



**Современный
Гуманитарный
Университет**

Дистанционное образование

Рабочий учебник

Фамилия, имя, отчество _____

Факультет _____

Номер контракта _____

ОСНОВЫ АВТОМАТИЗИРОВАННЫХ ИНФОРМАЦИОННЫХ СИСТЕМ

ЮНИТА 1

**ТЕОРЕТИЧЕСКИЕ ОСНОВЫ АВТОМАТИЗИРОВАННЫХ
ИНФОРМАЦИОННЫХ СИСТЕМ**

МОСКВА 1999

Разработано В.Н. Кузубовым

Рекомендовано Министерством общего
и профессионального образования
Российской Федерации в качестве
учебного пособия для студентов
высших учебных заведений

ОСНОВЫ АВТОМАТИЗИРОВАННЫХ ИНФОРМАЦИОННЫХ СИСТЕМ

Юнита 1. Теоретические основы автоматизированных информационных систем.

Юнита 2. Автоматизированные информационные системы (АИС).

Юнита 3. Обеспечение автоматизированных информационных систем.

ЮНИТА 1

Излагаются теоретические основы автоматизированных информационных систем (АИС), которые используются на протяжении всего жизненного цикла АИС. В число рассмотренных в этом качестве дисциплин входят: математическая логика, алгебра высказываний, понятие предиката. Элементы теории множеств и реляционной алгебры, основы теории формальных языков и алгоритмов.

Кроме того детально рассматриваются вопросы, связанные с использованием в АИС теории информации и кодирования, в том числе понятия: количество информации и энтропия сообщений. Уделено внимание вопросам применения общей теории систем, системного анализа и системотехники.

В качестве обоснования важности проблем использования АИС в современных условиях подробно рассматривается концепция информационного общества и места в нем информационных технологий и автоматизированных информационных систем.

Для студентов Современного Гуманитарного Университета

Юнита соответствует профессиональной образовательной программе № 1

(С) СОВРЕМЕННЫЙ ГУМАНИТАРНЫЙ УНИВЕРСИТЕТ, 1999

ОГЛАВЛЕНИЕ

ДИДАКТИЧЕСКИЙ ПЛАН	4
ЛИТЕРАТУРА	5
ПЕРЕЧЕНЬ УМЕНИЙ	6
ТЕМАТИЧЕСКИЙ ОБЗОР	7
1. Информационные системы и современное общество	7
1.1. Информационное общество	7
1.2. Информация, данные и информационные технологии	9
1.3. Автоматизированные информационные системы	11
1.4. Содержание и структура теории информационных систем ...	13
2. Логико-математические основы автоматизированных информационных систем	14
2.1. Элементы математической логики	14
2.2. Алгебра высказываний и логические связки	16
2.3. Понятие предиката	19
2.4. Элементы теории множеств, операции над множествами	20
2.5. Элементы алгебры отношений (реляционной алгебры)	24
3. Информационные и лингвистические основы автоматизированных информационных систем	30
3.1. Символьные конструкции	30
3.2. Формальные языки и грамматики	35
3.3. Элементы аналитической теории алгоритмов	39
3.4. Измерение и передача информации	49
3.5. Основы теории кодирования информации	58
3.6. Основы теории систем и системотехники	63
ЗАДАНИЯ ДЛЯ САМОСТОЯТЕЛЬНОЙ РАБОТЫ	67
ТРЕНИНГ УМЕНИЙ	70

* Глоссарий расположен в середине учебного пособия и предназначен для самостоятельного заучивания новых понятий.

ДИДАКТИЧЕСКИЙ ПЛАН

Информационное общество. Информация, сигналы, данные. Информационные технологии и системы. Автоматизированные информационные технологии и системы. Содержание и структура теории информационных систем. Математическая логика. Алгебра высказываний. Системы логических связей. Понятие предиката. Множества и подмножества. Операции над множествами. Отношения. Операции над отношениями. Реляционная алгебра. Символьные конструкции. Формальные языки и грамматики. Понятие алгоритма. Формальное определение алгоритма. Структурные и семантические схемы алгоритмов. Количество информации и энтропия сообщений. Передача информации. Кодирование и коды. Общая теория систем. Методы теории систем. Иерархические системы. Концептуализация. Формализация. Координация. Системный анализ. Системотехника.

ЛИТЕРАТУРА

Базовая

1. Мамиконов А.Г. Проектирование АСУ: Учебник. М., 1987.

Дополнительная

2. Биркгоф Г., Барти Т. Современная прикладная алгебра. М., 1976.
3. Макетирование. Проектирование и реализация диалоговых информационных систем. М., 1993.
4. Седегов Р.С., Орлова Н.М., Сидоров Ю.И. Оптимизация информационно-экономической системы предприятия. М., 1988.

ПЕРЕЧЕНЬ УМЕНИЙ

№ п/п	Умение	Алгоритмы
1	Расчет общего числа сообщений и количества информации при заданном наборе символов	<ol style="list-style-type: none"> 1. Найти количество символов в сообщении 2. Составить комбинаторную формулу 3. Подсчитать общее число сообщений 4. Вычислить количество информации
2	Определение максимальной возможности энтропии сообщения, состоящего из заданного числа символов при фиксированном алфавите	<ol style="list-style-type: none"> 1. Определить число возможных состояний системы сообщений 2. Записать выражение для подсчета энтропии 3. Преобразовать выражение к виду, удобному для расчета и вычислить
3	Определение свертки двух отношений (свертки де Моргана)	<ol style="list-style-type: none"> 1. Отобрать по одному кортежу из 1-го и 2-го отношения, имеющих совпадающие элементы. Последний элемент 1-го кортежа равен первому элементу второго 2. Отбросить совпадающие элементы, а оставшиеся объединить в один кортеж 3. Продолжить эту операцию пока не будут исчерпаны все кортежи 1-го отношения 4. Собрать все полученные кортежи в отношение, которое и будет сверткой

ТЕМАТИЧЕСКИЙ ОБЗОР*

1. ИНФОРМАЦИОННЫЕ СИСТЕМЫ И СОВРЕМЕННОЕ ОБЩЕСТВО

1.1. Информационное общество

Представление о новом “революционном” прорыве человеческого общества в области технологий, как о переходе к “постиндустриальному, информационному обществу” появилось в работах американских авторов в 80-х годах в результате анализа статистических данных о значительных изменениях в структуре занятого населения США. В частности к середине 80-х годов в США сложилась следующая структура занятости активного населения:

- около 4% активной рабочей силы было непосредственно занято в сельском хозяйстве;
- около 14% составляли собственно рабочие;
- более 60% активной рабочей силы составляли работники умственного труда – “информационные работники”).

Изменения в сфере производства, которые происходят в настоящее время, во многом сравнимы с изменениями, которые происходили в промышленности на первых этапах индустриализации около двух столетий назад.

Действительно, до начала прошлого века основная масса людей была занята в сфере сельскохозяйственного производства. В процессе индустриализации и массового переселения в города это соотношение постепенно изменялось. Уже в начале этого века больше половины рабочей силы большинства европейских стран и США было занято в сфере промышленного производства. К середине XX века “индустриальное общество” окончательно пришло на смену “аграрному”. Связанные с этим глобальные изменения нашли самое широкое отражение в литературе, живописи, театре. Всеобщее среднее образование, развитие масштабов высшего образования и превращение науки в производительную силу привели к следующему этапу глобальных изменений, которые мы наблюдаем сегодня.

В условиях радикального усложнения жизни общества, его технической и социальной инфраструктуры, решающим является изменение отношения людей и информации, которая становится таким же стратегическим ресурсом общества, как продукты питания в “аграрном”, а традиционные материальные или энергетические ресурсы в “индустриальном” обществе.

Информационное (постиндустриальное) общество – общество, основным фактором развития которого являются автоматизированные информационные технологии.

“Информационная революция”, как и предшествовавшие ей “аграрная” и “индустриальная” революции, решая одни проблемы, порождают новые.

Одной из таких проблем является “информационный взрыв”, избыток доступных многим современным людям данных, которых больше, чем в состоянии переварить человеческое сознание, служит причиной снижения качества мышления среди, в том числе, образованных членов современного общества. Информационная перегрузка – это реальность. Ежегодно публикуется более 40 000 научных статей. Специалисты жалуются, что они не в состоянии оценить всего, что относится к их предметной области. Огромные объемы информации, особенно собранные в виде статистических данных, являются полем для ошибочной и (или) преднамеренно ложной интерпретации.

* Жирным шрифтом выделены новые понятия, которые необходимо усвоить. Знание этих понятий будет проверяться при тестировании.

Опасность того, что немногочисленные “эксперты”, контролирующие информационные потоки, могут эксплуатировать менее образованную часть населения, становится реальностью. Те, кто не имеет достаточного доступа к информационным системам, будут находиться в гораздо менее выгодном положении.

Различие в уровне информационного обеспечения сегодня становится одной из существенных причин в дисбалансе экономического развития передовых и слабо развитых стран, порождает нестабильность в отношениях между странами. Информационная революция меняет глобальную экономику, преобразует национальную политику и предпринимательство, заставляет пересматривать внешнеполитические цели государств и методы их достижения. Неспособность старых политических структур решить задачи “информатизации общества” явилась одной из практических причин экономического и политического кризиса, который переживают сегодня многие страны.

Знания – вся совокупность полезной информации и процедур, которые можно к ней применить, чтобы произвести новую информацию. Знание всегда дает власть тем, кто им владеет и умеет им пользоваться. Увеличение объема и распространение информации среди огромной массы людей является предвестником изменений существующих структур власти. Политический и экономический кризис, который переживает сегодня мировое сообщество, во многом связан, кроме прочих причин, с начавшимся переходом многих стран к жизни в “информационном обществе”.

Информационная культура – совокупность методов, приемов и навыков по сбору, хранению, обработке и созданию информации.

Уровень информационной культуры – степень упорядоченности, системности и эффективности использования информационных технологий, а также относительный объем использования новых информационных технологий.

Будущее вторгается к нам уже сегодня. Глобальная информационная инфраструктура уже влияет на нашу жизнь. Эти изменения оказывают серьезное влияние на содержание, методы и средства современных социально-экономических, политических и других систем. В то же время влияние “информационного общества” на изменение целей и обновление содержания общего образования ощущается не слишком явно. Общественность слабо осознает проблемы, достоинства и опасности, которые несет с собой широкое распространение информационных технологий и с которыми придется столкнуться сегодняшним школьникам. Задача подготовки учащихся к использованию информационных технологий, к жизни полноценными гражданами в мире, который будет существенно отличаться от нашего прошлого и даже настоящего, должна быть не просто сформулирована, но и осознана.

В своем развитии человечество с неизбежностью проходило ряд кризисов, вызванных тем, что оно, как биологический вид уже давно стало монополистом. Последнее столетие активная деятельность людей меняет весь облик планеты. Ещё в начале века В.И. Вернадский говорил о том, что человек превращается в основную геолого-образующую силу планеты. Его монополизм стал беспрецедентным, поэтому экологические кризисы в истории человечества неизбежны. А поскольку сферой обитания человечества является вся планета, то эти кризисы носят глобальный характер и сказываются на судьбе всей биосферы, превращаясь во всепланетное явление.

Особенностью современного кризиса является то, что в конце XX века есть все основания думать, что ресурсы, необходимые для развития современной цивилизации, близки к исчерпанию. Чтобы человечество сохранилось, неизбежен пересмотр всех, ставших привычными, основ самосознания себя в этом мире. Информационные технологии являются

одной из тех основ, опираясь на которые человеческое общество дает ответ на вызовы окружающей среды.

1.2. Информация, данные и информационные технологии

Понятия “информация”, “знание”, “информационная система” следует в значительной степени считать интуитивными. До настоящего времени формальной теории автоматизированных информационных систем, подобной, допустим, теории кодирования и передачи информации или аналитической теории алгоритмов не существует.

В то же время существует инженерная дисциплина, которая базируется на хорошо формализованных теоретических построениях и имеет огромное значение в жизни современного общества. Эта дисциплина в различных странах имеет различные названия, но в содержательном плане ее предметная область является общепризнанной в мировом научном сообществе. В Европе эту дисциплину называют информатикой, а в США и других англоязычных странах – компьютерной наукой. Основными понятиями этой дисциплины являются следующие:

информация – любой вид сведений о предметах, фактах, понятиях предметной области, или сведения, неизвестные до их получения, являющиеся объектом хранения, передачи и обработки;

данные – представление информации в формализованном виде, удобном для пересылки, сбора, хранения и обработки;

сигнал – носитель данных (информации), может представлять собой физические сигналы или математические модели;

информационная технология (ИТ) – система научных и инженерных знаний, а также методов и средств, которая используется для создания, сбора, передачи, хранения и обработки информации в предметной области;

автоматизированная информационная технология (АИТ) – информационная технология, в которой для передачи, сбора, хранения и обработки данных используются методы и средства вычислительной техники и систем связи.

Альтернативы перехода мирового сообщества к информационному обществу на сегодня нет, более того этот переход уже активно осуществляется в большинстве стран мира.

Основой успешного перехода в информационное общество является владение всеми членами общества знаниями, умениями и навыками в использовании информационных технологий в быту, общественной и профессиональной деятельности. Эти задачи решают национальные системы образования.

Информационные технологии имеют ряд замечательных свойств, присущих только им. Основные из этих свойств перечислены ниже и они позволяют значительно увеличить эффективность всех процессов социально-экономического развития:

современный, научно-технический уровень ИТ таков, что им могут быть “перепоручены” практически все “нетворческие” процессы обработки информации, которые используются в человеческой деятельности;

основная масса навыков и приемов, которые используются в не творческих областях деятельности, хорошо алгоритмизируются в рамках ИТ;

современные средства связи, охватывающие весь земной шар, позволяют обеспечить доступ к различным видам ИТ в любой его точке;

на протяжении последних лет наблюдается неуклонное уменьшение абсолютной величины отношения стоимости к производительности у технических компонент автоматизированных информационных систем, что

повышает экономическую эффективность использования ИТ.

Вышеперечисленные (и ряд других) характеристики ИТ делают их незаменимым средством не только в борьбе с кризисными ситуациями в социально-экономических процессах, но и в повышении их качества и эффективности.

Рассмотрим более подробно, каким образом используются ИТ в различных областях человеческой деятельности.

Использование автоматизированных ИТ в управлении экономикой имеет столь же древнюю историю, как и история человечества. Это, в первую очередь, история создания автоматизированных систем управления (АСУ) различного назначения, от АСУ цехами и промышленными предприятиями – до АСУ национальными экономическими системами. В целом, можно считать, что методологическое, нормативно-техническое и прочие виды обеспечения ИТ, используемых в данных видах деятельности имеют хороший уровень проработки, а современный научно-технический уровень базовых ИТ позволяет создавать эффективные прикладные системы.

Проблемы ресурсных ограничений в создании и распространении таких систем в значительной степени снимаются переходом от индивидуального к типовому, массовому производству и внедрению. Сегодняшний уровень теоретической и инженерной проработки позволяет это делать. Эффективность использования ИТ по этим направлениям хорошо обосновывается.

В системах образования информационного общества информационные технологии должны заменить полностью или в значительной степени, такие средства обучения, как: перо, бумагу, книги и т.д. (в том числе и учебные аудитории). Здесь нужно рассматривать, как минимум, два вопроса:

готовность к этому преподавателей и учащихся;

наличие ИТ, которые могут это обеспечить.

На оба вопроса можно дать положительные ответы.

Опыт широкого использования ИТ в быту и различных областях человеческой деятельности показывает, что чем раньше человек начинает пользоваться этими средствами, тем легче и успешнее он ими овладевает. Таким образом, можно утверждать, что и основной контингент преподавателей и учащихся готовы, или могут быть достаточно быстро подготовлены к новой роли.

Методы и средства обучения при использовании информационных технологий претерпевают наиболее радикальные изменения по сравнению со всеми остальными направлениями деятельности в системе образования. ИТ оказывают решающее влияние на все этапы процесса обучения от предоставления учащимся знаний и навыков до контроля их усвоения, при этом обеспечиваются такие важнейшие характеристики обучения, как: качество, избирательность материала, учет индивидуальности, постоянный контроль усвояемости материала, высокая эффективность использования ресурсов преподавателей и т.д.

В информационном обществе наиболее важным средством для развития систем образования и повышения их эффективности являются информационные технологии. Их применение позволяет:

по всем направлениям применения информационных технологий значительно повысить эффективность работ;

при одинаковых затратах получать больший эффект;

обучение на базе ИТ позволит сократить разрыв между количеством людей, желающих получить образование и возможностью системы образования его предоставить.

Распространение телекоммуникационных технологий на базе средств вычислительной техники и средств связи вовлекает в сферу информационного

общества все новые слои населения и новые виды человеческой деятельности – от бытовых до общественно-политических.

Таким образом информационные технологии являются средством, которое позволяет достигать определенные цели, поставленные в различных предметных областях человеческой деятельности, а также повышают эффективность решения задач, направленных на достижение этих целей.

Информационные технологии, в первом приближении, принято разделять на базовые и прикладные.

Под **базовыми информационными технологиями** понимается система научных и инженерных знаний, а также методов и средств, которая используется для создания, сбора, хранения и обработки информации безотносительно к предметной области, в которой создается и используется данная информация. В состав этих технологий входят:

- операционные системы;
- языки программирования и технологии их использования (компиляторы, библиотеки, CASE-технологии и т.д.);
- системы управления базами данных;
- технологии работы в телекоммуникационных сетях;
- экспертные и другие интеллектуальные системы.

С помощью базовых ИТ, средств вычислительной техники и связи разрабатываются и поддерживаются **прикладные ИТ**, обеспечивающие эффективную поддержку конкретных областей человеческой деятельности.

1.3. Автоматизированные информационные системы

Процессы обработки информации всегда являлись основой человеческой деятельности и объединение таких процессов с информационными ресурсами, со временем стали называть **информационными системами (ИС)**. ИС – это комплекс, состоящий из информационной базы (хранилища информации) и процедур, позволяющих накапливать, хранить, корректировать, осуществлять поиск, обработку и выдачу информации. С появлением вычислительной техники ИС пережили качественный, революционный процесс развития превратившись в **автоматизированные информационные системы (АИС)**, т.е. – информационные системы, физической и функциональной компонентами которых является программно-технический комплекс и средства связи.

Современные АИС представляют собой чрезвычайно сложные человеко-машинные комплексы, интегрированные (неразрывно связанные) в национальную и мировую информационные среды. Именно эта интеграция и создает эффективную научно-техническую базу информационного общества, так как изолированные АИС в настоящее время малоэффективны.

Эффективность АИС во многом определяется их качеством и доверием к ним пользователей. Качество изделий, процессов проектирования, производства и услуг является одной из узловых проблем, определяющей уровень жизни человека и состояние народного хозяйства, что полностью относится и к области информационных технологий. В АИС входят следующие основные компоненты:

- аппаратные средства вычислительной техники;
- аппаратные средства телекоммуникации (связи);
- программные средства реализации функций АИС;
- информационные базы данных (БД);
- документация, регламентирующая функции и применение всех компонент АИС;
- специалисты, обслуживающие и использующие программно-технические средства.

Аппаратные компоненты АИС имеют достаточно универсальный характер и относительно слабо зависят от функционального назначения конкретной информационной технологии. Хотя при их выборе всегда учитывается ряд технических характеристик, анализ и испытания этих компонент могут проводиться достаточно традиционными методами и средствами, разработанными в области сложного приборостроения.

Остальные компоненты АИС составляют их интеллектуальную часть, определяющую назначение, функции и качество решения задач в конкретной области человеческой деятельности. Эти компоненты могут отличаться принципиальной новизной, большим разнообразием характеристик, которые трудно формализуются и требуют глубокого исследования методов проверки их значений.

Архитектурная, техническая и программно-информационная совместимость различных АИС может быть обеспечена только путем стандартизации и сертификации программно-технических средств в соответствии с требованиями международных и государственных стандартов. Для этого необходима стандартизация, сертификация и каталогизация средств, процессов и услуг, а также проведение единой технической политики при создании (приобретении) совместимых аппаратных и программных средств, организации взаимодействия и интеграции АИС различных уровней и назначения.

Любая реальная АИС действует в окружающей ее информационной среде, которую часто называют инфраструктурой.

Под инфраструктурой в экономике понимаются структуры, которые обеспечивают функционирование производственных систем, но непосредственно в технологических процессах производства продукции не участвуют. В их число входят: дороги, линии электропередач, системы снабжения ресурсами и т.д.

Под **инфраструктурой автоматизированных информационных систем** обычно понимают телекоммуникационные сети и связываемые ими объекты: серверы, автоматизированные рабочие места, каталоги сетевых информационных ресурсов и т.п. **Информационными ресурсами** являются информационные базы (банки и базы данных) различного назначения и другие информационные структуры.

В настоящее время особенно быстро развиваются телекоммуникационные компоненты инфраструктуры АИС. В области технологий передачи данных открывают новые горизонты в использовании сетей. Раньше, на начальной фазе внедрения таких технологий, возникало немало проблем – это и недостаток опыта использования, и относительно слабая совместимость, и ограниченный выбор оборудования. Но пожалуй основной преградой являлся изначально слабый пользовательский спрос, что в случае компьютерных сетей выражается в отсутствии адекватных технологии приложений.

Таким образом при создании сетей передачи данных нового поколения возникает необходимость решения сразу комплекса задач, лежащих как в области телекоммуникаций, так и в области информационных технологий. Современное состояние телекоммуникационных технологий представляет собой весьма пеструю картину. С одной стороны “всемирная паутина” (Internet) опутала весь земной шар и ближний космос, с другой – не иссякает критический поток со стороны пользователей данной “паутины”. Самый распространенный – что Интернет неудовлетворительно обеспечивает поиск нужной информации и доступ к ней, то есть не обеспечивает нужного качества тех услуг, ради которых она и создавалась. Кроме того имеется ощутимый разрыв в возможностях пользования этими услугами между различными группами населения и различными странами. Этот разрыв не только не сокращается, но наоборот увеличивается. Национальные правительства и

международные организации постоянно ощущают значительный дефицит ресурсов, которые бы они могли и хотели использовать для сокращения такого разрыва.

В связи с этим возникают проблемы безопасности использования АИС, которые здесь будут только обозначены, а детально будут рассматриваться во второй юните.

Быстрое развитие и использование информационных технологий не только открывает новые возможности, но и создает новые проблемы перед мировым сообществом, которые безусловно влияют на экономическую, социальную, культурную и образовательную деятельность и могут коренным образом повлиять на формы правления, творчество, сотрудничество, обмен идеями и знаниями, а также на повседневную жизнь (см. п. 1.1.). Такими проблемами являются:

- психобиологические, оказывающие отрицательное психологическое и физическое воздействие на пользователей;

- культурные, угрожающие национальной культурной самобытности пользователей;

- социально-экономические, увеличивающие неравенство возможностей получения доступа к качественным ИТ;

- политические, способствующие разрушению гражданского общества в национальных государствах;

- бесконтрольное и несанкционированное использование чужой интеллектуальной собственности;

- технологические угрозы нанесения ущерба или разрушения самим АИС.

1.4. Содержание и структура теории информационных систем

Огромные средства затрачиваются во всем мире на разработку многочисленных конкретных, прикладных систем и совершенно недостаточное внимание уделяется теоретическим вопросам. В принципе, такой же путь прошли в свое время и работы в области создания автоматизированных систем управления и телекоммуникационных технологий, вероятно это необходимый этап – “болезнь роста”.

Необходимость концептуального, системного осмысления положения дел в некоторой предметной области возникает в силу того, что на определенном этапе развития этой области накапливается большое количество знаний, фактов, задач и интересов, которые слабо увязаны между собой. Такое положение дел периодически возникает во всех развивающихся областях человеческих знаний, в быстроразвивающихся областях процессы могут принимать кризисный характер, вплоть до смены понятийного аппарата предметной области.

В частности при разработке сложных перспективных автоматизированных информационных систем процесс создания включает большое разнообразие видов деятельности и требует тесного взаимодействия между представителями научно-технических профессий и лицами, принимающими политические и экономические решения. Возникает необходимость сведения воедино огромных объемов разнообразной информации, согласования большого числа различных и зачастую противоречивых целей и интересов.

Этими обстоятельствами и вызвана необходимость закладывать в основу инженерной дисциплины “Автоматизированные информационные системы” хорошо проработанные разделы фундаментальных и прикладных научных дисциплин. Так как компоненты АИС (см. 1.3.) принадлежат к различным предметным областям, то и теоретические основы АИС включают в себя положения из различных научных дисциплин. Далее будут рассмотрены

теоретические основы компонент АИС.

При анализе и проектировании дискретных электронных схем для вычислительной техники и современных средств связи используют математический аппарат теории множеств, двоичной логики (булева алгебра) и теории кодирования. При разработке цифровых систем связи используются также методы математической статистики.

В разработках программного обеспечения и языков программирования используются методы автоматически-лингвистических моделей, аналитическая теория алгоритмов, модели исчисления предикатов, графовые модели и оптимизационные методы.

При разработке баз данных и информационных языков поиска и запроса данных в информационных базах используются методы реляционной алгебры, модели исчисления предикатов и математической лингвистики, а также графовые модели.

При анализе эффективности, работоспособности и надежности компонентов АИС и АИС в целом используются методы теории массового обслуживания и математическая статистика.

Наконец в информационно-поисковых АИС используются методы теории информации, основными понятиями которых являются **документ** – материальный объект с информацией, закрепленной созданным человеком способом для ее передачи во времени и пространстве и **классификатор** – официальный документ, представляющий систематизированный свод наименований и кодов кодификационных группировок и (или) объектов классификации.

Далее, в этой юните теоретические основы АИС рассматриваются более подробно. Все эти методы интегрируются с помощью общей теории систем и ее прикладной частью – системотехникой.

2. ЛОГИКО-МАТЕМАТИЧЕСКИЕ ОСНОВЫ АВТОМАТИЗИРОВАННЫХ ИНФОРМАЦИОННЫХ СИСТЕМ

2.1. Элементы математической логики

Информационные системы предназначены для накопления сведений, хранения их и выдачи по мере необходимости. Сведения эти представляют собой описания предметов реального мира или абстрактных предметов, возникающих в различных областях деятельности, и представляют собой некоторые “истинные” утверждения или сообщения. С течением времени или в результате ошибок они могут становиться “ложными”. Естественно, что одной из дисциплин, лежащих в основе теории информационных систем, должна быть и является математическая логика. В процессах анализа и разработки АИС широко используется математическое моделирование, в основе которого лежит понятие **экспликации** – строгой (математической) формулировки содержательного или интуитивного понятия.

Математическими дисциплинами, пригодными для описания совокупностей предметов и их свойств, являются:

математическая логика – методика и теория математических доказательств;

алгебра высказываний – операции над высказываниями, которые являются элементами множеств;

реляционная алгебра – совокупность множества отношений и множества операций над ними.

Как видим, основными понятиями, которые используются во всех этих дисциплинах являются понятия **множества** – объединение некоторого

количества определенных, отличных друг от друга объектов, которые называются элементами множества и, **подмножества** – А и В, два множества, если каждый элемент множества А является также элементом множества В, то А называется подмножеством В.

Таким образом, в основе теории информационных систем лежат математическая логика, теория множеств, реляционная алгебра, теория формальных языков, теория алгоритмов и теория, сложных систем.

Важным этапом развития математической логики является применение в ней самой ряда приемов, разработанных в математике, например использование буквенных (и вообще символьных) обозначений, применение математической абстракции, математического обобщения, понятий операции и логического значения (родственного понятию числа) и т.п. В результате математическая логика приобрела все черты математической дисциплины.

В связи с появлением ЭВМ и развитием теории алгоритмов математическая логика получила (уже как раздел математики) непосредственные практические приложения.

Предложения какого-либо языка могут являться описаниями предметов, явлений или отношений, существующих, например, в реальном мире. При этом оказывается, что кроме своей структуры и смыслового содержания они различаются еще степенью соответствия их содержания описываемому объекту. В простейшем случае последняя характеристика сводится к тому, что они признаются либо истинными, либо ложными.

Определение. Логическими значениями называются два абстрактных объекта: истина и ложь.

Одинаково распространены три способа кодирования (условного обозначения) логических значений (см. табл. 2.1). В АИС чаще всего пользуются кодом №3.

Таблица 2.1

Коды логических значений

Код № 1	Код № 2	Код № 3
ложь	Л	0
истина	И	1

Иногда логические значения определяют эмпирически или экспериментально, иногда же их задают по произволу (например, делая какое-либо допущение). В определенных случаях их получают по правилам математической логики.

Абстрагируясь от смыслового содержания предложений, приходим к идее **высказывания** — некоторого символа, обладающего логическим значением. Приведем несколько примеров.

Примеры.

1) Запись всякого предложения можно считать символом. Рассматривая этот символ вместе с поставленным ему в соответствие логическим значением, получаем высказывание. Например, высказываниями можно считать совокупности: а) “Москва – столица России” (истина); б) “Кислород легче водорода” (ложь). Первое из логических значений получено эмпирически, а второе – экспериментально. Можно считать высказыванием также пару: “Через точку, лежащую вне прямой, можно провести только одну прямую, параллельную данной” (И). В последнем случае логическое значение задано произвольно; ни эмпирически, ни экспериментально получить его нельзя, так

как это потребовало бы проверки того, что на всем своем бесконечном протяжении прямые не пересекаются. Не удастся получить его и средствами математической логики.

2) Не следует рассматривать как высказывания предложения “Иди сюда”, “Который час?”, так как по своему содержанию они не являются ни истинными, ни ложными. Тем не менее, если записи каждого из них поставить в соответствие логическое значение, с точки зрения математической логики будут получены высказывания.

2.2. Алгебра высказываний и логические связи

Частью математической логики является алгебра высказываний.

Определение. **Высказываниями** называются

- 1) элементарные высказывания;
- 2) составные высказывания.

Определение. **Элементарным высказыванием** называется символ, которому поставлено в соответствие логическое значение.

Замечание 1. Приведенное определение можно сформулировать и так: элементарным высказыванием называется совокупность символа и логического значения.

Замечание 2. Говоря об элементарных высказываниях, обычно указывают только символ, а логическое значение подразумевают. Такая манера говорить неявно предъявляет требование к совокупности рассматриваемых высказываний, заключающееся в следующем: в одной и той же области применения нельзя рассматривать высказывания, имеющие одинаковые символы и различные логические значения.

Пример. Элементарными являются высказывания, приведенные выше в примере. Кроме того, к их числу относятся: А (истина), В (ложь), С (И), Х (Л), У(1), Z (0).

Именно для элементарных высказываний характерно то, что их логические значения возникают вне математической логики. Уместно подчеркнуть, что не только с математической точки зрения, но и в реальных условиях логическое значение не определяется ни структурой, ни содержанием элементарного высказывания. Например, наряду с элементарным высказыванием “Снег – бел” (1), поскольку существует феномен красного снега, возможно и высказывание “Снег – бел” (0).

Определение. **Логическими связками** называются знаки.

$\neg, \wedge, \vee, \rightarrow, \sim, \neg\sim$ читаемые соответственно как “не”, “и”, “или”, “влечет”, “эквивалентно”, “неэквивалентно”. С помощью логических связок осуществляют построение сложных высказываний.

Замечание. Те же самые знаки, которыми пользуются как логическими связками, применяют как условные обозначения операций над логическими значениями.

Для логических значений определены следующие операции.

- 1) Операция отрицания; выполняется в соответствии с табл. 2.2.
- 2) Операция логического умножения; выполняется в соответствии с табл. 2.3.
- 3) Операция логического сложения; выполняется в соответствии с табл. 2.4.
- 4) Операция оценки импликации; определяется в соответствии с табл. 2.5.
- 5) Операция оценки эквивалентности; определяется в соответствии с табл. 2.6.
- 6) Операция оценки неэквивалентности, выполняется в соответствии с табл. 2.7.

Определение. Если А и В – высказывания, то составным высказыванием называется совокупность любой из нижеприведенных записей и ее логического значения: 1) $\neg(A)$ – называется отрицанием А и читается “не А”; 2) $(A) \wedge (B)$ – называется конъюнкцией А и В и читается “А и В”; 3) $(A) \vee (B)$ – называется

Таблица 2.2

Отрицание

$\neg 0 = 1$
$\neg 1 = 0$

Таблица 2.3

Логическое
умножение

$0 \wedge 0 = 0$
$0 \wedge 1 = 0$
$1 \wedge 0 = 0$
$1 \wedge 1 = 1$

Таблица 2.4

Логическое
сложение

$0 \vee 0 = 0$
$0 \vee 1 = 1$
$1 \vee 0 = 1$
$1 \vee 1 = 1$

Таблица 2.5

Оценка
импликации

$0 \rightarrow 0 = 1$
$0 \rightarrow 1 = 1$
$1 \rightarrow 0 = 0$
$1 \rightarrow 1 = 1$

Таблица 2.6

Оценка
эквивалентности

$0 \sim 0 = 1$
$0 \sim 1 = 0$
$1 \sim 0 = 0$
$1 \sim 1 = 1$

Таблица 2.7

Оценка
неэквивалентности

$0 \neg \sim 0 = 0$
$0 \neg \sim 1 = 1$
$1 \neg \sim 0 = 1$
$1 \neg \sim 1 = 0$

дизъюнкцией А и В и читается “А или В”; 4) $(A) \rightarrow (B)$ – называется импликацией А и В и читается “А влечет В”; 5) $(A) \sim (B)$ – называется эквивалентностью А и В и читается “А эквивалентно В” 6) $(A) \neg \sim (B)$ – называется отрицанием эквивалентности А и В и читается “А не эквивалентно В”.

Логическое значение каждого из перечисленных сложных высказываний вычисляют по логическим значениям А и В с помощью операции, знак которой совпадает с логической связкой, присутствующей в вышеприведенной записи.

Приведенные выше операции над логическими значениями известны под названием **булевых операций**.

Пример. Если заданы высказывания А(0), В(1), С(1), то $\neg A$, $A \wedge B$, $B \vee C$ являются сложными высказываниями (а значит, и просто высказываниями). Их логические значения легко вычислить; они соответственно равны 1, 0, 1. Очевидно, $(\neg A) \wedge (A \wedge B)$ и $(A \wedge B) \vee (B \wedge C)$ тоже являются высказываниями, причем первое имеет логическое значение 0, а второе – логическое значение 1 (см. табл. 2.2 – 2.4).

Определение. Два высказывания называются равнозначными, если их логические значения равны (одинаковы). Отношение равнозначности обозначается символом \approx .

Легко видеть, что отношение равнозначности является

- а) рефлексивным, так как для любого высказывания А справедливо $A \approx A$;
- б) симметричным: из $A \approx B$ следует $B \approx A$;
- в) транзитивным: из $A \approx B$ и $B \approx C$ следует $A \approx C$. Другими словами, это отношение относится к числу отношений эквивалентности.

Строку, в которой записаны символы, обозначающие два высказывания, соединенные знаком \approx , будем называть равнозначностью.

С помощью табл. 2.2 легко установить равнозначность

$$\neg(\neg A) \approx A, \quad (2.1)$$

проверяя ее для обоих возможных значений А. Аналогичным способом получим

$$A \wedge B \approx B \wedge A. \quad (2.2)$$

Проверку последней равнозначности для всевозможных сочетаний логических значений A и B удобно свести в таблицу.

A	B	$A \wedge B$	$B \wedge A$
0	0	0	0
0	1	0	0
1	0	0	0
1	1	1	1

Равнозначность (2.2) выражает перестановочный (коммутативный) закон для конъюнкции.

Справедливость остальных равнозначностей данного раздела предоставляем проверить читателю. Заметим лишь, что такая проверка является совершенно строгим доказательством. Возможность доказательств проверкой возникает благодаря тому, что логических значений всего два. В арифметике мы не могли пользоваться таким способом доказательства, потому что натуральных чисел бесконечно много.

Равнозначность

$$(A \wedge B) \wedge C \approx A \wedge (B \wedge C) \quad (2.3)$$

выражает ассоциативный (сочетательный) закон для конъюнкции. Она позволяет в многочисленных конъюнкциях опускать скобки и употреблять выражения вида $A \wedge B \wedge C$. Далее следует указать равнозначности

$$A \wedge A \approx A, \quad (2.4)$$

$$A \wedge 1 \approx A, \quad (2.5)$$

$$A \wedge 0 \approx 0. \quad (2.6)$$

В последних двух формулах 0 и 1 означают соответственно произвольное заведомо ложное и произвольное заведомо истинное высказывания.

Нетрудно также убедиться в правильности равнозначностей

$$A \vee B \approx B, \quad (2.7)$$

$$(A \vee B) \vee C \approx A \vee (B \vee C), \quad (2.8)$$

$$A \vee A = A, \quad (2.9)$$

$$A \vee 1 = 1, \quad (2.10)$$

$$A \vee 0 = A. \quad (2.11)$$

Равнозначность (2.7) выражает перестановочный закон; (2.8) выражает сочетательный закон для дизъюнкции, в силу которого в многочисленных дизъюнкциях можно опускать скобки и писать, например, $A \vee B \vee C$.

Равнозначность

$$A \wedge (B \vee C) \approx (A \wedge B) \vee (A \wedge C) \quad (2.12)$$

утверждает, что конъюнкция по отношению к дизъюнкции дистрибутивна, подобна тому, как в арифметике умножение дистрибутивно относительно сложения, что выражается формулой

$$a(b + c) = ab + ac.$$

и дизъюнкция дистрибутивна относительно конъюнкции

$$A \vee (B \wedge C) = (A \vee B) \wedge (A \vee C), \quad (2.13)$$

что не имеет аналогии в арифметике, так как формула $(a+bc) = (a+b)(a+c)$ неверна.

Уменьшение числа скобок в составных высказываниях достигают еще тем, что принимают соглашение считать связь \neg сильнее, чем \wedge , а эту связь – сильнее, чем \vee , которая сильнее связей \rightarrow , \sim , $\neg\sim$, имеющих одинаковую силу. Из этого соглашения и формул (2.3) и (2.8) следует, например, что вместо $((A \wedge B) \vee C) \rightarrow D$ можно писать $A \wedge B \vee C \rightarrow D$.

Большое значение имеют также равнозначности

$$A \wedge B \approx \neg(\neg A \vee \neg B), \quad (2.14)$$

$$A \vee B \approx \neg(\neg A \wedge \neg B), \quad (2.15)$$

$$A \rightarrow B \approx \neg A \vee B, \quad (2.16)$$

$$A \sim B \approx (A \rightarrow B) \wedge (B \rightarrow A), \quad (2.17)$$

$$A \neg\sim B \approx \neg(A \sim B). \quad (2.18)$$

В заключение этого раздела заметим, что высказывания ведут себя как логические константы.

Если рассматривать равнозначности, приведенные выше, как правила преобразования составных высказываний, в результате которых из одних высказываний получаются другие, имеющие новую структуру, но равнозначные исходным, то из формул (2.14), (2.16), (2.17) и (2.18) легко усмотреть, что любое высказывание можно привести к виду, содержащему только связи \neg и \vee . Таким образом, **логические связи дизъюнкции и отрицания образуют полную систему**.

Рассматривая несколько видоизмененный набор формул (2.15), (2.16), (2.17) и (2.18), мы видим, что любые логические связи можно выразить через связи \neg и \wedge . Это значит, что логические связи отрицания и конъюнкции тоже образуют полную систему.

В принципе, вместо рассмотрения всех логических связей и получаемых с их помощью составных высказываний, можно было бы обойтись одной из полных систем. Однако нужно помнить, что равнозначные высказывания не тождественны. Они имеют лишь одинаковые логические значения, но как предложения некоторого языка могут иметь не одинаковый смысл.

2.3. Понятие предиката

Слова или тексты, являющиеся собирательными именами (групповыми именами) предметов, обозначим x, y, \dots, z . Групповое имя обозначает произвольный предмет, принадлежащий некоторой группе предметов, имеющих собственные имена. Например, текст “житель Москвы” является групповым именем людей, каждый из которых является конкретным предметом и имеет свое собственное имя (для простоты примера мы отвлекаемся от факта наличия жителей Москвы, имеющих одинаковые фамилию, имя и отчество).

Пусть $P(x, y, \dots, z)$ означает некоторый текст, содержащий в своем составе групповые имена x, y, \dots, z и обладающий следующим свойством: если в этом тексте каждое групповое имя всюду, где оно входит, заменить допустимым

индивидуальным (собственным) именем (одним и тем же), то данный текст превратится в символическую часть некоторого высказывания.

Текст $P(x, y, \dots, z)$ называется **предикатом**; входящие в него групповые имена называются **предметными переменными**; о предметах, соответствующих групповому имени, являющемуся предметной переменной, говорят, что они принадлежат предметной области данной переменной; собственные имена указанных предметов называют **значениями предметной переменной**; логические значения получаемых высказываний называются значениями предиката. Количество различных предметных переменных, входящих в состав текста $P(x, y, \dots, z)$, называется **рангом предиката** (иногда – его “местностью” или “арностью”). Например, бывают предикаты первого ранга (одноместные, унарные), второго ранга (двухместные, бинарные), третьего ранга (трехместные, тернарные) и т.д.

Если высказывание является логической константой, то предикат – логической функцией.

Пример. Предметные переменные не обязательно обозначаются отдельными буквами, они могут обозначаться и целыми текстами. Одноместными предикатами являются $x > 10$ или “целое число больше 10”. Если предметной областью является область целых чисел, то эти предикаты отличаются только своей формой, но являются одинаковыми (или тождественно равными, равнозначными) логическими функциями. Если положить $x=8$, то оба предиката соответственно примут вид: $8 > 10$ и “8 больше 10”. И то и другое высказывание после проверки оказывается ложным, так что значением предикатов оказывается ложь (Л, 0). В данном случае при подстановке в предикаты вместо предметной переменной некоторого ее значения получилось элементарное высказывание. Но это – не обязательно. Например, предикат $x > 10 \vee x < 5$ при подстановке $x=6$ превращается в символическую часть высказывания $6 > 10 \vee 6 < 5$, которое оказывается ложным, в чем мы убеждаемся после вычисления его значения.

Предикат $x^2 + y^2 = z^2$, если предметные области между собой совпадают и совпадают с областью действительных чисел, при $x=1,5$; $y=-2$; $z=8$ превращается в символическую часть высказывания $1,5^2 + (-2)^2 = 8^2$. Логическое значение этого высказывания нетрудно установить. Оно есть ложь.

Последний предикат является тернарным (имеет третий ранг).

С формальной точки зрения высказывание является символом, обозначающим логическое значение (в математике говорят: совокупностью символа и определенного логического значения). Содержательное высказывание является повествовательным предложением, содержащим утверждение о некотором предмете (некоторых предметах). Предикат в содержательном смысле оказывается повествовательным предложением, содержащим утверждение о произвольных (переменных, любых) предметах, принадлежащих соответствующим предметным областям.

2.4. Элементы теории множеств, операции над множествами

В результате многовекового естественного отбора нервная система человека и высших животных сложилась так, что окружающий мир воспринимается ею как состоящий из отдельных друг от друга предметов. Восприятие части мира как отдельного предмета связано с бессознательно совершаемым актом абстрагирования, в результате которого слабые влияния частей предмета на другие предметы (и их части) либо остаются незамеченными, либо не учитываются; более сильные влияния частей предмета принимаются за влияния всего предмета в целом.

Абстракцию, в результате которой некоторая часть мира признается

предметом, будем называть **абстракцией предмета**. Из представлений о реальном мире понятие предмета было перенесено и в науку. Пользуясь этим понятием и применяя построенные на его основе научные результаты к решению практических задач, нужно не забывать о связанной с ним абстракцией и, следовательно, о его приближенном характере.

Математическое понятие множества, иллюстрациями которого в реальном мире являются различные собрания предметов, коллекции, совокупности (например, стадо овец, куча песка, совокупность типографских знаков, расположенных в данной строке, и т. п.) связано еще с одной абстракцией, которую будем называть абстракцией множества. Сущность **абстракции множества** заключается в том, что действительно существующие связи объединяемых в нем предметов между собой и с другими предметами игнорируются, а вместо них объединяемым предметам приписывают новые связи друг с другом, выражающие их принадлежность множеству. При этом считается, что два предмета, ничем не отличающиеся друг от друга, являются одним и тем же предметом. Поэтому все предметы, образующие множество, между собой различны. Само множество является новым предметом.

Если предмет a входит в множество M , то говорят: " a является элементом M " и пишут $a \in M$.

Знак \in называют знаком включения. Если о каком-нибудь предмете b известно, что он не является элементом множества M , то пишут $b \notin M$.

Удобно считать, что возможны: а) множество, не содержащее ни одного элемента, называемое **пустым**, и б) множество, состоящее из одного элемента, называемое **одноэлементным**. Пустое множество обозначают символом \emptyset . Допуская существование таких множеств, мы принимаем соглашение, упрощающее целый ряд формулировок.

Пример. Допустимо говорить о множестве звезд 100-й величины хотя таких звезд (из-за их малой яркости) нельзя увидеть даже в самый мощный телескоп. Если таких звезд вообще нет, то их множество является пустым.

Пример. Элементами множества могут быть предметы, существующие в разное время. Например, можно говорить о множестве високосных годов XX века.

Пусть A и B – два множества. Если каждый элемент множества A является также элементом множества B , то A называют подмножеством множества B и пишут $A \subseteq B$.

Очевидно, что всякое множество является своим подмножеством. Пустое множество является подмножеством любого множества.

Если $A \subseteq B$ и существует такой элемент множества B , который не является элементом множества A , то A называется правильной частью B . При этом пишут $A \subset B$.

Если $A \subseteq B$ и $B \subseteq A$, то говорят, что множества A и B равны, и пишут $A = B$.

Из приведенного определения вытекает, что равными являются только множества, состоящие из одних и тех же элементов.

Пример. Множество членов некоторой семьи и множество людей, присутствующих в некоторой комнате, вообще говоря, считаются различными, но если в упомянутой комнате присутствуют все члены названной семьи и нет никого более, то эти множества равны.

Пусть по-прежнему A и B – множества. Множество C тех элементов, которые принадлежат и A и B , называется пересечением или теоретико-множественным произведением множеств A и B . При этом пишут $C = A \cap B$.

Знак \cap называется знаком теоретико-множественного умножения. Операция построения $A \cap B$ по заданным A и B называется теоретико-множественным умножением. Для этой операции, как легко сообразить, справедлив перестановочный закон

$$A \cap B = B \cap A,$$

а также – сочетательный закон

$$(A \cap B) \cap C = A \cap (B \cap C).$$

Последнее равенство позволяет в многочленных теоретико-множественных произведениях опускать скобки, например, можно писать $A \cap B \cap C$.

Множество D всех элементов, каждый из которых принадлежит хотя бы одному из множеств A и B , называется теоретико-множественной суммой или объединением множеств A и B . При этом пишут $D = A \cup B$.

Знак \cup называется знаком теоретико-множественного сложения. Операция, дающая $A \cup B$ по заданным A и B , называется теоретико-множественным сложением. Для нее справедливы перестановочный закон

$$A \cup B = B \cup A$$

и сочетательный закон

$$(A \cup B) \cup C = A \cup (B \cup C).$$

Последнее равенство позволяет в многочленных теоретико-множественных суммах опускать скобки. Например, можно писать $A \cup B \cup C$.

В теории множеств существует два распределительных закона: умножения относительно сложения

$$A \cap (B \cup C) = (A \cap B) \cup (A \cap C)$$

и сложения относительно умножения

$$A \cup (B \cap C) = (A \cup B) \cap (A \cup C).$$

Множество E тех элементов, принадлежащих A , которые не принадлежат B , называется теоретико-множественной разностью A и B . При этом пишут $E = A \setminus B$.

Операция, позволяющая получить $A \setminus B$ по заданным A и B , называется теоретико-множественным вычитанием.

Отметим особенности этой операции:

- 1) если $A \cap B = \emptyset$, то $A \setminus B = A$;
- 2) вообще же всегда $A \setminus B = A \setminus (A \cap B)$.

Образуем все возможные пары, в которых первая компонента является элементом множества A , а вторая – элементом множества B . Множество этих пар называется декартовым или геометрическим произведением множеств A и B и обозначается $A \times B$.

Очевидно, $A \times B \neq B \times A$, т.е. для геометрического произведения переместительный закон не имеет места. Однако сочетательный закон

$$(A \times D) \times C = A \times (D \times C)$$

для него справедлив.

Пары, являющиеся элементами множества $A \times B$, будем обозначать (a, b) , где $a \in A$, $b \in B$. При этом будем иметь в виду, что скобки служат только для обозначения того факта, что a и b рассматриваются в совокупности. Образуя декартово произведение множеств $A \times B$ и C , его элементами будем считать (если $c \in C$) пары вида

$$((a, b), c) = (a, b, c),$$

т.е. тройки, образованные в определенном порядке из элементов множеств

А, В и С. Если число множеств сомножителей равно n , то элементами их декартова произведения будут кортежи (упорядоченные наборы) по n элементов.

В силу сочетательного закона в многочленных декартовых произведениях скобки можно опускать, т.е., например, можно писать $A \times B \times C$.

Если члены декартова произведения между собой равны, то декартово произведение называется декартовой степенью и обозначается

$$A \times A \times \dots \times A = A^n.$$

Элементами n -й декартовой степени множества A являются всевозможные кортежи из n элементов множества A .

Пусть существует однозначная функция $f\{x\}$ одного переменного, определенная для каждого значения x , являющегося элементом некоторого множества A . Когда x пробегает множество A , каждое ее значение принадлежит некоторому множеству B , элементы которого удовлетворяют условию $f(x) \in B$, если $x \in A$.

Множество B называется образом (или точнее f -образом) множества A .

Пусть теперь $P(x)$ – одноместный предикат, предметной областью которого является множество A . Т.е. элементы множества A , для которых указанный предикат принимает значение истины, образуют подмножество множества A . Получение подмножества с помощью предиката называется выделением подмножества (предикатом P).

Два множества A и B , называются равномошными, если элементам одного из них можно поставить во взаимно однозначное соответствие элементы второго.

Определение. Множество называется бесконечным, если оно равномошно правильной своей части. В противном случае оно называется конечным.

Легко видеть, что множество N натуральных чисел бесконечно, ибо множество M квадратов натуральных чисел есть его подмножество (его выделяет предикат " x есть квадрат натурального числа"), и эти множества равномошны. Взаимно однозначное соответствие между элементами этих множеств можно задать в виде функции $y=x^2$, где $x \in N$, $y \in M$.

Обозначим через N_n подмножество, выделяемое из множества N предикатом $x \leq n$. Множество N_n конечно, так как между элементами этого множества и элементами какой-либо его правильной части установить взаимно однозначное соответствие нельзя.

Мощностью множества N_n , а также мощностью любого равномошного ему множества, называется натуральное число n . Мощность конечного множества равна числу его элементов.

Множество N натуральных чисел не равномошно ни одному конечному множеству. Множество натуральных чисел называют счетным, и его мощность обозначают символом \aleph (алеф). Любое множество, равномошное N , также называют счетным.

Если конечное множество, хотя бы в принципе, можно задать в виде перечня его элементов, то бесконечное множество задать так невозможно. Даже натуральный ряд задать так невозможно, так как запись 1, 2, 3, ... перечисляет не все натуральные числа, а лишь некоторые.

Однако счетное множество можно задать в виде совокупности некоторого конечного множества элементов, называемых базой, и конечного множества функций (операций), называемого порождающей системой (функций или операций). При этом порождаемое множество M определяют так: если значениями аргументов порождающей функции являются элементы базы или множества M , то значение порождающей функции является элементом множества M . Никаких других элементов M не содержит.

Пример. Базой множества натуральных чисел можно считать одноэлементное множество, содержащее число 0, а системой порождающих функций – одноэлементное множество, содержащее функцию $\lambda(x)=x+1$. Описанием множества натуральных чисел будет теперь конечная строка: 0; $\lambda(x)$.

Если A – одноэлементное множество и $a \in A$, то пишут $A=\{a\}$.

Если A состоит из конечного числа элементов a_1, a_2, \dots, a_n , то пишут $A=\{a_1, a_2, \dots, a_n\}=\{a_i\}_{i=1}^n$.

Если x – некоторое собирательное (групповое) имя, применимое к каждому из элементов множества A , причем A является множеством всех x , то часто пишут $A=\{x\}$.

Если B является подмножеством, выделенным с помощью предиката $P(x)$ из этого множества A , то пишут $B=\{x|P(x)\}$.

Если B получено из $A = \{x\}$ с помощью функции $f(x)$ и является образом множества A , то пишут $B=\{f(x)|x \in A\}$.

Аналогичная запись возможна и при порождении множества с помощью системы функций. Например, множество натуральных чисел можно описать так: $N=\{\lambda(y)|y = 0 \vee y \in N\}$.

2.5. Элементы алгебры отношений (реляционной алгебры)

Понятие отношения

Между предметами реального мира существуют связи, обычно называемые отношениями. Например, человеческое общество можно считать разбитым на группы людей семьи, между членами которых существуют определенные правовые, социальные и моральные связи. Определенные связи, уже другого вида, существуют между подразделениями учреждений. Определенные связи (отношения) существуют и между частями механизма.

Описание каждой области связанных между собой предметов требует не только перечисления предметов, но и описания связей между ними.

Теория множеств дает некоторые возможности такого описания.

Пусть A_1, A_2, \dots, A_n – множества, не обязательно различные. Рассмотрим их декартово произведение

$$D = A_1 \times A_2 \times \dots \times A_n.$$

Предположим, что a_1, a_2, \dots, a_n – элементы соответственно множеств A_1, A_2, \dots, A_n . Кортеж (a_1, a_2, \dots, a_n) является элементом множества D . Предположим теперь, что F является подмножеством множества D , т.е. $F \subseteq D$. Если вышеуказанный кортеж удовлетворяет условию

$$(a_1, a_2, \dots, a_n) \in F,$$

то говорят, что его элементы находятся в отношении F и пишут $(a_1, a_2, \dots, a_n; F)$ или $F(a_1, a_2, \dots, a_n)$. Отметим, что при этом символом F называют и отношение и множество (кортежей, члены которых связаны отношением). Последнее множество иногда называют графиком отношения или множеством-отношением. Иногда вместо слов “находятся в отношении F ” говорят “удовлетворяют условию F ”.

Условимся два символа, обозначающие один и тот же элемент рассматриваемого множества, называть равными. Если символы x и y равны, то будем писать $x=y$.

Определение. Отношение F называется функциональным относительно i -го аргумента, если для любых двух наборов его графика

$$(a_1, a_2, \dots, a_n) \text{ и } (a'_1, a'_2, \dots, a'_n)$$

из того, что $a_1 = a'_1, a_2 = a'_2, \dots, a_n = a'_n$, следует, что $a_i = a'_i$.

Функциональное отношение задает некоторую функцию от $n-1$ переменных, значения которой можно получать так. Если задан набор значений исходных данных x_1, x_2, \dots, x_n , то находят кортеж данного отношения, имеющий вид

$$(x_1, x_2, \dots, x_{n-1})$$

и значением функции считают a .

В частном случае может оказаться, что $A_1 = A_2 = \dots = A_n = A$. В этом случае $D = A^n$; отношение F , имеющее график F , удовлетворяющий условию $F \subseteq D$, называют заданным на множестве A . Если F – функциональное отношение, то говорят, что оно определяет некоторую функцию от $(n-1)$ переменных на множестве A .

Пример. Предположим, что

$$A_1 = A_2 = A_3 = N = \{1, 2, 3, 4, 5, \dots\}.$$

В данном случае множество предметов бесконечно. Будет бесконечно и множество $D = A_1 \times A_2 \times A_3 = N^3$; его элементами являются всевозможные тройки целых чисел. Иногда удается указать признак, по которому можно определить принадлежность тройки натуральных чисел множеству F . Этот признак определяет F , а значит, может даже служить его названием. Допустим, что в данном случае такой признак гласит: “третий элемент кортежа равен сумме первого его элемента и квадрата второго его элемента”. Тогда тройка $(1, 2, 5)$ принадлежит множеству F , а тройка $(1, 2, 6)$ – не принадлежит. Значит, числа 1, 2, 5 удовлетворяют, а числа 1, 2, 6 – не удовлетворяют указанному отношению. Это отношение является функциональным и соответствует функции $z = x + y^2$.

Число элементов в кортежах множества-отношения называется **рангом отношения**. Иногда отношения n -го ранга называют n -местными или n -арными. На практике часто используются отношения 2-го ранга (бинарные, двухместные). В последних двух примерах приведены отношения 3-го ранга (трехместные, тернарные).

Иногда удобно пользоваться понятием отношения 1-го ранга (унарного, одноместного). Отношению первого ранга соответствует некоторое подмножество множества предметов, так как $A^1 = A$.

Остановимся подробнее на бинарных отношениях $\varphi \subseteq A \times A$, заданных на множестве A . Такие отношения часто обозначают $a_1 \varphi a_2$;

вместо (a_1, a_2, φ) . Отношение φ называют:

- 1) рефлексивным, если для любого $a \in A$ справедливо $a \varphi a$;
- 2) симметричным, если из $a_1 \varphi a_2$, следует $a_2 \varphi a_1$;
- 3) антисимметричным, если из $a_1 \varphi a_2$ следует невозможность $a_2 \varphi a_1$;
- 4) транзитивным, если из $a_1 \varphi a_2$ и $a_2 \varphi a_3$ следует $a_1 \varphi a_3$.

Про всякое бинарное отношение, заданное на A , говорят, что оно имеет тип эквивалентности, если оно рефлексивно, симметрично и транзитивно. Примером отношения типа эквивалентности может служить отношение равенства (чисел), заданное на каком-либо числовом множестве.

Рассмотрим теперь бинарное отношение, которое обладает тем свойством, что для любых $a_1 \in A, a_2 \in A$ имеет место либо $a_1 \varphi a_2$, либо $a_2 \varphi a_1$, но не сразу оба эти условия. Если, кроме того, рассматриваемое бинарное отношение не рефлексивно ни для одного $a \in A$, антисимметрично и транзитивно, то говорят, что φ имеет тип строгого порядка. Известное из математики отношение $<$ между числами имеет тип строгого порядка.

Множество A , на котором задано отношение типа строгого порядка, называется (строго) упорядоченным (этим отношением).

Если бинарное отношение φ , заданное на A , таково, что для любого $a \in A$, кроме может быть некоторого $a=a_0$, всегда существует $a \in A$ такое, что $a' \varphi a$ и, кроме того φ не рефлексивно ни для одного $a \in A$, антисимметрично и транзитивно, то отношение φ называется отношением типа древовидного порядка. Отношение типа древовидного порядка иллюстрируется генеалогиями “по мужской линии” царствовавших династий. Вместо термина “древовидный порядок” часто применяют “иерархический порядок”.

В дальнейшем под словами “порядок” и “иерархия” всегда будем понимать соответственно строгий порядок и иерархический (древовидный) порядок.

Для строгого и иерархического порядков, если $a \varphi a'$, то говорят, что a' следует за a . Если, кроме того, нет такого a'' , что $a \varphi a''$ и $a'' \varphi a'$, то говорят, что a' непосредственно следует за a . При строгом порядке, за каждым $a \in A$, кроме может быть только одного, непосредственно следует один и только один элемент. При иерархическом порядке есть элементы, не имеющие непосредственно следующих, но могут быть и имеющие несколько непосредственно следующих.

Операции над отношениями

Прежде всего рассмотрим операции над отношениями одинакового ранга. Пусть F и Φ – отношения ранга n между элементами из A_1, A_2, \dots, A_n . Те же буквы пусть обозначают соответствующие отношениям множества кортежей.

Теоретико-множественная сумма $F \cup \Phi$ определяет новое множество и тем самым новое отношение, которое обозначают $F \cup \Phi$ и читают “ F или Φ ”. Это отношение называется **суммой отношений** F и Φ .

Пересечение множеств $F \cap \Phi$ определяет новое множество кортежей n -го порядка и тем самым новое отношение, которое называют **произведением отношений** F и Φ , обозначают $F \cap \Phi$ и читают “ F и Φ ”.

С помощью теоретико-множественного вычитания множеств $F \setminus \Phi$ получают новое множество кортежей n -го порядка и, следовательно, новое отношение, которое называют разностью отношений F и Φ , обозначают $F \setminus \Phi$ и читают “ F , но не Φ ”. Если F является универсальным отношением, т.е. если множество F содержит все мыслимые кортежи n -го порядка, образованные из элементов множеств A_1, A_2, \dots, A_n , т.е. если $F=D$, то результат операции вычитания отношений обозначают Φ и читают “не Φ ”.

Операции над отношениями, описанные ниже, являются преобразованиями кортежей, входящих в состав множеств-отношений. Пусть числа $1, 2, \dots, n$ – номера элементов внутри кортежа, а i и j – целые числа, причем $1 \leq i < j \leq n$. **Обменом позиций** i и j называется операция над отношением, заключающаяся в том, что в каждом кортеже множества-отношения меняются местами элементы с номерами i и j . Если исходным отношением было F , то результат описанной операции обозначают $(i \leftrightarrow j)F$.

Если в названии отношения F (или в его описании) фигурируют названия позиций, имеющих номера i и j , то в описании $(i \leftrightarrow j)F$ эти названия меняются местами.

Пример. Если описанием бинарного отношения F служит фраза “1-е число больше, чем 2-е”, то после выполнения операции обмена позициями получится отношение $(1 \leftrightarrow 2)F$, описанием которого может служить фраза “2-е число больше, чем 1-е”. Правда, существует и другая фраза, которая могла бы служить описанием нового отношения: “1-е число меньше, чем 2-е”, но чтобы получить это описание, нужно уже привлечь знание естественного языка.

Операция расширения отношения над множеством F заключается в том,

что к каждому кортежу его множества-отношения слева добавляют один и тот же элемент a . При этом номера позиций всех “старых” элементов каждого кортежа увеличиваются на 1 и вновь получаемое отношение имеет ранг $n+1$. Расширение отношения применяют, если нужно уравнивать ранги двух отношений. Кроме того, эта операция используется при так называемой композиции отношений, имеющих одинаковый ранг.

Операция “проекция отношения” называется также исключением позиции. Сущность ее заключается в том, что из всех кортежей множества-отношения F исключают элемент, занимающий позицию номер i . После этого ликвидируют повторы получившихся более коротких кортежей. В результате возникает новое множество кортежей и соответственно новое отношение, которое обозначают $(i)F$. Описанная операция может быть обобщена на случай нескольких одновременно выполняемых проекций (т.е. исключения сразу нескольких позиций). При этом для ее обозначения употребляют запись $(i, j, \dots, k)F$,

где в скобках перечислены номера исключенных позиций. Иногда удобнее указывать все позиции, кроме исключенных. Тогда применяют запись

$$[l, m, \dots, n]F.$$

В некоторых случаях бывает полезной операция над отношением ранга n , заключающаяся в том, что в каждый кортеж отношения включается дополнительный элемент (предположим для определенности, что он займет позицию с номером $n+1$), равный элементу, занимающему в этом кортеже позицию с номером j . Если исходное отношение обозначено F , то результат выполнения над ним операции удвоения j -й позиции обозначают D_jF .

Пример. Пусть дано

$$F = \{(1, 2), (3, 7), (9, 2), (11, 5)\}.$$

Тогда

$$D_1F = \{(1, 2, 1), (3, 7, 3), (9, 2, 9), (11, 5, 11)\}.$$

Отношение D_1F имеет ранг 3.

Свертка двух отношений, известна также под названием свертки де Моргана. Пусть даны два отношения

$$F \subseteq A_1 \times A_2 \times \dots \times A_n \times A \text{ и } \Phi \subseteq A \times A_{n+1} \times A_{n+2} \times \dots \times A_{n+m}.$$

Их сверткой называется новое отношение

$$\Theta = A_1 \times A_2 \times \dots \times A_{n+m},$$

кортежи которого получаются из кортежей исходных отношений следующим путем: отбирается пара кортежей из первого и второго отношений, имеющих соответственно одинаковые первый и последний элементы; эти элементы отбрасываются, а остатки объединяются в один кортеж.

Операцию свертки обозначают символом $*$. Эта операция ассоциативна, т.е.

$$(F*\Phi)*\Psi = F*(\Phi*\Psi),$$

но не коммутативна, так как возможно, что

$$F*\Phi \neq \Phi*F.$$

Пример. Если

$$F = \{(a, 1), (a, 2), (a, 3), (b, 1), (b, 2), (v, 1), (v, 4)\}, \\ \Phi = \{(1, X, A), (1, Y, B), (2, Z, A), (3, X, B)\}.$$

Первый кортеж множества F образует с кортежами из Φ следующие сочетания:

$$\begin{aligned}(a, 1) &- (1, X, A), \\ (a, 1) &- (1, Y, B),\end{aligned}$$

из которых получатся кортежи свертки

$$(a, X, A), (a, Y, B).$$

Второй кортеж множества F даст только одно сочетание

$$(a, 2) - (2, Z, A),$$

из которого получится новый кортеж (a, Z, A) . Третий кортеж множества F опять сочетается с одним кортежем из Φ :

$$(a, 3) - (3, X, B).$$

Это сочетание даст кортеж (a, X, B) . Четвертый и шестой кортежи из F дадут сочетания

$$\begin{aligned}(\bar{b}, 1) &- (1, X, A), (\bar{b}, 1) - (1, X, A), \\ (\bar{b}, 1) &- (1, Y, B), (\bar{b}, 1) - (1, Y, B).\end{aligned}$$

из которых получим (\bar{b}, X, A) , (\bar{b}, Y, B) , (\bar{b}, X, A) , (\bar{b}, Y, B) .

Наконец, пятый кортеж множества F даст одно сочетание

$$(\bar{b}, 2) - (2, Z, A),$$

откуда получится новый кортеж (\bar{b}, Z, A) .

Седьмой кортеж из F не образует ни одного сочетания. Собирая новые кортежи, получим

$$F * \Phi = \{(a, X, A), (a, Y, B), (a, Z, A), (a, X, B), (\bar{b}, X, A), (\bar{b}, Y, B), (\bar{b}, X, A), (\bar{b}, Y, B), (\bar{b}, Z, A)\}.$$

Операция, называемая склеиванием двух отношений, отличается от свертки де Моргана тем, что при построении составных кортежей отбрасываются не оба одинаковых элемента, а лишь один. Обозначим операцию склеивания σ . Если

$$F \subseteq A_1 \times A_2 \times \dots \times A_n \times A_{n+1}$$

и

$$\Phi \subseteq A_{n+1} \times A_{n+2} \times \dots \times A_N,$$

то

$$F \sigma \Phi = (D_{n+1} F) * \Phi; F \sigma \Phi \subseteq A_1 \times A_2 \times \dots \times A_N.$$

Пусть F – n-арное отношение. Номера позиций элементов, образующих кортежи этого отношения, можно считать именами позиций и групповыми именами элементов, занимающих одну и ту же позицию в разных кортежах. Перечень имен всех позиций данного отношения будем называть списком позиций. Строку, которая получится, если из списка позиций удалить какое-либо число (в частности – нуль) имен позиций, но не все, назовем подсписком (позиций) данного отношения.

О каждом элементе какого-либо кортежа, принадлежащего отношению, говорят, что он поименован в подписке S, если имя занимаемой им позиции содержится в этом подписке.

Пусть F, S_1 , S_2 – произвольное отношение и два его подписка. Говорят,

что S_2 функционально зависит от S_1 , если для любых двух кортежей, принадлежащих отношению, из попарного равенства одноименных элементов, поименованных в S_1 , следует попарное равенство одноименных элементов, поименованных в S_2 .

Понятие функциональной зависимости между подписками является обобщением понятия функциональности отношения.

Если в отношении F подписок S_2 функционально зависим от подписка S_1 , то обратное – функциональная зависимость подписка S_1 от подписка S_2 – может быть, а может и не быть. При наличии как прямой, так и обратной функциональной зависимости между двумя подписками говорят, что каждый из них взаимно функционально зависит от другого.

Ключом отношения F называется его подписок, от которого функционально зависит полный список, и который не содержит в качестве своей правильной части другого подписка, от которого функционально зависит полный список.

Отметим некоторые свойства ключа произвольного отношения F :

1) подкортеж элементов отношения, соответствующий ключу, однозначно определяет один единственный кортеж отношения F , иногда говорят, что ключ идентифицирует кортежи отношения;

2) отношение может иметь несколько ключей;

3) всякое отношение имеет по крайней мере один ключ; это вытекает из того, что если отношение не имеет ключа, являющегося правильным подписком, то ключом оказывается весь список позиций отношения;

4) любые два ключа отношения F взаимно функционально зависимы.

Если отношение F представлено в виде множества своих (неповторяющихся) кортежей, то говорят, что оно представлено в первой нормальной форме.

Если отношение представлено в первой нормальной форме и любая его позиция, не входящая ни в один ключ, функционально полно зависит от каждого ключа, то говорят, что отношение имеет вторую нормальную форму.

Очевидно отношение, все ключи которого являются однопозиционными, имеет вторую нормальную форму.

Наконец, если отношение имеет вторую нормальную форму и каждая его позиция, не принадлежащая ни одному ключу, нетранзитивно зависит от каждого ключа, то говорят, что отношение имеет третью нормальную форму. Переход к нормальной форме высшего порядка, если отношение еще не имеет такой формы, можно получить путем расщепления отношения на несколько отношений с помощью соответствующим образом выбранных операций проекции.

Рассмотрим далее понятие **реляционная алгебра**. Предположим, что задано конечное или счетное множество конечных или счетных множеств

$$C = \{A', A'', \dots\}.$$

Задавая число n , мы можем выбрать из множества C всевозможными способами некоторое количество (не превосходящее числа n) входящих в него множеств и образовать из них последовательность A_1, A_2, \dots, A_n , в которой могут быть и равные множества. Опираясь на такие последовательности множеств, можно построить конечное или счетное множество n -арных отношений. Заставляя n пробегать значения 1, 2, 3, мы можем получить счетное множество конечных или счетных множеств различных отношений, всех возможных “арностей”, т.е. опять счетное множество. Обозначим совокупность всех полученных отношений через R . На множестве R определены операции (над отношениями), результаты которых тоже принадлежат R . Эти операции описаны выше.

В математической дисциплине, называемой абстрактной алгеброй, совокупность, образованная некоторым множеством объектов и множеством заданных на нем операций (не выводящих из него), называется алгеброй (т.е. термин “алгебра” применяется как для наименования математической дисциплины, так и для наименования одного из изучаемых ею предметов; этот термин является омонимом). В нашем случае совокупность множества R отношений и множества операций, описанных выше, называется реляционной алгеброй.

3. ИНФОРМАЦИОННЫЕ И ЛИНГВИСТИЧЕСКИЕ ОСНОВЫ АВТОМАТИЗИРОВАННЫХ ИНФОРМАЦИОННЫХ СИСТЕМ

3.1. Символьные конструкции

Символьные конструкции являются средством представления информации, ее носителями – совокупностями дискретных сигналов. В математике они представляют так называемое конструктивное направление. Теория символьных конструкций является начальным разделом теории формальных языков и теории алгоритмов и представляет собой раздел **лингвистики** - науки о языке, общих законах строения и функционирования языка. Прежде чем перейти к изложению символьных конструкций, определим основные понятия лингвистики, являющиеся основой данного раздела:

язык – форма существования знания в виде системы знаков плюс правила функционирования этих знаков, служащая средством человеческого общения, мышления и выражения;

общение – двусторонний процесс передачи информации (с определенными целями и по определенным правилам), выраженной на языке, понятном участникам общения;

информационный символ – символ сообщения (записи), который является частью его содержания, в отличие от служебных (управляющих, разделителей) символов;

знак – способ обозначения определенного понятия, предмета, свойства и используемый для приобретения, хранения, обработки и передачи информации;

языковая знаковая система – знаки, которые не функционируют независимо друг от друга, а образуют систему, правила которой определяют закономерности их построения, осмысления и употребления (грамматика, правила смысла);

синтактика – изучение структурных аспектов сочетаний знаков данной системы, правила их образования и преобразования безотносительно к их значениям и функциям;

прагматика – изучение отношения, воспринимающего знаковую систему (интерпретатор или адресат) к самой знаковой системе;

денотат (референт) – предмет, обозначаемый знаком;

концепт – информация, которую знак несет о возможных денотатах, об их положении в системе реальных, об их месте в универсуме;

слово – законченная последовательность знаков определенной длины, воспринимаемая как элемент обработки с определенным семантическим содержанием;

слог - часть слова, допускающая независимое обращение и обработку;

словосочетание – смысловое и грамматическое объединение нескольких значимых слов;

предложение – базовая единица языка, обладающая определенной для данного языка синтаксической и смысловой законченностью.

дискурс (связный текст) – два или более предложений, находящиеся друг с другом в смысловой связи.

Основными “атомами”, из которых строятся символьные конструкции, являются так называемые буквы. Буквы связываются между собой так называемыми связями; получаемые при этом конструкции могут быть заключены в оболочки, после чего могут быть использованы наряду с буквами для построения еще более сложных конструкций и т.д.

Понятия буквы и связи являются первичными и не могут быть выражены через другие математические понятия. Их можно достаточно точно описать. Такие описания представляют собой содержательные аксиомы. Понятие оболочки уже можно выразить через понятие букв и связей.

Символьные конструкции всегда применяются в некоторой “области применения”.

Буквы и связи неизменны и неделимы в области их применения. Буквы всегда являются символами, а связи иногда передаются в виде символов, а иногда другими способами, например в виде специального взаимного расположения букв. При желании всегда можно перейти к символьному изображению связей; в дальнейшем связи считаются символами, которые в некоторых случаях присутствуют неявно, т.е. подразумеваются.

В отношении всякой буквы (соответственно связи) всегда известно, что это буква (связь). Отсюда вытекает, что любая буква отличается от любой связи.

В отношении любых двух букв (любых двух связей) всегда известно, одинаковы они или различны. Буквы существуют сами по себе и могут быть связуемыми объектами, а связи существуют только вместе со связуемыми ими объектами. Связи их связывают, объединяют, сцепляют, “скрепляют”, устанавливают между ними определенные отношения.

Каждая буква может быть связана любым конечным количеством связей. Каждая связь может связывать лишь строго определенное количество конструктивных элементов, число которых называется **рангом связи**.

Связь можно для наглядности представить себе графически в виде кружка, внутри которого записано имя связи; из кружка исходят r ребер (линий), каждое из которых помечено целым положительным числом (названием жанра ветви), не превосходящим r (ранга связи). Помеченные ребра будем называть ветвями связи, а пометку каждого из них – жанром соответствующей ветви. Ветви связи изображают ее способности связывать конструктивные элементы. Имя связи называют также ее типом. Полным описанием связи является последовательность

$$t, a_1(n_1), a_2(n_2), \dots, a_k(n_k),$$

называемая ее характеристикой, в которой t – тип связи, k – жанр связи, a_i – название жанра ветви (целое число), n_i – число ветвей жанра a_i . При этом $n_1 + n_2 + \dots + n_k = r$, где r – ранг связи. Характеристики связей одного и того же типа должны быть одинаковы.

Все одинаковые буквы считаются буквами одного и того же типа. Каждая область применения допускает лишь конечное число типов букв и типов связей.

Простейшими связями являются так называемые связи следования букв. Существует три типа таких связей: начинающая связь $2(1)$, т.е. имеющая ранг 1 и один жанр ветви, имя которого 2-й, продолжающая связь $1(1)$, $2(1)$, т.е. имеющая ранг 2, один жанр 1-й и один жанр 2-й ветвей, и заканчивающая связь $1(1)$, т.е. имеющая ранг 1 и один жанр ветви, имя которого 1-й. Введем для обозначения этих связей соответственно значки

$$\bullet \rightarrow, \rightarrow, \dots \rightarrow \bullet$$

Пример: Буквы русского алфавита, строчные и заглавные, являются буквами в смысле данной теории. Однако знаки препинания тоже являются буквами, и арабские цифры – тоже. Следовательно, понятие буквы в теории символьных конструкций не совпадает с обиходным понятием буквы.

Пример: В русской письменности связи следования букв передаются специальным расположением букв. Считается, что начинающая связь (неявно) присутствует слева от самой первой (левой) буквы слова; продолжающая связь помещается между каждыми двумя буквами, стоящими рядом; заканчивающая связь размещена справа от последней буквы слова. Например, в слове

“аппарат”

связи расположены так:

•→а→п→п→а→р→а→т→•

Пример. В известной двоично-десятичной системе счисления двоичные коды 0000, 0001, 0010, 0011, 0100, 0101, 0110, 0111, 1000, 1001 являются буквами. Они соответствуют арабским цифрам 0, 1, 2, 3, 4, 5, 6, 7, 8, 9. В двоично-десятичной системе счисления они являются целыми и неизменными.

Выберем некоторую букву (тип), фиксированную в данной области применения, и будем применять ее только как группировочную. Обозначим ее условно S. Кроме того, выберем некоторый тип связи. Условно обозначим его σ. Пусть это будет связь 2-го ранга и 2-го жанра. Условимся группировочную букву связывать только ветвями первого жанра группировочных связей, а группируемые конструктивные элементы – только ветвями второго жанра этих связей. **Оболочкой** называется либо сама буква S, если в оболочке не содержится ни одного конструктивного элемента, либо совокупность буквы S и всех связей типа σ, связывающих ее своими ветвями первого жанра. Условно оболочку изображают в виде замкнутой линии, внутри которой расположены заключенные в нее (т.е. сгруппированные ею) конструктивные элементы. В дальнейшем, в каждой области применения используется только один тип группировочной буквы и, следовательно, только один тип оболочек. Группировка всегда будет производиться так, что оболочки будут “непересекающимися”.

Определение конструкции является рекурсивным. Для **рекурсивных определений** характерно, что в них содержатся прямая часть, задающая некоторые частные случаи, и циклическая часть, которая по уже определенным частным случаям строит новые частные случаи. Путем применения циклической части к частным случаям, которые были уже получены с ее помощью, можно получить любой возможный частный случай и потенциально исчерпать определяемое понятие. Определению конструкции предположим некоторые вспомогательные определения.

Определение 1. Конструктивный элемент является либо отдельной буквой, либо конструкцией, заключенной в оболочку.

Замечание. После определения конструкции, которое будет дано несколько ниже, возникает представление о процессе построения из нескольких конструктивных элементов нового конструктивного элемента, который в свою очередь может быть применен для аналогичного построения и т. д. Получается иерархия конструктивных элементов. Для каждого из них все конструктивные элементы, которые были применены на различных этапах его построения, называются внутренними. Если x и y – два конструктивных элемента и x – либо внутренний для y, либо совпадает с y, то говорят, что x является внутренним, для y.

Определение 2. Если x и y – два конструктивных элемента, для которых u и v являются соответственно невнешними, и если u и v связаны некоторой связью, то x и y называются непосредственно соединенными (указанной связью).

Определение 3. Два конструктивных элемента называются соединенными (с помощью связей), если они либо непосредственно соединены, либо один из них соединен с конструктивным элементом, который непосредственно соединен с другим из них.

Определение 4. Предположим, что задано несколько конструктивных элементов. Связь s **насыщена**, если каждой ее ветви сопоставлен конструктивный элемент из числа заданных или внутренних для заданных.

Определение 5. (определение конструкции). **Конструкцией** называется совокупность конструктивных элементов и связей, которая либо является пустой, либо содержит не менее одного конструктивного элемента и тогда удовлетворяет условию: любая связь в ней насыщена, либо содержит не менее двух конструктивных элементов и тогда, кроме того, любые два конструктивных элемента в ней соединены.

Определение 6. Конструкцией, заключенной в оболочку, называется либо группировочная буква S , если конструкция пустая, либо результат группировки оболочкой всех конструктивных элементов конструкции, невнутренних друг для друга.

Определение 7. Если конструктивный элемент является конструкцией, заключенной в оболочку, то связывание его какой-либо связью представляет собой связывание группированной буквы указанной оболочкой.

Замечание. **Полным определением конструкции** является совокупность семи приведенных выше частных определений.

Наиболее простой (после отдельной буквы) и практически важной конструкцией является так называемое слово. При образовании слов используются буквы и связи следования.

Определение. **Словом** называется конструкция, состоящая из букв и связей следования, или пустая конструкция. Непустое слово содержит ровно одну начинающую и одну заканчивающую связь и может содержать продолжающие связи. При этом каждая буква слова связана ровно двумя связями следования и ветви связей, соответствующие одной и той же букве, принадлежат различным жанрам. Длиной слова называется число его букв; длина пустого слова равна нулю.

Пример. Пустое слово обведено рамочкой; кроме него приводим два непустые слова:

•→и→•

•→с→у→м→м→а→•

Длины приведенных слов равны соответственно 0; 1; 5. В областях применения, в которых никакие связи кроме связей следования не используются, эти связи передают путем особого расположения букв. При этом второе и третье слова будут иметь вид

и, сумма.

Знаки препинания проставлены ради соблюдения традиционных правил грамматики русского языка и в состав описанных слов не входят.

В каждой области применения допускают лишь конечное число видов букв и видов связей. Допустимые типы букв и типы связей задают с помощью так называемых алфавитов.

Алфавит букв представляет собой слово, не содержащее группировочной буквы, все буквы которого попарно различны.

Пример. Дано слово “кот”. Оно состоит из попарно различных букв, но является ли оно алфавитом, по его виду ответить невозможно. Если оно

служит для указания допустимых в нашей области применения букв, то оно является алфавитом. В остальных случаях – не является. Понятие алфавита связано с понятием области применения.

Алфавит связей представляет собой слово, образованное из связей как из букв (или из символов, являющихся обозначениями связей), все буквы которого попарно различны. Условимся группировочные буквы и связи не включать в эти алфавиты.

Кроме алфавита букв и алфавита связей может потребоваться и алфавит оболочек. Он может быть пустым, если оболочки не применяются, или иметь длину 1, так как мы условились считать все оболочки принадлежащими одному и тому же типу.

Определение. Если буква (связь) одинакова с одной из букв алфавита букв A (с одной из связей, перечисленных в алфавите связей B), то она называется буквой в A (связью в B).

Пример. Рассмотрим два алфавита букв:

$$A_1 = 123456789 \text{ и } A_2 = +2 -13456879.$$

Буква “+” является буквой в A_2 и не является буквой в A_1 .

Алфавиты, являясь словами, если они не пусты, задают не только перечень допустимых букв или связей, но и определенный порядок между ними, называемый алфавитным. Это свойство алфавитов существенно используется в теории алгоритмов.

Класс конструкций, допустимых в данной области применения, задается как совокупность (A, B, Σ) или, если Σ является пустым, – как (A, B) , где A – алфавит букв, B – алфавит связей и Σ – алфавит оболочек. Конструкция считается принадлежащей соответствующему классу, если в ней применяются только буквы в A , связи в B , если Σ – непустой алфавит, может быть, оболочки. В течение долгого времени из всех символьных конструкций рассматривались только слова, причем связи следования подразумевались. Для выделения класса слов задавали один только алфавит букв. Этот прием остается в силе и теперь. Слово, все буквы которого являются буквами в A , называется словом в A . Заметим, что класс (A, B) является подклассом для класса (A, B, Σ) .

Если два алфавита не содержат одинаковых символов, они называются **непересекающимися**. При этом говорят, что их пересечение пусто. В противном случае они называются пересекающимися.

Пересечением алфавитов A и B называется третий алфавит, который получится, если, просматривая A в порядке его букв, вычеркнуть все буквы, не являющиеся буквами в B .

Объединением алфавитов A и B называется третий алфавит, который получится, если к алфавиту A приписать те буквы в B , которые не являются буквами в A , просматривая для их выявления алфавит B в порядке следования его букв.

Пересечение алфавитов A и B будем обозначать $A \cap B$, а их объединение $A \cup B$. Эти действия над алфавитами не коммутативны, т.е. зависят от порядка, в котором рассматриваются алфавиты.

Пример. Даны алфавиты букв

$$A = a\ 1\ 2\ б\ в, B = a\ 2\ 1\ г, C = 4\ г\ 5.$$

Тогда

$$A \cap B = a\ 1\ 2, B \cap A = a\ 2\ 1, \\ A \cup B = a\ 1\ 2\ б\ в\ г, B \cup A = a\ 2\ 1\ г\ б\ в.$$

Алфавиты A и C не пересекаются. При этом пишут

$$A \cap C = \emptyset.$$

Все сказанное об алфавитах справедливо как для алфавитов букв, так и для алфавитов связей.

Замечание. Алфавит связей предполагает знание всех характеристик связей.

3.2. Формальные языки и грамматики

Формальный язык – совокупность исходных знаков, принятых за неделимые, и правил построения из них слов и словосочетаний без всякой связи с их возможной семантикой, а **теория формальных языков** – наука о формальных языках, в частности о структуре и способах представления бесконечных классов формальных языков.

Формальные языки являются математическими объектами, в определенном смысле аналогичными естественным языкам. Для формальных языков сохраняются такие термины, как грамматика и предложение. Символьную конструкцию, построенную по правилам грамматики (формального языка), называют продвижением этого формального языка. Формальные языки, в конечном счете, нужны нам как надежные средства представления информации, допускающие автоматическую ее переработку без участия человека, понимающего смысл предложений. Такая переработка информации возможна только путем переработки символьных конструкций, являющихся предложениями. Следовательно, смысл предложений должен однозначно определяться структурой этих предложений и структурой предложений, запасенных в системе и привлекаемых в процессе переработки. В конечном счете, смысл должен определяться формой. Отсюда происходит термин “формальный язык”.

Вводя понятие формального языка, принимают ряд предосторожностей. Прежде всего условимся четко различать формальный язык, определение которого производится, и тот язык, который при этом применяется как средство. Определяемый язык будем называть **языком-объектом**, а язык, уже нам известный, с помощью которого производится определение языка-объекта, назовем **метаязыком**.

Вторая предосторожность будет заключаться в том, чтобы правила образования предложений формального языка не зависели от смысла частей, из которых предложения строятся. В естественных языках это свойство грамматики часто отсутствует. Например, два предложения:

- 1) “Я увидел белые высокие каменные дома”;
- 2) “Я увидел белые, желтые, красные дома”;

имеют разную структуру (обратите внимание на запятые), и это различие зависит от смысла прилагательных (однородные ли свойства они выражают).

Последнее требование, если мы обеспечим его выполнение, позволит строить предложения языка независимо от их смысла. Класс конструкций, являющихся предложениями, обычно называют формальным языком, а класс предложений, рассматриваемых вместе с соответствующим им “смыслом”, – **формальным языком, наделенным семантикой**. (При условии, конечно, что между формой предложений и их “смыслом” существует однозначное соответствие).

В принципе, при построении формальных языков можно пользоваться самыми различными метаязыками. Но для того чтобы избавить себя от необходимости всевозможных проверок, целесообразно применять метаязыки, соответствующие некоторым уже изученным схемам. Такие схемы, известные под названиями формальных грамматик, соответственно дедуктивно и

индуктивно порождающих формальный язык. Каждый раз, когда для некоторого класса конструкций построена одна из таких грамматик, доказано, что этот класс конструкций является формальным языком. Отсюда **грамматика (формальная)** – один из основных подходов к описанию бесконечного формального языка конечными средствами.

Важным классом формальных грамматик является дедуктивная **порождающая грамматика** – формальная грамматика, позволяющая построить любую правильную цепочку символов.

Такая грамматика задана, если указаны класс (A, B, Σ) конструкций, которые могут быть предложениями языка объекта, алфавит N так называемых нетерминальных букв, в котором выделена некоторая буква a , и конечный перечень P так называемых синтаксических правил. Алфавиты букв A и N должны быть непересекающимися;

Каждое синтаксическое правило является названием некоторой операции первого ранга (одноместной), исходными данными и результатами которой являются конструкции класса $(A \cup N, B, \Sigma)$. Записями синтаксических правил могут быть конструкции, содержащие буквы и связи, не указанные в названных выше алфавитах и называемые собственными буквами и связями грамматики.

Итак, **дедуктивная порождающая грамматика** – это совокупность

$$D = ((A, B, \Sigma), N, P, a),$$

где a – буква в N .

Исходным элементом для порождения предложения формального языка является буква a . **Процесс порождения называется выводом** и состоит из следующих многократно выполняемых действий:

1) Если преобразуемая конструкция принадлежит классу (A, B, Σ) , то она является предложением определяемого формального языка и процесс окончен; в противном случае переходим к следующему пункту.

2) Выбираем из перечня P любое синтаксическое правило, которое применимо к преобразуемой конструкции; выполняем над преобразуемой конструкцией указанную в правиле операцию; полученный результат считаем преобразуемой конструкцией: переходим к п. 1.

Процесс вывода либо заканчивается при выполнении п. 1, либо никогда не заканчивается, либо обрывается в процессе выполнения п. 2. Искомый результат получается только в первом случае.

Если воспользоваться терминологией теории множеств, то можно сказать, что дедуктивно порождаемый формальный язык – это множество конструкций класса (A, B, Σ) , выводимых из начального символа a .

Важным частным случаем дедуктивных грамматик является случай, в котором допустимы лишь конструкции, являющиеся словами. При этом

$$D = (A, N, P, a),$$

где a – буква в N .

Предложения языка, порождаемого такой грамматикой, являются словами в A .

В еще более частном случае в качестве синтаксических правил допускаются только операции частного вида, называемые подстановками или продукциями; каждое правило при этом записывается в виде $p_i \rightarrow q_i$, где p_i и q_i являются словами в $A \cup N$. Выполнение таких правил заключается в том, что любой фрагмент преобразуемого слова, одинаковый с p_i , заменяется фрагментом q_i . Дедуктивные грамматики описанного вида известны в теории формальных языков как **грамматики продукций**.

Наиболее известным частным случаем грамматики продукций является так называемая **контекстно свободная грамматика (КС-грамматика)**. В

КС-грамматике все подстановки в левых частях содержат лишь однобуквенные слова, образованные из нетерминальных символов.

Индуктивные порождающие грамматики более напоминают грамматики естественных языков. Такая грамматика задана, если заданы:

класс (A, B, Σ) конструкций, которые могут быть предложениями языка объекта;

набор C конструкций этого класса, называемых морфемами;

вспомогательный алфавит N , не пересекающийся с A , одна из букв которого β выделена;

перечень P синтаксических правил.

Часть вспомогательных букв являются собственными именами морфем, другая часть, в том числе и β , групповыми (собирательными) именами символьных конструкций, получаемых в процессе построения предложений языка, третья часть – функциональными знаками. Синтаксические правила имеют либо вид

$$x_i = x_j^0,$$

либо вид

$$x_i = f_k(x_1, x_2, \dots),$$

где x_1, x_2, \dots, x_i – групповые имена, а x_j^0 – индивидуальные имена. Откуда **индуктивная порождающая грамматика** – это совокупность

$$J = ((A, B, \Sigma), C, N, P, \beta),$$

где β – буква в N .

Процесс построения предложения языка-объекта по правилам индуктивной порождающей грамматики называется построением предложения из морфем. Заключается он в следующем.

1. Выбирают некоторое количество формул, в правых частях которых стоят индивидуальные имена морфем. Считают, что левые части этих формул являются групповыми именами соответствующих символьных конструкций (морфем). Переходят к п. 2.

2. Из перечня P выбирают любую формулу второго вида, в которой все стоящие в правой части групповые имена уже получили некоторые значения. Выполняют операцию, которую обозначает правая часть формулы над значениями указанных групповых имен. Полученную символьную конструкцию считают (еще одним) значением группового имени, стоящего в левой части формулы. Переходят к п. 3.

3. Если левая часть последней выполненной формулы не есть β , то перейти к п. 2. В противном случае полученная конструкция является предложением языка-объекта. Процесс можно либо закончить, либо, если буква β встречается в правых частях формул, продолжить, перейдя к п. 2.

Замечание. Групповые имена, стоящие на разных местах в правой части записи общего вида $t = f(x, y, \dots, z)$, между собой независимы. Даже если они между собой одинаковы, при выполнении п. 2 процесса построения им можно приписывать неодинаковые значения.

Пользуясь терминологией теории множеств, можно сказать, что **индуктивно порождаемый формальный язык** – это множество тех символьных конструкций β (т.е. носящих групповое имя β), которые построены с помощью синтаксических правил грамматики из ее морфем.

Множество морфем иногда называют **базой грамматики или ее словарным фондом**.

Важный частный случай индуктивной порождающей грамматики получится, если в качестве морфем взять слова и в качестве операций, называемых в

синтаксических правилах, – операций, результатами которых являются слова. При этом грамматика представит собой следующую совокупность:

$$I = (A, C, N, P, \beta),$$

где β – буква в N.

В этой записи алфавиты V и Σ опущены (они подразумеваются). Такая грамматика определяет язык, предложениями которого являются слова.

Если в формулах перечня синтаксических правил индуктивной грамматики последнего вида в правые части вместо индивидуальных имен морфем подставить сами морфемы (а это возможно, так как морфемы являются словами), то будет получена так называемая система **универсальных метаформул**.

Система универсальных метаформул, если указать метасимвол, выделенный в индуктивной грамматике, содержит всю информацию, которую содержит сама грамматика.

На практике обычно знак равенства, разделяющий каждую метаформулу на две части, заменяют знаком $::=$, который читается “по определению есть”. Это связано с тем, что сам знак $=$ часто является одной из букв языка объекта. Кроме того, для уменьшения числа метаформул применяют следующий прием. Метаформулы, имеющие одинаковые левые части, объединяют в новые формулы, в которых левая часть присутствует один раз, а правые части объединяемых метаформул перечислены в правой части и разделены символами “|”, которые читаются как “или”.

Наиболее известным и часто применяемым случаем универсальных метаформул является так называемая **нотация Бекуса**. Мы придем к ней, если ограничимся только операциями соединения слов

$$S_1(P) = P, S_2(P_1, P_2) = P_1 P_2, \dots, S_i(P_1, P_2, \dots, P_i) = P_1 P_2 \dots P_i,$$

где P, P_i – слова, а результат соединения слов обозначен строкой метасимволов, обозначающих соединяемые слова.

Как видно из этого определения, результаты операций соединения слов можно получать из функциональных записей отбрасыванием функциональных знаков, скобок и запятых. Это позволяет в метаформулах отбросить функциональные знаки, скобки и запятые.

В полученных после этого формулах имена морфем, являющихся в данном случае словами, следует заменить самими морфемами.

Наконец, если в качестве метасимволов применять фразы естественного языка, заключенные в угловые скобки $< >$, то будет получена **запись формальной грамматики в виде нотации Бекуса**.

Фразы, заключенные в угловые скобки, – это метасимволы, вспомогательные буквы, а не какие-либо “понятия”. С этими метасимволами и обращаться нужно, как с буквами: в своей области применения они должны быть целыми и неизменными. Применение фразы в составе вспомогательной буквы – это прием, облегчающий запоминание роли, которую играет метасимвол в грамматике.

Пример. Опишем десятичную позиционную систему счисления как формальный язык, пользуясь для этого нотацией Бекуса:

$\langle \text{ненулевая цифра} \rangle ::= 1 \mid 2 \mid 3 \mid 4 \mid 5 \mid 6 \mid 7 \mid 8 \mid 9$

$\langle \text{цифра} \rangle ::= 0 \mid \langle \text{ненулевая цифра} \rangle$

$\langle \text{строка цифр} \rangle ::= \langle \text{ненулевая цифра} \rangle \mid \langle \text{строка цифр} \rangle \langle \text{цифра} \rangle$

$\langle \text{целое число} \rangle ::= \langle \text{цифра} \rangle \mid \langle \text{строка цифр} \rangle$

Можно было бы ограничиться только двумя формулами:

$\langle \text{цифра} \rangle ::= 0 \mid 1 \mid 2 \mid 3 \mid 4 \mid 5 \mid 6 \mid 7 \mid 8 \mid 9$

$\langle \text{целое число} \rangle ::= \langle \text{цифра} \rangle \mid \langle \text{целое число} \rangle \langle \text{цифра} \rangle$

Но в таком случае появлялись бы записи чисел с незначащими нулями слева. Например, вместе с числом 125 было бы возможным и число 0125, и 000125.

Заметим, что никаких данных о смысле получаемых построений нотация Бекуса дать не может.

Для теории программирования важными являются следующие понятия, базирующиеся на теории формальных языков и грамматик:

автомат – это абстрактная, кибернетическая машина, обрабатывающая входную последовательность и определяющая ее принадлежность некоторому формальному языку или выдающая некоторую выходную последовательность;

конечный автомат – это простая разновидность автомата, которая однократно считывает входную строку слева направо, при этом в любой момент времени конечный автомат находится в некотором внутреннем состоянии, меняющемся после считывания очередного символа;

регулярный язык – язык, распознаваемый конечным автоматом;

регулярная грамматика – грамматика, порождающая регулярные языки;

трансляция – преобразование исходной системы данных в другую рабочую систему данных с аналогичными отношениями;

интерпретация – реализация смысла некоторого синтаксически законченного текста, представленного на конкретном языке;

понятность – синтаксическая, семантическая и прагматическая однозначность информации, требующая общности языка и знаний участников о предметной области общения.

3.3. Элементы аналитической теории алгоритмов

Теория алгоритмов возникла в математической логике как вспомогательная дисциплина, предназначенная для внутренних потребностей математики. Основными задачами, которые стояли перед ней, были исследование феномена неразрешимости и обоснование математики.

Практические применения теории алгоритмов первоначально основывались на интуитивном понятии алгоритма, поскольку математическое понятие алгоритма было разработано только для некоторых частных случаев, далеких от того, с чем приходится иметь дело на практике. Аналитическая теория алгоритмов, изучающая все их многообразие, возникла лишь в последние годы.

Интуитивное понятие алгоритма рисует нам его как строго сформулированное правило, следуя которому можно осуществить процесс решения задачи, причем осуществить механически, максимально экономя усилия, т.е. **алгоритм** – конечный набор предписаний, определяющий решение задачи посредством конечного количества операций.

По А.А. Маркову, **алгоритм** – это предписание, ведущее от исходных данных к искомому результату и обладающее свойствами:

определенности (общепонятности и точности, не оставляющей места для произвола);

массовости (возможности применения ко всем процессам данного класса);

результативности (возможности получения правильного результата за конечное число шагов).

Такое представление об алгоритме еще очень неясно, но его нетрудно уточнить. Такое уточнение нужно делать, во-первых, конкретизируя сказанное только что и, во-вторых, анализируя классические теории алгоритмов. Ограничимся приведением результата уточнения без его обоснования.

Алгоритм задается как предложение формального языка (это – запись

алгоритма) и определяет дискретный процесс преобразования исходного данного, запись которого является тоже предложением (другого) формального языка, в искомый результат (тоже предложение формального языка). Понятность алгоритма заключается в том, что для его выполнения должен быть задан другой алгоритм, называемый алгоритмом выполнения (или интерпретации), для которого исходным данным является совокупность записи алгоритма и записи операнда (объединяющая их символьная конструкция). От особенностей алгоритма выполнения зависит процесс, определяемый алгоритмом. Для некоторых вариантов исходных данных этот процесс содержит конечное число шагов (это – **результативность алгоритма**), но возможны и такие исходные данные, для которых алгоритмический процесс безрезультативно обрывается или никогда не заканчивается. В первом случае говорят, что алгоритм применим к исходному данному, а в двух последних, что – неприменим.

Алгоритмический процесс состоит из шагов, каждый из которых представляет собой выполнение достаточно простых операций. Существует набор простейших операций, называемых натуральными, все же остальные операции – это отображения, индуцируемые алгоритмами. При определении нового класса алгоритмов достаточно простыми считаются операции, которые уже получены.

Такое понимание алгоритма неминуемо ведет к тому, что его определение принимает вид **рекурсивного определения** (новые алгоритмы строятся с помощью уже имеющихся). Процесс рекурсивного определения должен начинаться с определения каких-то алгоритмов, не опирающегося на понятие алгоритма, иначе все определение в целом являлось бы так называемым порочным кругом. В соответствии с вышесказанным широкое формальное определение алгоритма начинается с того, что вводится понятие первичных алгоритмов. Однако, поскольку понятие первичных алгоритмов связано с понятием операции, остановимся сперва на этом понятии.

Операциями называются:

- 1) натуральные операции (см. ниже);
- 2) линеаризация и делинеаризация (см. ниже);
- 3) двухместная операция соединения слов (конкатенация);
- 4) всякое объявленное операцией отображение, осуществляемое алгоритмом.

Больше никаких операций нет.

Предварительно определим одну частную символьную конструкцию, которая нам потребуется.

Введем в рассмотрение связь первого ранга и первого жанра, неодинаковую с начинающей связью следования, и назовем ее выделяющей связью.

Определение. Конструкция, которая получится, если в непустом слове одну из букв выделить с помощью выделяющей связи (т. е. связать ее этой связью), называется **квазисловом**. Началом квазислова называется его буква, связанная начинающей связью следования, а его концом – связанная заканчивающей связью следования.

Пример. Из слова

●→Н→О→С→●

могут быть получены следующие квазислова:

$$\begin{array}{c} \downarrow \downarrow \downarrow \\ \bullet \rightarrow \text{Н} \rightarrow \text{О} \rightarrow \text{С} \rightarrow \bullet \quad \bullet \rightarrow \text{Н} \rightarrow \text{О} \rightarrow \text{С} \rightarrow \bullet \quad \bullet \rightarrow \text{Н} \rightarrow \text{О} \rightarrow \text{С} \rightarrow \bullet \end{array}$$

Если связи следования явно не указывать, то квазислова будут иметь вид

↓ ↓ ↓
нос нос нос

В качестве натуральных операций выбраны операции, умение выполнять которые при построении теории алгоритмов обычно постулируют, считая, что каждый человек умеет выполнять эти операции. Некоторые натуральные операции являются в действительности классами операций, причем для выделения индивидуальной операции должна быть указана буква языка операндов, которую мы будем в описаниях обозначать метасимволом *a* (см. табл. 3.1).

Натуральные операции делятся на натуральные действия и натуральные условия. В свою очередь натуральные действия делятся на четыре группы:

- в группу I входят преобразования слов в слова;
- в группу II – преобразования слов в квазислова;
- в группу III – преобразования квазислов в квазислова;
- в группу IV – преобразование квазислов в слова;

V группу натуральных операций образуют условия, выполнение которых не изменяет операнда, но ставит ему в соответствие логическое значение (истина, если условие выполнено, или ложь, если не выполнено).

Пример. Операция ж-генерации преобразует пустое слово, изображенное между кавычками “” >, в однобуквенное слово “ж”. Операция аннигиляции может быть применена только к однобуквенным словам и преобразует их в пустые слова. Например, однобуквенное слово “с” она преобразует в пустое слово “”. Операция нахождения начала слова может быть применена только к словам, и притом непустым. Выделяющую связь будем изображать в виде знака “!”, поставленного над выделенной буквой. Слова “ж”

•→ж→•

и “мама”

•→м→а→м→а→•
!

она перерабатывает в квазислова “ж”

!
•→ж→•
!

и “мама”

!
•→м→а→м→а→•

Операция продвижения вперед применима только к квазисловам, в которых выделена не последняя буква (иначе некуда продвигаться). Например, квазислово

!
аргумент

она преобразует в квазислово

!
аргумент

Таблица 3.1

Перечень натуральных операций

Номер группы	Название операции	Описание
I	a – генерация	Преобразование пустого слова в однобуквенное слово, состоящее из буквы a
	Аннигиляция	Преобразование однобуквенного слова в пустое слово
II	Нахождение начала слова	Преобразование непустого слова в квазислово путем выделения его начала
III	Продвижение вперед	Перенос в квазислове выделяющей связи на одну букву правее
	Продвижение назад	Перенос в квазислове выделяющей связи на одну букву левее
	Удлинение вперед без продвижения	Преобразование квазислова с выделенным концом путем присоединения буквы, одинаковой с выделенной
	Отбрасывание конца	Преобразование квазислова в новое путем отбрасывания букв, следующих за выделенной, которая при этом становится концом
	Замена буквы на a	Замена выделенной буквы квазислова буквой a , которая становится выделенной
IV	Отключение	Квазислово преобразуется в непустое слово путем отбрасывания выделяющей связи
V	Натуральные условия:	
	Условие непустоты	Проверяется предикат “рассматриваемое слово не пусто”
	Условие начала	Проверяется предикат “в рассматриваемом квазислове выделена первая буква”
	Условие конца	Проверяется предикат «в рассматриваемом квазислове выделена последняя буква»
	Условие тождества букв	Проверяется предикат «в рассматриваемом квазислове выделена буква, одинаковая с a »

Операция продвижения назад применима только к квазисловам, в которых выделена не первая буква. Например, квазислово

!
функция

она преобразует в квазислово

!
функция

Операция удлинения вперед применима только к квазисловам с выделенной последней буквой. Например, квазислово

!
сорт

она преобразует (если $a = t$) в квазислово

!
сортт

Операция отбрасывания конца применима к любым квазисловам и только к квазисловам. Например, она преобразует

! ! !
ж парк сорт

соответственно в квазислова

! ! !
ж пар сор

Операция замены буквы на a в конкретном случае может быть заменой буквы на a . Такая операция преобразовала бы квазислово

!
крен

в квазислово

!
кран

Операция отключения неприменима к словам, а только к квазисловам. Последнее из вышеприведенных квазислов она преобразовала бы в слово кран.

Прежде чем приступить к разъяснению операции линеаризации и делинеаризации, рассмотрим проблему получения всевозможных нумераций произвольной совокупности предметов, если известна какая-нибудь их нумерация. Итак, предположим, что некоторая конечная совокупность предметов перенумерована числами

1, 2, ..., n, n + 1, n + 2.

Такая совокупность упорядочена. В ней всегда можно просмотреть все ее предметы, выбрав сперва 1-й, а затем, переходя от одного к другому, в порядке их номеров. Существует алгоритмический процесс получения всех возможных нумераций по заданной нумерации, а значит, и алгоритм в интуитивном смысле для выполнения этой операции. Он заключается в следующем.

1) Составим строку s из одного элемента 1; перейдем к п. 2.
 2) Параметру j присвоим (временно) значение 1; перейдем к п. 3.
 3) Если $j=p$, то перейдем к п. 4, иначе – к п. 6.
 4) Припишем в конце каждой из имеющихся последовательностей s число $j+1$. Нами получены новые последовательности, обменивая в каждой из них элемент $j+1$ столько раз, сколько возможно, с предыдущим элементом, из каждой новой последовательности получим еще j новых последовательностей. Каждую из новых последовательностей будем называть именем s (старых последовательностей уже нет). Перейдем к п. 5.

5) Увеличим значение j на 1 и перейдем к п. 3.

6) Образует пары из полученных последовательностей, в каждой из которых 1-м членом является последовательность 1, 2, . . . , p . Всего таких пар будет $v = p! - 1$. Считаем, что в каждой паре элементы (числа), одинаково расположенные в ее членах, взаимно однозначно соответствуют друг другу. Таким образом получено v целочисленных функций, каждая из которых позволяет преобразовать исходную нумерацию в некоторую новую. Перейдем к п. 7.

7) Последовательно применяя полученные функции к исходной нумерации, получаем из нее (и кроме нее) еще v новых нумераций. Процесс окончен.

Существенной чертой описанного процесса является то обстоятельство, что кроме описания его и исходной нумерации для получения всех возможных нумераций никаких дополнительных данных не нужно. Полученные нами функции мы строим тогда, когда они могут потребоваться. Иметь их в запасе не нужно. Это важно потому, что такой запас, пригодный на любой случай, был бы бесконечен, и процедура получения всех возможных нумераций не была бы эффективной. При $p = 1$ заданная нумерация есть единственно возможная.

Пример. Продемонстрируем процесс получения всех возможных нумераций для случая трех предметов. Итак, предположим, что даны предметы a, b, c , перенумерованные следующим образом:

a, b, c
1 2 3

В соответствии с п.1 описания процесса составляем последовательность из одного числа 1. Полагаем далее $j=1$. Так как $j \neq 3$, то приписываем к каждой из имеющихся последовательностей (их всего – одна) число $j+1=2$ (т.е. выполняем п. 4), получаем новую последовательность 1, 2. Переставляя последний член с предшествующим ему, получаем еще одну последовательность 2, 1. После этого (в соответствии с п. 5) увеличим на единицу параметр j и получим $j=2$. Так как $j \neq 3$ то, приписывая $j+1=3$ к каждой из имеющихся двух последовательностей (в соответствии с п. 4), получим две новые последовательности 1, 2, 3 и 2, 1, 3, из которых с помощью перестановок чисел, вместе с последними двумя, всего получим 6 последовательностей:

1, 2, 3; 1, 3, 2; 3, 1, 2;
2, 1, 3; 2, 3, 1; 3, 2, 1.

Теперь, увеличивая параметр, мы получим $j=3$ и, в соответствии с п. 3, перейдем к построению нужных нам функций (в соответствии с п. 6). Эти функции можно представить в виде табличек:

123	123	123	123	123
<hr/>				
132	312	213	231	321

Верхняя строка каждой таблички содержит заданные номера, а нижняя – номера, получающиеся в результате изменения нумерации. Мы видим, что табличек всего $v = 3! - 1 = 5$. Вместе с исходной всего возможно 6 нумераций. Нумерация, получающаяся из исходной с помощью, например, 3-й таблички (функции):

a, b, c или, что то же, b, a, c
2 1 3 1 2 3

Линеаризацией называется операция, с помощью которой произвольная конструкция класса (A, B, Σ) преобразуется в слово определенного специального вида в специальном расширении алфавита A . Прежде всего опишем нужное нам расширение алфавита A .

Возьмем произвольный алфавит букв C , не пересекающийся ни с A , ни с B , и имеющий число букв, равное определенному нами максимальному рангу связей, перечисленных в B . Наконец, возьмем алфавит D , не пересекающийся ни с A , ни с B , ни с C , число букв которого равно 3. Для определенности буквами в D будем считать символы $(“”)$ и $“1”$, которые будем называть скобками и единицей (цифрой 1). В качестве расширения алфавита A возьмем $A' = B \cup A \cup C \cup D$. Слова, образованные из единиц, мы будем при выполнении нумераций применять как записи целых чисел в единичной системе счисления.

В дальнейшем, если некоторая буква предшествует в алфавите другой букве, будем говорить, что она младше этой второй буквы.

Ниже, при описании процесса линеаризации, мы будем произвольно нумеровать те или другие элементы конструкции, а затем учитывать все возможные результаты произвольной нумерации. Для этого будем репродуцировать обрабатываемую конструкцию $n!-1$ раз (где n – наибольший из произвольных номеров) и производить преобразование произвольных номеров (представленных как строки вида $1...1$) способом, описанным выше. После каждого такого шага число обрабатываемых конструкций будет увеличиваться. Дальнейшие действия будут применяться к каждой из $n!$ полученных конструкций в предположении, что каждая из них имеет то же имя, что исходная.

Линеаризация состоит из следующих этапов.

Этап I. Преобразуемую конструкцию K заключают в оболочку.

Этап II. Находят в K для каждой буквы β_i , ($i=1, 2, \dots$) алфавита и все связи, отвечающие этой букве. Производят их произвольную нумерацию и помечают словами $1, 11$, и т. д. Если произвольная нумерация привела к наибольшему номеру, который >2 , то определяют все возможные нумерации, увеличивая число экземпляров обрабатываемой конструкции и преобразуя вышеописанным способом номера. Считают, что каждая из полученных конструкций имеет имя K . Далее, переходя от связи к связи в каждой K , в порядке, который задают приписанные связям номера, упорядочивают ветви каждой связи аналогично тому, как это делалось для связей, с той лишь разницей, что вместо алфавита B используют алфавит C , считая, что ветвям i -го жанра отвечает i -я буква этого алфавита. Число конструкций снова возрастает (после каждой произвольной нумерации), а имя K закрепляется за каждой из полученных конструкций (это позволяет, говоря о K , иметь в виду каждую из них).

После окончания этого этапа все связи и ветви связей в некотором числе конструкций, каждая из которых называется K , помечены буквами и номерами.

Описанием каждой ветви любой связи называют слово, в начале которого стоит буква в B , затем несколько букв $“1”$, затем — буква в C и, наконец, опять несколько единиц. Считают, что связи распались на отдельные ветви.

Этап III. В преобразуемой конструкции находят оболочку, внутри которой нет других оболочек. Пусть K' - заключенная в ней конструкция. Выписывают все конструктивные элементы, из которых образована K' , в произвольном порядке, а перед каждым из них в лексикографическом порядке выписывают описания ветвей связей, для которых данный конструктивный элемент является связуемым объектом. Конструктивным элементом может быть (при повторных выполнениях этапа IV) либо буква в A , либо слово в A' , заключенное в скобки. Конструктивный элемент со всеми предшествующими ему описаниями ветвей назовем фрагментом. Каждый фрагмент является словом в A' . Все выписанные фрагменты в лексикографическом порядке объединим в одно слово и заключим в скобки. Полученным результатом в K заменим K' вместе с содержащей ее оболочкой. Все ветви связей, которые прежде связывали оболочку, объемлющую K' , теперь отнесем к открывающей скобке. Заметим, что если K' является пустой, то и слово, составленное из фрагментов, будет пустым.

Слово, которое будет получено после многократного выполнения данного этапа над каждой K и после отбрасывания содержащих его открывающей и закрывающей скобок, является одним из предварительных результатов, совокупность которых будет конечна.

Этап IV. Из всех предварительных результатов выбирают один, который лексикографически не старше остальных. Это и есть результат линеаризации, который однозначен, конечно, при заданных алфавитах B , C и D .

Делинеаризацией называется процесс восстановления конструкции по слову, имеющему соответствующую структуру, коротко говоря, - являющемуся ее описанием.

Две конструкции класса (A, B, Σ) называются одинаковыми, если они при одинаковом A' дают один и тот же результат линеаризации.

Предположим, что задан некоторый формальный язык L_1 , предложения которого являются исходными данными для определенных преобразований. L_1 будем называть входным языком операндов. Предположим, далее, что нам задан конечный набор операций, из которых некоторые могут быть применены к некоторым предложениям (символьным конструкциям) языка L_1 . Операции будем делить на действия и условия. Выполнение действия заключается в том, что конструкция — операнд заменяется конструкцией, получаемой из этого операнда путем выполнения операции; то, что получится, в дальнейшем считается операндом. Если операция к исходному операнду не применима, то никакого результата не получается и никакая информация о его отсутствии не вырабатывается. Выполнение условия заключается в проверке, справедливо ли оно для операнда, или нет. В первом случае результатом проверки является логическое значение истина, во втором случае — ложь. Операнд остается неизменным. Условия являются предикатами.

Построим контекстно свободный язык L_2 , который будем называть **алгоритмическим**. Для этого воспользуемся нотацией Бекуса. Предложение языка L_2 будем обозначать метасимволом (запись первичного алгоритма):

<запись первичного алгоритма> ::= <безусловный приказ> | <условный приказ>

<безусловный приказ> ::= <метка> <разделитель I> <знак действия> <разделитель II> <отсылка> <разделитель III>

<условный приказ> ::= <метка> <разделитель IV> <знак условия> <разделитель V> <отсылка> <разделитель VI> <отсылка> <разделитель III>

<отсылка> ::= <метка> | <стоп>

Для того чтобы L_2 был полностью определен, необходимо задать с помощью метаформул значения метасимволов:

<метка> <разделитель IV>

<стоп> <разделитель V>
 <разделитель I> <разделитель VI>
 <разделитель II> <знак действия>
 <разделитель III> <знак условия>

Мы этого не будем делать, определяя тем самым не один язык, а класс языков. В качестве L_2 , может быть принят любой из конкретных языков этого класса. Заметим только, что разделители должны быть выбраны так, чтобы они однозначно разделяли приказы на соответствующие части, а разделитель III должен быть отличим от разделителя VI, а также от любой отсылки и метки.

Для того чтобы наделить запись первичного алгоритма смыслом, зададим следующее правило.

Правило выполнения первичного а л г о р и т м а.

1) Просматривая запись первичного алгоритма с начала, найти первый приказ; перейти к п. 2.

2) Если рассматриваемый приказ является безусловным, перейти к п. 3, иначе — к п. 5.

3) Применить операцию, соответствующую знаку действия данного приказа, к операнду; найти отсылку в данном приказе; перейти к п. 4.

4) Если выбранная отсылка имеет вид (стоп), то процесс окончен; иначе, просматривая запись алгоритма с начала, найти приказ, метка которого одинакова с отсылкой, и перейти к п. 2.

5) Если операнд удовлетворяет условию, соответствующему знаку условия данного приказа, то перейти к п. 6, иначе — к п. 7.

6) Найти первую отсылку данного приказа; перейти к п. 4.

7) Найти вторую отсылку данного приказа; перейти к п. 4.

Правило выполнения зависит от языка L_2 и от заранее выбранного набора операций. Это правило является алгоритмом в интуитивном смысле слова, однако сформулированным настолько точно, что при его выполнении не может возникнуть никаких неясностей.

Запись первичного алгоритма, рассматриваемая вместе с правилом его выполнения, называется первичным алгоритмом. Первичные алгоритмы предполагают наличие двух языков: входного языка операндов и алгоритмического (для первичных алгоритмов, контекстно свободного) языка.

Пример. Приведем пример несложного семейства первичных алгоритмов, алгоритмический язык которых близок к подмножеству русского языка, но при этом является формальным. Для этого полностью определим алгоритмический язык L_2 следующим образом:

<метка> ::= <цифра> | <метка> <цифра>

<цифра> ::= 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9

<стоп> ::= остановиться

<разделитель I> ::=)

<разделитель II> ::= и | и перейти к п.

<разделитель III> ::= ;

<разделитель IV> ::=) Если

<разделитель V> ::= , то перейти к п.

<разделитель VI> ::= , иначе — к п.

<знак действия> ::= Величину <буква> уменьшить на значение <буква> |

Полагать, что <буква> равно значению <буква>

<знак условия> ::= <буква> <знак отношения> <буква>

<знак отношения> ::= >

<буква> ::= x | y | z | u | v | w

Теперь язык L_2 определен полностью. Разделитель II определен у нас в двух вариантах. Оба варианта эквивалентны, но первый из них мы будем

употреблять только перед отсылкой <стоп>, а второй во всех остальных случаях. Будем считать, что заданы только два действия: вычитание и присвоение значения, и только одно условие, гласящее “первая величина больше второй”.

Теперь приведем пример первичного алгоритма:

- 1) Если $x > y$, то перейти к п. 4, иначе — к п. 2;
- 2) Если $y > x$, то перейти к п. 5, иначе — к п. 3;
- 3) Полагать, что z равно значению x и остановиться;
- 4) Величину x уменьшить на значение y и перейти к п. 1;
- 5) Величину y уменьшить на значение x и перейти к п. 1;

Исходным данным является запись вида

$x=a$; $y=b$,

где a и b — любые целые неотрицательные числа. Описанный первичный

алгоритм является алгоритмом вычисления общего наибольшего делителя, т.е. одним из класса алгоритмов Евклида. Алгоритмический язык мы выбрали так, что приказы первичного алгоритма совпадают по виду и по смыслу (задаваемому правилом выполнения первичного алгоритма) с соответствующими фразами русского языка. Отметим, что такое сходство представляет собой редкий случай и вовсе не обязательно.

Применение правила выполнения первичного алгоритма к совокупности, образованной из записи первичного алгоритма и записи операнда, порождает процесс, называемый алгоритмическим. Этот процесс может либо заканчиваться при выполнении п. 2 правила, и тогда полученный операнд называется искомым результатом, либо обрываться из-за невыполнимости какого-либо другого пункта правила (безрезультатно остановиться), либо продолжаться неограниченно (не останавливаться).

В первом случае говорят, что первичный алгоритм применим к данному операнду, а во втором и третьем случаях — что неприменим.

Следует обратить внимание на то, что алгоритмический процесс представляет собой процесс совместного преобразования совокупности записей алгоритма и операнда. Частной особенностью этого преобразования является неизменность символьной конструкции, являющейся записью алгоритма.

Подклассом первичных алгоритмов являются **натуральные алгоритмы**, так называются первичные алгоритмы, для выполнения которых требуется только знание натуральных операций и, может быть, линеаризации и делинеаризации.

Рассмотрим теперь **широкое формальное определение алгоритма**. Класс первичных алгоритмов задан, если заданы два языка: L_1 и L_2 , причем предложения первого из них объявлены записями алгоритмов, а предложения второго — записями операндов и, если задано правило выполнения первичных алгоритмов, применимое к парам записей алгоритма, записей операнда.

Первый пункт общего определения алгоритма гласит:

- 1) Первичные алгоритмы — это алгоритмы.

Но общее понятие алгоритма существенно шире. Его второй пункт гласит:

- 2) Если заданы два языка L_1 и L_2 , причем предложения первого объявлены записями алгоритмов, а предложения второго — записями операндов, и если задан некоторый алгоритм W , операндами которого являются конструкции, получаемые связыванием каждого предложения из L_1 с n предложениями языка L_2 , при помощи вполне определенной связи ранга $n+1$, то задано семейство n -местных алгоритмов. W называется алгоритмом выполнения этих n -местных алгоритмов.

Третий пункт общего определения гласит:

- 3) n -местные алгоритмы — тоже алгоритмы.

Таким образом, каждый алгоритм в широком формальном смысле, если он не первичный, имеет свой алгоритм выполнения. Если бы не были определены первичные алгоритмы, то невозможно было бы дать строгое определение и остальных алгоритмов, так как невозможно было бы указать ни одного алгоритма выполнения. Но после того, как построен хотя бы один алгоритм выполнения, можно строить многие другие, уже не привлекая первичные алгоритмы.

Функции, порождаемые алгоритмами, называют вычислимыми.

Под **операндом** обычно понимают предложение формального языка, наделенное смыслом. Поэтому естественно отличать операнд от его записи. Обозначая операнд какой-либо буквой, будем его запись тоже обозначать той же буквой с крышкой (черточкой) наверху.

Широкое формальное определение понятия алгоритма дает нам обширную иерархию семейства алгоритмов, различных по своей структуре и возможностям.

При анализе или конструировании алгоритмов используют различные способы их описания, отражающие только те их свойства, которые существенны для конкретных целей анализа или конструирования. Такие описания алгоритмов называют их схемами.

Схемы, отражающие функциональные свойства алгоритмов, называются **семантическими**, а описания, отражающие структуру алгоритмов как символьных конструкций называются **структурными** схемами.

3.4. Измерение и передача информации

Информационный сигнал

Понятие информации – одно из основных понятий АИС, так как при автоматизации любых процессов на первый план выдвигаются процессы преобразования информации. Как уже отмечалось, строго формализованного понятия информации не существует. То понятие информации, которое обычно используется, заимствовано из теории связи. Основная его особенность состоит в абстрагировании от смыслового содержания информации, использовании ее количественной меры по Шеннону. Однако для разрешения многих задач необходимо оперировать именно количественными характеристиками смыслового или семантического содержания информации.

Тем не менее, в ряде случаев, классическое понятие количества информации бывает полезно при изучении процессов управления. Именно отвлечение от смыслового содержания информации позволяет получать обобщенные характеристики по загрузке каналов связи, памяти ЭВМ, каналов преобразования информации в АИС. Поэтому далее будут рассмотрены вопросы теории информации, характерные для теории связи и основанные на вероятностном подходе к процессам преобразования информации.

Теория информации имеет дело с определенной моделью системы связи. На рис. 1 представлена типичная структурная схема передачи сообщений, используемая в теории связи. Система связи начинается с источника информации, создающего сообщение или их последовательность для передачи по линии связи. Сообщениями могут быть:

человеческая речь (в этом случае радио или телефонное сообщение представляет собой некоторую функцию времени);

последовательность букв и цифр (телеграф);

некоторая функция двух координат и времени (черно-белое телевидение), которая представляет интенсивность света в точке передаваемого изображения и т.д.

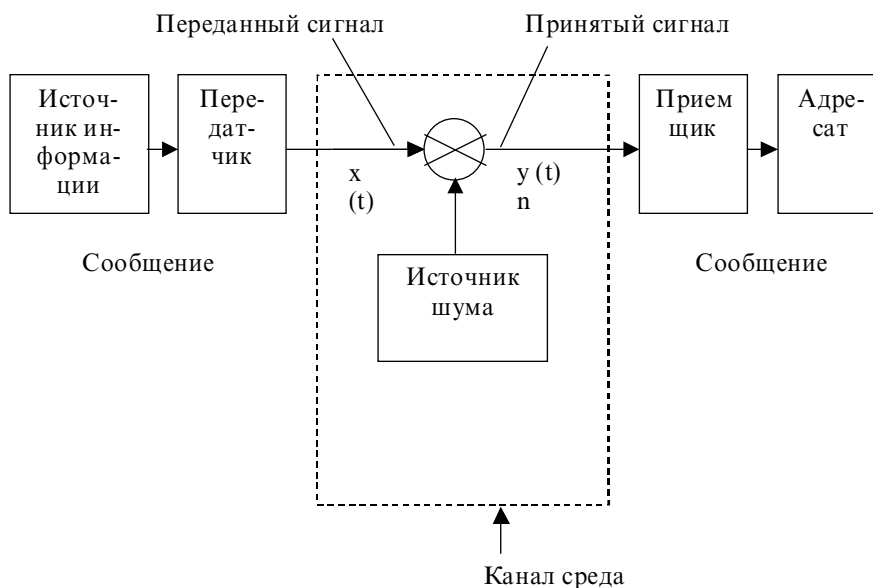


Рис. 1

Сообщения передаются на передатчик, который перерабатывает их в сигналы в соответствии с данным каналом связи. В простейшем случае (телефон) акустические волны человеческого голоса преобразуются в электрический ток. При радиоприеме эти низкочастотные сигналы наполняются радиочастотой, усиливаются по мощности, а при радиорелейной передаче человеческого голоса необходима специальная система импульсного кодирования. В общем случае передающее устройство имеет преобразователь (микрофон), кодирующее устройство, модулятор, передатчик, выходные устройства передатчика (антенну).

Из передатчика сигналы поступают в канал передачи или среду, в качестве которой могут выступать пара проводов (телефонная связь), коаксиальный кабель (телевизионная передача), полоса радиочастот (радиоприем), луч света видимого или инфракрасного диапазона или лазер. В канале связи (среде), как правило, на сигнал действуют помехи, которые искажают его, поэтому прием-восстановление информации осуществляется в условиях шумов (помех). Шумы при радиоприеме – это искровые разряды в атмосфере, промышленные помехи, искрение контактов в транспортных средствах и др. Работе радиолокационных систем мешают помехи, специально создаваемые противником.

В приемнике сообщение восстанавливается, здесь обычно выполняются операции, обратные тем, которые имеют место в передатчике: усиление сигнала, демодуляция, декодирование. С выхода приемника расшифрованное сообщение поступает адресату. Рассмотренную модель системы связи можно принять как некоторую основу для анализа процессов приема передачи сигналов.

Анализ общей структурной схемы связи показывает, что классическая теория информации в основном состоит из двух частей:

теории преобразования сообщений и сигналов, основную долю в которой составляют вопросы кодирования и декодирования;

теории передачи сообщений и сигналов без шумов и с шумами в канале связи.

Носителем сообщения или информации является **сигнал**. Следует различать физические (реальные) сигналы и их математические модели. Разнообразие реальных сигналов много, однако большинство из них описывается сравнительно небольшим количеством математических моделей. Очевидно, что при таком описании допускается какая-то погрешность, которая в одних случаях (моделях) существенно, а в других несущественно искажает природу реального сигнала. Сложность сведения реального сигнала к его математической модели состоит в том, что для этого нет достаточно общих правил. Кроме того, в зависимости от задачи при формализации уровень детализации может изменяться.

Рассмотрим импульсный сигнал, отраженный от цели в импульсной радиолокации. Радиолокатор периодически, с периодом T , посылает импульсы

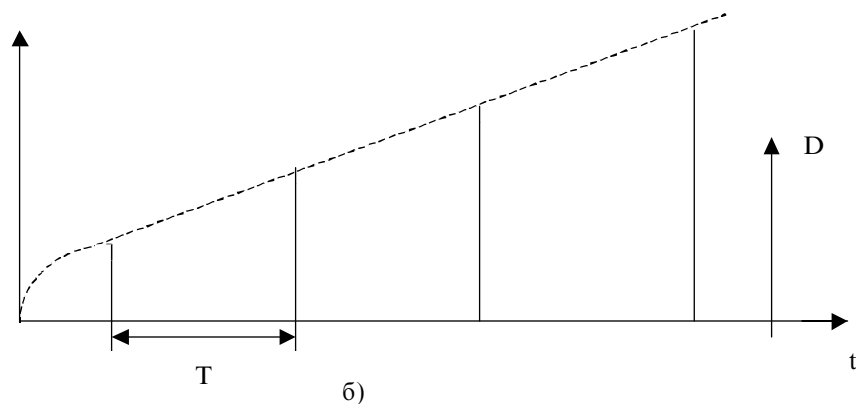
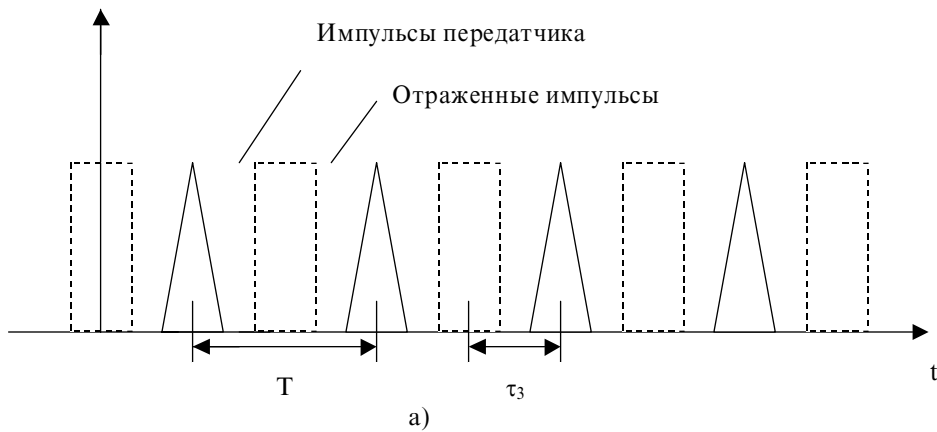


Рис. 2

определенной формы в сторону цели, которые, отразившись, возвращаются на вход приемника (рис. 2, а). Задержка τ_z этих импульсов относительно излученных пропорциональна дальности до цели D. Построение математической модели реального физического сигнала заключается в замене его амплитудно-модулированным сигналом (рис.2, б), состоящим из последовательности δ -импульсов, амплитуда которых равна дискретным значениям дальности до целей или времени задержки отраженных импульсов относительно излученных.

Количество информации

В классической теории информации известно по крайней мере два определения количества информации. Оба определения очень близки между собой, принципиальное различие между ними появляется лишь при попытке ввести смысловое содержание информации. Первое определение (по Хартли) использует комбинаторный подход, второе (по Шеннону) распространяет на передачу и переработку информации вероятностную точку зрения. Чем больше неопределенности в принятом сообщении, тем больше информации в нем содержится.

Поэтому количество информации определяют следующим образом:

$$I = \log_2(P_1/P_0),$$

где: I – количество информации;

P_1 – вероятность данного события после поступления сообщения на вход приемника;

P_0 – вероятность данного события до поступления сообщения на вход приемника.

Если шумы отсутствуют, то можно считать, что вероятность данного события после поступления на вход приемника сообщения о нем равна единице.

За основание логарифма принимают чаще всего цифру 2, но иногда используют также десятичный и натуральный логарифмы.

Для измерения количества информации введена специальная единица измерения, которая называется бит (bit) информации. Количество информации в битах равно правой части приведенного выше равенства.

Сообщение, как правило, набирается или составляется из символов или элементов: буквенного алфавита, цифр, слов или фраз, названий цвета, предметов и т.д. Обозначим общее число символов в алфавите через m . Если сообщение формируется из двух независимо и равновероятно появляющихся символов, то нетрудно видеть, что число возможных комбинаций равно m^2 . Действительно, зафиксировав один из двух символов сообщения ($n = 2$) и комбинируя его со всеми возможными m символами алфавита, получим m различных сообщений. После этого фиксируем следующий символ алфавита и снова его комбинируем со всеми символами алфавита, получим еще m сообщений. С учетом предыдущего имеем $2m$ сообщений. Продолжив этот процесс до тех пор пока будет зафиксирован последний из m символов алфавита, получим всего $mm = m^2$ сообщений. В общем случае, если сообщение содержит n элементов (n – длина сообщения), число возможных сообщений

$$N = m^n.$$

Пример. Предположим, имеется набор из трех букв А, В, С, а сообщение формируется из двух.

Согласно формуле число возможных сообщений будет АА, ВА, СА, АВ, ВВ, СВ, АС, ВС, СС, т. е. $N = 3^2 = 9$.

Однако нетрудно видеть, что комбинаторное выражение неудобно брать в качестве меры количества информации, во-первых, если все множество или ансамбль возможных сообщений состоит из одного сообщения ($N = 1$), то информация в нем должна отсутствовать, во-вторых, если есть два, независимых источника сообщений, каждый из которых имеет в своем ансамбле N_1 и N_2 сообщений, то общее число возможных сообщений от этих двух источников

$$N = N_1 N_2,$$

т.е. является произведением, тогда как количества информации должны складываться, и общее количество должно быть прямо пропорционально числу символов в сообщении. Поэтому за количество информации берут логарифм числа возможных сообщений

$$I = \log N = n \log m.$$

По существу при выводе этого соотношения считалось, что появление символов в сообщении равновероятно и они статистически независимы. Чаше бывает наоборот. Так, в русском языке одни буквы встречаются чаще, другие – реже, после согласных, как правило, следуют гласные. Очевидно, в этом случае информации будет меньше.

Как правило, при приеме по телеграфу первых слов можно с достаточной точностью предсказать следующие слова. Поэтому говорят о взаимосвязи элементов в сообщении. Связь понимается в вероятностном смысле, т.е. существует условная вероятность появления (при данном алфавите) символа A вслед символом B :

$$P\{A|B\}.$$

Так, в русском тексте после гласной не может следовать мягкий знак или подряд четыре гласные буквы, т.е. условная вероятность равна нулю. Если понятно, какие символы последуют дальше, сообщение представляет мало интереса и содержит меньше информации, чем оно содержало бы, если бы взаимная связь его элементов не была очевидна. В качестве примера взаимных связей можно привести прямой порядок слов в предложении, согласно которому после подлежащего должно следовать сказуемое: если принимается сообщение “Идет дождь”, то достаточно принять “Идет дожд...” и прием следующих символов уже не добавит информации. Это свойство сообщений характеризуется величиной, называемой избыточностью.

Количество информации может уменьшаться также из-за того, что в силу особенностей языка различные символы с разной вероятностью появляются в тексте сообщения. Так, свойство буквы E в английском языке встречаться чаще буквы I позволяет предсказывать, предопределять сообщение, т.е. неравновероятное, неравномерное появление символов в сообщении (если, конечно, оно заранее известно) уменьшает количество сведений, количество информации в принимаемом сообщении.

Современный развитый язык насчитывает в своем составе до 100 тысяч слов. Однако не все они одинаково часто употребляются. В среднем достаточно знать несколько тысяч слов, чтобы изъясняться. Слова в языке также обладают разной вероятностью появления. Очевидно, что неравномерное распределение вероятностей появления отдельных слов в языке (максимум одних и минимум других) также уменьшает количество информации, так как можно предсказать появление тех или иных слов в сообщении.

Так же как в теории вероятностей, в теории информации к вероятности возможны два подхода. В первом случае, который эквивалентен усреднению по времени, рассматриваются бесконечно длинные (а практически просто длинные) сообщения. В процессе наблюдения во времени за длинным сообщением исследуется статистика появления отдельных символов или их комбинаций. Во втором случае рассматривается множество (теоретически бесконечно большое) конечных сообщений и статистические характеристики определяются путем усреднения по ансамблю сообщений. При этом всегда предполагается, что число сообщений (или символов) такое большое, что применим закон больших чисел. Приведем простейший вывод выражения для количества информации, предполагая отсутствие связей элементов. Пусть имеем алфавит, состоящий из m элементов (символов) h_1, h_2, \dots, h_m .

Вероятности появления этих элементов в сообщении соответственно равны p_1, p_2, \dots, p_m . Составим из этих элементов сообщение, содержащее n элементов. Среди них будет n_1 элементов h_1, n_2 элементов h_2, \dots, n_m элементов h_m . Вероятность появления каждой комбинации из n элементов выразится произведением вероятностей отдельных элементов, так как предполагается, что появление каждого элемента есть независимое событие. С учетом повторяющихся элементов вероятность некоторого сообщения

$$p = p_1^{n_1} p_2^{n_2} \dots p_m^{n_m} = \prod_{i=1}^m p_i^{n_i}$$

вероятность появления i -го символа

$$p_i = n_i / n.$$

Можно считать, что все N возможных сообщений (все перестановки) равновероятны. Поэтому

$$p = 1 / \prod_{i=1}^m p_i^{n_i},$$

откуда число возможных сообщений

$$p = 1 / \prod_{i=1}^m p_i^{n_i}.$$

Логарифмируя, получаем количество информации в сообщении длиной n при неравновероятности его элементов:

$$I = \log N = -n \sum_{i=1}^m p_i \log p_i.$$

Пример. Имеются символы а, в, с. В этом случае $m = 3, n_1 = 2, n_2 = 3, n_3 = 1, n = 2 + 3 + 1 = 6$. Возможные сообщения будут выглядеть следующим образом: 1 – ааввс; 2 – вааввс; 3 – вваас; 4 – ввваас; ... Общее число сообщений

$$N = n! / (n_1! n_2! n_3!);$$

$$N = 6! / (2! 3! 1!) = (5 \cdot 4 \cdot 3 \cdot 2) / 2 = 60.$$

Отсюда при длине сообщения $n = 6$ количество информации будет равно:

$$I = \log_2 60 = 6 \text{ бит}.$$

Энтропия информационных сообщений

В теории информации вводится также понятие **энтропии** H - характеристики данного ансамбля сообщений с заданным алфавитом, которая является мерой неопределенности, имеющейся в ансамбле сообщений.

$$H = I / n = - \sum_{i=1}^m p_i \log p_i$$

Различие между количеством информации и энтропией в ряде случаев носит условный характер. Так, если рассматривать энтропию всех возможных сообщений длиной n , то эта величина будет совпадать с количеством информации, задаваемой формулой

$$I = \log N = -n \sum_{i=1}^m p_i \log p_i$$

И наоборот, если нас интересует информация на один символ, содержащаяся в ансамбле сообщений, то эта величина определяется величиной энтропии этого ансамбля. При передаче сообщений необходимо дать математическую формулировку определения количества информации как меры снятия неопределенности, которая численно равна разности априорной (до получения сигнала) и апостериорной (после получения сигнала) энтропии принимаемого сигнала:

$$I = H - H_0,$$

где H – априорная, H_0 – апостериорная энтропии. При этом определении, которое иногда называется третьим определением количества информации (имея в виду, что первое – по Хартли, второе – по Шеннону), считается, что информация относится к одному символу сообщения или энтропия – к определенной длине n сообщения. Это определение одинаково справедливо для дискретных и непрерывных сообщений. В последнем случае вообще трудно установить существенную разницу между количеством информации и энтропией. Если обозначить через

$H(x)$ – энтропию множества передаваемых символов (на входе канала связи);

$H(y)$ – энтропию множества принимаемых символов;

$H(x, y)$ – энтропию множества всевозможных пар (x_i, y_k) ;

$H(x | y_k)$ – энтропию множества отправляемых символов, оставшуюся после приема символа y_k ,

$H(y | x_i)$ – энтропию множества принимаемых символов при условии, что известен отправленный символ;

$H(x|y)$ и $H(y|x)$ – математические ожидания величин $H(x|y_k)$ и $H(y|x_i)$, то количество информации при приеме символа y_k определяется по формуле

$$I_k = H(x) - H(x|y_k).$$

Очевидно, что величина I_k является случайной и необходимо ее усреднить по всему множеству принимаемых символов.

После усреднения получим:

$$I = H(x) - H(x|y).$$

Эти формулы определяют свойство симметрии, заключающееся в том, что средняя неопределенность того, какой символ будет получен, снимаемая при посылке конкретного символа, равна средней неопределенности того, какой символ был отправлен, снимаемой при приеме символа.

В частности, величину $I(x, y)$ можно интерпретировать, как относительную информацию объекта x относительно объекта y (или объекта y относительно объекта x), равную разности априорной и апостериорной энтропии объекта y (или x). Причем оба объекта могут быть или дискретными, или непрерывными, один из объектов x может быть дискретен, а другой y непрерывен. В этом случае определяется количество информации, равное разности априорной и апостериорной энтропии дискретного объекта x или разности априорной и апостериорной дифференциальных энтропии непрерывного объекта y

$$I(x, y) = H(x) - M[H(x|y)] = H\varepsilon(y) - M[H\varepsilon(x|y)] =$$

$$= \sum_{i=1}^m \int_{(x)} p_{(x|y)} \log p(x_i|y) / [p(x_i) p(y)],$$

где: ε – уровень квантования сигнала,

M – математическое ожидание.

Симметрия является важным свойством информации, которое заключается в том, что $I(x, y)$ одинаковым образом зависит от x и y и количество информации об объекте x , содержащееся в объекте y , равно количеству информации об объекте y , содержащемуся в объекте x . Отсюда можно сформулировать четвертое определение количества информации: среднее количество информации есть мера соответствия двух случайных объектов.

Передача информации и пропускная способность канала связи

Теперь можно оценить **пропускную способность каналов связи**. Рассмотрим канал связи, представленный на рис.1. На его передающий конец подается сигнал $x(t)$, который поступает на вход приемника в искаженном шумом $n(t)$ виде $y(t)$. Введем понятие пропускной способности канала связи. **Пропускная способность канала связи** определяется как максимальная величина относительной информации выходного сигнала относительно входного:

$$C = \lim_{t_0 \rightarrow \infty} [I_{\max}(x, y) / t_0],$$

где $I(x, y)$ относительная информация, причем все сигналы рассматриваются как эквивалентные дискретные (рис. 3).

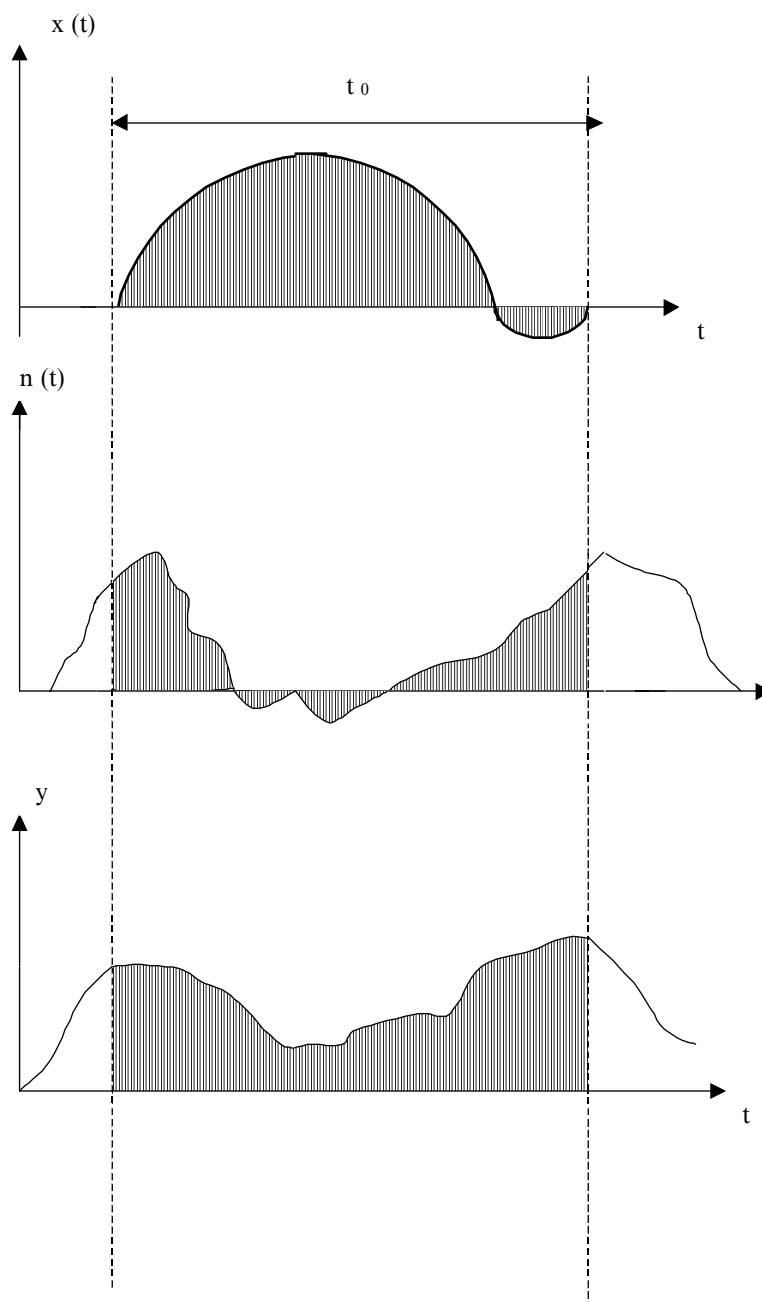


Рис. 3

Современный Гуманитарный Университет

Иногда величина $V = I(x, y)/t_0$ называется скоростью передачи информации по каналу связи. Эта величина равна количеству относительной информации, передаваемой в единицу времени. За единицу времени при дискретном канале связи удобно считать время передачи одного символа. В этом случае в формулах для скорости передачи информации понимают энтропии и количества информации на один символ. Для непрерывных каналов связи используются две единицы измерения или обычная единица (к примеру, секунда), или интервал времени между отчетами.

3.5. Основы теории кодирования информации

Сообщения передаются в виде сигналов, имеющих определенную форму и последовательность. В телеграфе сообщение обычно передается при помощи алфавитов, цифр или алфавита и цифр вместе. Сигналы следуют в определенной последовательности. Например, в коде Морзе каждой букве и цифре соответствует некоторая последовательность кратких (точки) и длинных (тире) посылок тока, разделяемых кратковременными паузами по длительности такими же, как и точки. Пробел между буквами при этом изображается выключением тока на три единицы времени, а пробел между словами – на шесть единиц времени (рис. 4, а). Если обозначить тире 1, а точки 0, то можно записать:

А	Б	В	Г	Д
01	1000	011	110	100

По коду Бодо, применяемому в буквопечатающих аппаратах, каждой букве соответствует сигнал из пяти импульсов одинаковой длительности и формы (рис. 4, б):

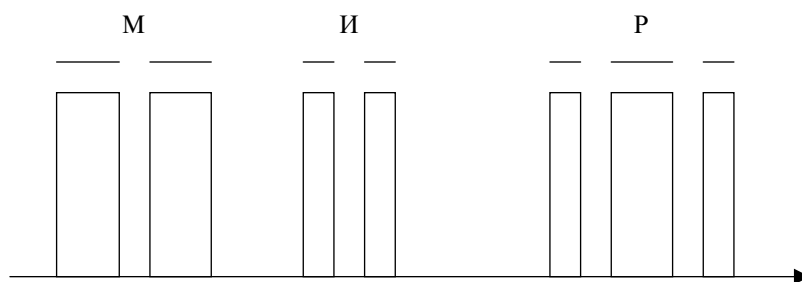
А	10000
Б	00110
В	01101
Г	01010
Д	11110

Этот код равномерен, так как на каждый символ требуется одинаковое время.

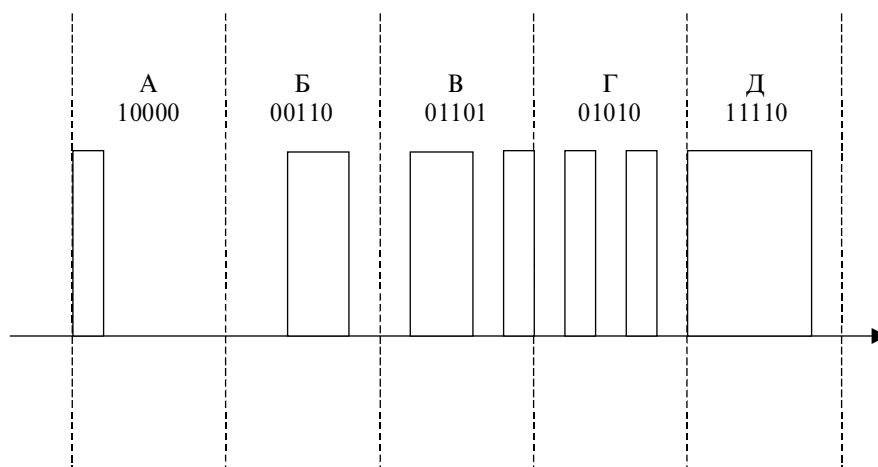
Следует различать способ кодирования и способ модуляции сигнала или сообщения. Так, рассмотренные коды Морзе и Бодо – двоичные, т.е. имеют в основании 2: сообщения передаются с помощью посылки сигнала (тока) или его отсутствия. Могут быть троичные коды, когда используются посылки положительного и отрицательного знака и отсутствие посылки, пятеричные и т.д. Единичная посылка сигнала может быть в виде напряжения постоянного или переменного тока (радиоимпульса). Наконец, можно передавать тире тремя импульсами, а точку – отсутствием импульсов. Во всех этих случаях код один и тот же, а модуляции разные.

В первом случае (рис. 5, а) имеет место модулированный сигнал постоянного тока, во втором (рис. 5, б) – сигнал переменного тока, модулированный прямоугольным напряжением, так называемый радиоимпульс (если частота заполнения лежит в диапазоне радиочастот). В последнем случае (рис. 5, в) имеет место кодоимпульсная модуляция. Помимо рассмотренных могут быть и другие виды модуляции.

По физической природе сигналы могут быть электрические, акустические, механические, радиолокационные и пр. В данном случае не исследуются ни



а)



б)

Рис. 4

физическая природа сигналов, ни виды модуляции, а будут рассмотрены виды кодирования.

Мощным средством повышения помехоустойчивости систем являются **корректирующие коды**, которые возникли главным образом в связи с развитием ЭВМ. Их основная идея заключается в том, что наряду с кодовой группой, несущей полезную информацию, передаются дополнительные знаки, с помощью которых удастся обнаруживать и исправлять ошибки, вводить коррекцию. Такая процедура вносит избыточность, снижает эффективность системы, но повышает ее помехоустойчивость.

В простейшем случае такой код получается добавлением к кодовой комбинации единицы с тем, чтобы сумма всех единиц была четной (нечетность означает появление ошибки). Построим его на примере кода Бодо.

Буквы А Б В Г Д Е

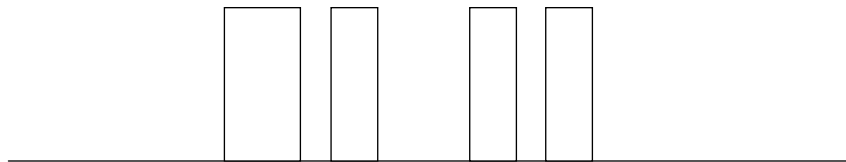
Обычный код 10000 00110 01101 01010 11110 01000

Дополнительный код 100001 001100 011011 010100 111100 010001

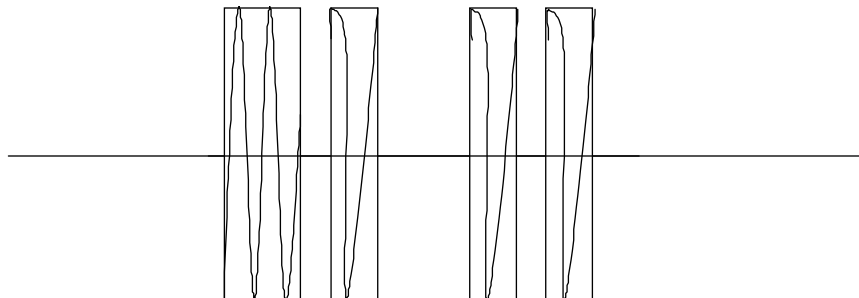
Такой код позволяет обнаружить одиночную ошибку, но не в состоянии ее локализовать и исправить.

Не только обнаружить, но и исправить ошибку можно с помощью более мощных кодов, которые строятся следующим образом. Пусть имеется n_0 - значный двоичный код. Общее число комбинаций $N=2^{n_0}$.

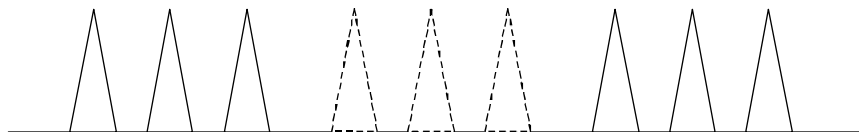
Каждый из таких кодов отличается один от другого хотя бы одним знаком. Дополним код еще одним знаком, а число кодовых комбинаций оставим неизменным, тогда $N=2^{n_0+1} = 2 \cdot 2^{n_0}$ и можно так подобрать кодовые комбинации, что они будут отличаться двумя знаками. При этом будет использована только половина всех возможных комбинаций от 2^{n_0+1} , вторая половина образует запрещенные комбинации: любое появление одиночной ошибки превращает ее в запрещенную и тем самым ошибка обнаруживается. Дополним теперь



а)



б)



в)

Рис. 5

код таким количеством знаков, которое даст возможность двум кодовым комбинациям отличаться тремя знаками при неизменном числе $N=2^{n_0}$. Такой код позволит не только обнаружить, но и исправить одиночную ошибку.

Действительно, если случилась одиночная ошибка в какой-то комбинации, то эта комбинация от других будет отличаться на два знака, а от своей – на один и ее легко исправить.

Определим общее число дополнительных знаков, необходимых для обнаружения и исправления одиночных ошибок. Пусть из общего числа позиций n для передачи информации используется n_0 , которое будем считать фиксированным. Остальные позиции $k = n - n_0$ используются в качестве проверочных. Символы, которые ставятся на k проверочных позициях, определяются при кодировании проверкой на четность каждой из k групп информационных символов. Как будет далее показано, на каждой проверочной позиции при кодировании ставится 0 или 1, смотря по тому, какая сумма единиц – четная или нечетная получается при каждой из n проверок на четность. Сигнал кодируется так, чтобы в результате в каждой из n проверок получалось четное число. На приемном конце появляются на некоторых позициях единицы вместо нулей и, наоборот, нули вместо единиц. При приеме также производится проверка на четность.

Построим код, который позволял бы обнаруживать и исправлять одиночную ошибку (если произошли две ошибки сразу, код бессилён). Пусть принята кодовая комбинация с ошибкой или без нее. Произведем в ней последовательно k проверок. После каждой проверки запишем 0, если результат свидетельствует об отсутствии ошибки на проверяемых позициях (сумма единиц четная), и 1, если результат свидетельствует о наличии ошибки (сумма единиц нечетная). Запись справа налево полученной последовательности единиц и нулей дает двоичное число. Потребуем, чтобы это число, называемое проверочным, давало номер позиции, на которой произошло искажение. Отсутствию ошибки в принятой кодовой комбинации будет соответствовать число, составленное из нулей. Проверочное число должно описывать $(n_0 + k + 1)$ событий. Следовательно, число k определяется на основании неравенства

$$2^k \geq (n_0 + k + 1)$$

и так как $n = n_0 + k$, то умножив неравенства на $2^{n_0}/n + 1$ получим

$$2^n / n + 1 \geq 2^{n_0}$$

Это соотношение позволяет определить максимальное n_0 при данном n или минимальное n для данного n_0 . Приведем соответствующие значения в табл. 3.2

Таблица 3.2

n	n_0	k	n	n_0	k
1	0	1	6	3	3
2	0	2	7	4	3
3	1	2	8	4	4
4	1	3	9	5	4
5	2	3	10	6	4

Определим теперь позиции, которые надлежит проверить в каждой из k проверок. Если ошибок нет, то на всех проверяемых позициях будет 0, если в низшем разряде числа стоит 1, это значит, что в результате первой проверки обнаружена ошибка. Будем при первой проверке проверять те номера позиций, двоичные представления которых имеют в первом разряде единицы, т.е.

1 = 1
3 = 11
5 = 101
7 = 111
9 = 1001 и т. д.

Таким образом, первая проверка охватывает позиции 1, 3, 5, 7, 9.

Для второй проверки выберем такие позиции, двоичные представления которых имеют единицу во втором разряде:

2 = 10
3 = 11
6 = 110
7 = 111
10 = 1010

Для третьей проверки имеем:

4 = 100
5 = 101
6 = 110
7 = 111
12 = 1100
13 = 1101
14 = 1110

Такой выбор проверяемых позиций дает возможность определить номер позиции, в которой произошла одиночная ошибка. Напомним, что при каждой проверке сумма единиц должна быть четной. Если произошла ошибка на одной из позиций первой проверки, то в проверочном числе в низшем (правом) разряде появится единица, как это и должно быть.

Дальнейшую расшифровку проверочного числа дает вторая проверка: если среди всех позиций второй проверки ошибок нет, то будет появляться нуль. Если на третьей позиции произошла ошибка, нарушившая четность как в первой, так и во второй проверке, то после двух проверок в двух низших разрядах появятся единицы.

В третьей и следующих проверках третья позиция уже отсутствует. Таким образом, в нашем примере проверочное число равно 0011 = 3, что и дает номер позиции, на которой произошла ошибка. Аналогично можно убедиться, что любая одиночная ошибка на любой позиции может быть устранена проверками, дающими проверочное число, равное номеру позиции, на которой произошла ошибка.

В третьей и следующих проверках третья позиция уже отсутствует. Таким образом, в нашем примере проверочное число равно 0011 = 3, что и дает номер позиции, на которой произошла ошибка. Аналогично можно убедиться, что любая одиночная ошибка на любой позиции может быть устранена проверками, дающими проверочное число, равное номеру позиции, на которой произошла ошибка.

Остается решить, какие позиции использовать под проверочные символы. Выбор для проверки позиций 1, 2, 4, 8 ... обеспечивает появление хотя бы одной из этих позиций при каждой проверке, и это позволяет независимо от знаков передаваемого числа получить при каждой проверке четное число единиц. Соответствующие проверяемые числа приведены в табл. 3.3.

Таблица 3.3

Порядковый номер	Проверочная позиция	Проверяемые числа
1	1	1, 3, 5, 7, 9, 11, 13, 15, 17, ...
2	2	2, 3, 6, 7, 10, 11, 14, 15, 18, ...
3	4	4, 5, 6, 7, 12, 13, 14, 15, 20, ...
4	8	8, 9, 10, 11, 12, 13, 14, 15, 25, ...

3.6. Основы теории систем и системотехники

Анализ развития АИС за более чем 40-летний срок позволяет выявить интересные закономерности, а именно, рождение и гибель одних идей, связанных с первыми успехами и постепенное оформление и закрепление в сознании специалистов других, которые используются и сегодня. Эти идеи, прошедшие проверку временем, составляют ядро методов создания и использования автоматизированных систем управления.

Основой, которая позволяет интегрировать все вышеперечисленные методы анализа, описания и разработки АИС, является системный подход. Важнейшие понятия **теории систем** перечислены ниже:

система – множество взаимосвязанных элементов, каждый из которых связан прямо или косвенно с каждым другим элементом, а два любые подмножества этого множества не могут быть независимыми не нарушая целостность, единство системы;

элемент (системы) – простейшая структурная составляющая системы, которая в рамках данной системы не структурируется;

структура (системы) – совокупность устойчивых связей, способов взаимодействия элементов системы, определяющая ее целостность и единство;

интеграция (в системе или систем) – восстановление и (или) повышение качественного уровня взаимосвязей между элементами системы, а также процесс создания из нескольких разнородных систем единой системы, с целью исключения (до технически необходимого минимума) функциональной и структурной избыточности и повышения общей эффективности функционирования;

иерархическая система – система, подсистемы которой занимают последовательные вертикальные уровни и определен приоритет действий (право вмешательства) подсистем верхнего уровня по отношению к нижним уровням.

Согласно общей теории систем, первым этапом при создании любой большой системы является этап разработки **концепции** – основного замысла в научной, технической и других видах человеческой деятельности.

Затем следует этап **формализации**, который заключается в следующем.

1. Вводится определение функциональной системы как отображения:

$S:X \rightarrow Y$ абстрактного множества X в абстрактное множество Y , где X и Y

представляют соответственно множество входов x и множество выходов y . В более общей формулировке система есть отношение $S \subseteq X \times Y$ над абстрактными множествами X и Y . Выражение системы в виде отношения над абстрактными множествами дает формализованное математическое представление интуитивного понятия системы.

2. Вводится формальное определение системы принятия решений (или “решающей системы”). В такой системе сигнал, приходящий на ее вход, конкретизирует “свободные” параметры решаемой проблемы (бывшие ранее неопределенными), и результатом работы системы является решение поставленной проблемы, получаемое на ее выходе.
3. Полученные формализмы используются для описания различных видов иерархических систем, из которых будет строиться проектируемая система.

В иерархических системах особенно важное значение придается проблеме **координации** подсистем, означающей такое воздействие на подсистемы, которое заставляет их действовать согласованно, подобно тому как обычно координируется деятельность индивидуумов или групп внутри некоторой организации. Чтобы сделать такое представление о координации операциональным, нужно более четко определить, что именно подразумевается под словами “действовать согласованно”. В общем случае координация осуществляется в связи с определенной целью или задачей; деятельность частей системы координируется ради общей цели так, чтобы вся система в целом достигла поставленной цели.

Координация – это сфера деятельности или задача вышестоящей управляющей системы, в ходе которой она пытается добиться, чтобы нижестоящие системы управления функционировали согласованно. Успех вышестоящей управляющей системы в осуществлении надлежащей координации оценивается по отношению к общей глобальной цели, подставленной перед всей системой. Так как нижестоящие управляющие системы действуют так, чтобы достичь своих собственных индивидуальных целей, то, вообще говоря, между ними возникает конфликт, который приводит к тому, что скорее всего глобальная цель не будет достигнута. Действия координатора направлены как раз на последствия такого внутрисистемного конфликта, которые он должен постараться, если не полностью исключить, то по крайней мере уменьшить.

Из общей теории систем известно о существовании **синергетического эффекта**, то есть появлении нового качества у сложной системы, которое в явном виде отсутствует у составляющих ее компонентов (элементов и подсистем). Тем не менее, свойства системных компонентов влияют на характеристики синергетического эффекта. Правда, зависимость здесь далеко не прямая. Так, хорошо известно, что используя самые совершенные программно-технические средства для создания прикладных систем, можно получить отрицательный эффект, поэтому анализ и оценка влияния базовых информационных технологий на качество прикладных ИТ имеет первостепенное значение.

Любая АИС представляет собой сложную модель реального объекта из конкретной области человеческой деятельности. При создании АИС, на всех этапах используются методы системного анализа. Согласно представлениям **системного анализа**, объект исследования, который должен быть автоматизирован, выглядит следующим образом. Объект исследования представляет собой сложную динамическую систему, которая включает в себя большое число разнообразных взаимодействующих подсистем. Состояние такой системы, как правило, неустойчиво. При этом отклонения от устойчивого состояния имеют тенденцию накапливаться и в определенное время малые воздействия на систему могут вызвать значительные изменения ее состояния.

Сложная динамическая система может быть линейной или нелинейной. Объект, который имеет нелинейный характер, может находиться в различных, в том числе и неустойчивых стационарных состояниях, соответствующих различным законам поведения. Поведение таких систем нельзя уложить в одну теоретическую схему. Наблюдаемые параметры таких систем подвержены случайным отклонениям от средних значений – флуктуациям, при этом в области неустойчивости флуктуации подвержены положительной обратной связи, что может привести к разрушению системы. Такой критический момент в функционировании системы называется **точкой бифуркации**.

На основании вышесказанного формулируются методы, применяемые при системном подходе к созданию АИС, последовательность которых перечислена ниже:

- объект автоматизации представляет собой сложную, динамическую, нелинейную систему;

- проводится системный анализ его функционирования в окружающей среде;

- информационные технологии рассматриваются как один из самых мощных факторов развития и функционирования объекта;

- проводится системный анализ целей, задач и проблем системы и свойств информационных технологий (фактора развития) и разрабатывается **концепция** и рекомендации по использованию ИТ;

- концепция и рекомендации, методы и средства должны быть использованы при разработке научно-технической программы – официального, руководящего документа, в котором представлен основной замысел решения проблем с помощью информационных технологий, а также система взаимоувязанных целей, задач, направлений, методов и средств информатизации системы на прогнозируемый период.

Для реализации масштабных и сложных проектов, к которым относятся программы создания АИС мировая и отечественная наука и практика выработали ряд основополагающих принципов:

- 1) программа должна состоять из творческих и перспективных решений и быть направлена на одновременную реализацию политических, экономических, социальных и научно-технических идеалов;

- 2) программа должна быть четко структурирована, а результаты реализации промежуточных этапов должны быть самооценны сами по себе и инициировать другие проекты;

- 3) должен быть принят достаточно широкий взгляд на программу, чтобы охватить все элементы, средства и методы с единых интеграционных позиций;

- 4) необходимо держать в поле зрения национальные и зарубежные проекты, реализуемые в данной предметной области и по возможности кооперироваться с этими проектами;

- 5) основная цель программы должна быть сбалансирована по политическим, социальным, научно-техническим требованиям и экономическим факторам, при этом необходимо установить промежуточные краткосрочные цели, с которыми можно соотноситься как с критериями оценки проекта;

- 6) необходимо четко уяснить соотношение между составляющими программы создания АИС, исследовательскими темами и программой в целом, на каждом этапе должны рассматриваться несколько альтернатив и выбор обосновываться;

- 7) для непрерывного управления и контроля за выполнением программы следует создать организационное ядро из представителей различных структур, имеющих отношение к проблеме, эта группа должна состоять из небольшого числа лиц и сохраняться неизменной на весь период выполнения программы;

8) должен соблюдаться принцип целостности, основные идеи должны быть ясно выражены и задокументированы, никаких изменений не должно производиться, пока не будет выявлена очевидная ошибка.

Прикладные вопросы общей теории систем являются предметом инженерной дисциплины системотехники. **Системотехника** – это совокупность методов и средств для создания больших технических систем, которые основываются на теории систем. Разработка обычных технических систем опиралась на детальный анализ и углубляющуюся специализацию задач и методов. Большие системы, напротив, предполагают интеграцию, рассмотрение различных сторон явлений, о чем и было сказано выше.

ЗАДАНИЯ ДЛЯ САМОСТОЯТЕЛЬНОЙ РАБОТЫ

- 1. Составьте логическую схему базы знаний по теме юниты.*

2. Найдите и исправьте ошибку в записи сочетательного закона теоретико-множественного умножения $(A \cap B) \cap C = A \cap (B \cup C)$.

3. Запишите условия равенства двух множеств A и B .

4. Дополните приведенную таблицу равнозначности логических значений A и B .

A	B	$A \wedge B$	$A \vee B$
0	0		
0	1		
1	0		
1	1		

5. Дополните приведенную таблицу логического сложения и умножения

$$\begin{array}{ll}
 0 \vee 0 = & 0 \wedge 0 = \\
 0 \vee 1 = & 0 \wedge 1 = \\
 1 \vee 0 = & 1 \wedge 0 = \\
 1 \vee 1 = & 1 \wedge 1 =
 \end{array}$$

6. Запишите десятичную позиционную систему счисления в виде формального языка, используя для этого нотацию Бекуса.

7. Определите максимально возможную энтропию системы из трех элементов, каждый из которых может находиться в 4-х возможных состояниях.

8. Дополните таблицу оценки импликации

$$0 \rightarrow 0 =$$

$$0 \rightarrow 1 =$$

$$1 \rightarrow 0 =$$

$$1 \rightarrow 1 =$$

9. Дополните таблицу оценки эквивалентности

$$0 \sim 0 =$$

$$0 \sim 1 =$$

$$1 \sim 0 =$$

$$1 \sim 1 =$$

ТРЕНИНГ УМЕНИЙ

1. Пример выполнения упражнения тренинга на умение № 1

Задание

Рассчитать общее число сообщений и количество информации при заданном наборе символов.

Заданы символы a, b, c ; число символов $m = 3$; веса символов в сообщении $n_1=2, n_2=3, n_3=1$.

Решение

Предварительно заполните таблицу, подобрав к каждому алгоритму конкретное соответствие из данного задания.

№ п/п	Алгоритм	Конкретное действие, соответствующее предложенному алгоритму
1	Найти количество символов в сообщении	Количество символов в сообщении $n = n_1 + n_2 + n_3 = 2 + 3 + 1 = 6$
2	Составить комбинаторную формулу числа сообщений	Комбинаторная формула числа сообщений N равна: $N = \frac{n!}{n_1! \cdot n_2! \cdot n_3!}$
3	Подсчитать общее число сообщений	Общее число сообщений равно $N = \frac{1 \cdot 2 \cdot 3 \cdot 4 \cdot 5 \cdot 6}{1 \cdot 1 \cdot 2 \cdot 1 \cdot 2 \cdot 3} = 3 \cdot 4 \cdot 5 = 60$
4	Вычислить количество информации при заданных параметрах	Количество информации при длине сообщения $n = 6$ будет равно: $\log_2 60 \approx 6 \text{ бит}$

Решите самостоятельно следующие задания:

Задание 1.1

Рассчитать общее число сообщений и количество информации, если заданы символы: a, b, c, d, число символов $m = 4$ и веса символов - $n_1 = 2, n_2 = 1, n_3 = 4, n_4 = 3$.

Задание 1.2

Рассчитать общее число сообщений и количество информации, если заданы символы: a, b, c, d, число символов $m = 4$ и веса символов - $n_1=4$, $n_2=2$, $n_3= 3$, $n_4=2$.

Задание 1.3

Рассчитать общее число сообщений и количество информации, если заданы символы: a, b, c, число символов $m = 3$ и веса символов - $n_1=5$, $n_2=3$, $n_3= 2$.

2. Пример выполнения упражнения тренинга на умение № 2

Задание

Определить максимально возможную энтропию сообщения, состоящего из заданного числа символов при фиксированном алфавите.

Задано 5 символов, при алфавите - 32 символа.

Решение

Предварительно заполните таблицу, подобрав к каждому алгоритму конкретное соответствие из данного задания.

№ п/п	Алгоритм	Конкретное действие, соответствующее предложенному алгоритму
1	Определить число возможных состояний системы	Число возможных состояний системы N равно: $N = 32^5$
2	Записать выражение для подсчета энтропии	Выражение для энтропии будет: $H = \log_2 N = \log_2 32^5$
3	Преобразовать выражение и вычислить	$H = \log_2 32^5 = 5 \cdot \log_2 32 = 5 \log_2 2^5 = 25 \cdot \log_2 2 = 25$ бит

Решите самостоятельно следующие задания:

Задание 2.1

Определить максимально возможную энтропию сообщения из 6 символов, при алфавите - 64 символа.

Задание 2.2

Определить максимально возможную энтропию сообщения при 4 символах и при алфавите - 128 символов.

Задание 2.3

Определить максимально возможную энтропию сообщения из 6 символов, при алфавите - 16 символов.

3. Пример выполнения упражнения тренинга на умение № 3

Задание

Произвести свертку двух отношений (свертку де Моргана). Свертку обозначить знаком *

Заданы отношения:

$$F_1 = \{(a, 1), (a, 2), (a, 3), (b, 1), (b, 2), (v, 1), (v, 4)\}$$

$$F_2 = \{(1, x, A), (1, y, B), (2, z, A), (3, x, B)\}$$

Решение

Предварительно заполните таблицу, подобрав к каждому алгоритму конкретное соответствие из данного задания.

№ п/п	Алгоритм	Конкретное действие, соответствующее предложенному алгоритму
1	Отобрать по одному кортежу из 1-го и 2-го отношений с совпадающими последним и первым элементами	$(a, 1) - (1, x, A); (a, 1) - (1, y, B)$
2	Отбросить совпадающие элементы, а оставшиеся объединить в новый кортеж	Кортежи свертки: $(a, x, A); (a, y, B)$
3	Продолжать операцию до исчерпания всех кортежей 1-го отношения	2-й кортеж из F_1 : $(a, 2) - (2, z, A)$ дает (a, z, A) 3-й кортеж из F_1 : $(a, 3) - (3, x, B)$ дает (a, x, B) 4-й кортеж из F_1 : $(b, 1) - (1, x, A); (b, 1) - (1, y, B)$ дает: $(b, x, A); (b, y, B)$ 5-й кортеж из F_1 : $(b, 2) - (2, z, A)$ дает (b, z, A) 6-й кортеж из F_1 : $(v, 1) - (1, x, A); (v, 1) - (1, y, B)$ дает $(v, x, A); (v, y, B)$ 7-й кортеж из F_1 (b, y) не образует кортежа свертки (нет пары)
4	Собрать все полученные кортежи в отношение свертки	$F_1 * F_2 = \{(a, x, A), (1, B, y), (a, z, A), (a, x, B), (b, x, A), (b, y, B), (b, z, A), (v, x, A), (v, y, B)\}$

Решите самостоятельно следующие задания:

Задание 3.1

Произвести свертку двух нижезаданных отношений (свертку де Моргана)

Задание 3.2

Произвести свертку двух нижезаданных отношений (свертку де Моргана)

Задание 3.3

Произвести свертку двух нижезаданных отношений (свертку де Моргана)

ОСНОВЫ АВТОМАТИЗИРОВАННЫХ ИНФОРМАЦИОННЫХ СИСТЕМ

ЮНИТА 1

ТЕОРЕТИЧЕСКИЕ ОСНОВЫ АВТОМАТИЗИРОВАННЫХ ИНФОРМАЦИОННЫХ СИСТЕМ

Редактор Л.А. Савина
Оператор компьютерной верстки Д.В. Федотов

Изд. лиц. ЛР № 071765 от 07.12.1998
НОУ “Современный Гуманитарный Институт”
Тираж

Сдано в печать
Заказ

Современный Гуманитарный Университет