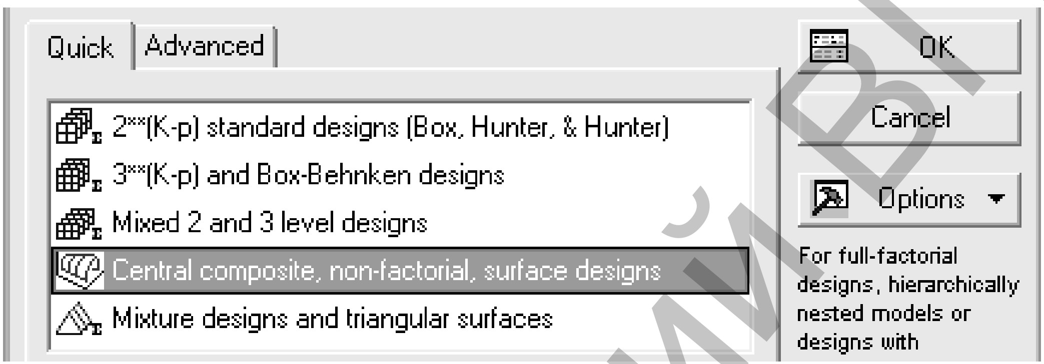


Рис.2.12. Установки вкладки ***Design&Analysis of Experiments***

Например, для построения ортогонального центрального композиционного плана второго порядка нужно во вкладке ***Add to design*** ввести в план дополнительный столбец для значений отклика, добавить одну центральную точку; во вкладке ***Design characteristic*** указать расчет α звездных точек для ортогональности плана (***Compute / use alpha for orthogonality***).

Анализ экспериментальных данных проводится аналогично рассмотренному в примере. Необходимо определить зависимую и независимые переменные, найти коэффициенты регрессионной модели второго порядка и уровни их значимости. Исключив статистически незначимые коэффициенты, получить уравнение регрессионной модели и изображение поверхности отклика исследуемой зависимости. Провести дисперсионный анализ, оценить степень влияния факторов на отклик и адекватность модели.

Автоматическое создание отчета

При статистической обработке данных в рабочем окне системы STATISTICA одновременно может находиться множество открытых документов: файл с исходными данными, таблицы с результатами анализа, графики. Подобная ситуация типична для большинства видов статистического анализа. Если же расчеты содержат много промежуточных результатов, которые генерируют набор таблиц и графиков, то это приводит к тому, что все рабочее пространство будет заполнено, что крайне неудобно.

В связи с этим в системе STATISTICA реализована возможность автоматического создания файла регистрации результатов анализа. Такой файл называется автоотчетом. Для создания автоотчета необходимо вызвать пункт меню ***File > Output Manager***, в окне ***Options*** во вкладке ***Менеджер вывода*** установить метку в окне ***Также послать окно отчета***, а во вкладке ***Отчеты*** задать параметры автоматического создания отчета. Все таблицы и графики, появляющиеся на экране, будут автоматически выводиться в файл с автоотчетом, который открывается в отдельном окне на рабочем пространстве системы STATISTICA.

6.3. Контрольные вопросы по теме

1. Каковы цели стратегического и тактического планирования компьютерных экспериментов?
2. Какие виды факторов бывают в имитационном эксперименте?
3. Что называется полным факторным экспериментом? Объясните процедуру формирования плана эксперимента.
4. Каким образом формируется факторный план 2k? Как учитываются эффекты взаимодействия факторов в функции отклика?
5. Что такое «ядро» эксперимента? Какие точки нужно добавить к плану эксперимента, чтобы можно было построить модель второго порядка?
6. Что такое поверхность отклика? Как построить регрессионную модель?
7. Как оценить адекватность полученной модели?
8. Какие методы математической статистики используются для анализа результатов имитационного моделирования систем?
9. Какие средства пакета STATISTICA используются для обработки данных эксперимента?
10. Какими возможностями располагает система STATISTICA? Какие статистические модули имеются в пакете?
11. Назовите выборочные числовые характеристики. Как получить описательную статистику с помощью пакета STATISTICA?
12. Какие типы графиков имеются в пакете STATISTICA?
13. Как построить матрицу планирования полного факторного эксперимента в пакете STATISTICA? Что представляют собой зависимые и независимые переменные?
14. Назовите порядок операций, которые необходимо выполнить для построения уравнения регрессии.
15. С помощью каких статистических критериев оценивается значимость коэффициентов уравнения регрессии?
16. Что представляет собой коэффициент детерминации? Для чего он используется?
17. Как можно оценить степень влияния факторов на отклик модели с помощью дисперсионного анализа?
18. Что называют остатками? Как оценить адекватность модели с помощью

анализа остатков?

19. Что представляет собой процедура центрального композиционного планирования?
20. Как создать автоотчет? Какую информацию содержит отчет?

Лабораторная работа № 8. Тема: планирование эксперимента

Задание. Рассмотреть ПФЭ типа 2^3, где число факторов k=3, число уровней p=2, число опытов N=8, число повторных наблюдений в каждом опыте n=3. Построить математическую модель в виде неполного квадратичного полинома и оценить ее параметры по результатам проведения ПФЭ 2^3.

В таблице 1 представлена матрица планирования ПФЭ. В таблице 2-N (N – номер варианта) занесены отклики объекта в трех сериях опытов: каждая колонка соответствует одной серии опытов, каждая строка – номеру эксперимента (точке плана), таблицы 1.

Таблицы значений критерия Стьюдента для расчета статистической значимости коэффициентов модели и критические значения критерия Фишера для оценки адекватности модели даны в приложении.

Матрица планирования Таблица 1

№ точки плана	X_1	X_2	X_3	X_1X_2	X_1X_3	X_2X_3	$X_1X_2X_3$	$Y_{1,j}$	$Y_{2,j}$	$Y_{3,j}$	Y_j сред
1											
…											
8											

1. Заполнить таблицу, вычислить коэффициенты регрессии для многофакторной модели.
2. Получив границы доверительного интервала, исключить статистически незначимые коэффициенты (если это необходимо).
3. Рассчитать теоретические значения отклика для моделей, включающих:
   1) линейные коэффициенты;
   2) линейные коэффициенты и коэффициенты двойного взаимодействия;
   3) линейные коэффициенты, коэффициенты двойного взаимодействия и коэффициент взаимодействия трех факторов.
4. Вычислить значения дисперсии воспроизводимости и остаточных дисперсий для каждой из моделей. По критерию Фишера проверить гипотезу об адекватности полученных моделей.

Пример

Матрица планирования и значения отклика в каждом наблюдении приведены в таблице (верхний уровень фактора кодируется как «1», нижний как «-1»).

N	X_1	X_2	X_3	Y_{j1}	Y_{j2}	Y_{j3}
1	1	1	-1	7,8	8,5	7,7
2	-1	1	-1	1,8	2,5	2
3	1	-1	-1	4,4	4,8	5,3
4	-1	-1	-1	4,3	4,2	5
5	1	1	1	9,7	10,4	11,4
6	-1	1	1	4,2	4,4	4,5
7	1	-1	1	3,7	3,4	4
8	-1	-1	1	3,8	5,1	4,8

Будем искать уравнение функции отклика в виде следующего выражения:

$$y = b_0 + b_1x_1 + b_2x_2 + b_3x_3 + b_{12}x_1x_2 + b_{13}x_1x_3 + b_{23}x_2x_3 + b_{123}x_1x_2x_3,$$

где b_0 – значение свободного члена; b_1, b_2, b_3 – линейные коэффициенты; b_{12}, b_{13}, b_{23} – коэффициенты двойного взаимодействия факторов, x_1, x_2, x_3 – кодированные значения факторов.

Внесем в таблицу расширенную матрицу планирования и исходные данные. Для вычисления среднего значения отклика по трем наблюдениям в опыте будем использовать функцию СРЗНАЧ(диапазон ячеек). Результирующая таблица будет иметь вид:

	A	B	C	D	E	F	G	H	I	J	K	L
1	Матрица планирования											
2	№ точ. плана	X_1	X_2	X_3	X_1X_2	X_1X_3	X_2X_3	$X_1X_2X_3$	$Y_{1,j}$	$Y_{2,j}$	$Y_{3,j}$	**Y_j сред**
3	1	1	1	-1	1	-1	-1	-1	7,8	8,5	7,7	**8,00**
4	2	-1	1	-1	-1	1	-1	1	1,8	2,5	2	**2,10**
5	3	1	-1	-1	-1	-1	1	1	4,4	4,8	5,3	**4,83**
6	4	-1	-1	-1	1	1	1	-1	4,3	4,2	5	**4,50**
7	5	1	1	1	1	1	1	1	9,7	10,4	11,4	**10,50**
8	6	-1	1	1	-1	-1	1	-1	4,2	4,4	4,5	**4,37**
9	7	1	-1	1	-1	1	-1	-1	3,7	3,4	4	**3,70**
10	8	-1	-1	1	1	-1	-1	1	3,8	5,1	4,8	**4,57**

Вычислим коэффициенты. Для расчета коэффициента b_0 достаточно просто вычислить среднее значение по столбцу J. Для вычисления остальных коэффициентов нужно найти сумму произведений факторов на средние значения отклика по всем опытам. Для выполнения этой операции можно использовать функцию СУММПРОИЗВ(массив1;массив2;...), которая перемножает соответствующие элементы заданных массивов и возвращает сумму произведений.

	A	B	C	D	E	F	G	H	I	J	K	L
1	Матрица планирования											
2	№ точ. плана	X_1	X_2	X_3	X_1X_2	X_1X_3	X_2X_3	$X_1X_2X_3$	$Y_{1,j}$	$Y_{2,j}$	$Y_{3,j}$	**$Y_{j\ сред}$**
					…	….	…					
12	Коэфф:	b1	b2	b3	b12	b13	b23	b123				b0
13		1,438	0,921	0,463	1,571	-0,121	0,729	0,179				5,321

Таким образом, уравнение модели отклика с линейными коэффициентами будет иметь следующий вид:

$$\textbf{(1)}\quad y = 5{,}321 + 1{,}438\,x_1 + 0{,}921\,x_2 + 0{,}463\,x_3$$

Модель отклика, учитывающая двойные взаимодействия факторов:

$$\textbf{(2)}\quad y = 5{,}321 + 1{,}438\,x_1 + 0{,}921\,x_2 + 0{,}463\,x_3 + 1{,}571\,x_1x_2 - 0{,}121\,x_1x_3 + 0{,}729\,x_2x_3$$

Модель отклика, учитывающая тройное взаимодействие факторов:

(3)

$$y = 5{,}321 + 1{,}438\,x_1 + 0{,}921\,x_2 + 0{,}463\,x_3 + 1{,}571\,x_1x_2 - 0{,}121\,x_1x_3 + 0{,}729\,x_2x_3 + 0{,}179\,x_1x_2x_3$$

Для того чтобы оценить качество (адекватность) каждой из моделей, нужно вычислить еще две величины: дисперсию воспроизводимости S_B^2 (ДВ) и дисперсию адекватности $S_{ад}^2$ (ДА). Будем считать модель адекватной, если дисперсия адекватности меньше дисперсии воспроизводимости $S_{ад}^2 < S_B^2$.

Продолжим заполнение таблицы. Введем в столбец М формулы для расчета дисперсии воспроизводимости. Для этого сначала с помощью функции ДИСП(диапазон) нужно рассчитать дисперсию для каждого опыта, затем найти среднее: вычислить сумму дисперсий и разделить на число опытов N=8. Результат: вычисленное значение дисперсии воспроизводимости помещается в ячейке М13.

K	L	M	N	O	P	Q	R	S
			Линейная модель		ЛМ + коэф.дв.взаим.		ЛМ + коэф.тр.взаим.	
$Y_{3,j}$	**$Y_{j\ сред}$**	ДВ	1 теор. отклик	1 ДА	2 теор. отклик	2 ДА	3 теор. отклик	3 ДА
7,7	**8,00**	0,19	7,22	0,61	8,18	0,03	8,00	0
2	**2,10**	0,13	4,34	5,03	1,92	0,03	2,10	7,89E-31
5,3	**4,83**	0,20	5,38	0,29	4,65	0,03	4,83	0
5	**4,50**	0,19	2,50	4,00	4,68	0,03	4,50	0
11,4	**10,50**	0,73	8,14	5,56	10,32	0,03	10,50	0
4,5	**4,37**	0,02	5,27	0,81	4,55	0,03	4,37	0
4	**3,70**	0,09	6,30	6,76	3,88	0,03	3,70	1,97E-31
4,8	**4,57**	0,46	3,43	1,30	4,39	0,03	4,57	7,89E-31
	b0	ДВ		1 ДА		2 ДА		3 ДА
	5,321	0,253		1,523		0,016		1,11E-31

Для получения теоретических значений отклика нужно подставить значения коэффициентов в первое модельное уравнение (расчеты по каждой точке в столбце N), затем второе (столбец P) и третье (столбец R) модельное уравнение.

В столбцах O, Q и S вычисляются значения дисперсии адекватности. Сначала для каждого опыта вычисляется квадрат отклонения теоретического значения, которое находится в столбце N, P, R от среднего экспериментального (столбец L). Затем находится сумма квадратов отклонений и делится на число 8(3-1)=16. Результатs находятся в ячейках O13, Q13 и S13.

Выводы: для модели (1) вычисленное значение дисперсии адекватности больше дисперсии воспроизводимости, следовательно, модель не является адекватной. Для моделей (2) и (3) значения дисперсии адекватности меньше дисперсии воспроизводимости, следовательно, эти модели адекватны.

Варианты заданий для самостоятельного выполнения

Таблица 1

2,124	2,95	2,139
3,382	3,394	3,368
2,105	2,652	2,155
4,307	4,242	4,276
3,807	3,089	3,096
5,981	5,148	5,723
3,948	3,901	3,914
6,173	6,92	6,832

Таблица 2

3,651	3,605	3,653
6,547	6,514	6,535
4,761	4,793	4,816
9,515	9,566	9,534
5,828	5,847	5,842
13,041	13,081	13,051
8,364	8,371	8,338
25,575	25,563	25,611

Таблица 3

8,171	8,93	9,117
13,972	13,962	13,565
11,065	10,751	11,381
17,681	17,153	17,981
12,8	13,089	12,536
21,774	21,967	21,505
16,097	16,733	16,378
29,58	30,092	30,179

Таблица 4

2,588	2,597	2,542
4,191	4,165	4,152
3,201	3,231	3,202
5,509	5,453	5,448
3,793	3,83	3,85
6,718	6,752	6,76
4,263	4,966	5,001
9,738	9,753	9,702

Таблица 5

3,072	3,028	3,08
5,193	5,159	5,163
3,932	3,155	3,893
7,094	7,126	7,249
4,74	4,704	4,668
9,163	9,167	9,16
6,336	6,396	6,369
14,676	14,668	14,725

Таблица 6

4,292	4,285	4,333
8,385	8,390	8,404
5,881	5,886	5,847
13,349	13,332	13,357
7,389	7,368	7,439
20,252	20,271	20,271
11,282	11,269	11,293
66,571	66,613	66,562

Таблица 7

4,307	4,284	4,284
8,387	8,996	8,43
5,832	5,873	5,856
13,329	13,904	13,328
7,379	7,415	7,415
19,255	20,278	21,304
11,226	11,238	11,271
63,599	65,605	69,588

Таблица 8

1,983	1,951	1,569
3,004	3,024	2,984
2,435	2,415	2,428
3,767	3,794	3,784
2,788	2,123	2,815
4,491	4,467	4,492
3,485	3,151	3,515
5,683	5,879	5,863

Таблица 9

3,254	3,932	3,124
5,147	5,17	5,378
3,926	3,895	3,937
7,117	7,121	7,901
4,101	4,682	4,69
9,715	9,159	9,115
6,39	6,383	6,384
11,677	14,67	12,718

Таблица 10

2,549	2,537	2,563
4,118	4,164	4,155
3,236	3,220	3,202
5,445	5,485	5,449
3,825	3,812	3,790
6,721	6,714	6,741
4,951	4,989	4,955
9,735	9,693	9,705

Таблица 11

2,164	2,165	2,145
3,347	3,338	3,322
2,639	2,658	2,651
4,281	4,251	4,296
3,086	3,084	3,081
5,082	5,128	5,117
3,95	3,932	3,908
5,855	6,87	6,875

Таблица 12

3,583	3,605	3,623
6,555	6,564	6,523
4,795	4,790	4,776
9,504	9,530	9,524
5,855	5,839	5,827
13,040	13,011	13,045
8,328	8,301	8,303
25,586	25,544	25,578

Таблица 13

2,544	2,157	2,235
3,374	3,372	3,256
2,474	2,913	2,545
4,317	4,255	4,658
3,919	3,137	3,654
5,092	5,673	5,012
3,951	3,919	3,255
6,858	6,869	6,542

Таблица 14

3,592	3,627	3,254
6,562	6,581	6,025
4,792	4,801	4,157
9,552	9,528	9,256
5,905	5,886	5,255
13,089	13,063	13,269
8,365	8,366	8,965
25,578	25,534	25,112

Таблица 15

8,609	8,231	8,758
13,602	13,956	14,022
11,198	11,021	10,934
18,578	18,125	17,979
12,906	12,025	12,514
22,287	22,984	21,517
16,708	16,214	16,327
29,648	29,235	29,924

Таблица 16

2,537	2,561	2,539
4,129	4,012	4,138
3,199	3,268	3,248
5,511	5,987	5,445
3,789	3,658	3,852
6,709	6,254	6,743
4,111	4,257	5,007
9,746	9,125	9,737

Тема 7. Инструментальные средства моделирования систем Система моделирования GPSS

7.1. Краткая характеристика системы GPSS

Система имитационного моделирования GPSS (General Purpose Simulation System) предназначена для описания и исследования дискретных моделей систем массового обслуживания (СМО). Описание на языке GPSS представляет собой совокупность операторов (блоков), характеризующих процессы обработки заявок. В языке имеется более 40 блоков с различным функциональным назначением. Для детализации функций, выполняемых блоком, у каждого из них имеется набор параметров.

Моделирование системы на GPSS представляет собой имитацию последовательности переходов системы из одного состояния в другое в особые моменты времени (событийное моделирование).

Объекты GPSS подразделяются на 7 категорий.

Динамическими объектами в СМО являются транзакты (сообщения, заявки), которые представляют собой единицы исследуемых потоков и производят ряд определенных действий. Отличие одних транзактов от других состоит в назначаемом им наборе параметров. Для имитации процессов, протекающих в моделируемой системе, все транзакты, порождаемые в процессе моделирования, образуют списки, в которых транзакты отсортированы по времени, при равных временах у транзактов – по приоритетам.

В начальный момент времени в GPSS-модели нет ни одного транзакта. В процессе моделирования транзакты входят в модель (генерируются) в соответствии с потребностями, которые возникают в моделируемой системе. В общем случае в модели существует большое число транзактов, но в один момент времени двигается только один. Если транзакт начал свое движение, он перемещается от блока к блоку по пути, заданному алгоритмом моделирования. Продвижение текущего транзакта продолжается по блокам модели до тех пор, пока не произойдет одно из следующих событий:

- транзакт входит в блок задержки и переходит в список будущих событий;
- транзакт входит в один из блоков проверки условий, и условие не позволяет транзакту перемещаться дальше (наступает условие блокировки), тогда транзакт переводится в список будущих событий;
- транзакт входит в блок удаления транзакта из модели.

Если возникло одно из описанных условий, то транзакт остается на месте, и начинается перемещение в модели другого транзакта, который выбирается из списка текущих событий. Если становится невозможным продвижение всех транзактов из списка текущих событий, то изменяется модельное время, то есть наступает время следующего события или группы событий, и алгоритм повторяется. Изменение последовательного перемещения транзакта по модели может быть нарушено оператором, определяющим для данного транзакта номер следующего блока.

Объекты аппаратной категории – это устройства, памяти и логические ключи. Воздействуя на эти объекты, транзакты могут изменять их состояние и влиять на движение других транзактов. *Устройства* моделируют реальные объекты, в которых может происходить обработка транзактов (сообщений). Устройство является аналогом обслуживающего прибора системы массового обслуживания. В любой момент времени устройство может быть занято только одним транзактом. *Памяти* описывают такие устройства, которые обслуживают одновременно несколько транзактов. Количество одновременно обслуживаемых транзактов определяют емкость памяти. *Ключи* используются для блокировки или изменения движения транзактов в зависимости от ранее наступивших событий. Наличие в модели объектов аппаратной категории позволяет автоматически регистрировать статистическую информацию.

Операционные объекты, то есть *блоки*, задают логику функционирования модели системы и определяют пути движения транзактов между объектами аппаратной категории. Описание системы на GPSS представляет собой последовательность блоков, каждый из которых соответствует некоторому оператору. Каждый блок имеет определенное количество реквизитов, называемых полями, которые отделяются друг от друга запятой (это аналоги параметров процедур и функций в языках программирования), но положение полей строго фиксировано и отсутствие некоторого поля отмечается запятой. Формат записи блоков имеет следующий вид:

[метка] ОПЕРАТОР A,B,C,D,...; Комментарии

Метка блока должна начинаться с буквы и содержать не более пяти алфавитно-цифровых символов. Для некоторых блоков метка не ставится. ОПЕРАТОР – это ключевое слово, указывающее конкретную функцию, выполняемую данным блоком. Далее после имени блока через пробел располагаются операнды, уточняющие и конкретизирующие выполнение функции, определённой в поле операции. Операнды разделяются запятыми. Количество операндов определяется типом блока и его функциональным назначением в данной программе. Некоторые операции вообще не имеют операндов, а в некоторых операнды могут быть пропущены. Если некоторое частичное поле остается пустым, т.е. не нужен соответствующий аргумент, это отмечается еще одной запятой. Комментарий можно поместить в рабочей строке после точки с запятой или в строке, в первой позиции которой будет находиться звездочка (*).

Объекты вычислительной категории описывают связи между элементами системы, задаваемые с помощью аналитических или логических соотношений. Они могут служить для задания вероятностных законов распределения случайных величин в имитационной модели, для численного или логического описания условий, определяющих движение транзактов. В качестве объектов вычислительной категории введены арифметические и булевские переменные и функции.

К ***статистическим объектам*** относятся *очереди* и *таблицы*, вводимые для оценки характеристик поведения системы. Транзакт помещается в очередь в том случае, когда некоторое устройство не в состоянии обслужить его немедленно (например, устройство занято, либо память переполнена). Статистические таблицы используются для получения характеристик и частотных распределений определенных аргументов, например, времени задержки транзакта в модели в целом или в отдельных ее частях, длин очередей, содержимого памяти и так далее. В конце эксперимента с моделью результаты в таблицах выводятся на печать.

Запоминающие объекты служат для задания условий моделирования, хранения, накопления и обработки информации, получение которой не предусмотрено стандартными средствами GPSS. К объектам запоминающей категории относятся ячейки и матрицы.

Объекты группирующей категории содержат информацию о транзактах, находящихся в модели. К ним относятся списки и группы.

Стандартные числовые атрибуты. Все элементы, используемые в GPSS, имеют специальные числовые характеристики, которые называются *стандартными числовыми атрибутами* (СЧА). Например, для транзактов можно применять такие СЧА, как приоритет или время пребывания в модели. В GPSS СЧА имеют следующие объекты: транзакты, блоки, устройства, многоканальные устройства, очереди, таблицы, ячейки и матрицы ячеек сохраняемых величин, списки и группы, вычислительные объекты (СЧА вычисленного значения функции, СЧА вычисленного значения переменной). С помощью СЧА пользователь получает доступ к характеристикам состояния системы в процессе моделирования.

Следует отметить, что символьные обозначения СЧА нельзя использовать в качестве меток операторов и блоков программы GPSS.

7.2. Моделирование одноканальных устройств

Пример 1. Заявки поступают в СМО через фиксированное время (8 минут). Время обработки (обслуживания) каждой заявки распределено равномерно в интервале времени от 5 до 15 минут. После обработки заявки покидают систему. Провести обработку 100 заявок.

Для создания нового файла используется команда ***File>New>Model***.

Текст программы на GPSS:

```
GENERATE 8        ; генерация транзактов каждые 8 минут
SEIZE 1           ; занятие прибора обслуживания 1
ADVANCE 10,5      ; задержка транзакта на время от 5 до 15 минут
RELEASE 1         ; освобождение прибора 1
TERMINATE 1       ; уничтожение транзакта
START 100         ; запуск процесса моделирования для 100 транзактов
```

Рассмотрим назначение основных блоков модели.

Для создания транзактов (заявок), входящих в модель, служит блок GENERATE (генерировать), имеющий следующий формат:

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

В поле <A> задается среднее значение интервала времени между моментами поступления в модель двух последовательных транзактов. Если этот интервал постоянен, то поле <B> не используется, иначе в поле <B> указывается разброс интервала времени относительно среднего значения. Разброс может быть задан в виде модификатора-интервала или модификатора-функции. Модификатор-интервал используется, когда интервал поступления транзактов является случайной величиной с равномерным законом распределения (диапазон изменения интервала поступления транзактов имеет границы A-B, A+B). Модификатор-функция используется, если закон распределения интервала поступления отличен от равномерного. В поле <C> задается время появления первого транзакта. Если это поле пусто или равно 0, то момент появления первого транзакта определяется операндами <A> и <B>. Поле <D> задает количество транзактов, которое должно быть сгенерировано, после чего генерация транзактов прекращается. Если это поле пусто, то блок генерирует неограниченное число транзактов до завершения моделирования. В поле <E> указывается значение приоритета генерируемых транзактов, которое может лежать в диапазоне от 0 до 127, причем большее значение соответствует более высокому приоритету. Если поле <E> пусто, то генерируемые транзакты имеют нулевой приоритет. СЧА с названием PR позволяет ссылаться на приоритет транзакта.

Вход транзактов в блок GENERATE не допускается. В этом случае происходит ошибка исполнения и прерывание работы программы.

Блок занятия устройства SEIZE имеет следующий формат**:**

SEIZE A

В поле <A> задается номер (имя) занимаемого устройства. Устройство остается занятым до тех пор, пока занимающее его сообщение не войдет в соответствующий парный блок RELEASE.

Блок освобождения устройства RELEASE имеет следующий формат**:**

RELEASE A

В поле <A> задается номер (имя) освобождаемого устройства.

Транзакты обслуживаются устройствами в течение некоторого промежутка времени. Для моделирования такого обслуживания, то есть для задержки транзактов на определенный отрезок модельного времени, служит блок задержки транзактов ADVANCE, имеющий следующий формат:

ADVANCE A,[B]

Операнды в полях <A> и <B> задают среднее значение и разброс интервала времени, на которое транзакт задерживается в устройстве. Транзакты, входящие в блок ADVANCE, переводятся из списка текущих событий в список будущих событий, а по истечении вычисленного времени задержки возвращаются назад в список текущих событий и их продвижение в модели продолжается. Если вычисленное время задержки равно 0, то транзакт в тот же момент модельного времени переходит в следующий блок, оставаясь в списке текущих событий.

В примере 1 транзакты, поступающие в модель из блока GENERATE каждые 8 минут, попадают в блок SEIZE и занимают устройство с номером 1. Далее в блоке ADVANCE определяется случайное время задержки транзакта, имеющее равномерное распределение на отрезке [5;15], и транзакт переводится в список будущих событий. По истечении времени задержки транзакт возвращается в список текущих событий, входит в блок RELEASE и освобождает устройство 1.

Для удаления транзактов из модели и подсчета удаленных транзактов служит блок TERMINATE (завершить), имеющий следующий формат:

TERMINATE [A]

Значение поля <A> указывает, на сколько единиц уменьшается содержимое счетчика завершений при входе транзакта в данный блок TERMINATE. Каждый транзакт, проходящий через блок TERMINATE, вычитает из счетчика A единиц. Если поле <A> не определено, то оно считается равным 0, и транзакты, проходящие через такой блок, не уменьшают содержимого счетчика завершений.

Начальное значение счетчика завершений устанавливается в поле <A> управляющего оператора START, предназначенного для запуска прогона модели. Прогон модели заканчивается, когда содержимое счетчика завершений обращается в 0. Счетчик завершения задает число транзактов, которые выйдут из модели, в отличие от значения поля <D> блока GENERATE, определяющего число транзактов, которые войдут в модель. По окончании моделирования транзакты могут оставаться в модели.

Участок модели, связанный с парой блоков GENERATE-TERMINATE, называется сегментом. Простые модели могут состоять из одного сегмента, в сложных моделях может быть несколько сегментов.

Для запуска программы на выполнение необходимо выбрать пункт меню ***Command>Create Simulation***. В процессе выполнения программы собирается стандартная статистическая информация, которая автоматически выводится на печать по окончании моделирования. Рассмотрим результаты моделирования по стандартному отчету примера 1 (основные обозначения приведены в таблице 1.1):

```
         START TIME           END TIME  BLOCKS  FACILITIES  STORAGES
              0.000           1023.219     5        1           0

LABEL                LOC  BLOCK TYPE     ENTRY COUNT CURRENT COUNT RETRY
                      1    GENERATE          127             26      0
                      2    SEIZE             101              1      0
                      3    ADVANCE           100              0      0
                      4    RELEASE           100              0      0
                      5    TERMINATE         100              0      0

FACILITY ENTRIES UTIL.  AVE. TIME AVAIL. OWNER PEND INTER RETRY DELAY
 1          101   0.992    10.052    1     101    0    0     0     26

CEC XN   PRI          M1      ASSEM  CURRENT  NEXT  PARAMETER    VALUE
   101    0      808.000     101       2       3

FEC XN   PRI         BDT      ASSEM  CURRENT  NEXT  PARAMETER    VALUE
   128    0     1024.000     128       0       1
```

Таблица 1.1

Обозначение	**Описание**
	Общая статистика
START TIME	время начала моделирования
END TIME	время окончания моделирования
BLOCKS	количество блоков, используемых в программе
FACILITIES	количество устройств, используемых в программе
STORAGES	количество многоканальных устройств
LOC	номер блока, назначенный системой
BLOCK TYPE	название блока
ENTRY COUNT	общее количество транзактов, прошедших через блок за время моделирования
CURRENT COUNT	количество транзактов, задержанных в блоке на момент конца моделирования
RETRY	количество транзактов, ожидающих специальных условий для прохождения через данный блок
	Отчет о работе устройств
FACILITY	название устройства
ENTRIES	количество транзактов, прошедших через устройство
UTIL	вероятность загрузки устройства (часть периода моделирования, когда устройство было занято)
AVE. TIME	среднее время обработки одного транзакта устройством
AVAIL	состояние готовности устройства на момент конца моделирования (1 – готово к обслуживанию очередной заявки; 0 – не готово)
OWNER	номер последнего транзакта занимающего устройство

Обозначение	Описание
	(если не занималось, то значение 0)
PEND	количество транзактов, находящихся в режиме прерывания, ожидающих устройство
INTER	количество транзактов, прерывающих устройство в данный момент
RETRY	количество транзактов, ожидающих специальных условий, зависящих от состояния устройства
DELAY	количество транзактов, ожидающих занятия или освобождения устройства
Статистические данные об очередях	
QUEUE	имя очереди
MAX	максимальная длина очереди
CONT.	текущее значение длины очереди
ENTRY	общее количество входов в очередь
ENTRY(0)	общее количество входов в очередь без последующего ожидания («нулевые» входы)
AVE.CONT.	средняя длина очереди
AVE.TIME	среднее время пребывания транзактов в очереди
AVE.(-0)	среднее время пребывания в очереди с учетом всех входов и без учета «нулевых» входов
RETRY	количество транзактов, ожидающих специальных условий
Информация о списке текущих событий *CEC (Current Events Chain)*	
XN	номер транзакта
PRI	приоритет транзакта (по умолчанию - 0)
M1	время пребывания транзакта в системе с момента начала моделирования
ASSEM	номер семейства транзактов
CURRENT	номер блока, в котором находится транзакт
NEXT	номер блока, в который перейдет транзакт далее
PARAMETER	номер или имя параметра транзакта
VALUE	значение параметра
Информация о списке будущих событий *FEC (Future Events Chain)*	
XN	номер транзакта
PRI	приоритет транзакта
BDT	таблица модельных событий – абсолютное модельное время выхода транзакта из списка будущих событий (и перехода транзакта в список текущих событий)
ASSEM	номер семейства транзактов
CURRENT	номер блока, в котором находится транзакт (0 – если транзакт не вошел в модель)
NEXT	номер блока, в который перейдет транзакт далее

Обозначение	Описание
PARAMETER	номер или имя параметра транзакта
VALUE	значение параметра

В рассмотренном примере количество сгенерированных транзактов равно 127. Так как время генерирования транзактов меньше, чем среднее время их обработки, обработано и выведено из системы только 100 транзактов. На момент окончания моделирования в устройство поступил 101-й по счету транзакт. Это число показано в поле OWNER статистики устройства. 26 транзактов ждут освобождения устройства. Среднее время обработки одного транзакта устройством 10.052, вероятность загрузки устройства близка к единице.

Статистические данные об очередях могут быть получены с помощью блоков занятия и освобождения очереди QUEUE (вход в очередь) и DEPART (выход из очереди). Формат записи блока QUEUE**:**

QUEUE A,[B]

В поле <A> задается номер или имя очереди. Поле <B> определяет количество мест в очереди, занимаемое транзактом (по умолчанию 1). Поскольку очередь обычно используется для измерения времени ожидания, за блоком QUEUE обычно следуют такой блок как SEIZE, который может задержать сообщение. Формат блока DEPART:

DEPART A,[B]

В поле <A> задается номер или имя очереди. Поле <B> определяет количество мест в очереди, освобождаемое транзактом. Это число не должно превышать текущую длину очереди. Если поле <B> пусто, длина очереди уменьшается на единицу.

Внесите следующие изменения в программу примера 1:

```
GENERATE 8        ; генерация транзактов каждые 8 минут
QUEUE OCH1        ; занятие очереди och1
SEIZE 1           ; занятие прибора обслуживания 1
DEPART OCH1       ; освобождение очереди och1
ADVANCE 10,5      ; задержка транзакта на время от 5 до 15 минут
RELEASE 1         ; освобождение прибора 1
TERMINATE 1       ; уничтожение транзакта
START 100         ; запуск процесса моделирования для 100 транзактов
```

В стандартном отчете должен появиться блок статистики по очередям:

```
QUEUE     MAX CONT. ENTRY ENTRY(0) AVE.CONT. AVE.TIME    AVE.(-0) RETRY
 OCH1      27   27    127       1    12.795    103.089     103.908   0
```

В приведенном примере точное значение таймера в момент завершения прогона модели неизвестно, так как продолжительность прогона устанавливается не по модельному времени, а по количеству транзактов, прошедших через модель. Если необходимо управлять продолжительностью прогона по модельному времени, то в модели используется специальный сегмент, называемый сегментом таймера.

<u>Пример 2.</u> В систему массового обслуживания поступают заявки, распределенные по равномерному закону в интервале от 3 до 7 минут. Обработка поступивших заявок осуществляется также по равномерному закону распределения в интервале от 5 до 9 минут. Необходимо смоделировать работу системы обслуживания в течение 120 минут.

```
GENERATE 5,2     ; генерация транзактов
SEIZE 1          ; занятие прибора обслуживания 1
ADVANCE 7,2      ; задержка транзакта на время от 5 до 9 минут
RELEASE          ; освобождение прибора 1
TERMINATE        ; уничтожение обработанных транзактов без уменьшения
                 ; счетчика
GENERATE 120     ; сегмент таймера: генерация транзакта через 120 мин
TERMINATE 1      ; сегмент таймера: уничтожение транзакта
START 1          ; задание числа счетчика завершений
```

Стандартный отчет:

```
          START TIME           END TIME  BLOCKS  FACILITIES  STORAGES
               0.000            120.000     7        1           0

LABEL              LOC  BLOCK TYPE     ENTRY COUNT CURRENT COUNT RETRY
                    1    GENERATE          25             9        0
                    2    SEIZE             16             0        0
                    3    ADVANCE           16             1        0
                    4    RELEASE           15             0        0
                    5    TERMINATE         15             0        0
                    6    GENERATE           1             0        0
                    7    TERMINATE          1             0        0

FACILITY ENTRIES  UTIL. AVE. TIME AVAIL. OWNER PEND INTER RETRY DELAY
 1          16    0.947   7.102    1      17    0     0     0      9
```

Как следует из отчета, за 120 минут было сгенерировано 25 транзактов в условном вложенном цикле, который отражает основную работу моделируемой системы. Обработано и выведено из системы 15 транзактов. На момент окончания моделирования в устройство поступил 17-й по счету транзакт внутреннего цикла. Это число показано в поле OWNER статистики устройства. Среднее время занятости устройства составляет 7.102 минут. Коэффициент использования устройства равен 0.947. Число транзактов, ожидающих специальных условий для устройства, равно 9. Поле DELAY статистики устройства заполнено (число 9) потому, что работа внутреннего цикла была прервана внешним циклом путем назначенного времени работы (число 120).

В приведенной модели из двух сегментов, первый (основной) сегмент выполняет те же функции, что и в предыдущем примере. Однако поле <A> блока TERMINATE в первом сегменте пусто, т.е. уничтожаемые транзакты не уменьшают содержимого счетчика завершений. Во втором сегменте блок GENERATE создаст первый транзакт в момент модельного времени, равный 120. Но этот транзакт окажется и последним в данном сегменте, так как, войдя в блок TERMINATE, он обратит в 0 содержимое счетчика завершений, установленное оператором START равным 1. Таким образом, в этой модели гарантируется завершение прогона в определенный момент модельного времени, а точное количество транзактов, прошедших через модель, не определено. Если по условию задачи требуется, чтобы все сгенерированные транзакты были обработаны, используется поле <D> блока GENERATE. Подобное условие задачи часто требуется для уточнения характера работы системы, когда обслуживание заявок осуществляется медленнее, чем поступление этих заявок в систему.

Пример 3. В систему массового обслуживания поступает фиксированное число заявок через интервалы времени, распределенные равномерно от 3 до 7 минут. Обработка заявок осуществляется также по равномерному закону в интервале времени от 5 до 9 минут. Необходимо смоделировать генерирование и обработку в системе 100 заявок.

```
GENERATE 5,2,,100  ; генерация 100 транзактов
SEIZE 1            ; занятие прибора обслуживания 1
ADVANCE 7,2        ; задержка транзакта на время от 5 до 9 минут
RELEASE 1          ; освобождение прибора 1
TERMINATE 1        ; уничтожение транзактов без уменьшения счетчика
START 100          ; задание числа счетчика завершений
```

Стандартный отчет:

```
          START TIME           END TIME  BLOCKS  FACILITIES  STORAGES
               0.000            715.916     5        1           0

LABEL              LOC  BLOCK TYPE     ENTRY COUNT CURRENT COUNT RETRY
                    1    GENERATE           100             0       0
                    2    SEIZE              100             0       0
                    3    ADVANCE            100             0       0
                    4    RELEASE            100             0       0
                    5    TERMINATE          100             0       0

FACILITY ENTRIES  UTIL.  AVE. TIME AVAIL. OWNER PEND INTER RETRY DELAY
 1        100     0.991    7.095     1      0    0     0     0     0
```

Число 100 установлено в поле <D> блока GENERATE и в поле <A> оператора START. Несмотря на то, что время генерации транзактов в примере меньше времени обработки, поле CURRENT_COUNT файла стандартного отчета нулевое, так как число сгенерированных транзактов равно числу счетчика завершений. Моделирование системы будет происходить до тех пор, пока не будут сгенерированы и выведены все 100 транзактов.

7.3. Моделирование многоканальных устройств

<u>Пример 4.</u> Пусть в вычислительной системе имеются два компьютера (интенсивность обработки заданий одинаковая). В вычислительную систему на обработку поступают задания. Интервал времени между двумя последовательными поступлениями заданий подчиняется равномерному закону распределения в интервале 1-11 минут. Перед вычислительной системой допустима очередь заданий, длина которой не ограничена. Время выполнения задания также равномерно распределено в интервале 1-19 минут.

Смоделировать обработку 100 заданий.

```
VS STORAGE 2      ; задать МКУ (емкость 2)
GENERATE 6,5      ; генерация транзактов каждые 1-11 минут
QUEUE OCH1        ; занятие очереди och1
ENTER VS          ; занятие МКУ
DEPART OCH1       ; освобождение очереди och1
ADVANCE 10,9      ; задержка транзакта на время от 1 до 19 минут
LEAVE VS          ; освобождение МКУ
TERMINATE 1       ; уничтожение транзакта
START 100         ; запуск процесса моделирования для 100 транзактов
```

Если параллельно работающие устройства являются одинаковыми, то для их моделирования может использоваться объект многоканальное устройство (МКУ) или накопитель, память. Формат оператора:

Метка STORAGE A

Метка – имя МКУ; <A> – емкость МКУ (количество однотипных устройств, входящих в МКУ). Если память имеет ёмкость равную единице, то она аналогична устройству, осуществляющему одноканальное обслуживание. Различие: обслуживание в устройстве может быть прервано транзактом с более высоким приоритетом, чего не допускается при работе с памятью.

Блоки SEIZE – RELEASE заменены соответственно на блоки ENTER – LEAVE, моделирующие работу многоканального устройства. Блок входа в устройство ENTER имеет следующий формат записи:

ENTER A,[B]

В поле <A> указывается номер или имя МКУ, поле <B> задает емкость МКУ (по умолчанию 1). Если значение поля <B> равно нулю, то транзакт не задерживается на входе, а блок рассматривается как нерабочий. Также транзакт может быть задержан на входе в блок, если многоканальное устройство заполнено или имеющейся емкости недостаточно, или устройство в данный момент недоступно.

Блок выхода из устройства LEAVE имеет следующий формат:

LEAVE A,[B]

Поле <A> определяет номер (имя) МКУ. Блок LEAVE освобождает заданное число единиц МКУ, определенное в поле <B> (по умолчанию 1). Занятый объем МКУ уменьшается на число освобождаемых единиц, оставшаяся емкость увеличивается на ту же величину. Число освобождаемых единиц не должно превышать текущее содержимое МКУ. Счетчик числа входов не изменяется.

Стандартный отчет (обозначения приведены в таблице 1.2):

```
         START TIME           END TIME  BLOCKS  FACILITIES  STORAGES
              0.000            599.856     7        0           1

            NAME                       VALUE
         OCH1                        10001.000
         VS                          10000.000

LABEL          LOC  BLOCK TYPE     ENTRY COUNT CURRENT COUNT RETRY
                 1    GENERATE           100             0       0
                 2    QUEUE              100             0       0
                 3    ENTER              100             0       0
                 4    DEPART             100             0       0
                 5    ADVANCE            100             0       0
                 6    LEAVE              100             0       0
                 7    TERMINATE          100             0       0

QUEUE     MAX CONT. ENTRY ENTRY(0) AVE.CONT. AVE.TIME   AVE.(-0) RETRY
 OCH1     4    0    100      28     0.727     4.359      6.054     0

STORAGE    CAP. REM. MIN. MAX.   ENTRIES AVL.  AVE.C. UTIL. RETRY DELAY
 VS          2    2    0    2      100    1    1.756  0.878    0     0
```

Таблица 1.2

Обозначение	**Описание**
STORAGE	имя МКУ
CAP.	емкость МКУ, заданная оператором STORAGE
REM.	количество единиц свободной емкости в конце периода моделирования
MIN.	минимальное количество емкости за используемый период
MAX.	максимальное количество емкости за используемый период
ENTRIES	количество входов в МКУ за период моделирования
AVL.	состояние готовности МКУ в конце периода моделирования (1 – МКУ готов, 0 – нет)
AVE.C.	среднее значение занятой емкости за период моделирования
UTIL.	средний коэффициент использования всех устройств, входящих в МКУ
RETRY	количество транзактов, ожидающих специальных условий, зависящих от состояния МКУ
DELAY	количество транзактов, ожидающих занятия или освобождения устройства МКУ

7.4. Моделирование систем с использованием блоков передачи управления, работа с логическими ключами

Блок передачи транзактов TRANSFER служит для изменения маршрута движения транзакта. Блок TRANSFER имеет следующий формат:

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

Поле <A> задает режим выбора следующего блока, к которому должно перейти сообщение. Поля <B> и <C> задают возможные значения номеров следующих блоков или их положение. Поле <D> представляет собой константу, используемую для вычисления возможных адресов движения транзактов. Оператор имеет пять основных режимов работы.

Безусловный режим выбора

Если операнд <A> отсутствует, то блок TRANSFER работает в безусловном режиме. Входящее в блок TRANSFER сообщение переходит к блоку, указанному в поле <B>. Если сообщение в этот блок войти не может, попытка направить сообщение к какому-либо другому блоку не производится.

<u>Пример 5.</u> На вход СМО поступают три потока заявок: заявки первого потока имеют наивысший приоритет 3 и поступают с интервалом 5-15 минут; заявки второго потока – приоритет 2 и интервал поступления 15-25 минут; третьего потока – наименьший приоритет 1 и интервал поступления 25-35 минут. Время обработки заявки от 4 до 10 минут. Интервалы времени между двумя последовательными поступлениями заявок и время обработки заявки подчиняются равномерному закону распределения. Требуется смоделировать процесс обработки 1000 заявок.

```
GENERATE 10,5,,,3     ; генерация первого потока заявок с высшим
                      ; приоритетом
TRANSFER ,MET1        ; безусловная передача управления на метку MET1
GENERATE 20,5,,,2     ; генерация второго потока заявок с приоритетом 2
TRANSFER ,MET1        ; безусловная передача управления на метку MET1
GENERATE 30,5,,,1     ; генерация третьего потока с низшим приоритетом
MET1 ASSIGN 1,PR      ; присваивание параметру 1 транзакта значения,
                      ; соответствующего его приоритету
QUEUE P1              ; занятие очереди, имеющей номер приоритета
                      ; транзакта
SEIZE 1               ; занятие устройства 1
DEPART P1             ; освобождение соответствующей очереди
ADVANCE 7,3           ; обработка транзакта
RELEASE 1             ; освобождение устройства 1
TERMINATE 1           ; удаление транзакта
START 1000            ; моделирование процесса прохождения 1000 заявок
```

Стандартный отчет показывает, что на момент окончания моделирования 272 сгенерированных заявки остались в очереди. Из них только одна заявка с наивысшим приоритетом не была обработана. Фрагмент отчета:

```
          START TIME         END TIME  BLOCKS  FACILITIES  STORAGES
             0.000           6953.277    12         1          0

LABEL                LOC  BLOCK TYPE     ENTRY COUNT CURRENT COUNT RETRY
                       1    GENERATE            696             0     0
                       2    TRANSFER            696             0     0
                       3    GENERATE            346             0     0
                       4    TRANSFER            346             0     0
                       5    GENERATE            231             0     0
MET1                   6    ASSIGN             1273             0     0
                       7    QUEUE              1273           272     0
                       8    SEIZE              1001             1     0
                       9    DEPART             1000             0     0
                      10    ADVANCE            1000             0     0
                      11    RELEASE            1000             0     0
                      12    TERMINATE          1000             0     0

FACILITY   ENTRIES UTIL. AVE.TIME AVAIL.  OWNER PEND  INTER RETRY DELAY
 1          1001   0.998   6.932    1      1273   0     0     0    272

QUEUE     MAX CONT. ENTRY ENTRY(0) AVE.CONT. AVE.TIME    AVE.(-0) RETRY
 1        230  230    231     0     114.302   3440.574   3440.574   0
 2         42   42    346     0      26.288    528.296    528.296   0
 3          2    1    696     2       0.427      4.268      4.281   0
```

Статистический режим выбора

Значение операнда <A> задает вероятность перехода транзакта к блоку, указанному в поле <C>. Остальные сообщения направляются по альтернативному адресу к блоку, указанному в поле <B>, или к следующему по номеру блоку, если операнд <B> пропущен. Если вычисленное значение аргумента меньше или равно нулю, то произойдет безусловная передача сообщения к блоку, указанному в поле <B>. Если же значение аргумента больше или равно 1000, произойдет безусловная передача сообщений к блоку, указанному в поле <C>.

Пример 6. Интервал времени поступления заявок в СМО распределен равномерно в диапазоне 3-7 минут. Обработка заявок осуществляется двумя обслуживающими каналами. Поступление заявок в первый или второй канал происходит с вероятностью 0,3 и 0,7 соответственно. Время обслуживания заявок каждым каналом распределено равномерно в интервале 5-9 минут. Необходимо осуществить обработку 100 заявок.

```
GENERATE 5,2,,100          ; генерация потока 100 заявок
TRANSFER .7,MET1,MET2      ; передача 70% заявок в устройство 2
MET1 SEIZE 1               ; занятие устройства 1
ADVANCE 7,2                ; обработка транзакта
```

```
RELEASE 1                  ; освобождение устройства 1
TRANSFER ,EX               ; безусловный переход на метку EX
MET2 SEIZE 2               ; занятие устройства 2
ADVANCE 7,2                ; обработка транзакта
RELEASE 2                  ; освобождение устройства 2
EX TERMINATE 1             ; удаление транзакта
START 100             ; моделирование процесса прохождения 100 заявок
```

Стандартный отчет показывает, что 66 заявок из 100 обработано вторым каналом, а 34 – первым. Фрагмент отчета:

FACILITY	ENTRIES	UTIL.	AVE. TIME	AVAIL.	OWNER	PEND	INTER	RETRY	DELAY
1	34	0.497	7.330	1	0	0	0	0	0
2	66	0.917	6.972	1	0	0	0	0	0

При исследовании различных систем методом имитационного моделирования часто возникает необходимость выполнения многократных прогонов модели. Для того, чтобы выполнить несколько прогонов модели, можно использовать оператор **CLEAR.** В примере 6 после оператора START 100 нужно добавить операторы

```
CLEAR
START 100         ; моделирование еще 100 заявок (второй прогон)
```

Фрагмент отчета второго прогона:

FACILITY	ENTRIES	UTIL.	AVE. TIME	AVAIL.	OWNER	PEND	INTER	RETRY	DELAY
1	31	0.399	6.819	1	0	0	0	0	0
2	69	0.891	6.833	1	0	0	0	0	0

Режим BOTH

Если в поле <A> стоит зарезервированное слово BOTH, блок TRANSFER работает в режиме условного перехода с одним альтернативным адресом. В этом режиме в поле <B> указывается блок, к которому направляется транзакт (основной адрес). Если это сделать не удается, сообщение пытается перейти к блоку, указанному в поле <C> (альтернативный адрес). Если сообщение не сможет перейти ни к одному, ни к другому блоку, оно остается в блоке TRANSFER и будет повторять в том же порядке попытки перехода при каждом просмотре списка текущих событий до тех пор, пока не сможет выйти из блока TRANSFER. Ниже приведен фрагмент программы, в котором сообщение сначала пытается перейти к блоку TRY1; если оно не может войти в этот блок, оно пытается войти в блок TRY2. Иначе сообщение остается в списке текущих событий и повторяет эти попытки до тех пор, пока не выйдет из блока TRANSFER.

```
TRANSFER  BOTH, TRY1, TRY2
TRY1  SEIZE  1
TRY2  SEIZE  2
```

Режим ALL

Если в поле <A> стоит зарезервированное слово **ALL**, блок TRANSFER работает в режиме условного перехода со многими альтернативами. В этом режиме каждое входящее сообщение прежде всего пытается перейти к блоку, указанному в поле <B> (основной адрес). Если сообщение в этот блок войти не может, то последовательно проверяются блоки в поисках первого, способного принять это сообщение, включая блок, указанный операндом <C> (последний адрес). Номер каждого проверяемого блока вычисляется как сумма номера предыдущего блока и шага, заданного операндом <D>:

$$\boldsymbol{n + m, n + 2m, n + 3m, \ldots l,}$$

где n – номер блока, указанного в поле <B>; m – значение шага, заданного в поле <D>; l – номер блока, указанного в поле <C>. Этот номер должен быть больше номера блока, указанного в поле <B>, на величину, кратную шагу m. Как только первый блок, способный принять сообщение, будет найден, сообщение входит в этот блок и оттуда продолжает свое дальнейшее движение. Если сообщение не может перейти ни к одному из указанных блоков, оно остается в блоке TRANSFER и повторяет данную процедуру при каждом просмотре списка текущих событий до тех пор, пока не выйдет из блока.

Если устройства в примере 6 равнозначны (вероятность поступления заявки в первый или второй канал равна 0,5), оператор, передающий заявку в свободное устройство, должен выглядеть следующим образом:

TRANSFER ALL,MET1,MET2,4

Блоки следует располагать таким образом, чтобы разность между номерами блоков, указанных в полях <B> и <C>, была кратна шагу, указанному в поле <D>. Например, при использовании оператора

TRANSFER ALL, 60, 120, 25

режим ALL недопустим, потому что разность между номерами блоков, записанных в полях <B> и <C>, не является кратной шагу, указанному в поле <D>.

Следует отметить, что каждый раз, когда интерпретатор при просмотре списка текущих событий обнаруживает сообщение, задержанное в блоках TRANSFER BOTH или TRANSFER ALL, он пытается продвинуть сообщение, начиная с блока, указанного в поле <B>. Следовательно, в режиме BOTH в тех случаях, когда возможен переход к обоим блокам (<B> и <C>), блок <B> имеет некоторое преимущество. Аналогично, в режиме ALL в случае, когда возможен переход к нескольким блокам, блоки с меньшими номерами имеют некоторое преимущество перед блоками с большими номерами.

Режим PICK

Если в поле <A> стоит зарезервированное слово PICK, блок TRANSFER случайным образом выбирает один блок, к которому должно быть направлено сообщение из последовательности блоков с номерами ***n*,**

n+1, n+2,...m (*n* - номер блока, указанного поле <B>, а *m* - номер блока, указанного в поле <C>). Все блоки, включая указанные в полях <B> и <C>, выбираются с одинаковой вероятностью, равной $1/((m\text{-}n)+1)$. Если сообщение не может сразу перейти к следующему блоку, то оно будет ждать в блоке TRANSFER до тех пор, пока не будет снято блокирующее условие. Номер блока в поле <C> должен быть больше или равен n+1. В примере:

TRANSFER PICK,10,19

сообщение, вошедшее в блок TRANSFER, пытается войти в один из 10 блоков (10,11,...19) с равной вероятностью: 1/10.

Помимо блока TRANSFER, потоком сообщений может управлять блок TEST, который определяет направление движения транзакта в зависимости от выполнения некоторого условия (алгебраического соотношения). Блок TEST имеет следующий формат:

TEST X A,B,[C]

Операнды <A> и <B> – сравниваемые величины, которые могут быть именем, любым целым числом или СЧА. Во вспомогательном поле операции <X> оператора описания блока TEST записывается знак логической операции – один из шести условных операторов:

L – меньше; отношение истинное, если значение аргумента поля <A> меньше значения аргумента поля <B>;
LE – меньше или равно;
E – равно;
NE – не равно;
G – больше;
GE – больше или равно.

Если отношение СЧА, заданных в полях <A> и <B>, истинно, сообщение переходит к следующему блоку. Если отношение ложно, сообщение переходит к блоку, номер которого задан полем <C>. Если поле <C> пусто, транзакт задерживается в блоке TEST до выполнения условия.

Например:

```
TEST G M1,500,MET
SEIZE 1
…
MET SEIZE 2
```

Если значение времени пребывания транзакта в модели больше 500, то переходим к следующему по номеру блоку, ложно – к метке MET.

<u>**Пример 7.**</u> Интервал времени поступления заявок в СМО распределен равномерно в диапазоне от 1 до 11 минут. Обработка заявок длится от 1 до 19 минут. Так как обслуживающее устройство не успевает обрабатывать заявки, образуется очередь. Если длина очереди больше трех, то заявка выводится из системы без обработки. Смоделировать

прохождение 100 заявок. Определить число заявок, покинувших систему необслуженными.

```
GENERATE 6,5        ; генерация заявок
TEST L Q$OCH,3,EX   ; если условие "длина очереди меньше 3" ложно,
                    ; удалить заявку
QUEUE OCH           ; занятие очереди
SEIZE 1             ; занятие устройства 1
DEPART OCH          ; освобождение очереди
ADVANCE 10,9        ; обслуживание заявки
RELEASE 1           ; освобождение устройства 1
EX TERMINATE 1      ; удаление транзакта
START 100           ; моделирование процесса прохождения 100 заявок
```

Стандартный отчет показывает, что при полной загрузке устройства было обслужено (прошло через блок RELEASE) 66 заявок, 34 получили отказ из-за заполнения очереди. Моделирование было остановлено, как только из системы было удалено 100 транзактов.

Блок проверки состояния элементов GATE разрешает движение транзактов при определенном состоянии устройств, накопителей, ключей. Формат записи блока GATE:

GATE X A, [B]

Поле <A> содержит имя или номер объекта, для которого проводится проверка во вспомогательном поле операции <X>. Поле <B> содержит номер или имя блока, в который переходит транзакт, если проверяемое условие не выполняется. Мнемонические обозначения проверяемого условия записываются непосредственно после поля GATE.

Состояние устройства описывается следующими условиями:

U – устройство используется, занято;
NU – устройство не используется, свободно;
I – устройство обслуживает прерывание;
NI – устройство работает без прерывания;
FV – устройство доступно;
FNV – устройство недоступно.

Состояние накопителя (памяти) описывается следующими условиями:

SE – накопитель пуст;
SNE – накопитель не пуст;
SF – накопитель заполнен;
SNF – накопитель не заполнен;
SV – накопитель доступен;
SNV – накопитель не доступен.

Состояние логического ключа описывается следующими условиями:

LR – ключ в состоянии «выключен»;
LS – ключ в состоянии «включен».

Существует два режима работы блока GATE: режим отказа, или условного входа и режим перехода, или безусловного входа. При работе в режиме отказа блок GATE блокирует транзакт, если соответствующий объект не находится в требуемом состоянии. Если же поставленное условие удовлетворяется, блок разрешает вход транзактов. Если в поле <B> указано имя (номер) блока, то вместо отказа блок GATE будет направлять транзакт на указанный блок.

Пример 8. Поток заявок поступает в СМО с двумя обслуживающими каналами равномерно с интервалом 4-6 минут. Если первый канал занят, то заявки поступают на обработку во второй канал. Время обработки первого канала распределено равномерно в промежутке 8-10 минут, второго – 12-14 минут. Смоделировать процесс обработки 100 заявок.

```
GENERATE 5,1       ; генерация потока 100 заявок
GATE NU 1,MET2     ; если устройство 1 занято, отправить заявку на
                   ; устройство 2
SEIZE 1            ; занятие устройства 1
ADVANCE 9,1        ; обработка транзакта
RELEASE 1          ; освобождение устройства 1
TRANSFER ,EX       ; безусловный переход на метку EX
MET2  SEIZE 2      ; занятие устройства 2
ADVANCE 13,1       ; обработка транзакта
RELEASE 2          ; освобождение устройства 2
EX TERMINATE 1     ; удаление транзакта
START 100          ; моделирование процесса прохождения 100 заявок
```

Для иллюстрации работы блока проверки состояния элементов GATE с накопителем (памятью) изменим условие предыдущего примера.

Пример 9. Поток заявок поступает в накопитель с допустимой емкостью, равной 3 единицам, равномерно с интервалом 4-6 минут. После накопителя заявки поступают на обработку в первый обслуживающий канал. Если накопитель заполнен, то заявки поступают на обработку во второй канал. Смоделировать процесс обработки 100 заявок.

```
NAK STORAGE 3         ; накопитель емкостью 3 единицы
GENERATE 5,1          ; генерация потока 100 заявок
GATE SNF NAK,MET2     ; если накопитель заполнен, отправить заявку на
                      ; устройство 2
ENTER NAK             ; добавление заявки в накопитель
QUEUE 1               ; занятие очереди к устройству 1
SEIZE 1               ; занятие устройства 1
DEPART 1              ; освобождение очереди к устройству 1
LEAVE NAK             ; удаление заявки из накопителя
```

```
ADVANCE 9,1          ; обработка транзакта
RELEASE 1            ; освобождение устройства 1
TRANSFER ,EX         ; безусловный переход на метку EX
MET2  QUEUE 2        ; занятие очереди к устройству 2
SEIZE 2              ; занятие устройства 2
DEPART 2             ; освобождение очереди к устройству 2
ADVANCE 13,1         ; обработка транзакта
RELEASE 2            ; освобождение устройства 2
EX TERMINATE 1       ; удаление транзакта
START 100            ; моделирование процесса прохождения 100 заявок
```

Анализ стандартного отчета показывает, что использование накопителя уменьшает общее время моделирования процесса прохождения заявок и увеличивает загрузку обслуживающих каналов.

Логические ключи

Логические ключи служат для моделирования объектов с двумя логическими состояниями «включено», «выключено». Управление ключами осуществляется транзактами, продвигающимися по модели. Транзакт может установить ключ в состояние «включено», сбросить его – перевести в состояние «выключено» или инвертировать, т. е. изменить состояние ключа на противоположное значение.

Оператор установки начальных значений INITIAL записывается вне тела основной программы. Формат записи оператора INITIAL:

INITIAL A,B

В поле <A> вводится имя устанавливаемой величины, в поле <B> – начальное значение. Для инициализации логического ключа в поле <A> записывается выражение LS$имя_ключа или LSj, где j – номер ключа; в поле <B> записывается 0 или 1. В начале моделирования все ключи находятся в состоянии «выключено». Для включения, выключения и инвертирования ключей служит оператор LOGIC.

Формат записи блока LOGIC имеет вид:

LOGIC X A

Во вспомогательном поле операций <X> задается мнемоническое обозначение операции. В поле <A> задается номер или имя логического ключа, ссылка на имя ключа может происходить в любом месте программы. В поле <X> могут быть заданы следующие мнемонические обозначения:

S – установить в состояние «включен» логический переключатель;

R – сбросить логический переключатель;

I – инвертировать логический переключатель.

Формат записи блока GATE для проверки логических ключей имеет вид:

GATE X A,B

В поле <X> задается один из следующих логических операторов:

LS – логический переключатель «включен»;

LR – логический переключатель «сброшен».

В поле <A> задается номер или имя логического ключа, состояние которого проверяется. В поле <B> задается номер блока, к которому переходит транзакт, если логический оператор вспомогательного поля <X> имеет значение «ложь». Если значение логического оператора «истина», транзакт переходит к следующему по номеру блоку.

Пример 10. На обработку в СМО поступает поток заявок с интервалом времени 9-11 минут. Нечетные по номеру заявки обрабатываются на первом канале, время обработки 15 минут, четные – на втором канале со временем 17 минут. Смоделировать процесс прохождения 101 заявки.

```
INITIAL LS$KEY1,0      ; инициализация ключа с именем KEY1 (эту строку
                       ; можно опустить)
GENERATE 10,1,,101     ; генерация потока заявок
LOGIC I KEY1           ; инвертирование ключа
GATE LS KEY1,MET2      ; разделение потока четных и нечетных заявок
MET1  SEIZE 1          ; занятие устройства 1
ADVANCE 15             ; обработка заявки
RELEASE 1              ; освобождение устройства 1
TRANSFER ,EX           ; безусловная передача управления
MET2  SEIZE 2          ; занятие устройства 2
ADVANCE 17             ; обработка заявки
RELEASE 2              ; освобождение устройства 2
EX TERMINATE 1         ; удаление транзакта
START 101              ; моделирование прохождения 101 заявки
```

Фрагмент стандартного отчета:

```
FACILITY   ENTRIES UTIL. AVE.TIME AVAIL. OWNER PEND INTER RETRY DELAY
 1            51   0.744  15.000     1      0     0     0     0      0
 2            50   0.827  17.000     1      0     0     0     0      0

LOGICSWITCH          VALUE        RETRY
 KEY1                  1             0
```

Формирование потока четных и нечетных по номеру заявок осуществляется блоками LOGIC и GATE, логический переключатель «включен». Через первое устройство проходит 51 нечетная по номеру заявка, а через второе – 50 четных по номеру заявок. Если в примере вместо блока GATE с полем LS поставить блок GATE с полем LR, то четный и нечетный потоки поменяются местами: первое устройство будет обслуживать четные заявки, а второе устройство – нечетные. Если же в поле <X> блока LOGIC поставить S или R, то обработка всех поступающих заявок будет осуществляться одним из устройств. В стандартном отчете появилось поле LOGICSWITCH, которое определяет заданное имя логического ключа KEY1. Поле VALUE определяет значение логического переключателя в конце периода моделирования. Поле RETRY определяет количество транзактов, ожидающих наступления специальных условий, зависящих от состояния логического переключателя.

Практическое задание № 1. Тема: моделирование систем средствами GPSS

Задание 1. В систему массового обслуживания поступают заявки, распределенные по равномерному закону в интервале от 5 до 15 мин. Обработка поступивших заявок осуществляется также по равномерному закону распределения в интервале от 8 до 16 мин. Необходимо смоделировать работу системы обслуживания в течение 3 часов.

Провести эксперимент с компьютерной моделью и ответить на следующие вопросы:

Сколько заявок будет обработано за 3 часа?

Сколько заявок покинут систему необслуженными?

Какова максимальная длина очереди при заданных условиях?

Как должно измениться время обработки заявок, чтобы очередь не накапливалась?

Внесите изменения в программу, чтобы в системе было сгенерировано и обработано ровно 50 заявок. Сколько времени потребуется для их обработки?

Задание 2. На станцию технического обслуживания, которая состоит из бокса для ремонта и бокса для техосмотра, каждые 15-35 минут поступают автомобили. Из них 73% требуют ремонта, который продолжается 35-55 минут, а 27% проходят техосмотр (9-25 минут). Промоделируйте 40 часов работы станции технического обслуживания.

Задание 3.. Вычислительная система состоит из 3-х компьютеров. С интервалом 3-5 мин в систему поступают задания. Если первый компьютер свободен, то задание поступает на обработку к первому компьютеру (5-7 мин), иначе ко второму (7-11 мин). В случае занятости второго компьютера проверяется, свободен ли третий. Если свободен, то задание обрабатывается с интервалом 8-12 мин. Промоделируйте обработку 100 заданий. Измените условие задачи: обработка заданий может осуществляться тремя компьютерами равновероятно.

Задание 4. Выполните в среде GPSS задание с использованием блока GATE.

СМО состоит из одного прибора и очереди перед ним. Обработка заявки в приборе занимает 20±5 единиц времени. Очередь ограничена длиной 4. Заявки приходят каждые 15±3 единиц времени и если в очереди нет свободных мест, то заявки покидают модель необслуженными. Промоделировать систему в течение 1000 единиц времени.

7.5. Обработка результатов моделирования средствами GPSS

Ряд числовых атрибутов, характеризующих функционирование моделируемой системы, можно получить с помощью средств для сбора статистики. Для получения оценок математического ожидания (среднего значения) и дисперсии последовательности значений случайных величин, полученных в результате моделирования системы, используются блоки сбора статистики TABLE и TABULATE. Блок TABLE имеет следующий формат:

<NAME> TABLE A,B,C,D

Блок TABLE определяет аргумент, а также число и ширину частотных интервалов. Метка NAME определяет имя таблицы. В поле <A> задается аргумент таблицы – имя переменной, значение которой будет табулироваться. В поле <B> задается верхняя граница первого интервала. В поле <C> задается ширина частотного интервала – разница между верхней и нижней границей каждого частотного класса. В поле <D> задается число частотных интервалов. Желательно так подбирать ширину и количество интервалов, чтобы в полученные интервалы попадало 100% значений аргумента таблицы. Помимо таблицы частот одновременно вычисляются оценки среднего и стандартного отклонения аргумента таблицы. Блоки описания таблиц помещаются в начале программы.

Блок, связанный с оператором TABLE – TABULATE. Для сбора элементов данных сообщение должно войти в блок TABULATE с тем же именем таблицы, что определено в блоке TABLE. Блок TABULATE имеет следующий формат:

TABULATE A,[B]

Блок TABULATE табулирует текущее значение заданного аргумента. Способ табуляции зависит от режима работы таблицы, который определяется оператором описания таблицы TABLE.

В поле <A> задается номер или имя таблицы, в которую табулируется значение аргумента. В поле <B> задается число единиц, которые должны быть занесены в тот частотный интервал, куда попало значение аргумента. Если поле <B> пусто, эта величина полагается равной единице.

Когда сообщение входит в блок TABULATE, оценивается аргумент таблицы (операнд <A> в операторе TABLE). Если он меньше или равен операнду <B> в операторе TABLE, то выбирается первый частотный класс таблицы. Если аргумент таблицы не подходит для этого класса, то класс выбирается путем деления значения аргумента на операнд <C> оператора TABLE. Нижняя граница частотного класса включается в предыдущий класс. Если таблицы недостаточно для размещения этого значения, то выбирается последний частотный интервал. Затем выбирается целое из частотного класса и счетчик увеличивается на величину, определяемую операндом <B> оператора TABULATE. По умолчанию увеличение происходит на 1. В конце работы оператора TABULATE изменяются значения среднего и стандартного отклонения аргумента таблицы.

Стандартные числовые атрибуты, связанные с описываемым оператором, следующие: TB – среднее значение аргумента; TC – число входов в таблицу; TD – стандартное отклонение.

<u>Пример 11.</u> Интервал поступления заявок в СМО подчиняется равномерному закону распределения с диапазоном 3-7 минут. Обработка заявок происходит параллельно в двух каналах. Время обслуживания заявки первым каналом 13-17 минут, вторым – 15-19 минут. Смоделировать процесс обслуживания поступающих заявок, собрать статистику об очередях при обслуживании 100 заявок. Построить гистограммы распределения длительности пребывания заявки в очереди к первому каналу и длины очереди ко второму каналу.

```
TAB1 TABLE QT1,5,30,12
TAB2 TABLE Q$OCH2,10,10,12
GENERATE 5,2
SPLIT 1,UST2        ; распределение заявок по каналам
UST1 QUEUE 1
SEIZE 1             ; обработка заявки на первым канале
DEPART 1
ADVANCE 15,2
RELEASE 1
TABULATE TAB1  ; статистика по времени ожидания в первой очереди
TRANSFER ,EX
UST2 QUEUE OCH2
SEIZE 2             ; обработка заявки на втором канале
DEPART OCH2
ADVANCE 17,2
RELEASE 2
TABULATE TAB2  ; сбор статистики по длине второй очереди
EX TERMINATE 1
START 100
```

По условию заявки распределяются поровну между двумя каналами. В связи с тем, что время обслуживания каждым каналом различно, следует ожидать, что количество обслуженных заявок каждым каналом будет различным. Статистика об очередях собирается автоматически при наличии в системе блоков QUEUE и DEPART. Табулирование значений СЧА осуществляется с помощью оператора TABLE и блока TABULATE.

Формат записи первого оператора TABLE:

TAB1 TABLE QT1, 5, 30, 12

В поле метки находится имя оператора TAB1, в поле <A> задан стандартный числовой атрибут QT1 – среднее время пребывания транзакта (заявки) в очереди под номером 1 (в соответствии с QUEUE 1), в поле <B> – задается верхний предел первого интервала (в примере число 5), в поле <C> задается ширина частотного интервала (в примере число 30), в поле <D> задается число частотных интервалов (12).

Формат записи второго оператора TABLE:

TAB2 TABLE Q$OCH2, 10, 10, 12

Второй оператор TABLE табулирует величину Q$OCH2 – длину очереди под именем OCH2. В стандартном отчете указываются следующие величины:

MEAN – среднее значение или оценка математического ожидания;
STD.DEV – оценка среднеквадратического отклонения;
RANGE – интервалы группирования;
FREQUENCY – количество наблюдений, попавших в каждый интервал;
CUM.% – накопленная частота в процентах.

Фрагмент стандартного отчета:

```
TABLE          MEAN      STD.DEV.       RANGE        RETRY       FREQUENCY  CUM.%
 TAB1       136.649     76.413                         0
                                  -         5.000                      1      1.89
                           5.000  -        35.000                      5     11.32
                          35.000  -        65.000                      6     22.64
                          65.000  -        95.000                      6     33.96
                          95.000  -       125.000                      6     45.28
                         125.000  -       155.000                      6     56.60
                         155.000  -       185.000                      6     67.92
                         185.000  -       215.000                      7     81.13
                         215.000  -       245.000                      6     92.45
                         245.000  -       275.000                      4    100.00
 TAB2        56.234     31.145                         0
                                  -        10.000                      3      6.38
                          10.000  -        20.000                      4     14.89
                          20.000  -        30.000                      5     25.53
                          30.000  -        40.000                      4     34.04
                          40.000  -        50.000                      5     44.68
                          50.000  -        60.000                      5     55.32
                          60.000  -        70.000                      4     63.83
                          70.000  -        80.000                      4     72.34
                          80.000  -        90.000                      5     82.98
                          90.000  -       100.000                      4     91.49
                         100.000  -       110.000                      4    100.00
```

Стандартный отчет показывает, что в интервал времени ожидания в первой очереди от 0 до 5 минут попала 1 заявка, от 5 до 35 минут – 5 заявок, от 35 до 65 – 6 заявок и т.д. В очереди длиной от 0 до 10 единиц находилось 3 заявки, от 10 до 20 – 4 заявки, от 20 до 30 – 5 заявок и т.п.

Для построения гистограммы необходимо выбрать после выполнения программы пункт меню Window/Simulation Window/Table

Window. Далее в открывшемся диалоговом окне задать имя таблицы (в данном примере TAB1 или TAB2). Результаты представлены на рис.1.1, 1.2.

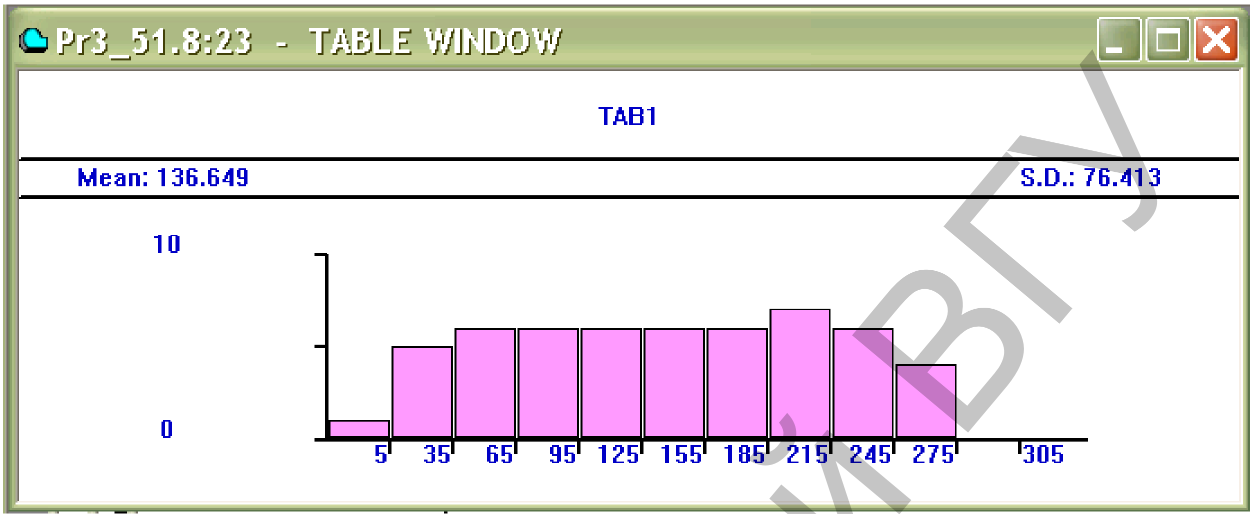


Рис.1.1. Гистограмма распределения длительности пребывания заявки в очереди к первому каналу

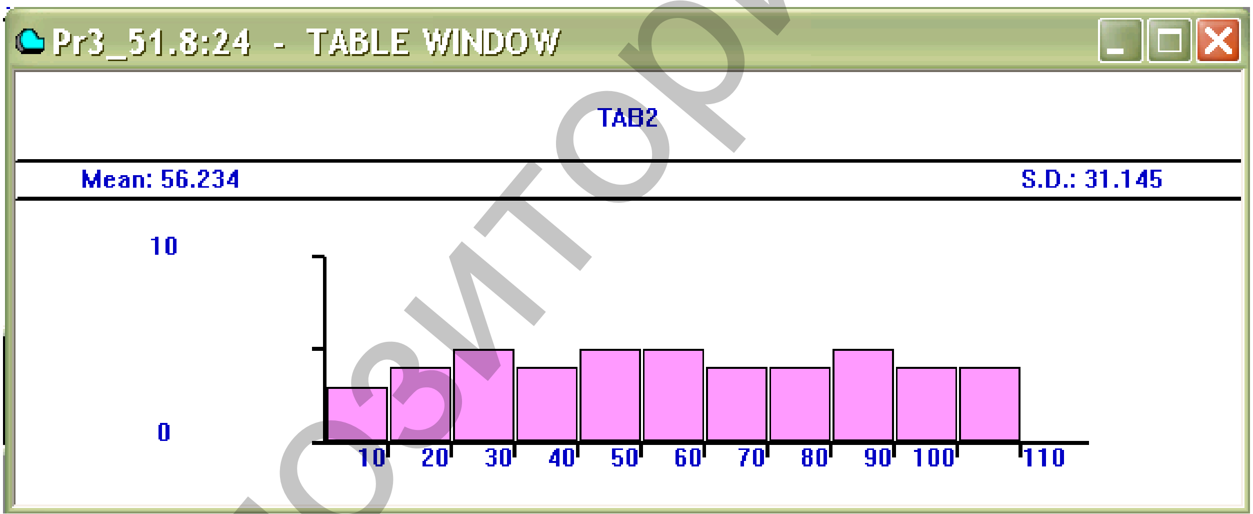


Рис.1.2. Гистограмма распределения длины очереди ко второму каналу

7.6. Моделирование последовательности значений случайных величин с заданным законом распределения

Моделирование последовательности значений случайных величин с заданным законом распределения реализуется на основе использования базовой случайной величины, имеющей равномерное распределение в интервале (0,1). В GPSS имеются встроенные датчики случайных чисел, генерирующие целые случайные числа, равномерно распределенные от 0 до 999. Для получения случайного числа путем обращения к такому генератору достаточно записать RN с номером генератора, например RN1. Полученное число необходимо привести к отрезку (0,1) делением на 1000.

Так как встроенные генераторы основаны на конгруэнтном методе получения случайных чисел, то при каждом запуске они выдают одну и ту же последовательность чисел. Команда RMULT позволяет изменить эту последовательность путем изменения базы генератора. Формат команды:

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

Операнды <A>, <B>, <C>, <D>, <E>, <F>, <G> задают начальные множители для соответствующих по номеру (1-7) генераторов случайных чисел. Например, RMULT 333,,3 устанавливает начальные состояния множителей генераторов случайных чисел с номерами 1 и 3.

В GPSS имеется три типа переменных: арифметические переменные (целые), арифметические переменные с «плавающей точкой» (действительные) и булевские переменные. Переменные определяются перед началом моделирования с помощью следующих операторов описания:

Имя VARIABLE выражение (для целой переменной);

Имя FVARIABLE выражение (для действительной переменной).

Имя BVARIABLE выражение (для булевской переменной).

Здесь имя – имя переменной, используемое для ссылок на нее, а выражение – арифметическое или логическое (для булевской переменной) выражение, определяющее переменную. Арифметическое выражение представляет собой комбинацию операндов, в качестве которых могут выступать константы, функции, знаки арифметических операций и круглых скобок. Знаком операции умножения в GPSS является символ #.

Например, генерацию 1000 значений равномерно распределенной на интервале [5, 15] случайной величины можно выполнить так (рис.1.3):

```
ZN FVARIABLE 10#(RN1/1000)+5 ; определение СВ
TAB1 TABLE V$ZN 5,1,12    ; определение таблицы для гистограммы СВ
GENERATE V$ZN       ; генерация значений СВ
TABULATE TAB1       ; табулирование таблицы значений СВ
TERMINATE 1
START 1000
```

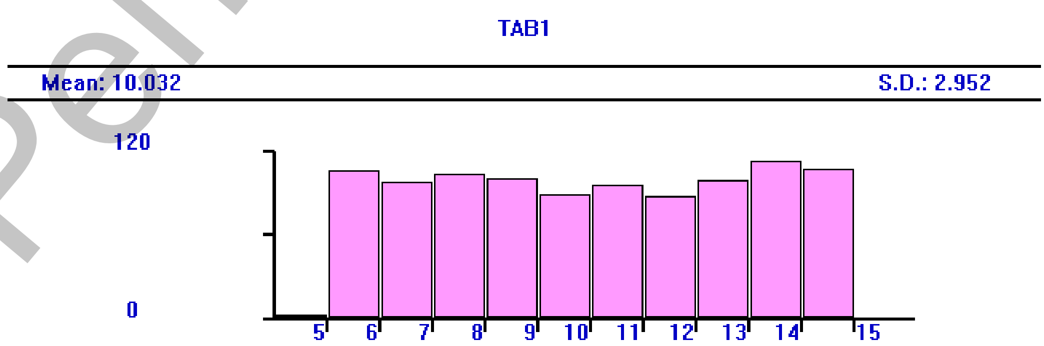


Рис.1.3. Гистограмма равномерно распределенной СВ

Для получения значений случайной величины Y с экспоненциальным (рис.1.4) законом можно воспользоваться соотношением $y = -\frac{1}{\lambda}\ln\gamma_i$, полученным на основе метода обратной функции. Для $\lambda = 1$:

```
ZN FVARIABLE (-1)#LOG((1+RN1)/1000)
TAB1 TABLE V$ZN 0.5,0.5,12
GENERATE V$ZN
TABULATE TAB1
TERMINATE 1
START 1000
```

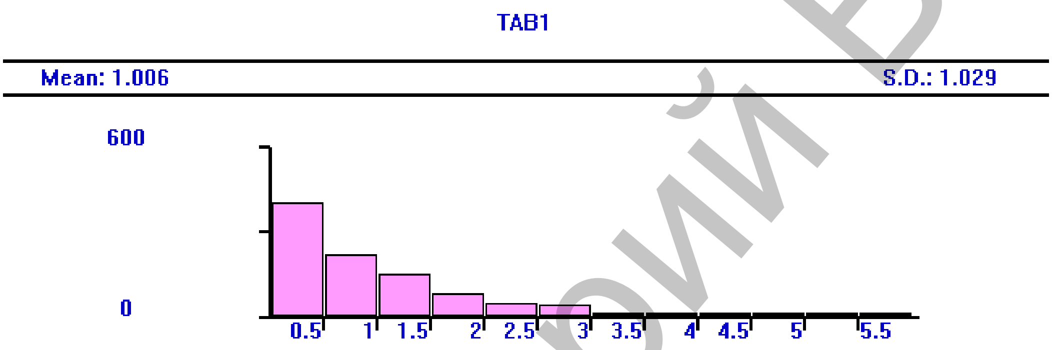


Рис.1.4. Гистограмма СВ с экспоненциальным законом распределения

Такой способ является достаточно трудоемким, так как требует обращения к математическим функциям, вычисление которых требует десятков машинных операций. Другим возможным способом является использование вычислительных объектов GPSS типа функция.

Функции используются для вычисления величин, заданных табличными зависимостями. Существует пять типов функций (n – количество точек, задающих функцию): 1) непрерывные числовые значения Cn; 2) дискретные числовые значения Dn; 3) перечень числовых значений Ln; 4) дискретные значения атрибутов En; 5) перечень значений атрибутов Mn.

Первые два типа функции являются основными.

Каждая функция определяется перед началом моделирования с помощью оператора определения FUNCTION, имеющего следующий формат:

Имя FUNCTION A,B

Здесь имя – имя функции, используемое для ссылок на нее; A – стандартный числовой атрибут, являющийся аргументом функции. При использовании непрерывной функции для генерирования случайных чисел ее аргументом должен быть один из генераторов случайных чисел RNj.; B – тип функции и число точек таблицы, определяющей функцию.

Тип непрерывной числовой функции кодируется буквой C. Так, например, в определении непрерывной числовой функции, таблица которой содержит 24 точки, поле <B> должно иметь значение C24.

Особенностью использования встроенных генераторов случайных чисел RNj в качестве аргументов функций является то, что их значения в этом контексте интерпретируются как дробные числа от 0 до 0,999999.

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

При использовании функции в поле <B> блоков GENERATE и ADVANCE вычисление интервала поступления или времени задержки производится путем умножения операнда <A> на вычисленное значение функции. Отсюда следует, что функция, используемая для генерирования случайных чисел с показательным (экспоненциальным) распределением $\lambda = 1$, должна описывать зависимость *y = – ln (1-x),* представленную в табличном виде.

Использование функций для получения случайных чисел с заданным законом распределения дает хотя и менее точный результат за счет погрешностей аппроксимации, но зато с меньшими вычислительными затратами (несколько машинных операций на выполнение линейной интерполяции). По сути, этот вариант реализации получения последовательности значений случайных величин соответствует методу ступенчатой аппроксимации функции плотности распределения вероятностей.

Функции всех типов имеют единственный СЧА с названием FN, значением которого является вычисленное значение функции. Вычисление функции производится при входе транзакта в блок, содержащий ссылку на СЧА FN с именем функции.

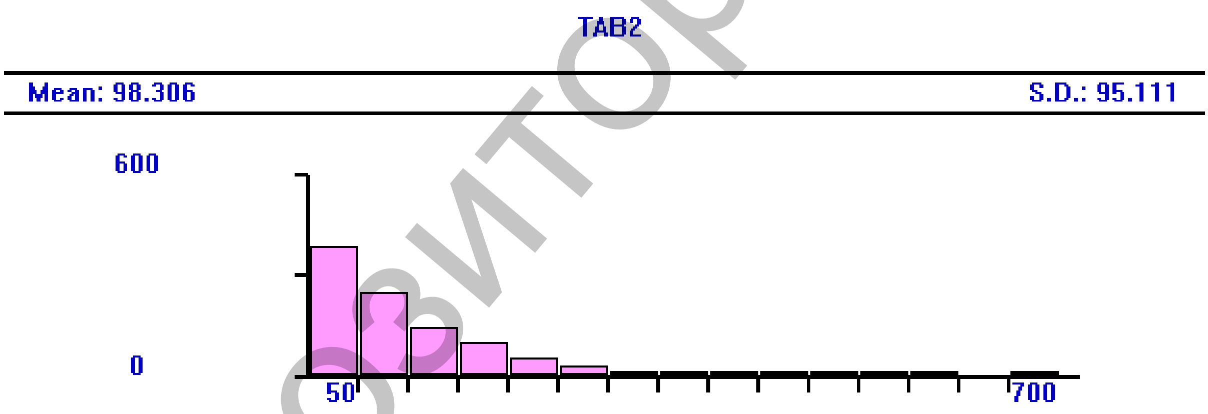
Пример 12. Выполнить моделирование случайных величин со следующими параметрами:

1) экспоненциальное распределение для λ=1;
2) экспоненциальное распределение для λ=0,01;
3) стандартное нормальное распределение со средним, равным 0 и стандартным квадратичным отклонением СКО=1;
4) нормальное распределение со средним, равным 30 и СКО=4;
5) дискретное распределение такое, чтобы СВ получала значения 1, 4, 5 с относительной частотой 0,40; 0,10; 0,50.

Решение заданий 1 и 2:

```
EXPON FUNCTION RN1,C24          ; задаем функцию
0,0/.1,.104/.2,.222/.3,.355/.4,.509/.5,.69/.6,.915
.7,1.2/.75,1.38/.8,1.6/.84,1.85/.88,2.12/.9,2.3
.92,2.52/.94,2.81/.95,2.99/.96,3.2/.97,3.5/.98,3.9
.99,4.6/.995,5.3/.998,6.2/.999,7/.9998,8
TAB1 TABLE FN$EXPON 0.5,0.5,12 ; таблица для СВ с распределением (1)
TAB2 TABLE V$EXPV 50,50,15      ; таблица для СВ с распределением (2)
EXPV FVARIABLE 100#FN$EXPON
GENERATE V$EXPV
* для генерации транзактов можно использовать вместо предыдущих двух строк:
* GENERATE 100,FN$EXPON
TABULATE TAB1
TABULATE TAB2
TERMINATE 1
START 1000
```

На рисунке 1.5 показано распределение случайной величины для интенсивности потока событий λ=0,01. Операнд <A> получен путем преобразования интенсивности в соответствующее ей среднее время между последовательными поступлениями заявок: 1 заявка в 100 единиц



времени.

Рис.1.5. Гистограмма СВ с экспоненциальным законом распределения, λ=0,01

Решение заданий 3 и 4: для того, чтобы с помощью таблицы задать нормальное распределение случайной величины, используется 25 точек для обеспечения достаточной точности аппроксимации. Данная таблица задает случайную величину Z с математическим ожиданием равным 0, и стандартным квадратичным отклонением равным 1. Для моделирования нормальной случайной величины X с другими значениями математического ожидания и СКО необходимо произвести вычисления по формуле:

$$X = m_x + \sigma_x Z ,$$

где m_x – математическое ожидание СВ, σ_x – СКО.

```
NORM FUNCTION RN2,C25
0,-5/.00003,-4/.00135,-3/.00621,-2.5/.02275,-2
.06681,-1.5/.11507,-1.2/.15866,-1/.21186,-.8/.27425,-.6
.34458,-.4/.42074,-.2/.5,0/.57926,.2/.65542,.4
.72575,.6/.78814,.8/.84134,1/.88493,1.2/.93319,1.5
.97725,2/.99379,2.5/.99865,3/.99997,4/1,5

TAB3 TABLE FN$NORM 0,0.5,10   ; таблица для СВ с распределением (3)
TAB4 TABLE V$NORMV 20,1,25    ; таблица для СВ с распределением (4)

NORMV FVARIABLE 30+4#FN$NORM
GENERATE V$NORMV
* или GENERATE (30+4#FN$NORM)      ; другой способ получения СВ
TABULATE TAB3
TABULATE TAB4
TERMINATE 1
START 1000
```

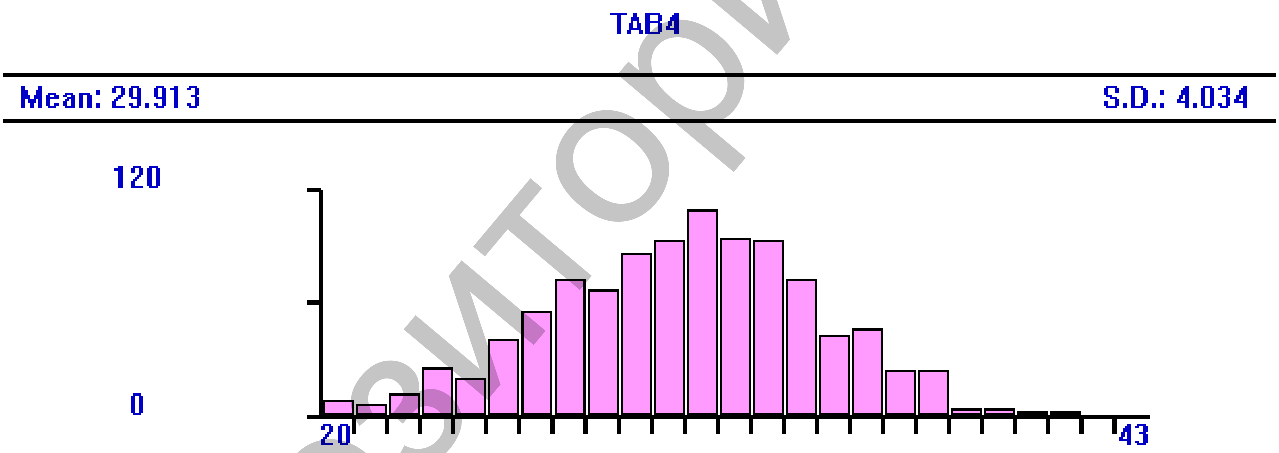


Рис.1.6. Гистограмма СВ с нормальным распределением, М=30, СКО=4

Решение задания 5:

```
DSV FUNCTION RN3,D3
0.4,1/.5,4/1,5
TAB5 TABLE FN$DSV,0,1,7
GENERATE FN$DSV
TABULATE TAB5
TERMINATE 1
START 1000
```

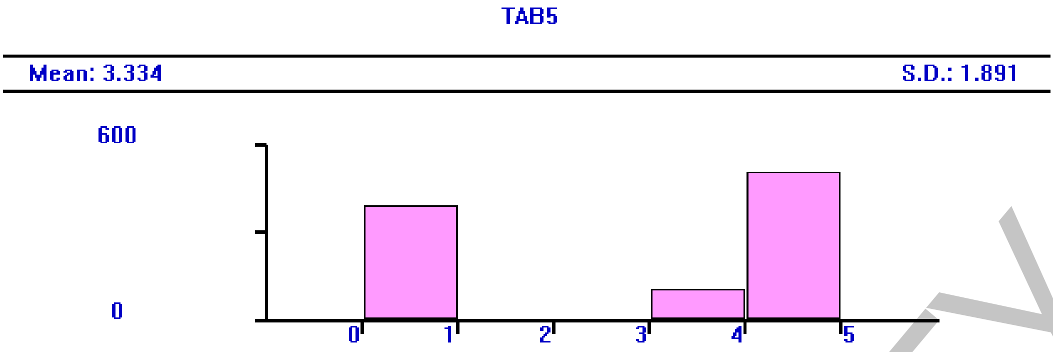


Рис.1.7. Гистограмма СВ с дискретным распределением

Практическое задание № 2. Тема: моделирование случайных величин с заданным законом распределения средствами GPSS

Задание.

Заявки поступают в СМО равномерно через 10 ± 2 единицы времени. Поток обработки заявок является пуассоновским с интенсивностью λ=0,1.

Построить частотное распределение, получить оценки математического ожидания и среднеквадратического отклонения для <u>времени пребывания заявки в модели</u> (для решения задачи можно использовать стандартный числовой атрибут M1, который связан с каждым транзактом и хранит время пребывания транзакта в модели). Сравнить полученные значения с теоретическими значениями для заданного значения λ.

Решить задачу двумя способами: моделируя время обработки заявки на основе метода обратной функции и метода ступенчатой аппроксимации.

Вычислить ошибки оценивания математического ожидания и дисперсии при использовании разных методов моделирования.

7.7. Контрольные вопросы по теме

1. В чем заключается достоинство имитационного моделирования как метода исследования сложных систем?
2. В чем преимущество использования языка имитационного моделирования?
3. Что представляют собой объекты GPSS-модели?
4. Что такое транзакт? Какова его структура и функции?
5. Какие блоки языка GPSS предназначены для создания и уничтожения транзактов?
6. Какие блоки языка GPSS осуществляют модификацию параметров транзактов?

7. Каким образом можно получить поток заявок в GPSS-модели?
8. Из каких элементов состоит стандартный отчет?
9. Какие средства языка GPSS используются для имитации одноканальных устройств?
10. Какие блоки языка GPSS предназначены для управления устройствами?
11. Для чего предназначены таблицы в языке GPSS?
12. Какую информацию о таблицах можно получить из отчета?
13. Какие блоки языка GPSS предназначены для управления накопителями?
14. При каких условиях на входе прибора или МКУ образуется очередь? Чем определяется факт наличия очереди?
15. Чем отличается захват прибора от занятия прибора?
16. Как моделируются параллельно работающие приборы?
17. Какие средства GPSS используется для моделирования МКУ?
18. В чем заключается сходство и различие между прибором и МКУ?
19. Как реализуется передача управления (изменение последовательности передвижения транзактов в модели) в GPSS-модели?
20. Каким образом в языке GPSS можно управлять ключами?
21. Какие блоки языка GPSS входят в группу синхронизации транзактов?
22. Какая случайная величина называется базовой?
23. Какие существуют распределения случайных величин?
24. Как определяются в языке GPSS функции для разыгрывания значений дискретной и непрерывной случайной величины?
25. Какие средства языка GPSS используются для имитации экспоненциально или нормально распределенных случайных величин?
26. Какие статистические объекты используются в GPSS?
27. Какие блоки языка GPSS предназначены для получения статистических результатов моделирования системы?
28. Каким образом можно строить гистограммы в языке GPSS?
29. Какие средства имеются в языке GPSS для организации экспериментов с моделью?
30. Каковы особенности имитационного эксперимента на ЭВМ с точки зрения обработки результатов?

Лабораторная работа № 9. Тема: моделирование систем массового обслуживания с использованием языка GPSS

Задание

Необходимо промоделировать работу участка цеха, обрабатывающего два потока деталей различного типа и состоящего из трех станков, которые выполняют 6 операций. Операции 1, 2, 3 выполняются для деталей первого типа, операции 4, 5, 6 – для деталей второго типа. В таблице 1 представлено распределение выполняемых операций по станкам А1, А2 и А3. Интервалы времени между поступлениями деталей и время выполнения операций распределены равномерно. Информация о временах поступления и выполнения операций заданы в таб.2 и таб.3.

Определить для рабочего дня (8 часов) и рабочей недели (5 дней при односменном режиме) среднюю загрузку каждого станка, среднее время обработки деталей каждого типа, длину очередей на обработку для каждого станка, размер склада для деталей каждого вида.

Проанализировать эффективность работы участка цеха и предложить способы его модификации с целью улучшения.

Порядок выполнения работы

1. Ознакомьтесь с текстом программы на GPSS для примера.
2. Внесите в программу необходимые изменения в соответствии с вашим вариантом.
3. Продумайте меры по улучшению выходных параметров участка цеха (повышение производительности, улучшение использования оборудования).
4. Выполните моделирование. Составьте отчет.

Содержание отчета:

- текст исходной программы по номеру варианта;
- результаты моделирования для рабочего дня и рабочей недели, их анализ;
- предлагаемые меры по улучшению работы участка цеха;
- текст программы с внесенными изменениями;
- результаты моделирования;
- объяснение влияния изменений на результаты моделирования.

Таблица 1. Распределение операций по станкам

Вариант	Операция 1	Операция 2	Операция 3	Операция 4	Операция 5	Операция 6
Пример	А1	А2	А3	А1	А3	А2
1	А1	А2	А3	А3	А2	А1
2	А1	А2	А3	А3	А1	А2
3	А1	А2	А3	А1	А2	А3
4	А1	А2	А3	А2	А1	А3
5	А1	А2	А3	А2	А3	А1
6	А2	А1	А3	А1	А2	А3
7	А2	А1	А3	А1	А3	А2
8	А2	А1	А3	А2	А1	А3
9	А2	А1	А3	А2	А3	А1
10	А2	А1	А3	А3	А1	А2

11	A2	A1	A3	A3	A2	A3
12	A3	A1	A2	A1	A2	A3
13	A3	A1	A2	A1	A2	A3
14	A3	A1	A2	A1	A3	A2
15	A3	A1	A2	A2	A1	A3
16	A3	A1	A2	A2	A3	A1

Таблица 2.

Вариант	Интервалы времени поступления деталей первого типа (мин.)	Интервалы времени поступления деталей второго типа (мин.)
Пример	30 + 5	20 + 5
1	25 + 4	25 + 6
2	20 + 3	30 + 7
3	15 + 5	35 + 8
4	10 + 4	20 + 5
5	30 + 5	10 + 3
6	15 + 4	15 + 6
7	30 + 10	15 + 3
8	20 + 5	20 + 5
9	25 + 4	10 + 3
10	45 + 5	15 + 5
11	20 + 4	15 + 3
12	10 + 3	15 + 5
13	20 + 3	30 + 7
14	15 + 5	35 + 8
15	10 + 4	20 + 5
16	30 + 5	10 + 3

Таблица 3.

Вариант	Интервал времени выполнения операции 1 (мин.)	Интервал времени выполнения операции 2 (мин.)	Интервал времени выполнения операции 3 (мин.)	Интервал времени выполнения операции 4 (мин.)	Интервал времени выполнения операции 5 (мин.)	Интервал времени выполнения операции 6 (мин.)
Пример	5 + 2	20 + 4	10 + 3	7 + 3	15 + 5	15 + 5
1	20 + 4	5 + 2	15 + 5	15 + 5	7 + 3	10 + 3
2	10 + 3	15 + 5	5 + 2	20 + 4	10 + 3	7 + 3
3	18 + 3	10 + 3	12 + 5	20 + 4	25 + 8	12 + 4
4	12 + 5	15 + 5	18 + 3	10 + 3	5 + 2	20 + 4
5	15 + 5	20 + 4	10 + 3	18 + 3	12 + 5	20 + 4
6	10 + 3	25 + 8	5 + 2	15 + 5	18 + 3	15 + 5
7	15 + 5	12 + 5	20 + 4	5 + 2	10 + 3	18 + 3

8	20 + 4	18 + 3	10 + 3	7 + 3	15 + 5	25 + 8
9	10 + 3	15 + 5	10 + 3	12 + 5	5 + 2	20 + 4
10	25 + 8	5 + 2	12 + 5	7 + 3	10 + 3	15 + 5
11	20 + 4	10 + 3	15 + 5	5 + 2	12 + 5	25 + 8
12	12 + 5	20 + 4	25 + 8	15 + 5	5 + 2	10 + 3
13	15 + 5	20 + 4	10 + 3	18 + 3	12 + 5	20 + 4
14	10 + 3	25 + 8	5 + 2	15 + 5	18 + 3	15 + 5
15	15 + 5	12 + 5	20 + 4	5 + 2	10 + 3	18 + 3
16	20 + 4	18 + 3	10 + 3	7 + 3	15 + 5	25 + 8

Пример

Исходные данные для структуры участка цеха и интервалы времени поступления деталей на станок, выполняющий первую операцию (для деталей первого типа) и на станок, выполняющий четвертую операцию (для деталей второго типа), а также интервалы времени обработки каждым станком заданы в таблицах 1, 2 и 3.

Таблица 4. Таблица определений для примера	
Элементы GPSS	Назначение
Транзакты: 1-й сегмент модели 2-й сегмент модели 3-й сегмент модели	 Детали первого типа Детали второго типа Таймер
Станки: stan1 stan2 stan3	операции 1 и операции 4 операции 2 и операции 6 операции 3 и операции 5
Очереди: och1 och2 och3	общая очередь к станку А1 общая очередь к станку А2 общая очередь к станку А3

Единица времени в модели - 1 мин.

Текст программы на GPSS

```
GENERATE 30,5 ; первый сегмент модели
QUEUE och1
SEIZE stan1
DEPART och1
ADVANCE 5,2
RELEASE stan1
QUEUE och2
SEIZE stan2
```

```
DEPART och2
ADVANCE 20,4
RELEASE stan2
QUEUE och3
SEIZE stan3
DEPART och3
ADVANCE 10,3
RELEASE stan3
TERMINATE

GENERATE 20,5 ; второй сегмент модели
QUEUE och1
SEIZE stan1
DEPART och1
ADVANCE 7,3
RELEASE stan1
QUEUE och3
SEIZE stan3
DEPART och3
ADVANCE 15,5
RELEASE stan3
QUEUE och2
SEIZE stan2
DEPART och2
ADVANCE 15,5
RELEASE stan2
TERMINATE

GENERATE 480  ; третий сегмент модели – таймер
TERMINATE 1
START 1
```

В данном примере таймер настроен на выполнение моделирования в течение 8 часового рабочего дня. Для выполнения моделирования в течение 5 дней таймер должен быть откорректирован.

Выходные данные

Распечатка выходных данных для моделирования работы участка цеха в течение рабочего дня (8 часов).

```
                GPSS World Simulation Report - Проверка лаб7.15.1
                     Monday, December 12, 2011 12:46:59

             START TIME           END TIME  BLOCKS  FACILITIES  STORAGES
```

0.000	480.000	36	3	0

NAME	VALUE
OCH1	10000.000
OCH2	10004.000
OCH3	10002.000
STAN1	10001.000
STAN2	10005.000
STAN3	10003.000

LABEL	LOC	BLOCK TYPE	ENTRY COUNT	CURRENT COUNT	RETRY
	1	GENERATE	15	0	0
	2	QUEUE	15	0	0
	3	SEIZE	15	0	0
	4	DEPART	15	0	0
	5	ADVANCE	15	0	0
	6	RELEASE	15	0	0
	7	QUEUE	15	4	0
	8	SEIZE	11	0	0
	9	DEPART	11	0	0
	10	ADVANCE	11	1	0
	11	RELEASE	10	0	0
	12	QUEUE	10	1	0
	13	SEIZE	9	0	0
	14	DEPART	9	0	0
	15	ADVANCE	9	0	0
	16	RELEASE	9	0	0
	17	TERMINATE	9	0	0
	18	GENERATE	25	0	0
	19	QUEUE	25	0	0
	20	SEIZE	25	0	0
	21	DEPART	25	0	0
	22	ADVANCE	25	0	0
	23	RELEASE	25	0	0
	24	QUEUE	25	2	0
	25	SEIZE	23	0	0
	26	DEPART	23	0	0
	27	ADVANCE	23	1	0
	28	RELEASE	22	0	0
	29	QUEUE	22	7	0
	30	SEIZE	15	0	0
	31	DEPART	15	0	0
	32	ADVANCE	15	0	0
	33	RELEASE	15	0	0
	34	TERMINATE	15	0	0
	35	GENERATE	1	0	0
	36	TERMINATE	1	0	0

FACILITY	ENTRIES	UTIL.	AVE. TIME	AVAIL.	OWNER	PEND	INTER	RETRY	DELAY
STAN1	40	0.530	6.355	1	0	0	0	0	0
STAN3	32	0.913	13.702	1	38	0	0	0	3
STAN2	26	0.924	17.057	1	29	0	0	0	11

QUEUE	MAX	CONT.	ENTRY	ENTRY(0)	AVE.CONT.	AVE.TIME	AVE.(-0)	RETRY
OCH1	1	0	40	28	0.082	0.988	3.293	0

```
OCH3              4    3     35     2     2.169     29.741     31.543   0
OCH2             11   11     37     1     4.189     54.342     55.851   0
```

Обсуждение результатов моделирования

Таблица 5. Средняя загрузка станков (в %)		
Станок	в течение 8 часов	в течение 5 рабочих дней
STAN 1	53,0	52
STAN 2	92,4	99,8
STAN 3	91,3	97,7

Таблица 6. Максимальная длина очередей к станкам		
Станок	в течение 8 часов	в течение 5 рабочих дней
OCH 1	1	1
OCH 2	11	588
OCH 3	4	8

Таблица 7. Среднее время обработки деталей на станках (в мин.)		
Станок	в течение 8 часов	в течение 5 рабочих дней
STAN 1	6,35	6,25
STAN 2	13,70	16,96
STAN 3	17,06	13,41

Общее число обработанных деталей в течение 8 часов равно 40, в течение рабочей недели – 2002. Эти данные могут служить основанием для расчета необходимого размера склада готовой продукции. Из результатов моделирования можно сделать вывод, что первый станок А1 загружен приблизительно на 50%, т.е. недогружен. Перегружен станок А2 (об этом говорит средний процент использования и длина очереди). Станок А3 загружен оптимально (коэффициент загрузки около 100%, очередь невелика).

Для повышения эффективности работы участка цеха при заданном потоке деталей можно использовать два станка А2. Для проверки данного предположения надо внести в программу модели изменения: определить многоканальное устройство и заменить в программе блоки SEIZE-RELEASE на ENTER-LEAVE.

По результатам данного варианта моделирования можно сделать вывод: использование двух станков А2 позволило ликвидировать очередь к данному станку, снизив среднюю загрузку станков А2 до 67% за пять часов работы (чего и следовало ожидать), очереди ко всем устройствам стали небольшими.

Однако при увеличении срока моделирования до 5 дней перераспределилась нагрузка на станок А3: при средней нагрузке в 98% максимальная очередь на обработку станком А3 возросла до 15. Следовательно, надо разгружать третий станок. Можно попробовать скорректировать время обработки детали на третьем станке, чтобы количество деталей в очереди уменьшилось. Таким же образом и далее можно продолжить анализ результатов моделирования и внесение исправлений в модель с целью определения оптимальной структуры участка цеха при заданном потоке деталей.

Если структуру цеха менять нельзя, то можно подобрать такие параметры потока деталей, которые давали бы возможность оптимально загружать данное оборудование.

Список рекомендуемой литературы

1. Айвазян С.А., Енюков И.С., Мешалкин Л.Д. Прикладная статистика: Основы моделирования и первичная обработка данных. – М.: «Финансы и статистика», 1983. – 471 с.
2. Боровиков В.П., Боровиков И.П. Statistica. Статистический анализ и обработка данных в среде Windows. – М.: Информационно-издательский дом "Филинъ", 1997.- 608 с.
3. Бусленко Н.П. Моделирование сложных систем. – М.: Наука, 1988.
4. Вентцель Е.С. Исследование операций. Задачи, принципы, методология. – М.: Дрофа, 2004.
5. Кузнецов А.В., Сакович В.А., Холод Н.И. Математическое программирование. – Мн.: Вышэйшая школа, 1994.
6. Лобач В.И., Кирлица В.П., Малюгин В.И., Сталевская С.Н. Имитационное и статистическое моделирование: Практикум для студентов мат. и экон. спец. – Мн.: БГУ, 2004. – 189 с.
7. Руководство по GPSS/PC. Minuteman Software, перевод с английского под ред. Якимова И. М. Казань, 1997 г. – 320 с.
8. Советов Б. Я., Яковлев С. А. Моделирование систем. Учебник для ВУЗов. М.: Высшая школа, 2007 г. – 343 с.
9. Советов Б.Я., Яковлев С.А. Моделирование систем. Практикум: Учеб. пособие для вузов по спец. "Автоматизир. системы обработки информ. и упр." – М.: Высш. шк., 1999. – 224с.
10. Советов Б.Я., Яковлев С.А. Моделирование систем. Учеб. для вузов. – М.: Высш. шк., 2001. – 343с.
11. Таха, Х. Введение в исследование операций. – Изд. дом «Вильямс», 2005.
12. Томашевский В., Жданова Е. Имитационное моделирование в среде GPSS. Серия «Факультет». М.: Бестселлер, 2003. – 416 с.
13. Тюрин Ю.П., Макаров А.А. Анализ данных на компьютере. – М.: ИНФРА-М, Финансы и статистика, 1995.– 384 с.
14. Хахулин Г.Ф. Основы конструирования имитационных моделей: учеб. пособие. – М.: НТК Поток, 2002. – 222 с.
15. Черненький В. М. Разработка САПР. Книга 9 – Имитационное моделирование: Практическое пособие. / Под редакцией Петрова А. В. / М.: Высшая школа, 1990 г. – 112 с.
16. Шеннон Р. Дж. Имитационное моделирование систем – искусство и наука. М.: Мир, 1978 г. – 418 с.
17. Шикин Е.В., Шикина Г.Е. Исследование операций: учеб. – М.: ТК Велби, Изд-во Проспект, 2006.
18. Шрайбер Т.Дж. Моделирование на GPSS. – М.: Машиностроение, 1980. – 593 с.
19. http://www.gpss.ru. Сайт, посвященный системе моделирования GPSS.
20. http://www.statsoft.ru. Статистический портал, содержащий информацию о пакете STATISTICA.

Приложение 1

Блоки GPSS

Оператор	*Поля*	*Описание*
Блоки для работы с транзактами		
GENERATE	A,B,C, D,E	Создание транзактов. А – среднее время между транзактами; В – половина диапазона или модификатор функции; С – начальная задержка; D – число транзактов, которое должно быть создано; E – приоритет транзакта
TERMINATE	A	Уничтожение транзакта. А – на сколько уменьшается счетчик, определяющий число проходов
SPLIT	A,B,C	Создание копий транзакта. А – число создаваемых копий; В – блок куда направляются копии; С – номер параметра транзакта, в котором производится нумерация получаемых транзактов
ASSEMBLE	A	Объединение транзактов, являющихся членами одного ансамбля А – число объединяемых транзактов, далее из блока выходит 1 транзакт
GATHER	A	Аналогично SPLIT, только из блока выходят все транзакты
MATCH	A	Синхронизация двух транзактов одного ансамбля. А – метка сопряженного блока, блок применяется парами
ADVANCE	A,B	Задержка транзакта. А – среднее время задержки; В – половина диапазона или модификатор функции
ASSIGN	A,B,C	Изменение параметра транзакта. А – номер параметра или имя, В – значение, С – функция
PRIORITY	A,B	Установить приоритет. А – новое значение приоритета
Блоки, изменяющие маршруты движения транзактов		
GATE X	A,B	Выбор направления в зависимости от состояния устройств Условие X: NU (прибор свободен), U (прибор занят), NI (прибор не прерван), I (прибор прерван), FV (прибор доступен), FNV (прибор не доступен), SE (накопитель пуст), SNE (накопитель не пуст), SF (накопитель полон), SNF (накопитель не полон), SV (накопитель доступен), SNV (накопитель не доступен), LS (логический ключ включен), LR (логический ключ выключен).
TEST X	A,B,C	Сравнение двух операндов и выбор направления движения Условие X: E =, G >, GE >=, L <, LE <=, NE !=; A,B – операнды, С – альтернативный путь при невыполнении условия (если отсутствует, то транзакт блокируется).
TRANSFER	A,B,C, D	Передача транзактов (логически, статистически, условно, безусловно). А – определяет режим: BOTH (один альтернативный адрес); PICK (случайным образом); ALL (много альтернатив, В – первый адрес, С – последний адрес, D содержит индексную константу); если А пустое, то безусловная передача в В, дробь в А задает вероятность выбора
Блоки для работы с приборами		
SEIZE	A	Занять прибор. А – имя или номер прибора

<table>
<tr><th>Оператор</th><th>Поля</th><th>Описание</th></tr>
<tr><td>RELEASE</td><td>A</td><td>Освободить прибор. А – имя или номер прибора</td></tr>
<tr><td>PREEMPT</td><td>A,B,C, D,E</td><td>Захватить прибор. А – имя или номер прибора; если в В находится PR, то захват в режиме приоритета; С определяет блок, куда передается прерванный транзакт, при этом прерванный транзакт продолжает претендовать на данное устройство; D определяет параметр; если в Е находится RE, то прерванный транзакт не претендует на устройство</td></tr>
<tr><td>RETURN</td><td>A</td><td>Окончание захвата. А – имя или номер прибора.</td></tr>
<tr><td>FUNAVAIL</td><td>A,B,C, D,E,F, G,H</td><td>Прибор недоступен. А – имя или номер прибора</td></tr>
<tr><td>FAVAIL</td><td>A</td><td>Прибор доступен. А – имя или номер прибора</td></tr>
<tr><td colspan="3" align="center">Блоки для работы с очередями</td></tr>
<tr><td>QUEUE</td><td>A,B</td><td>Занять очередь. А – имя очереди (номер), В – число занимаемых мест (по умолчанию 1)</td></tr>
<tr><td>DEPART</td><td>A,B</td><td>Освободить очередь. А – имя очереди (номер), В – число освобождаемых мест (по умолчанию 1)</td></tr>
<tr><td colspan="3" align="center">Блоки для работы с накопителями (многоканальными устройствами)</td></tr>
<tr><td>ENTER</td><td>A,B</td><td>Занять накопитель. А – имя накопителя (номер), В – число занимаемых ячеек (по умолчанию 1)</td></tr>
<tr><td>LEAVE</td><td>A,B</td><td>Освободить накопитель. А – имя накопителя (номер), В – число освобождаемых ячеек (по умолчанию 1)</td></tr>
<tr><td>SUNAVAIL</td><td>A</td><td>Накопитель недоступен. А – имя накопителя (номер)</td></tr>
<tr><td>SAVAIL</td><td>A</td><td>Накопитель доступен. А – имя накопителя (номер)</td></tr>
<tr><td colspan="3" align="center">Блок для сбора статистических данных</td></tr>
<tr><td>TABULATE</td><td>A,B</td><td>Сбор статистики. А – имя таблицы (номер), В – число единиц, добавляемых к интервалу наблюдения</td></tr>
<tr><td colspan="3" align="center">Блоки для изменения состояния логических ключей, матриц, величин</td></tr>
<tr><td>LOGIC X</td><td>A</td><td>Изменение состояния логического ключа. А – имя ключа. Условие X: S – включен, R – выключен, I – инвертирован</td></tr>
<tr><td>SAVEVALUE</td><td>A,B</td><td>А – имя сохраняемой величины, В – значение</td></tr>
<tr><td>MSAVEVALUE</td><td>A,B,C,D</td><td>А – имя матрицы, В –строка, С – столбец, D – значение</td></tr>
<tr><td colspan="3" align="center">Операторы описания данных</td></tr>
<tr><td>Имя VARIABLE</td><td>A</td><td>Определить целую переменную, А – выражение-инициализатор</td></tr>
<tr><td>Имя FVARIABLE</td><td>A</td><td>Определить вещественную переменную, А – выражение-инициализатор</td></tr>
<tr><td>Имя BVARIABLE</td><td>A</td><td>Определить булеву переменную, А – выражение-инициализатор</td></tr>
<tr><td>Имя MATRIX</td><td>A,B,C</td><td>Определить матрицу, А – не используется, В – количество строк, С – количество столбцов.</td></tr>
<tr><td>INITIAL</td><td>A,B</td><td>Инициализировать. А – инициализируемый элемент, В – выражение-инициализатор</td></tr>
<tr><td>Имя STORAGE</td><td>A</td><td>Определить накопитель, А – емкость накопителя</td></tr>
<tr><td>Имя TABLE</td><td>A,B,C, D</td><td>Определить таблицу (гистограмму), А – табулируемое значение, В – нижний предел, С – шаг, D – число интервалов</td></tr>
</table>

Оператор	*Поля*	*Описание*
Имя QTABLE	A,B,C, D	Определить таблицу (гистограмму) для очереди, табулируется автоматически время ожидания транзакта в очереди, A – имя очереди, далее аналогично с TABLE

Приложение 2

Стандартные числовые атрибуты

Обозначение	*Описание*
Общемодельные	
AC1	Значение абсолютного системного времени (время после выполнения последнего CLEAR)
C1	Значение относительного системного времени (время после выполнения последнего RESET)
TG1	Значение счетчика числа оставшихся проходов
RN	Случайное значение 0-999 генератора с заданным номером (RN1, RN2, RN3, …)
Транзакты	
P	Значение параметра
PR	Целое значение определяет приоритет транзакта
M1	Время пребывания транзакта в модели
MP	Время с момента входа в блок
Приборы	
F	Состояние прибора (1 – занят, 0 – свободен)
FC	Счетчик числа занятий
FI	1 – если прибор захвачен, 0 – в противном случае
FR	Коэффициент использования прибора
FT	Среднее время обслуживания транзакта
FV	1 – если прибор доступен, 0 – в противном случае.
Накопители	
R	Число свободных ячеек накопителя
S	Число используемых ячеек накопителя
SA	Среднее содержимое накопителя
SC	Целое число – счетчик входов
SE	1 – если накопитель пуст, 0 – в противном случае
SF	1 – если накопитель полон, 0 – в противном случае
SR	Коэффициент использования накопителя
SM	Максимальное содержимое накопителя
ST	Среднее время нахождения транзакта в накопителе
SV	1 – если накопитель доступен, 0 – в противном случае
Очереди	
Q	Текущая длина очереди
QA	Средняя длина очереди
QC	Счетчик общего числа входов
QM	Максимальная длина очереди
QT	Среднее время пребывания транзакта в очереди
QX	Среднее время пребывания транзакта в очереди без учета нулевых входов
QZ	Число элементов, не задержанных в очереди