

Министерство образования и науки РФ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
АМУРСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
(ФГБОУ ВО «АмГУ»)

**ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ
ОБРАБОТКИ И АНАЛИЗА ДАННЫХ**

сборник учебно-методических материалов

для направления подготовки 38.03.05. – Бизнес-информатика

Благовещенск, 2017

Печатается по решению
редакционно-издательского совета
факультета математики и информатики
Амурского государственного
Университета

Составитель: Жилиндина О.В.

Информационные технологии обработки и анализа данных: сборник учебно-методических материалов для направления подготовки 38.03.05. – Благовещенск: Амурский гос. ун-т, 2017.

© Амурский государственный университет, 2017
© Кафедра Информационных и управляющих систем, 2017
© Жилиндина. О.В., составление

1. Краткое изложение лекционного материала

1. Предпосылки использования информационных технологий в управлении. Введение в историю проблемы «человек-машина». Взгляды Н. Винера, А. Тьюринга. История дискуссии «Может ли машина мыслить?». Тест Тьюринга. Взлеты и падения искусственного интеллекта. Как человек и машина решают одну и ту же проблему. Понятие управления Теоретические подходы к понятию управления появились как естественное развитие системного анализа. Логически это понятно - после того как сформулировано понятие системы и выбран математический аппарат для описания системы, сразу же возникает вопрос о том, как можно системой управлять. Здесь сказал свое веское слово Норберт Винер. Норберт Винер – отец кибернетики Родители Норберта были еврейскими иммигрантами, выходцами из небольшого городка Белосток в Польше (тогда входила в состав Российской Империи). На исходе девятнадцатого столетия они покинули всё ещё внешне спокойную и вполне благополучную Россию, и перебрались в Штаты. В 4 года Винер уже был допущен к родительской библиотеке, а в 7 лет написал свой первый научный трактат по дарвинизму. Норберт никогда понастоящему не учился в средней школе. Зато 11 лет от роду он поступил в престижный Тафт-колледж, который закончил с отличием уже через три года получив степень бакалавра искусств. В 18 лет Норберт Винер уже числился доктором наук по специальности «математическая логика» в Корнельском и Гарвардском университетах. В девятнадцатилетнем возрасте доктор Винер был приглашён на кафедру математики Массачусетского технологического института. В 1913 году молодой Винер начинает своё путешествие по Европе, слушает лекции Рассела и Харди в Кембридже и Гильберта в Гёттингене. После начала войны он возвращается в Америку. Перед второй мировой войной Винер стал профессором Гарвардского, Корнельского, Колумбийского, Брауновского, Геттингенского университетов, получил в собственное безраздельное владение кафедру в Массачусетском институте, написал сотни статей по теории УМК.3(500) – 080502 – СД.08.7/9 – 2007 2 вероятностей и статистике, по рядам и интегралам Фурье, по теории потенциала и теории чисел, по обобщённому гармоническому анализу...

Практическое воплощение Во время второй мировой войны, на которую профессор пожелал быть призванным, он работает над математическим аппаратом для систем наведения зенитного огня (детерминированные стохастические модели по организации и управлению американскими силами противовоздушной обороны). Он разработал новую действенную вероятностную модель управления силами ПВО. Основной научный труд Винера - «Кибернетика» Винера увидела свет в 1948 году. Полное название главной книги Винера выглядит следующим образом «Кибернетика, или управление и связь в животном и машине». За пару месяцев до смерти Норберт Винер был удостоен Золотой Медали Учёного, высшей награды для человека науки в Америке. Он тихо умер 18 марта 1964 года в Стокгольме. Оставил после себя книги воспоминаний «Записки бывшего вундеркинда», «Я - математик» Кибернетика Объектом кибернетики являются все управляемые системы. Системы, не поддающиеся управлению, в принципе, не являются объектами изучения кибернетики. Кибернетика вводит такие понятия, как кибернетический подход, кибернетическая система и прочие. Кибернетика является междисциплинарной наукой, призванной объединить и систематизировать знания тех областей, которые до сих пор было принято считать различными и несовместимыми. Данная цель достигается в кибернетике за счёт анализа и выявления общих принципов и подходов в процессе научного познания.

Наиболее весомыми теориями, объединяемыми кибернетикой, можно назвать следующие: • теория передачи сигналов • теория информации • теория систем • теория управления • теория автоматов • теория принятия решений • синергетика Кроме средств анализа, кибернетика предлагает мощные инструменты для синтеза решений, предоставляемые аппаратами математического анализа, линейной алгебры, геометрии выпуклых множеств, теории вероятностей и математической статистики. А также более высокоприкладные области математики, такие, как математическое программирование, эконометрика, информатика и прочие производные дисциплины. Направления • Биологическая кибернетика и медицинская кибернетика • Техническая и инженерная кибернетика • Экономическая кибернетика, социальная кибернетика Одно из ключевых понятий теории управления - "цель управления", т. к. бесцельных управлений не бывает. В технических системах цель

не принадлежит системе, а задается, т. е. это внешний по отношению к системе фактор. 3 Управление - процедура выбора и реализации определенных целенаправленных действий. Кибернетика экономическая, научное направление, занимающееся приложением идей и методов кибернетики к экономическим системам. Экономическая кибернетика рассматривает экономику, а также её структурные и функциональные звенья как системы, в которых протекают процессы регулирования и управления, реализуемые движением и преобразованием информации. Методы экономической кибернетики дают возможность стандартизировать и унифицировать эту информацию, рационализировать получение, передачу и обработку экономической информации, обосновать структуру и состав технических средств её обработки. Именно такой подход определяет внутреннее единство и характер исследований в рамках экономической кибернетики. Они служат, в частности, теоретической основой создания автоматизированных систем управления (АСУ) и систем обработки данных (СОД).

Экономическая кибернетика развивается по трём основным направлениям, которые всё более тесно увязываются друг с другом. Теория экономических систем и моделей разрабатывает: методологию системного анализа экономики и её моделирования, отражения структуры и функционирования экономических систем в моделях; вопросы классификации и построения комплексов экономико-математических моделей; проблемы экономического регулирования, соотношения и взаимного согласования различных стимулов и воздействий в функционировании экономических систем; вопросы поведения людей и коллективов.

При исследовании этих проблем экономическая кибернетика, прежде всего, опирается на политическую экономию и общую теорию систем, а также на социологию и теорию регулирования, обобщает результаты разработки экономикоматематических методов и моделей. Теория экономической информации рассматривает экономику как информационную систему. Она изучает: потоки информации, циркулирующие в народном хозяйстве как коммуникации между его элементами и подсистемами, характеристики информационных каналов и передаваемых по ним сообщений; экономические измерения и вообще знаковые системы в экономике, то есть языки экономического управления, включая разработку комплексов хозяйственных показателей, правил их расчёта (эти вопросы выделяются в экономическую семиотику); процессы принятия решений и обработки данных в информационных системах народного хозяйства на всех его уровнях и вопросы наилучшей организации этих процессов. Здесь экономическая кибернетика тесно соприкасается с теорией информации, исследованиями по определению полезности или ценности информации, семиотикой, теорией программирования, информатикой.

Теория управляющих систем в экономике конкретизирует и сводит воедино исследования остальных разделов экономической кибернетики. Она направлена на комплексное изучение и совершенствование системы управления народным хозяйством и отдельными хозяйственными объектами, а в конечном счете, — на их оптимальное функционирование. Особое внимание уделяется: проблемам планирования и руководства реализацией планов — методологии, технологии и организации этих функций управления, использованию комплексов экономико-математических моделей и других научных методов в практике управления; разработке внутренне согласованного комплекса экономических, административных, правовых и других 4 стимулов и норм управления, построению организационных структур органов управления; изучению и учету человеческих факторов (социальнопсихологических и т. п.) в процессах хозяйственного управления, взаимодействию человека и машины в АСУ; проблемам проектирования и внедрения АСУ в целом. Экономическая кибернетика рассматривает АСУ не как «пристройку» к тем или иным органам управления для обработки данных, а как саму систему управления хозяйственным объектом, основанную на комплексном применении экономикоматематических методов и моделей, современной информационно-вычислительной техники — с соответствующей технологией и организацией её работы. Современное состояние

С 80-х годов можно считать, что технология решения задач, опирающаяся на идею использования знаний о предметной области, где возникла задача, и знаний о том, как решаются подобные задачи, характерная для работ по интеллектуальным системам, стала основной парадигмой для современной информатики. Но это уже относится к тому периоду, который наступил после завер-

шения в середине 70-х начального этапа развития информатики в нашей стране. Термин "информатика" в 80-е годы получает широкое распространение, а термин "кибернетика" постепенно исчезает из обращения, сохранившись лишь в названиях тех институтов, которые возникли в эпоху "кибернетического бума" конца 50-х - начала 60-х годов. В названиях новых организаций термин "кибернетика" уже не используется. Алан Тьюринг родился в 1912 году, 23 июня.

В школе он испытывал тягу к техническим наукам и в пятнадцать лет самостоятельно разобрался в теории относительности, при этом совершенно не хотел заниматься языками, латынью и пр. Дирекция школы терпела долго, но однажды все-таки направила матери Тьюринга записку следующего содержания: «Ваш сын, видимо, хочет быть только научным специалистом. Может быть, математиком — такие ученики, как он, рождаются раз в 200 лет. Но что он вообще забыл в Public School?». Машина Тьюринга Алана Тьюринга чрезвычайно вдохновляла идея о том, что для решения любой проблемы не надо думать, и уж тем более - угадывать. Достаточно разработать определенный метод. Если механически следовать этому порядку действий, правильное решение можно будет получить гораздо быстрее.

Тьюринг разработал свои собственные логические ходы и ввел понятие «определяющего метода», который позже получил название «алгоритм». В 1936 г. Алан построил логическую модель своей знаменитой машины Тьюринга. Надо сказать, что в те годы под словом Computer подразумевался человек, проводящий однообразные вычисления по определенным инструкциям. Например, так называли бухгалтеров, счетоводов и т.п. Идея Алана Тьюринга была в том, что для проведения подобных действий присутствие человека не требуется. Машина Тьюринга — это очень простое вычислительное устройство. Она состоит из ленты бесконечной длины, разделенной на ячейки, и головки, которая перемещается вдоль ленты и способна читать и записывать символы. Также у машины Тьюринга есть такая характеристика, как состояние, которое может выражаться целым числом от нуля до некоторой максимальной величины. В зависимости от состояния машина Тьюринга может выполнить одно из трех действий: записать символ в ячейку, передвинуться на одну ячейку вправо или влево и установить внутреннее состояние. 5 Устройство машины Тьюринга чрезвычайно просто, однако на ней можно выполнить практически любую программу.

Для выполнения всех этих действий предусмотрена специальная таблица правил, в которой прописано, что нужно делать при различных комбинациях текущих состояний и символов, прочитанных с ленты. До сих пор машина Тьюринга служит моделью для изучения теоретических проблем в области математических алгоритмов. Бесконечно длинная лента поделена на дискретные сегменты, в каждом из которых записан 0 или 1. "Головка для считывания и записи", которая может находиться в любом из нескольких внутренних состояний (здесь только два состояния: A и B), перемещается вдоль ленты. Каждый цикл начинается с того, что головка считывает один бит с сегмента ленты. Затем, в соответствии с фиксированным набором правил перехода, она записывает в сегмент ленты бит данных, изменяет свое внутреннее состояние и перемещается на одну позицию влево или вправо. Поскольку данная машина Тьюринга обладает всего двумя внутренними состояниями, ее возможности ограничиваются лишь тривиальными вычислениями. Более сложные машины с большим числом состояний способны смоделировать поведение любой ЭВМ, в том числе и значительно более сложной, чем они сами.

Это оказывается возможным благодаря тому, что они хранят полное представление логического состояния большей машины на бесконечной ленте и разбивают каждый вычислительный цикл на большое количество простых шагов. Показанная здесь машина логически обратима: мы всегда можем определить предшествующие состояния машины. Машины Тьюринга, обладающие другими правилами перехода, могут и не быть логически обратимыми. Вклад в криптографию (время Второй Мировой войны) Во время войны Тьюринг был призван на секретную работу в британское криптоаналитическое бюро (Государственный институт кодов и шифров). Бюро занималось перехватом и расшифровкой немецких шифровок. В то время как немецкие криптографы действовали по наитию, подходя к криптографии прежде всего как к искусству, Алан Тьюринг использовал научный и алгоритмический подход, основанный на статистическом анализе данных. Для шифрования немцы использовали специальный трехдисковый шифратор «Энигма», а британ-

ские ученые создали специальный вычислитель «Бомба», с помощью которого, начиная с 1940 года, расшифровывали все переговоры Люфтваффе. К концу войны они построили ряд более мощных вычислителей и расшифровали шифр Верховного главнокомандования.

Компьютерный интеллект Алан Тьюринг первым поставил вопрос о том, что такое компьютерный интеллект. В 1950 г. он опубликовал свою знаменитую статью под названием «Может ли машина думать?» (Can the machine think?). В ней Тьюринг рассмотрел два основополагающих определения — «машина» и «мыслить». Но если размышления ученого по первому пункту были актуальны лишь для его современников, то разработанный Тьюрингом критерий разумности вычислительных систем используется до сих пор. В упрощенном изложении тест Тьюринга формулируется таким образом: «Судья и подопытный общаются между собой с помощью телеграфа (чтобы исключить узнавание по голосу). Судья может задавать подопытному любые вопросы. Если на основании ответов подопытного он не может ответить на вопрос – кто общается с ним человек или компьютер, то компьютер обладает искусственным интеллектом». По поводу искусственного интеллекта Тьюринг истывал чрезвычайный оптимизм. «И лишь в одном случае нельзя будет построить мыслящую машину, — писал в своей статье Тьюринг, — если человеческое общение основано на психокинезе и способности к прорицанию. В этом случае науке просто предстоит изучить подобные явления и смоделировать их». Алан Тьюринг прогнозировал, что в ходе пятиминутного теста компьютер с памятью около 126 Мб сможет обмануть человека в 30% случаев. Что показало время Сегодня компьютеры далеко перешли рубеж в 128 Мб памяти. И даже существует множество чат-ботов, которые могут поддерживать общение. До сих пор остается открытым вопрос, насколько адекватно тест Тьюринга определяет разумность машины. В сегодняшней его интерпретации он позволяет оценить лишь имитационные способности программы — то есть насколько хорошо она притворяется человеком. В 2001 г. утешительный приз за человекоподобие достался программе ALICE Ричарда Уоллеса, которая смогла убедить более 30% судей в своей немашинной природе. Развитие подхода Термин «искусственный интеллект» датируется 1955 г., когда Джон МакКарти (John McCarthy) предложил его на конференции в Дартмутском университете. Казалось, что огромный успех неизбежен. Так, Герберт Саймон (Herbert Simon, 1965) смело предсказал, что «в пределах самого ближайшего будущего — значительно меньше, чем двадцать пять лет, — мы будем иметь техническую возможность заменить машиной любую функцию человека во всех организациях». Такое волнение первых лет подпитывалось рядом успешных демонстраций ИИ в ряде ограниченных областей. Авторы системы одной из систем ИИ Ньювелл и Саймон утверждали, что при достаточно высоком уровне обобщения все задачи выглядят одинаково: это всегда проблема перехода от начального состояния к желательному. Поэтому, имея проблемно-ориентированный набор операторов (каждый из которых может осуществлять переход от одного такого состояния к другому), с помощью единственного общего метода можно решить какую бы то ни было задачу, используя общую эвристику сокращения различий. Искусственный интеллект сейчас

Цель искусственного интеллекта (ИИ) состоит в построении компьютерных систем, которые могут продемонстрировать уровень интеллекта, подобный 7 человеческому разуму. Начиная с компьютерных программ, понимания естественного языка, игр, доказательства теорем, распознавания образца, обучающихся машин и робототехники, ИИ превратился в зрелую дисциплину, продукты которой имеют широкое практическое применение. Однако цель не была достигнута: каждая из систем ИИ обычно компетентна только в одной узкой предметной области. Экспертные системы интенсивно развиваются, начиная с 60-х годов прошлого века. Вот примеры промышленных экспертных систем, все ни основаны на формальной логике: • MICIN — экспертная система для медицинской диагностики. Разработана группой по инфекционным заболеваниям Стенфордского университета. Ставит соответствующий диагноз, исходя из представленных ей симптомов, и рекомендует курс медикаментозного лечения любой из диагностированных инфекций. База данных состоит из 450 правил. • PUFF — анализ нарушения дыхания.

Данная система представляет собой MICIN, из которой удалили данные по инфекциям и вставили данные о легочных заболеваниях. • DENDRAL — распознавание химических структур. Данная система старейшая, из имеющих звание экспертных. Первые версии данной системы поя-

вились еще в 1965 году во все том же Стенфордском университете. Пользователь дает системе DENDRAL некоторую информацию о веществе, а также данные спектроскопии (инфракрасной, ядерного магнитного резонанса и масс-спектрометрии), и та в свою очередь выдает диагноз в виде соответствующей химической структуры. • PROSPECTOR — экспертная система, созданная для содействия поиску коммерчески оправданных месторождений полезных ископаемых. К 90-м годам эти системы породили столь большую волну общественных ожиданий к ИИ, что вызвали инвестиционный бум в эту область знаний. Инвестиционный бум достаточно быстро прекратился по двум причинам: • неспособность экспертных систем дать сколько-нибудь полезные промышленные результаты, • появление конкурирующих промышленных технологий: Интернет, мобильная связь. В целом попытки разработки систем искусственного интеллекта выявили следующие проблемы. Принципиальные особенности человеческого мышления • формальная дедуктивная логика — довольно бедная модель человеческого мышления, (например человек легко оперирует с утверждениями «все птицы летают», «курица это птица» и «курица не летает», хотя с точки зрения 1 Основная цель (функция) логики всегда оставалась неизменной: исследование того, как из одних утверждений можно выводить другие.

При этом предполагается, что вывод зависит только от способа связи входящих в него утверждений и их строения, а не от их конкретного содержания. Изучая, «что из чего следует», логика выявляет наиболее общие или, как говорят, формальные условия правильного мышления. Формальная логика — наука об общих структурах и законах правильного мышления, образования и сочетания понятий и высказываний, о правилах умозаключений независимо от их конкретного содержания. Начало формальной логике положила силлогистика Аристотеля. Вот пример аристотелевского силлогизма «Всякий человек смертен», «Сократ — человек», следовательно «Сократ смертен». 8 формальной логики эти утверждения противоречивы), однако на сегодня другой модели мышления не существует; • формальная логика имеет свойство монотонности — это означает, по мере приближения к цели количество вариантов, которое надо просчитать для получения правильного ответа может только возрастать, человеческая логика принципиально немонотонна; • мы даже не знаем, откуда у нас появляются идеи, гипотезы, решения и схемы; • самые значительные успехи были достигнуты тогда, когда моделируемые миры были крошечными, но такие системы становились несовершенными, когда предпринимались попытки увеличить их до масштаба реального объема элемента, объединить данные большого количества датчиков или найти обобщение путем объединения нескольких предметных областей²; • сложно подобрать структурированные представления знаний, которые были бы достаточно гибкими для того, чтобы моделировать разумный человеческий навык оценки соответствия (человек может мгновенно загрузить огромные области знания в свою «рабочую память» или достать их оттуда, если что-то сделает их значимыми); • можно построить обучающуюся машину, но сегодняшние ИИ-системы требуют огромных человеческих вложений в обучение (в инжиниринг знаний) и настройку исполнения; • хотя сегодняшние настольные компьютеры предлагают такие объемы запоминающих устройств и производительность, которые в тысячи раз превышают показатели машин, использовавшихся пионерами, мы все еще рассматриваем ИИ, устроенный в миллионы раз проще, чем мозг человека. Хотя ум может в значительной степени компенсироваться мощностью, мы все еще пытаемся моделировать человеческий разум с процессорами таракана. Против ярких сторонников ИИ свой знаменитый аргумент «китайская комната» выдвинул Сирли (Searle, 1984).

По аналогии с компьютером он предположил комнату, в которой рабочие без знания китайского языка переводят тексты с китайского на английский, рабски следуя набору правил. Так как в соответствии с гипотезой ни одно существо внутри комнаты не понимало китайского, Сирли, проводя аналогию, говорил, что ни одна ИИ-программа не может понимать свою тематику. Тем не менее, поиск ИИ привел к значительным достижениям в области информатики и информационных технологий в дополнение к их широко распространенным практическим применениям. Это стало причиной интригующих философских проблем. Цель во многом изменилась.

Машинный интеллект теперь не рассматривается как копия человеческого разума, хотя многие все еще полагают, что мы однажды создадим интеллекты, превосходящие наши. ² Здесь уместно упомянуть провал широко разрекламированного в свое время (90-е годы XX века) японского

проекта по созданию компьютеров V поколения, известного в Европе как «японская перчатка». Речь шла о создании компьютеров, у которых элементарной единицей операции было бы логическое умозаключение (сейчас у любого компьютера элементарная операция – действие с целыми числами). Проект провалился по совершенно неожиданной причине: если мир высказываний не замкнут, то ни к одному высказыванию нельзя построить однозначно понимаемое отрицание. Например, есть высказывание «я читаю эту книгу». С формальной точки зрения, чтобы получить отрицание, но поставить в высказыванию частицу «не», но ее поставить в разных местах «не я читаю эту книгу», «я не читаю эту книгу», «я читаю не эту книгу», «я читаю эту (но) не книгу». С точки зрения формальной логики все эти отрицания правомерны, но имеют разный смысл. 9 Самые большие выгоды от разработок ИИ — это сопутствующие результаты. Непосредственно или косвенно разработки инспирировали успехи в компьютерных языках, проектировании баз данных, объектно-ориентированном программировании, параллелизме, нечеткой логике и распознавании образов, воспроизвели эволюцию, генетические алгоритмы и искусственную жизнь.

В области человеко-машинного взаимодействия никто не ведет речь о замене человека. Вектор развития изменился. Первым это положение зафиксировал Уолтер Эшби, который выдвинул идею «усилителя интеллекта». Затем эта идея появлялась под аналогичными названиями «партнерская система» и пр.

Тема 2. Информационные технологии как необходимое условие менеджмента Становление менеджмента как науки. Процесс принятия решения с точки зрения менеджмента. Стратегическая роль информации. Развитие инструментальных средств сбора и обработки информации. Концепция хранилищ данных - Data Warehouse. Распределенные информационные ресурсы. Этапы развития менеджмента в мире Первый этап развития менеджмента начался в начале двадцатого столетия и связан с учением Ф.Тейлора после публикации его книги "Принципы научного управления", в которой он впервые рассмотрел научные подходы и принципы построения системы управления. Используя систему управления, разработанную Ф.Тейлором, американские фирмы и Америка в целом наглядно продемонстрировали ее практическую значимость и влияние на развитие экономики. По оценке Питера Друкера: «С тех пор как Тейлор стал внедрять свои принципы, производительность труда в развитых странах увеличилась раз в пятьдесят.

Этот беспрецедентный рост и явился основой для повышения материального благосостояния и улучшения качества жизни населения передовых стран... К 1930 г. система научного управления Тейлора вопреки сопротивлению со стороны профсоюзов и интеллигенции получила широкое распространение во всех развитых странах...капитализм и промышленная революция принесли выгоды прежде всего рабочим, а не капиталистам. Этим и объясняется полный провал марксизма в высокоразвитых странах...» Интересно, что появившись в свет в 1911 году книга Фредерика Тейлора уже в 1912 году была издана в России на русском языке, в 1913 году заклеена В.И. Лениным как «Научная» система выжимания пота», а в 1918 году рекомендована им же для внедрения: "...Учиться работать — эту задачу Советская власть должна поставить перед народом во всем ее объеме. Последнее слово капитализма в этом отношении, система Тейлора, — как и все прогрессы капитализма, — соединяет в себе утонченное зверство буржуазной эксплуатации и ряд богатейших научных завоеваний в деле анализа механических движений при труде, изгнания лишних и неловких движений, выработки правильнейших приемов работы, введения наилучших систем учета и контроля и т. д.

Советская республика во что бы то ни стало должна перенять все ценное из завоеваний науки и техники в этой области.... Надо создать в России изучение и преподавание системы Тейлора, систематическое испытание и приспособление ее». Система Тейлора или тейлоризм получила широкое распространение в нашей стране в годы советской власти. Она была известна под названием НОТ – научная организация труда. 10 Естественно от системы Тейлора была отсечена ее капиталистическая составляющая – оплата труда. Сам Ф. Тейлор считал, что: «Главнейшей задачей управления предприятием должно быть обеспечение максимальной прибыли для предпринимателя, в соединении с максимальным благосостоянием для каждого занятого в предприятии работника».

Система Тейлора (тейлоризм) основана на глубоком разделении труда, рационализации трудовых движений. Тейлор разработал методы, по которым для каждого вида работы, для каждой операции с помощью хронометража и тщательного изучения движений рабочего устанавливался единственный, самый рациональный способ выполнения заданной работы. Учитывая важность применения на практике методов научной организации труда (НОТ), Ф.Тейлор сформулировал на основе этих методов новые обязанности администрации. Среди обязанностей администрации он выделил следующие: • выработка научного фундамента для каждого отдельного действия во всех разновидностях труда с установлением строгих правил для каждого движения, усовершенствование и стандартизация всех орудий и условий труда; • тщательный отбор рабочих, их последующая тренировка, обучение и развитие с целью получения высококвалифицированных работников; • сотрудничество с рабочими в целях достижения соответствия всех отдельных отраслей производства ранее выработанным администрацией научным принципам, а также обязательное поощрение рабочих за ускоренную работу и за точное выполнение ими производственных заданий; • равномерное распределение труда и ответственности между администрацией и рабочими (администрация берет на себя те отрасли труда, для которых она является лучше приспособленной).

Важной особенностью системы Тейлора является ее практическая реализация посредством конкретных методов, так называемая «техника системы», в которую входят: • калькуляция производственных затрат; • дифференциальная оплата труда; • введение инструкционных карт; • образование распределительных бюро — определение видов работы и расстановка по ним исполнителей; • стандартизация движений, рабочих операций, инструмента и орудий труда; • подбор функциональных мастеров — по регулированию конфликтов и дисциплине, плано-распределительным работам, ремонту оборудования, заработной плате и нормированию, проектированию работы; • определение и точный учет рабочего времени и решение в связи с этим проблемы нормирования труда.

Основным недостатком системы Ф.Тейлора представляется то, что она в своем первоначальном виде была рассчитана на дисциплинированных рабочих. Тем не менее, на современном этапе развития науки и практики управления персоналом принципы научного управления по-прежнему являются прогрессивными и актуальными. Кроме того, Тейлор пришел к важному выводу, что главная причина низкой производительности кроется в несовершенной системе стимулирования рабочих. Для поддержания у рабочего постоянного ожидания награды Тейлор предложил использовать «прогрессивную» систему оплаты труда. Другой представитель «организационной школы» - Г. Форд (1863—1947 гг.), в свое время названный «автомобильным королем». По мнению экспертов, именно он совершил «революцию в цехе», изобретя конвейер для производства автомобилей. Приоритет в этой системе отдавался технологии и технике, в которые «вписывали» человека. Главные идеи Г. Форда представлены в трудах «Моя жизнь, мой труд» (1922 г.), «Сегодня, завтра» (1926 г.), «Движение вперед» (1930 г.), «Эдисон, каким я его знал» (1930 г.). Следует заметить, что в эпоху индустриализации СССР отдельные руководители, инженеры и рабочие проходили стажировку на заводах Г. Форда, изучая технику, управление производством и организацию труда.

К числу главных принципов системы Г. Форда относятся: • экономический эффект системы; • ведущая роль технико-технологической системы; • точность как стандарт и качество продукции; • новая технология на базе поточного производства; • максимально высокие темпы работы; • подвижность и непрерывность процесса производства; • массовое производство шаблонной продукции на базе конвейера; • независимость производства от субъективных характеристик работника. Второй этап развития менеджмента связан с новыми подходами в развитии учения о менеджменте, на основе учения Ф.Тейлора, но с принципиально новыми подходами. Появится и апробируется на практике так называемая классическая (административная) школа управления, родоначальниками которой стали А.Файоль, П.Урвик, Д.Муни, П.Слоун. В частности, А.Файоль впервые предположили новую теорию менеджмента, раскрывающую его функции, принципы и необходимость теоретического изучения. Он предложил 14 принципов, обеспечивающих качественное управление, • Разделение труда. • Власть и ответственность. • Дисциплина. • Единоначалие. • Единство руководства. • Подчиненность частных интересов общественным. • Справедливое вознаграждение персо-

нала. • Централизация. • Скалярная цепь подчинения в иерархии. • Порядок. • Справедливость. • Стабильность работы для персонала. • Инициатива. • Корпоративный дух. Для наших целей наиболее важны 6 групп операций, который выделили Файоль: 12 • технические, • коммерческие, • финансовые, • страховые, • учетные, • административные, административные операции он разделил на о планирование, о организацию, о мотивацию, о контроль и координацию. Третий этап развития менеджмента стал называться "неоклассическим", нарождается и начинает развиваться школа "человеческих отношений", развитие которой связано с именами ученых А.Файоля, Д.Муни, П.Слоуна, Э.Мейо.

На этом этапе апробируется социологическая концепция групповых решений. Четвертый этап развития менеджмента относится к периоду 1940 – 1960 гг. В эти годы происходит эволюция управленческой мысли, которая направлена на развитие теории организации менеджмента на основе достижений психологической и социологической наук, оказывающих решающее воздействие на человека в системе управления. Пятый этап развития управленческой мысли отличается от всех предыдущих тем, что происходит становление современных количественных методов обоснования управленческих решений под воздействием широкого использования в практике экономико-математических методов и электронновычислительной техники.

Этот процесс успешно развивается по настоящее время. Так, Д.Макгрегор впервые обосновал свою теорию и доказал, что отношение менеджера к своим подчиненным существенно влияет на их поведение и на рабочий климат в организации. В теории "Х" - утверждение приоритета контролирующего менеджера, в теории "У" - принцип распределения объективности. (Годы развития этой концепции – 1950 – 1960 г.)

Шестой этап развития менеджмента можно отнести к периоду 1970 – 1980 гг. Учеными – управленцами вырабатываются новые подходы в развитии теории управления, смысл которых сводится к тому, что организация – это открытая система, приспособляющаяся к внутренней среде (организации) нужно искать во внешней среде. Исходя из такого посыла проследили установления взаимосвязей между типами сред и различными моделями управления. К этому периоду относятся теории: "стратегического менеджмента" И.Ансоффа, "теория властных структур между организациями" Г.Саланчика, "конкретной стратегии, конкурентоспособности, потребительских качеств продукции и ресурсов" Портера и т.д.

Седьмой этап относится к 80-м гг., которые ознаменовались появлением новых подвидов в управлениях, неожиданным для многих открытием "организационной структуры" как мощного механизма управления, особенно успешно использованного Японией и другими странами, с важнейшими по силе воздействия управленческими методами. 13 Восьмой этап развития менеджмента относится к 90-м гг. На этом этапе просматриваются три основные тенденции: • возврат к прошлому - осознание значения материальной, технической базы современного производства; • создание социальных поведенческих элементов - это усиление внимания не только к организационной культуре, но и к различным формам демократизации управления, участие рядовых работников в прибылях, в осуществлении управленческих функций в других сферах деятельности; • усиление международного характера управления. Переход многих стран к открытой экономике, участие в конкурентной борьбе, организации современной деятельности. Школа науки управления (количественная школа)

С точки зрения информатизации управления эта школа внесла, пожалуй, наибольший вклад. Заслуга школы науки управления заключается в том, что она сумела определить основные внутренние и внешние переменные (факторы), влияющие на организацию. Становление школы науки управления связано с развитием математики, статистики, инженерных наук и других смежных с ними областей знаний. Школа науки управления сформировалась в начале 50-х гг. и успешно функционирует и в настоящее время. В школе науки управления различают два главных направления: • рассмотрение производства как «социальной системы» с использованием системного, процессного и ситуационного подходов. • исследование проблем управления на основе системного анализа и использования кибернетического подхода, включая применение математических методов и ЭВМ. Подходы к процессу управления • Системный подход предполагает, что каждый из элементов, составляющих систему (рассматриваемую организацию), имеет свои определенные це-

ли. • Процессный подход основывается на положении о том, что все функции управления зависят друг от друга. • Ситуационный подход непосредственно связан с системным и процессным подходами и расширяет их применение на практике. Сущность его заключается в определении понятия ситуации, под которой подразумевается конкретный набор обстоятельств, переменных, оказывающих влияние на организацию в определенное время. Взаимодействие математики и управления Второе направление школы науки управления связано с развитием точных наук и, прежде всего, математики.

Начало применения математических методов в экономических исследованиях в XIX в. связывают с именем французского экономиста А. Каунота (1801–1877). Возможность использования математики для решения экономических проблем вызвала большой интерес в России. Ряд крупных специалистов, таких как В.К.Дмитриев, Г.А.Фельдман, Л.В.Канторович, внесли большой вклад в разработку и развитие экономикоматематических методов (ЭММ). Особое место принадлежит Д.Е. Слуцкому, известному своими работами по теории вероятности и математической статистике. В 1915 г. он опубликовал статью «К теории сбалансированности бюджета потребителя», которая оказала большое влияние на развитие экономико-математической теории. Через 20 лет эта статья получила мировое 14 признание. Первая в стране Лаборатория экономико-математических методов была создана в 1958 г. в Академии наук В.С. Немчиновым. В 1930 г. в г. Кливленде (США) была образована ассоциация «Международное общество для развития экономической теории в связи со статистикой и математикой», в которую входили известные буржуазные экономисты И.Шумпетер, И.Фишер, Р.Фриш, М.Калецкий, Я.Тинберген и др.

Ассоциация стала выпускать журнал «Эконометрика». Образование этой ассоциации послужило отправным моментом создания математической школы экономистов. Отличительной особенностью науки управления является использование моделей. Модели приобретают особенно важное значение, когда необходимо принимать решения в сложных ситуациях, требующих оценки нескольких альтернатив. Понятие управленческого решения. Таким образом, 50-е гг. XX в. характеризуются формированием нового этапа в развитии управленческой мысли. На основе синтеза идей, выдвинутых в предшествующие периоды, исследователи пришли к пониманию необходимости комплексного подхода к управлению. Кроме того, была сформулирована идея о том, что управление – это не только наука, но и искусство. Осознается и формулируется понятие управленческого решения и как последовательности, состоящей из следующих шагов: • сбор информации и диагностика проблемы, • выявление альтернатив, • оценка альтернатив, • окончательный выбор Стратегическая роль информации в управлении Работа с информацией (информационная культура) является одним из важнейших средств для управления изменениями в компании. Мы можем говорить о ценностях, установках и поведении, которые влияют на процессы осознания, сбора, организации, обработки, распространения и использования информации. Есть три принципиальных причины, в силу которых менеджер сегодня должен заботиться об информационной культуре своей компании. • Во-первых, она уже не является частью общей организационной культуры компании. Все больше компаний понимают необходимость преобразований, ориентированных на свои отрасли и рынки.

Чтобы влиять на будущее, нужно представлять себе, на что оно будет похоже, для чего, в свою очередь, нужно работать с разнообразнейшей деловой, рыночной, политической, технологической и социальной информацией. • Во-вторых, информационные технологии делают возможным создание в компаниях компьютерных сетей, с помощью которых идет общение между менеджерами, но важно знать, как люди используют эту информацию. Само по себе создание такой сети со всеми ее рабочими станциями и мультимедийными возможностями не гарантирует того, что информация будет использоваться более разумно и более эффективно. • В-третьих, для разных функциональных служб, подразделений и рабочих групп информационная культура различна, а это означает различие подходов к процессам осознания, сбора, организации, обработки, распространения и использования информации. Поэтому многие менеджеры согласятся с тем, что информационная культура важна для выработки стратегии и осуществления перемен, но нет определенности в вопросе — как повлиять на использование информации. Четыре типа информационной

культуры Сегодня в компаниях можно встретить четыре разновидности информационной культуры

Каждая влияет на способ использования информации - информационное поведение - и отражает приоритеты руководителей компании в использовании информации для достижения успеха или предотвращения провалов. Первая — это функциональная культура, где информацию используют для влияния на других. Эта культура в наибольшей степени присуща жестко иерархизированным компаниям, где информация служит прежде всего для управления и контроля. Характерным для этой культуры информационным поведением является контроль. Здесь этот термин используется не в отрицательном смысле, а просто как обозначение необходимой для компании деятельности. Многие деловые процессы используются для контроля во всех областях - от бухгалтерского учета до снабжения. Вопрос в том, является ли контроль положительной характеристикой культуры корпорации или он возвращает негибкость и подозрительность. Швейцарско-шведская конструкторская компания АВВ хорошо известна своей системой финансового учета, которая охватывает более 1600 разбросанных по миру отделений. Чтобы управлять такой корпорацией при наличии довольно маленького центрального аппарата управления, нужно иметь точную и полную информацию о деятельности отделений. Без положительного отношения к контролю АВВ была бы просто неуправляемой. В культуре взаимодействия менеджеры и специалисты в достаточной степени доверяют друг другу и потому могут обмениваться информацией, важной для совершенствования процессов и роста эффективности. Прямой обмен информацией о возможных срывах и провалах необходим для устранения проблем и адаптации к изменениям. Правда, есть немало компаний, внедряющих программы Тотального управления качеством или Перестройки управления, и которые одновременно наказывают служащих и менеджеров, решившихся заговорить о недостатках и ошибках. Но есть и компании вроде Boss Corporation, где такая информация рассматривается как необходимое условие успеха. Они полагают, что доводя такую информацию до своих служб и рабочих групп, до потребителей и поставщиков, компания помогает устранению проблем и совершенствованию производства. В культуре исследования менеджеры и служащие стремятся к пониманию будущих тенденций и нахождению лучшего способа отразить возможную угрозу. Здесь господствующим информационным поведением является предвидение. Сегодня во многих компаниях существуют анклавы исследовательской культуры в службах, связанных с обслуживанием клиентов, с исследованиями рынка, с технологическими исследованиями и разработками и со сбором информации. Но для многих отраслей, таких как производство полупроводников и разработка программного обеспечения, этого недостаточно.

Все в компании — от генерального директора до низовых звеньев — должны быть бдительными. Эндрю Гров, генеральный директор Intel, заявил, что “выживают только параноики”, потому что смена технологий в производстве полупроводников происходит примерно каждые 18 месяцев, а инвестиции на создание нового производства огромны — от 1,5 до 2 млрд. дол. Наконец, есть культура открытости. Здесь служащие и менеджеры открыты для нового понимания природы кризисов и радикальных перемен. Эти компании сознательно отбрасывают старые подходы к бизнесу, чтобы освободиться для поиска новых перспектив и идей, обещающих создание новых продуктов и услуг, которые могли бы изменить условия конкуренции поверх рынков и отраслей. Корпорация Microsoft одновременно конкурирует на рынках онлайн-информации, развлечений и продажи видеопродукции, чем радикально изменила традиционное представление о производителе программного обеспечения. Компания не просто предвидит изменения или адаптируется к ним, но перекраивает саму базу конкуренции в самых разных отраслях. Немало компаний обладают анклавами культуры открытости, где собирают и обрабатывают информацию, разрабатывают новые продукты и сценарии развития бизнеса, добиваются партнерских отношений с потребителями и поставщиками. Но есть только считанное число таких компаний, как Microsoft, которые сделали культуру открытий неотъемлемой частью своей стратегии. Инструментальные средства сбора и обработки информации Исторически первыми инструментальными средствами сбора и обработки информации стали сначала файлы, в которых хранились данные, затем системы управления базами данных (СУБД).

Предпосылкой введения СУБД являясь формализация подходов к структуре информации: реляционное исчисление, модели «сущность-связь», онтологические модели. С помощью СУБД накоплены и обрабатываются гигантские массивы данных. Концепция «хранилищ данных» (Data Warehouse) Согласно классическому определению Б. Инмона, DW есть предметно ориентированный, интегрированный, неизменный, поддерживающий хронологию набор данных, предназначенный для поддержки принятия решений. Следует отметить, что в этом определении соединены две различные функции: а) сбор, организация и подготовка данных для анализа в виде постоянно наращиваемой базы данных; б) собственно анализ как элемент принятия решений. Принятие решений в качестве сферы применения DW существенно сужает определение. Если в определении оставить лишь анализ (как элемент научных, технологических и экологических систем), круг использования данной концепции может быть значительно расширен.

Очень важен основной принцип действия DW: единожды занесенные в DW данные затем многократно извлекаются из него и используются для анализа. Отсюда вытекает одно из основных преимуществ использования DW в работе предприятия - контроль за критически важной информацией, полученной из различных источников, как за производственным ресурсом. Суть определения DW проиллюстрирована на рисунке. Исторически концепция хранилищ данных возникла после того, как на предприятиях накопилось большое количество баз данных в различных подразделениях: бухгалтерии, финансовом отделе, отделе кадров и появилась необходимость связать эти сведения в единое целое для построения общей картины. Для технического исполнения хранилища данных используются интернет/интранет технологии и СУБД. Хранилище данных – технология для руководителя (руководителей) Выгоды от использования хранилищ данных на предприятиях Практика показывает, что внедрение хранилищ данных дает положительный результат в следующих областях: • сегментация рынка; • планирование продаж, прогнозирование и управление; • забота о клиенте (CRM системы); • разработка схем лояльности; • проектирование и разработка новых видов продукции; • интеграция цепочки поставок; • интеллектуальные технологии в организации бизнеса. • распространение DW из области стратегического планирования на текущие операции; Трудности внедрения хранилищ данных Причины, которые затрудняют внедрение технологии хранилища данных на предприятии: • Разнобой в используемых технологических средствах. Независимо от выбора СУБД в рамках каждого подразделения формируется различная корпоративная технология обработки данных, по-разному распределяются нагрузки на сервер и клиентскую часть и т. д. Единообразие подхода в какой-то степени обеспечивает ритмичность проведения работ. • Организационно-технологические факторы.

С одной стороны, подразделения могут быть не только не готовы к работе по новой информационной технологии (что в порядке вещей), но и вправе отказаться от нововведений по ряду объективных или субъективных причин (явление также обычное). С другой стороны, недоучет разработки организационно - технологического обеспечения может привести к постоянному переносу сроков реализации проекта (а иногда просто к срыву всех сроков). Степень подготовленности персонала на всех уровнях иерархии должна контролироваться на всех этапах разработки и внедрения системы. • Отсутствие лингвистического обеспечения. Это особенно критично для построения хранилищ данных.

Обычно при создании системы предполагается, что заказчику контекстуально все понятно и технологические процессы (бизнес-процедуры) не вызывают затруднений. В качестве примера можно привести сроки освоения бухгалтерских программ, построенных с использованием специальных компьютерных языков и предполагающих, что бухгалтер достаточно хорошо владеет основами информатики и программирования в рамках университетского курса. Для успешного использования DW огромное значение имеют метаданные, на основе которых пользователь получает доступ к данным. Семантика и смысл всех данных DW должны быть ясно и точно определены. На практике, если это не доминирующая цель системы, разработке лингвистического обеспечения при проектировании не уделяют внимания, относя это на потом. Построение DW следует начинать с создания лингвистического обеспечения. • фактор руководителя имеет очень большое значение, поскольку DW создается чаще всего для задач принятия и поддержки решений. Если руководи-

тель не уверен в том, какие информационные задачи следует решать, ему лучше подождать, пока его конкуренты или партнеры не обзаведутся собственными DW.

Тогда-то он и позаимствуете у них опыт. • Объединение плохо совместимого. Например, не следует объединять в программах типа "Торговый дом" бухгалтерский и управленческий учет в рамках одной технологии разработки. Это будет долгий и трудный процесс, поскольку подходы к моделированию и проектированию этих частей существенно различаются. Реализовать их по отдельности и проще, и быстрее. Однако есть и другие примеры, когда для разработки используются концепции более высокого уровня абстракции. Удачны, например, решения в Ваап или ROSS system, при разработке которых была отработана методология слияния транзакционной и аналитической частей в рамках единой КИС. Зато они и стоят дорого. • Корректность его данных, полученных из разных источников. Данные перед загрузкой в хранилище должны быть либо "очищены от шума", либо обработаны методами нечеткой логики, допускающей наличие противоречивых фактов. Например, данные о предприятии-партнере могут быть получены от разных экспертов, чьи оценки порой бывают диаметрально противоположными. Распределенные информационные ресурсы Концепция хранилищ данных позволяет достаточно успешно решать проблемы интеграции распределенных информационных ресурсов в пределах одного предприятия. Но развитие интернет технологий поставило задачу интеграции информационных ресурсов и создания распределенных баз данных более остро.

Одна из главных особенностей современных информационных систем - распределенный характер, информационные системы охватывают все более количество структурных единиц. Примером распределенной системы может послужить система резервирования билетов крупной авиакомпании, имеющей свои филиалы в различных частях Земли. Главная проблема таких систем - организация обработки распределенных данных. Данные находятся на компьютерах различных моделей и производителей, функционирующих под управлением различных операционных систем, а доступ к данным осуществляется разнородным программным обеспечением. Сами компьютеры территориально удалены друг от друга и находятся в различных географических точках планеты. Ответом на задачи реальной жизни стали две технологии: технология распределенных баз данных (Distributed Database) и технология тиражирования данных (Data Replication). Под распределенной базой данных подразумевают базу, включающую фрагменты из нескольких баз данных, которые располагаются на различных узлах сети компьютеров, и, возможно, управляются различными СУБД. Распределенная база данных выглядит с точки зрения пользователей и прикладных программ как обычная локальная база.

В этом смысле слово "распределенная" отражает способ 19 организации базы данных, но не внешнюю ее характеристику ("распределенность" базы не должна быть видна извне). В отличие от распределенных баз, тиражирование данных предполагает отказ от их физического распределения и опирается на идею дублирования данных в различных узлах сети компьютеров. Ниже будут изложены детали, преимущества и недостатки каждой технологии. Тема 3. Информационные технологии на этапе принятия решений Выявление альтернатив для принятия управленческого решения - технологии OLAP и Data Mining. Интеллектуальный анализ данных. Визуализация данных. Тематические цифровые карты. Оценка альтернатив – сценарное моделирование «что-если». Избыток информации вреден. Ранее мы рассмотрели технологии, которые позволяют накапливать и обрабатывать большие объемы информации. При этом выявилась следующая проблема: чрезмерное количество данных в виде обширных таблиц затрудняет принятие управленческого решения – особенностью человеческого мышления является то, что для принятия решения человеку нужна компактная, сжатая информация, лучше в графическом виде. Стали возникать различные способы решения проблемы сжатия данных.

Практика использования и внедрения хранилищ данных показала, что анализировать данные оперативных систем напрямую невозможно или очень затруднительно. Это объясняется различными причинами: • разрозненностью данных, хранением их в форматах различных СУБД и в разных "уголках" корпоративной сети, • даже если на предприятии все данные хранятся на центральном сервере БД (что бывает крайне редко), руководитель (аналитик) почти наверняка не разберется в их сложных, подчас запутанных структурах. • задача технологии OLAP - предоставить "сы-

рье" для анализа в одном месте и в простой, понятной структуре • появление отдельного хранилища целесообразно потому, что сложные аналитические запросы к оперативной информации тормозят текущую работу компании, надолго блокируя таблицы и захватывая ресурсы сервера. Концепция OLAP Централизация и удобное структурирование - это далеко не все, что нужно руководителю (аналитику). Ему ведь еще требуется инструмент для просмотра, визуализации информации. Традиционные отчеты, даже построенные на основе единого хранилища, лишены одного - гибкости. Их нельзя "покрутить", "развернуть" или "свернуть", чтобы получить желаемое представление данных. Конечно, можно вызвать программиста (если он захочет прийти), и он (если не занят) сделает новый отчет достаточно быстро - скажем, в течение часа (пишу и сам не верю - так быстро в жизни не бывает; давайте дадим ему часа три).

Получается, что аналитик может проверить за день не более двух идей. А ему (если он хороший аналитик) таких идей может приходиться в голову по нескольку в час. И чем больше "срезов" и "разрезов" данных аналитик видит, тем больше у него идей, которые, в свою очередь, для проверки требуют все новых и новых "срезов". Вот бы ему такой инструмент, который позволил бы разворачивать и сворачивать данные просто и удобно! В качестве такого инструмента и выступает OLAP. OLAP - это Online Analytical Processing, т. е. оперативный анализ данных. Хотя OLAP и не представляет собой необходимый атрибут хранилища данных, он все чаще и чаще применяется для анализа накопленных в этом хранилище сведений. Структура хранилища данных Оперативные данные собираются из различных источников, очищаются, интегрируются и складываются в реляционное хранилище.

При этом они уже доступны для анализа при помощи различных средств построения отчетов. Затем данные (полностью или частично) подготавливаются для OLAP-анализа. Они могут быть загружены в специальную БД OLAP или оставлены в реляционном хранилище. Важнейшим его элементом являются метаданные, т. е. информация о структуре, размещении и трансформации данных. Благодаря им обеспечивается эффективное взаимодействие различных компонентов хранилища. Логическая структура OLAP данных OLAP = многомерное представление = Куб OLAP предоставляет удобные быстродействующие средства доступа, просмотра и анализа деловой информации. Пользователь получает естественную, интуитивно понятную модель данных, организуя их в виде многомерных кубов (Cubes). Осями многомерной системы координат служат основные атрибуты анализируемого бизнес-процесса. Например, для продаж это могут быть товар, регион, тип покупателя. В качестве одного из измерений используется время. На пересечениях осей - измерений (Dimensions) - находятся данные, количественно характеризующие процесс - меры (Measures). Это могут быть объемы продаж в штуках или в денежном выражении, остатки на складе, издержки и т. п. Пользователь, анализирующий информацию, может "разрезать" куб по разным направлениям, получать сводные (например, по годам) или, наоборот, детальные 21 (по неделям) сведения и осуществлять прочие манипуляции, которые ему придут в голову в процессе анализа. В качестве мер в трехмерном кубе, изображенном на рисунке, использованы суммы продаж, а в качестве измерений - время, товар и магазин. Измерения представлены на определенных уровнях группировки: товары группируются по категориям, магазины - по странам, а данные о времени совершения операций - по месяцам. Чуть позже мы рассмотрим уровни группировки (иерархии) подробнее.

Пример куба "Разрезание" куба Даже трехмерный куб сложно отобразить на экране компьютера так, чтобы были видны значения интересующих мер. Что уж говорить о кубах с количеством измерений, большим трех? Для визуализации данных, хранящихся в кубе, применяются, как правило, привычные двумерные, т. е. табличные, представления, имеющие сложные иерархические заголовки строк и столбцов. Двумерное представление куба можно получить, "разрезав" его поперек одной или нескольких осей (измерений): мы фиксируем значения всех измерений, кроме двух, - и получаем обычную двумерную таблицу. В горизонтальной оси таблицы (заголовки столбцов) представлено одно измерение, в вертикальной (заголовки строк) - другое, а в ячейках таблицы - значения мер. При этом набор мер фактически рассматривается как одно из измерений - мы либо выбираем для показа одну меру (и тогда можем разместить в заголовках строк и столбцов два измерения), либо показываем несколько мер (и тогда одну из осей таблицы займут названия мер, а

другую - значения единственного "неразрезанного" измерения). На первом рисунке изображен двумерный срез куба для одной меры - Unit Sales (продано штук) и двух "неразрезанных" измерений - Store (Магазин) и Время (Time). На следующем представлено лишь одно "неразрезанное" измерение - Store, но зато здесь отображаются значения нескольких мер - Unit Sales (продано штук), Store Sales (сумма продажи) и Store Cost (расходы магазина). Двумерное представление куба возможно и тогда, когда "неразрезанными" остаются и более двух измерений. При этом на осях среза (строках и столбцах) будут размещены два или более измерений "разрезаемого" куба. Значения, "откладываемые" вдоль измерений, называются членами или метками (members). Метки используются как для "разрезания" куба, так и для ограничения (фильтрации) выбираемых данных - когда в измерении, остающемся "неразрезанным", нас интересуют не все значения, а их подмножество, например 22 три города из нескольких десятков.

Значения меток отображаются в двумерном представлении куба как заголовки строк и столбцов. Архитектура OLAP-приложений Все, что говорилось выше про OLAP, по сути, относилось к многомерному представлению данных. То, как данные хранятся, грубо говоря, не волнует ни конечного пользователя, ни разработчиков инструмента, которым клиент пользуется. Многомерность в OLAP-приложениях может быть разделена на три уровня: • Многомерное представление данных - средства конечного пользователя, обеспечивающие многомерную визуализацию и манипулирование данными; слой многомерного представления абстрагирован от физической структуры данных и воспринимает данные как многомерные. • Многомерная обработка - средство (язык) формулирования многомерных запросов (традиционный реляционный язык SQL здесь оказывается непригодным) и процессор, умеющий обработать и выполнить такой запрос. • Многомерное хранение - средства физической организации данных, обеспечивающие эффективное выполнение многомерных запросов.

Первые два уровня в обязательном порядке присутствуют во всех OLAP-средствах. Третий уровень, хотя и является широко распространенным, не обязателен, так как данные для многомерного представления могут извлекаться и из обычных реляционных структур; процессор многомерных запросов в этом случае транслирует многомерные запросы в SQL-запросы, которые выполняются реляционной СУБД. Конкретные OLAP-продукты, как правило, представляют собой либо средство многомерного представления данных, OLAP-клиент (например, Pivot Tables в Excel 2000 фирмы Microsoft или ProClarity фирмы Knosys), либо многомерную серверную СУБД, OLAP-сервер (например, Oracle Express Server или Microsoft OLAP Services). Слой многомерной обработки обычно бывает встроен в OLAP-клиент и/или в OLAP-сервер, но может быть выделен в чистом виде, как, например, компонент Pivot Table Service фирмы Microsoft. Технические аспекты многомерного хранения данных Как уже говорилось выше, средства OLAP-анализа могут извлекать данные и непосредственно из реляционных систем. Такой подход был более привлекательным в те времена, когда OLAP-серверы отсутствовали в прайс-листах ведущих производителей СУБД. Но сегодня и Oracle, и Informix, и Microsoft предлагают полноценные OLAP-серверы, и даже те IT-менеджеры, которые не любят разводить в своих сетях "зоопарк" из ПО разных производителей, могут купить (точнее, обратиться с соответствующей просьбой к руководству компании) OLAP-сервер той же марки, что и основной сервер баз данных. OLAP-серверы, или серверы многомерных БД, могут хранить свои многомерные данные по-разному. Прежде чем рассмотреть эти способы, нам нужно поговорить о таком важном аспекте, как хранение агрегатов (промежуточных итогов, сумм). Дело в том, что в любом хранилище данных - и в обычном, и в многомерном - наряду с детальными данными, извлекаемыми из оперативных систем, хранятся и суммарные показатели (агрегированные показатели, агрегаты), такие, как суммы объемов продаж по месяцам, по категориям товаров и т. п.

Агрегаты хранятся в явном виде с единственной целью - ускорить выполнение запросов. Ведь, с одной стороны, в хранилище накапливается, как правило, очень большой объем данных, а с другой - аналитиков 23 в большинстве случаев интересуют не детальные, а обобщенные показатели. И если каждый раз для вычисления суммы продаж за год пришлось бы суммировать миллионы индивидуальных продаж, скорость, скорее всего, была бы неприемлемой.

Поэтому при загрузке данных в многомерную БД вычисляются и сохраняются все суммарные показатели или их часть. Но, как известно, за все надо платить. И за скорость обработки запросов к суммарным данным приходится платить увеличением объемов данных и времени на их загрузку. Причем увеличение объема может стать буквально катастрофическим - в одном из опубликованных стандартных тестов полный подсчет агрегатов для 10 Мб исходных данных потребовал 2,4 Гб, т. е. данные выросли в 240 раз! Степень "разбухания" данных при вычислении агрегатов зависит от количества измерений куба и структуры этих измерений, т. е. соотношения количества "отцов" и "детей" на разных уровнях измерения. Для решения проблемы хранения агрегатов применяются подчас сложные схемы, позволяющие при вычислении далеко не всех возможных агрегатов достигать значительного повышения производительности выполнения запросов. Технология «Data Mining» Технология Data Mining (буквально «добыча информации») явилась попыткой борьбы с чрезмерно большими массивами информации. Технология развивалась из направления исследований «искусственный интеллект», получившего названия в англоязычной литературе “data mining” и “knowledge discovery”. Под “knowledge discovery in databases” (обнаружение знаний в базах данных) (КДД) понимают какой-либо нетривиальный процесс идентификации достоверных, новых, потенциально полезных и хорошо понимаемых образцов (структур, patterns) в данных. Ключевое достоинство «Data Mining» по сравнению с предшествующими методами - возможность автоматического порождения гипотез о взаимосвязи между различными параметрами или компонентами данных.

Сравнение формулировок задач при использовании методов OLAP и Data Mining OLAP Data Mining Каковы средние показатели травматизма для курящих и некурящих? Встречаются ли точные шаблоны в описаниях людей, подверженных повышенному травматизму? Каковы средние размеры телефонных счетов существующих клиентов в сравнении со счетами бывших клиентов (отказавшихся от услуг телефонной компании)? Имеются ли характерные портреты клиентов, которые, по всей вероятности, собираются отказаться от услуг телефонной компании? Какова средняя величина ежедневных покупок по украденной и не украденной кредитной карточке? Существуют ли стереотипные схемы покупок для случаев мошенничества с кредитными карточками? Основные методы извлечения фактов

Методы извлечения новых знаний из баз фактов, применяемые в Data Mining, весьма различны – это и статистические процедуры, генетические алгоритмы, нейронные сети, деревья решений, индуктивное логическое программирование и т.д. Общим обстоятельством в различных реализациях Data Mining является то, что 24 данные недостаточно формализованы, но извлекаемость из них посредством компьютерных программ новых полезных знаний возможна. Выделяют пять стандартных типов закономерностей (эвристик), которые позволяют выявлять методы Data Mining: • ассоциация (выбор типовых сочетаний), • последовательность (определение типовой последовательности) • классификация • кластеризация • прогнозирование Интеллектуальная информационная система. Для воплощения технологии извлечения знаний используется информационная система специального типа – «интеллектуальная система», «советующая система», «партнерская система». Уточним термин «интеллектуальная система». ИС есть компьютерная система для решения классов задач, которые или не могут быть решены человеком в реальное время, или же их решение требует автоматизированной поддержки, или же их решение дает результаты сопоставимые по информативности с решениями человека.

Характеризация компьютерной системы как интеллектуальной будет неполной, если не будут уточнены как природа решаемых задач, так и средства их решения, реализуемые благодаря определенной архитектуре компьютерной системы. Типовые задачи Data Mining в розничной торговле Предприятия розничной торговли сегодня собирают подробную информацию о каждой отдельной покупке, используя кредитные карточки с маркой магазина и компьютеризованные системы контроля. Типичные задачи, которые можно решать с помощью Data Mining в сфере розничной торговли, это • анализ покупательской корзины, • исследование временных шаблонов, • создание прогнозирующих моделей. Анализ покупательской корзины (анализ сходства) предназначен для выявления товаров, которые покупатели стремятся приобретать вместе. Знание покупательской корзины необходимо для улучшения рекламы, выработки стратегии создания запасов товаров

и способов их раскладки в торговых залах. Исследование временных шаблонов помогает торговым предприятиям принимать решения о создании товарных запасов. Оно дает ответы на вопросы типа: «Если сегодня покупатель приобрел видеокамеру, то через какое время он вероятнее всего купит новые батарейки и пленку?». Создание прогнозирующих моделей дает возможность торговым предприятиям узнавать характер потребностей различных категорий клиентов с определенным поведением, например, покупающих товары известных дизайнеров или посещающих распродажи.

Эти знания нужны для разработки точно направленных, экономичных мероприятий по продвижению товаров. Типовые задачи Data Mining в банковском деле Выявление мошенничества с кредитными карточками. Путем анализа прошлых транзакций, которые впоследствии оказались мошенническими, банк выявляет некоторые стереотипы такого мошенничества. 25 Сегментация клиентов. Разбивая клиентов на различные категории, банки делают свою маркетинговую политику более целенаправленной и результативной, предлагая различные виды услуг разным группам клиентов. Прогнозирование изменений клиентуры. Data Mining помогает банкам строить прогнозные модели ценности своих клиентов и соответствующим образом обслуживать каждую категорию. Типовые задачи Data Mining в страховой деятельности Страховые компании в течение ряда лет накапливают большие объемы данных. Здесь также можно использовать методы Data Mining: для выявления мошенничества и анализа риска. Выявление мошенничества. Страховые компании могут снизить уровень мошенничества, отыскивая определенные стереотипы в заявлениях о выплате страхового возмещения, характеризующих взаимоотношения между юристами, врачами и заявителями.

Анализ риска. Путем выявления сочетаний факторов, связанных с оплаченными заявлениями, страховщики могут уменьшить свои потери по обязательствам. Известен случай, когда в СИ-ТА крупная страховая компания обнаружила, что суммы, выплаченные по заявлениям людей, состоящих в браке, вдвое превышают суммы по заявлениям одиноких людей. Компания отреагировала на это новое знание пересмотром своей общей политики предоставления скидок семейным клиентам. Программные средства Извлечение данных (Data Mining - DM) - одно из самых ценных новшеств SQL Server 2000. В версии SQL Server 7.0 специалисты Microsoft впервые реализовали аналитическую службу OLAP, предоставляющую возможности составления нерегламентированных (гибких) запросов и анализа данных. В процессе работы с нерегламентированными запросами аналитик точно знает, на какие вопросы клиент хотел бы получить ответы, и просто извлекает нужную информацию из куба OLAP. SQL Server 2000 применяет для предоставления возможностей DM новый интерфейс приложений (API), называемый OLE DB for Data Mining (OLE DB for DM). В состав SQL Server 2000 вошли два алгоритма DM, так называемые деревья принятия решений и алгоритм кластеризации. Визуализация данных Визуализация данных – наглядное представление данных для лица, принимающего решение. представление числовой и текстовой информации в виде графиков, диаграмм, структурных схем, таблиц, карт и т.д. Современные компьютерные технологии используют широкий спектр методов визуализации информации. Легкость построения графиков и диаграмм с помощью ЭВМ все заметнее меняет когнитивные навыки исследователя.

Современные пакеты анализа социологической информации позволяют строить сотни типов различных графиков и диаграмм. Исследователь может одним взглядом обнаружить особенности, выявить закономерности и аномалии в больших объемах информации. Мощным средством анализа информации являются интерактивные средства модификации графических представлений. Особенно широко графические методы используются в разведочном анализе данных, позволяя выявлять закономерности в многомерных массивах информации. Современные методы визуализации информации широко используются для представления и анализа результатов компьютерного моделирования. Так в методологии иконологического моделирования визуализация позволяет 26 пользователю выявлять различные формы пространственной и временной самоорганизации, анализировать поведение нелинейных систем и процессов. Благодаря компьютеризации визуализация информации играет все большую роль в повышении эффективности коммуникаций. Использование слайд-фильмов и современных средств вывода информации на большой экран позволяет су-

щественно повысить эффективность лекций, докладов и презентаций. Отметим, что визуализация информации позволяет повысить эффективность коммуникаций и без помощи компьютерных технологий. Так в методологиях исследования мягких систем и качественного анализа данных используются образные схемы. В этих схемах участники проблемных ситуаций выражают свои представления в произвольной форме с помощью примитивных рисунков, графиков, текстовых подписей.

Оказалось, что отсутствие стандартизации элементов рисунка не создает коммуникативных затруднений для участников обсуждений. Наоборот, использование невербальных форм представления информации позволяет легко концентрировать внимание на узловых точках проблемы. Результаты междисциплинарных исследований позволяют уверенно утверждать, что визуализация является одним из наиболее перспективных направлений повышения эффективности методов анализа и представления информации. Карта Математически определенное, уменьшенное, генерализованное изображение поверхности Земли, другого небесного тела, или космического пространства, показывающее расположенные, или спроецированные на них объекты в принятой системе условных знаков. Карта рассматривается как образно-знаковая модель, обладающая высокой информативностью, пространственно-временным подобием относительно оригинала, метричностью, особой обзорностью и наглядностью, что делает ее важнейшим средством познания в науках о Земле и социальноэкономических науках. По масштабу различают крупномасштабные карты (large scale maps) [1:100 000 и крупнее], среднемасштабные карты (medium scale maps) [1:200 000–1:1 000 000] и мелкомасштабные карты (small scale maps) [мельче 1:1 000 000]. В соответствии с содержанием различают следующие группы (виды) карт: общегеографические карты (general map), тематические карты, в т.ч. карты природы (natural map), социально-экономические карты (social and economical map), карты взаимодействия природы и общества (maps of nature and society interaction), а также специальные карты. Все они могут быть аналитическими, комплексными или синтетическими картами. По практической специализации различают несколько типов карт: инвентаризационные карты (inventory maps), показывающие наличие и локализацию объектов; оценочные карты (evaluative maps), характеризующие объекты (например, природные ресурсы) по их пригодности для каких-либо видов хозяйственной деятельности; рекомендательные карты (recommendative maps), показывающие размещение мероприятий, предлагаемых для охраны, улучшения природных условий и оптимального использования ресурсов; прогнозные карты (prognostic maps, forecast maps), содержащие научное предвидение явлений, не существующих или неизвестных в настоящее время. Тематическая карта Тематическая карта - карта, отражающая один сюжет (тему, объект, явление, отрасль) или сочетание сюжетов. Различают тематические карты природных и общественных явлений, а также их взаимодействия. Все сведения о природных и общественных явлениях, нанесенные на карту в соответствии с ее темой, составляют ее специальное содержание. Обязате 27 тематических карт является их географическая основа. Различают тематические карты природных, общественных явлений и их взаимодействия (например, карты геологические, этнографические, социально-экономические, экологические и т.п.). По степени обобщения изображаемых явлений выделяют аналитические, комплексные и синтетические карты. Легенда карты Свод условных обозначений, использованных на карте, с текстовыми пояснениями к ним. Обычно, легенды карт создаются на основе классификаций изображаемых объектов и явлений, они становятся их графической моделью и часто служат для построения классификаторов. Большие и сложные легенды карт делятся на разделы и подразделы, причем графические средства и надписи подчеркивают их иерархическую соподчиненность Карта-схема Карта с неточно выдержанными масштабом и проекцией, упрощенным и обобщенным изображением элементов содержания.

Геоинформационная система, ГИС 1. Информационная система, обеспечивающая сбор, хранение, обработку, доступ, отображение и распространение пространственно-координированных данных (пространственных данных). ГИС содержит данные о пространственных объектах в форме их цифровых представлений (векторных, растровых, квадротомических и иных); 2. Программное средство ГИС (1) - программный продукт, в котором реализованы функциональные возможности ГИС. Поддерживается программным, аппаратным, информационным, нормативно-правовым, кад-

ровым и организационным обеспечением. По территориальному охвату различают глобальные, или планетарные ГИС (global GIS), субконтинентальные ГИС, национальные ГИС, зачастую имеющие статус государственных, региональные ГИС (regional GIS), субрегиональные ГИС и локальные, или местные ГИС (local GIS). ГИС различаются предметной областью информационного моделирования, например, городские ГИС, или муниципальные ГИС, МГИС (urban GIS), природоохранные ГИС (environmental GIS) и т.п.; среди них особое наименование, как особо широко распространенные, получили земельные информационные системы. Проблемная ориентация ГИС определяется решаемыми в ней задачами (научными и прикладными), среди них инвентаризация ресурсов (в том числе кадастр), анализ, оценка, мониторинг, управление и планирование, поддержка принятия решений.

Интегрированные ГИС, ИГИС (integrated GIS, IGIS) совмещают функциональные возможности ГИС и систем цифровой обработки изображений (данных дистанционного зондирования) - см. обработка снимков - в единой интегрированной среде. Полимасштабные, или масштабнонезависимые ГИС (multiscale GIS) основаны на множественных, или полимасштабных представлениях пространственных объектов (multiple representation, multiscale representation), обеспечивая графическое, или картографическое воспроизведение данных на любом из избранных уровней масштабного ряда на основе единственного набора данных с наибольшим пространственным разрешением. Пространственно-временные ГИС (spatiotemporal GIS) оперируют пространственно-временными данными. Реализация геоинформационных проектов (GIS project), создание ГИС в широком смысле слова включает этапы: предпроектных исследований (feasibility study), в том числе изучение требований пользователя (user requirements) и функциональных возможностей используемых программных средств ГИС, техникоэкономическое обоснование, оценку соотношения "затраты/прибыль" (costs/benefits); системное проектирование ГИС (GIS designing), включая стадию пилот-проекта (pilot-project), разработку ГИС (GIS development); ее тестирование на небольшом территориальном фрагменте, или тестовом участке (test area), прототипирование, или создание опытного образца, или прототипа (prototype); внедрение ГИС (GIS implementation); эксплуатацию и использование. Научные, технические, технологические и прикладные аспекты проектирования, создания и использования ГИС изучаются геоинформатикой. Оценка альтернатив «что-если». В условиях рыночной экономики степень неопределенности экономического поведения субъектов рынка достаточно высока. В связи с этим большое практическое значение приобретают методы перспективного анализа, когда нужно принимать управленческие решения, оценивая возможные ситуации и делая выбор из нескольких альтернативных вариантов. Теоретически существует четыре типа ситуаций, в которых необходимо проводить анализ и принимать управленческие решения, в том числе и на уровне предприятия: в условиях определенности, риска, неопределенности, конфликта.

Рассмотрим каждый из этих случаев. Анализ и принятие управленческих решений в условиях определенности. Это самый простой случай: известно количество возможных ситуаций (вариантов) и их исходы. Нужно выбрать один из возможных вариантов. Степень сложности процедуры выбора в данном случае определяется лишь количеством альтернативных вариантов. Рассмотрим две возможные ситуации: а) Имеется два возможных варианта ($n=2$). В данном случае аналитик должен выбрать (или рекомендовать к выбору) один из двух возможных вариантов. Последовательность действий здесь следующая: • определяется критерий, по которому будет делаться выбор; • методом "прямого счета" исчисляются значения критерия для сравниваемых вариантов; • вариант с лучшим значением критерия рекомендуется к отбору. Возможны различные методы решения этой задачи. Как правило, они подразделяются на две группы: • методы, основанные на дисконтированных оценках; • методы, основанные на учетных оценках. Первая группа методов основывается на следующей идее. Денежные доходы, поступающие на предприятие в различные моменты времени, не должны суммироваться непосредственно; можно суммировать лишь элементы приведенного потока. Как правило для расчетов применяется ЭВМ. При проведении анализа в условиях определенности могут успешно применяться методы машинной имитации, предполагающие множественные расчеты на ЭВМ. В этом случае строится имитационная модель объекта или процесса (компьютерная программа), содержащая b -е число факторов и переменных, значения

которых в разных комбинациях подвергается варьированию. Таким образом, машинная имитация - это эксперимент, но не в реальных, а в искусственных условиях. По результатам этого эксперимента отбирается один или несколько вариантов, являющихся базовыми для принятия окончательного решения на основе дополнительных формальных и неформальных критериев. Анализ и принятие управленческих решений в условиях риска. Эта ситуация встречается на практике наиболее часто. Здесь пользуются вероятностным подходом, предполагающим прогнозирование возможных исходов и присвоение им вероятностей. При этом пользуются: • известными, типовыми ситуациями (типа - вероятность появления герба при бросании монеты равна 0.5); • предыдущими распределениями вероятностей (например, из выборочных обследований или статистики предшествующих периодов известна вероятность появления бракованной детали); • субъективными оценками, сделанными аналитиком самостоятельно либо с привлечением группы экспертов.

Анализ и принятие управленческих решений в условиях неопределенности. Эта ситуация разработана в теории, однако на практике формализованные алгоритмы анализа применяются достаточно редко. Основная трудность здесь состоит в том, что невозможно оценить вероятности исходов. Основной критерий - максимизация прибыли - здесь не срабатывает, поэтому применяют другие критерии: • максимин (максимизация минимальной прибыли) • минимакс (минимизация максимальных потерь) • максимакс (максимизация максимальной прибыли) и др. Анализ и принятие управленческих решений в условиях конфликта. Наиболее сложный и мало разработанный с практической точки зрения анализ. Подобные ситуации рассматриваются в теории игр. Безусловно, на практике эта и предыдущая ситуации встречаются достаточно часто. В таких случаях их пытаются свести к одной из первых двух ситуаций либо используют для принятия решения неформализованные методы. Оценки, полученные в результате применения формализованных методов, являются лишь базой для принятия окончательного решения; при этом могут приниматься во внимание дополнительные критерии, в том числе и неформального характера.

2. Методические рекомендации (указания) к практическим занятиям

1. Классификация автоматизированных информационных систем и технологий
 - Автоматизированные системы управления
 - Системы поддержки принятия решения
 - Автоматизированные информационно-вычислительные системы
 - Автоматизированные системы обучения
 - Автоматизированные информационно-справочные системы

2. И.Р. экономической и научно-технической информации.
 - И.Р. федеральных и муниципальных органов власти.
 - И.Р. государственной системы статистики

3. Модели процесса обработки информации.
 - Модели информационных процессов представления данных
 - Извлечение информации.
 - Обработка информации.
 - Хранение информации.
 - Представление и использование информации.

4. Системы сбора информации. Системы статистических наблюдений
 - Способы считывания информации в распределенных средах
 - Сканирующие системы сбора информации
 - Многоточечные системы сбора информации
 - Мультиплицированные системы сбора информации
 - Системы сбора информации параллельного действия

5. Принципы и математические основы анализа данных.
 - Математико-статистические таблицы.
 - Обработка табличных данных

3. Методические рекомендации (указания) к лабораторным занятиям

Текстовый редактор Microsoft Word. Электронная анкета.

Форма – это бланк для заполнения конкретными данными. Форма Word представляет собой электронную версию формы в виде документа с незаполненными областями, в которые пользователь должен на компьютере вводить данные. Форма предназначена для сокращения трудозатрат на оформление документов за счет предварительного создания шаблона, на базе которого создаются новые документы, например формы различных справок установленного образца, бланков документов, таблиц и т.д.

В Word можно создать следующие виды форм:

- формы, которые печатаются, а затем заполняются на бумаге. Для ввода данных используется панель инструментов Формы для вставки текстовых полей и флажков со списками возможных ответов, например «Да» и «Нет». Это обычный документ, который не требует защиты при работе;

- электронные формы, которые можно распространять через электронную почту или по сети. Для вставки поля формы используется панель инструментов Формы, а для упрощения заполнения к полям форм можно добавить подсказки. Эти формы требуют защиты от несанкционированного доступа по изменению вида;

- настраиваемые электронные формы, которые представляют большие возможности для ввода данных, выбора ответов из списка и записи ответов. Эти формы также можно распространять через электронную почту или по сети. Для них также необходима защита от несанкционированного доступа по изменению ее вида.

Форма состоит из *постоянной* (неизменной) и *переменной* (изменяемой) при вводе данных части документа. В состав формы могут входить:

- произвольный текст;
- графика (рисунки, диаграммы);
- текстовые поля для ввода информации:
 - Обычный текст
 - Число
 - Дата
 - Текущая дата
 - Текущее время
 - Вычисляемое значение
- списки альтернативных значений, предлагаемых для выбора при заполнении формы;
- кнопка <Флажок>.

К элементам форм могут применяться все основные команды меню Формат.

К *постоянной* части формы относится неизменяемый при вводе текст, таблицы фиксированного содержания, оформительские элементы: рисунки, графика, линии разметки, элементы оформления – заполнение, обрамление. Текст формы и включаемые в нее элементы могут быть организованы в табличном виде.

Переменная часть документа включает поля, списки, переключатели. Настройка элементов переменной части формы осуществляется в соответствующих диалоговых окнах <<Параметры текстового поля>>, <<Параметры со списком>>, <<Параметры флажка>>. Вызов этих окон производится командой Свойства из контекстного меню (таблица 1).

Для полей, списков и флажков могут выполняться макрокоманды при входе и выходе курсора из этого элемента. Макрокоманды создаются предварительно.

В строке-состояния или по клавише <F1> можно сделать индикацию справки по работе с элементами формы. Справка содержит произвольный текст, подготовленный разработчиком формы; максимальная длина текста справки – 255 символов.

Каждый элемент формы автоматически помечается закладкой. Закладка - это специальная пометка в документе, которая позволяет ссылаться на определенное место в документе или фрагмент текста по имени закладки. Такие ссылки могут использоваться для быстрого перемещения по тексту, в формулах, для перекрестных ссылок.

Таблица 1.1 – Параметры полей переменной части формы

п/п	Элемент поля	Тип поля	Параметры поля
	Текст	Обычный текст	Максимальная длина текста Текст по умолчанию
		Число	Ограничение длины Число по умолчанию Формат числа
		Дата (время)	Ограничение длины Дата по умолчанию Формат даты
		Текущая дата	Формат текущей даты
		Текущее время	Формат текущего времени
		Вычисление	Выражение (формула вычисления, начинающаяся со знака =, содержит имена закладок, константы, знаки операций) Ограничение длины Формат выражения
	Списки	Элементы списка	Значение каждого элемента списка Порядок следования элементов списка
	Флажок		Размер флажка (в пунктах) Состояние по умолчанию (снят, установлен)

Элемент формы (поле) может иметь статус только для чтения на момент заполнения формы, это устанавливается при снятии отметки Разрешить изменение.

Чтобы пользователи могли вводить данные только в предназначенные для этого поля, но не могли изменять вид формы, необходимо установить защиту, выполнив следующие действия:

- выберите команду сервис, установить защиту;
- установите переключатель Запретить любые изменения, кроме в положение ввода данных в полях форм;
- введите пароль в поле Пароль. Пользователи, не знающие пароля, смогут вводить данные в полях, но не смогут изменить форму;
- нажмите кнопку <ОК>.

Для установки и снятия защиты во время разработки или изменения формы можно нажать кнопку <Защита формы> на панели инструментов Формы.

Задание 1

1. Разработайте шаблон для подготовки документов в форме бланка-заказа следующего содержания, оставляя незаполненными выделенные серым цветом поля текста:

Лист 1

Название организации
 Страна, индекс, город
 Адрес организации.....
 Телефон, факс Оплата:

ДОГОВОР ПОСТАВКИ (СЧЕТ-ЗАКАЗ) № (номер) (текущая дата)
 Счет следует оплатить и сообщить номер платежного поручения до (дата)
Покупатель:

Продавец:

Наименование	Кол-во единиц	Цена за единицу	Сум	
			ма	...
.....
.....
.....
.....
		Товарная сумма
		Сумма за доставку
		Итого к оплате
		НДС

Условия доставки:

Тип оплаты:

Особые условия:

Лист 2

Название организации

Страна, индекс, город,,

Адрес организации

Телефон, факс, Оплата:

УСЛОВИЯ ДОГОВОРА ПОСТАВКИ (СЧЕТА-ЗАКАЗА)

№ (номер) (текущая дата)

1. (Название организации), именуемый в дальнейшем Поставщик, с одной стороны, и (Название организации), именуемый в дальнейшем Покупатель, с другой стороны, заключили настоящий договор о нижеследующем.

2. Настоящие условия продажи и поставки являются обязательными к исполнению Продавцом и Покупателем по наименованиям, количеству, ценам, срокам, порядку оплаты, согласованными сторонами в настоящем договоре поставке (счете-заказе).

3. Покупатель имеет право предъявлять претензии по количеству и качеству товаров в соответствии с договором поставки и действующим законодательством.

4. Продавец не несет ответственности за какой-либо сопутствующий ущерб, включая какие-либо убытки, связанные с товарооборотом, потерей прибыли.

5. Поставка товара осуществляется согласно договору поставки и условиям, указанным в настоящем договоре поставки (счете-заказе).

6. Покупатель обязан оплатить товары в соответствии с договором поставки и условиями, указанными в настоящем договоре поставки (счете-заказе). В случае просрочки платежа за поставляемые товары Покупатель обязан уплатить Продавцу неустойку, размер которой указан в договоре поставки.

Покупатель: _____

Продавец: _____

2. Сохраните шаблон в файл под именем «Шаблон_ФИО».

3. Используя разработанный шаблон бланка-заказа в качестве формы, установите на поле формы следующие поля и заполните один документ, используя разработанную форму. Заполненный документ на основе разработанного шаблона сохраните в файл под именем «Заказ_ФИО».

Пояснения по заполнению формы:

- *текущая дата*, в параметрах необходимо указать тип поля – текущая дата и выбрать формат даты;

Заказ) № 123 20.05.2009

- *дата оплаты*, в параметрах необходимо указать тип поля – дата и выбрать формат даты; оплатить и сообщить номер поручения до

- *наименование товара*, тип поля – обычный текст;

- *количество единиц*, закладкам необходимо присвоить имя Quantity_ и т.д. (в дальнейшем их можно будет использовать при разработке макроса, обслуживающего разрабатываемый шаблон), тип поля – числовой и указать формат числа

Кол-во еди- ниц
.....

- *цена за единицу*, закладкам необходимо присвоить имя Price_1 и т.д., тип поля – числовой и указать тип числа;

- *сумма*, закладке присвоить имя Amount_1 и т.д., а также вставить формулу для подсчета суммы и указать тип числа;

- *товарная сумма заказа*, закладке присвоить имя Subtotal, а также вставить формулу для подсчета товарной суммы заказа и указать формат числа;

- *сумма за доставку*, закладке присвоить имя Shipping и указать формат числа;

- *итого к оплате*, закладке присвоить имя Total, установить формулу для подсчета итоговой суммы и указать формат числа;

- *НДС*, закладке присвоить имя SalesTax и указать формат числа;

- *условия доставки*, вставить поле со списком, который должен содержать следующие элементы: самовывоз, ж/д транспорт, автотранспорт;

- *тип оплаты*, необходимо вставить поле со списком, который должен содержать следующие элементы: наличный, безналичный;

- *особые условия*, тип поля – обычный текст;

- *остальные поля*, закрашенные серым цветом, - текстовые поля формы с ограниченной длиной текста.

Задание 2

Разработайте форму **Электронная анкета** (приложение 1), которая позволит в интерактивном режиме работы вводить номер курса, выбирать названия факультета, давать ответы на поставленные в анкете вопросы путем заполнения полей или выбором соответствующих переключателей.

Форма документа должна быть разработана как шаблон, который состоит из нескольких разнотипных по технологии разработки зон:

- текст, поясняющий технологию работы с формой;

- рисунок;

- поле ввода номера курса;

- список наименований факультетов (добавление нового факультета или корректировка его названия в процессе ввода невозможны);

- таблица А, которая состоит из постоянной части виде фиксированных названий строк (1-7) и переменной части виде полей ввода во 2-ю колонку и ячейки итогов;

- таблица В, которая состоит из постоянной части виде фиксированных названий строк (1-8) и переменной части виде полей ввода во 2-ю колонку и ячейки итогов;
- таблицы Уровень дохода, Дополнительные заработки, Семейное положение. Каждая таблица состоит из постоянной части в виде заголовка и фиксированных названий строк, а также переменной части в виде флажков.

Выполнение задания рекомендуется производить по этапам:

- 1 этап – создание постоянной части формы в виде текста и таблиц;
- 2 этап – внедрение рисунка;
- 3 этап – создание переменной части формы;
- 4 этап – установка защиты и сохранение формы.

Контрольные вопросы

1. Что такое форма и для чего она предназначена
2. Виды форм
3. Из каких частей состоит форма
4. Что относится к постоянной части формы
5. Что относится к переменной части формы
6. Что такое закладка и для чего она предназначена

Табличный процессор Microsoft Excel. Создание макросов.

Если в процессе работы в Microsoft Excel вы часто сталкиваетесь с необходимостью выполнять повторяющиеся процедуры, такие как применение курсива и больших букв в заголовках или запись одних и тех же категорий в каждый из рабочих листов книги, вы можете сэкономить время, автоматизировав выполнение таких задач. Вы можете создать макросы почти для всех последовательностей команд Microsoft Excel.

Наиболее простым способом создания макроса является его запись, заключающаяся в перечислении действий. Для этого включите процедуру записи макроса, определите последовательность команд макроса, а затем выключите запись макроса. В процессе записи макроса включенные в него команды протоколируются, автоматически переводятся на язык Visual Basic и хранятся в отдельном листе модуля.

После завершения записи макроса вы можете запустить его, выбрав его имя в диалоговом окне Макрос (Macro). Допускается также вызов макроса с помощью специально созданной кнопки или определенной комбинации клавиш.

Еще до начала записи макроса необходимо определить, какие команды будут выполняться с помощью макроса, и очередность их выполнения. После того как вы выберете команду Макрос - Начать запись в меню Сервис, все, что вы будете вводить с клавиатуры, и все команды будут записаны. Вам также необходимо продумать имя и описание макроса, соответствующие его назначению.

В записи макросов по умолчанию используются абсолютные ссылки. При создании макроса, результатом выполнения которого будет ввод данных, вам может потребоваться поменять ссылки на относительные, иначе данные будут вводиться в одни и те же ячейки, независимо от того, какая ячейка была выделена перед запуском макроса.

По умолчанию запустить макрос можно только в том случае, если рабочая книга, в которой он создан, открыта. Для того чтобы иметь доступ к макросу в любое время, в диалоговом окне Запись макроса в списке Сохранить в выберите Личная книга макросов. Личная книга макросов представляет собой скрытый файл, который Microsoft Excel создает при выборе соответствующей опции, и при каждом последующем запуске Microsoft Excel этот файл открывается автоматически.

Перед тем как сохранить макрос, всегда проверяйте правильность его выполнения. Для запуска макроса в меню Сервис выберите команду Макрос - Макросы. Откроется диалоговое окно Макрос. В списке поля имен макросов выделите его имя. Запуская макрос, вы выполняете записанные в нем действия. В некоторых случаях перед началом запуска макроса необходимо выделить ячейку или несколько ячеек, которые подлежат обработке с помощью макроса.

Чтобы другие пользователи могли разобраться в работе макросов, к макросам также следует добавлять комментарии или документировать их. Документирование макроса начинается с присвоения имени и описания. Иногда необходимо заглянуть также и внутрь макроса, чтобы понять, какие действия выполняет каждый его шаг.

Некоторые фрагменты текста в записи макроса выделены зеленым цветом, некоторые - синим, а некоторые - черным. Зеленым цветом выделяются комментарии. Сведения, содержащиеся в комментариях, игнорируются в процессе выполнения макросов. Ключевые слова, то есть слова со специальными значениями в Visual Basic, выделяются синим цветом (в вашем модуле это начало и конец макроса). Сами инструкции или шаги макроса записываются черным цветом, а ошибки, допущенные при записи инструкций, выделяются красным. Имя макроса и его описание записаны как комментарии и выделены зеленым цветом. В тексте макроса комментарии всегда начинаются с апострофа или со слова «Rem», что позволяет приложению Microsoft Excel идентифицировать их как комментарии.

Документировать макрос можно, добавляя комментарии к каждой инструкции. Комментарии могут располагаться как непосредственно перед инструкцией, так и после нее. Нет ограничений на содержание и размер комментариев, необходимо лишь начинать ввод комментариев с апострофа, что позволит приложению Microsoft Excel отличить их от инструкций. Текст комментария будет выделен в модуле зеленым цветом автоматически. Добавление комментариев не влияет на выполнение макроса. Если вы забыли поставить перед строкой комментария символ апострофа, то при попытке запуска макроса вы получите сообщение об ошибке.

Любой макрос можно запустить из диалогового окна Макрос, но это не самый быстрый способ. Если какой-либо макрос вы часто запускаете при работе с определенной таблицей данных, то для его запуска можно создать специальную кнопку. Тогда вместо выбора команды Макрос из меню Сервис макрос можно будет запускать, просто щелкнув на этой кнопке.

Для создания кнопки макроса сначала необходимо нарисовать эту кнопку, пользуясь графическими возможностями Microsoft Excel. Щелкните на кнопке Кнопка панели инструментов Формы и, перетаскивая указатель мыши, нарисуйте рамку кнопки на рабочем листе. В дальнейшем размер кнопки и ее расположение можно будет изменить. Создавая кнопку, вы одновременно связываете с ней один из макросов вашей рабочей книги, поэтому имя кнопки должно быть достаточно информативным, чтобы по нему можно было определить назначение макроса.

Создавая кнопку, вам не обязательно сразу же точно определять ее местоположение и размеры, так как эти параметры можно будет изменить позднее. Для редактирования кнопки ее необходимо сначала выделить. Однако в отличие от большинства объектов рабочего листа, для выделения кнопки макроса недостаточно просто щелкнуть на ней (так как в этом случае происходит вызов макроса). Чтобы выделить кнопку макроса, нужно нажать клавишу [Ctrl] и, не отпуская ее, щелкнуть на кнопке.

Для того чтобы запустить макрос с помощью кнопки, достаточно один раз щелкнуть на ней. Если этот макрос обрабатывает диапазон ячеек, то перед тем, как запустить его, необходимо выделить соответствующую ячейку или диапазон.

Обеспечить удобный доступ к макросу можно также с помощью назначения для него специальной комбинации клавиш. Можно связать макрос с любой комбинацией клавиш (например, [Ctrl] + ключ или [Ctrl] + [Shift] + ключ), которая не задействована для других команд. Назначение макросу комбинации клавиш осуществляется после вызова кнопки Параметры в диалоговом окне Макрос (Macro).

Задание 1

1. Создайте макрокоманду, автоматически вставляющую в таблицу заголовков из четырех строк.

Создание первого макроса – Вставка_строк.

1.1. Создайте новый документ: Пуск – Программы – Microsoft Excel. Открылась рабочая книга с чистым рабочим листом.

1.2. Выберите команды меню Вид – Панели инструментов – Настройка. Установите флажок опции Остановка записи, затем щелкните по кнопке Закрыть, если он не установлен.

1.3. Выберите команды меню Сервис – Макрос – Начать запись.

1.4. В диалоговом окне Запись макроса введите имя первого макроса Вставка_строк, в поле Сочетание клавиш – введите N (латинский алфавит), в поле Сохранить в введите Эта книга, в поле Описание введите Макрос вставляет строки. Нажмите ОК.

1.5. На панели инструментов Остановка записи щелкните кнопку Относительная ссылка.

Далее приступим к написанию макроса:

1.6. Выделите первую строку. Для этого: щелкните мышью по номеру строки 1.

1.7. Выполните команды меню Вставка – Строки, снимите выделение.

1.8. Еще раз выделите первую строку и повторите команды меню Вставка – Строки, снимите выделение.

1.9. Повторите действия еще два раза (вставьте еще две строки -всего четыре).

1.10. Щелкните по кнопке Остановить запись. Макрос записан.

2. Создайте макрокоманду, набирающую и форматирующую заголовки.

Создание второго макроса – Создание_заголовка

2.1. Повторите все операции по созданию макроса . Введите имя второго макроса – Создание_заголовка, Сочетание клавиш – М (латинский алфавит), в поле Описание введите Макрос вставляет заголовок. Кнопка Относительная ссылка должна быть нажата.

Далее приступим к написанию макроса:

2.2. Выделите диапазон ячеек A1:G1 и щелкните по кнопке Объединить и поместить по центру, установите начертание – Полуужирный, размер – 12. Введите текст: Докладная записка. Снимите выделение.

Выделите диапазон ячеек A2:G2 и повторите действия форматирования. Введите текст: Учет квартальных продаж.

Выделите диапазон ячеек A3:G3 и повторите действия форматирования. Введите текст: ООО "Здоровье".

Аналогично произведите форматирование четвертой строки. Введите текст: Коммерческий директор Иванов И.И.

2.3. Щелкните по кнопке Остановить запись. Макрос записан.

3. Использование макросов

3.1. На новом листе книги активизируйте ячейку A1. Выполните команды меню Сервис–Макрос– Макросы. Из диалогового окна выберите имя первого макроса Вставка_строк, нажмите кнопку Выполнить. На листе появились четыре новые строки.

3.2. Активизируйте ячейку A1 и запустите второй макрос Создание_заголовка другим способом: нажмите клавиши <Ctrl>+<M>. Заголовок вставлен.

Задание 2

Создайте в табличном редакторе Excel документ (приложение 2) и макрос для заполнения полей этого документа.

Контрольные вопросы

1. Для чего предназначены макросы
2. Как создаются макросы
3. Каким образом можно записать макрос
4. Где можно записать макрос
5. Как запустить макрос
6. Как запустить макрос с помощью комбинации клавиш
7. Как запустить макрос с помощью кнопки

Табличный процессор Microsoft Excel. Решение вычислительных задач. Модель межотраслевого баланса.

Систему (1) вида

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n = b_1, \\ a_{21}x_1 + a_{22}x_2 + \dots + a_{2n}x_n = b_2, \\ \dots, \\ a_{n1}x_1 + a_{n2}x_2 + \dots + a_{nn}x_n = b_n, \end{cases} \quad (3.1)$$

принято называть *системой n линейных алгебраических уравнений (СЛАУ) с n неизвестными*. При этом произвольные числа a_{ij} ($i = 1, 2, \dots, n; j = 1, 2, \dots, n$) называются *коэффициентами системы* (коэффициентами при неизвестных), а числа b_i ($i = 1, 2, \dots, n$) – *свободными членами*. Такая форма записи (3.1) алгебраической линейной системы называется *нормальной*. *Решением СЛАУ (3.1)* называется совокупность чисел x_i ($i = 1, 2, \dots, n$), при подстановке которых в систему каждое из ее уравнений обращается в тождество.

Систему (3.1) можно записать в матричной форме

$$A \times X = B, \quad (3.2)$$

где A – матрица коэффициентов при неизвестных (*матрица системы*):

$$A = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \dots & \dots & \dots & \dots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{pmatrix}, \quad (3.3)$$

X – вектор-столбец неизвестных $X = (x_1, x_2, \dots, x_n)^T$:

$$X = \begin{pmatrix} x_1 \\ x_2 \\ \dots \\ x_n \end{pmatrix}, \quad (3.4)$$

B – вектор-столбец свободных членов:

$$B = \begin{pmatrix} b_1 \\ b_2 \\ \dots \\ b_n \end{pmatrix}, \quad (3.5)$$

или $B = (b_1, b_2, \dots, b_n)^T$. Целое число n называется *размерностью системы*.

Система (3.2) может быть записана в развернутом виде

$$\begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \dots & \dots & \dots & \dots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{pmatrix} \times \begin{pmatrix} x_1 \\ x_2 \\ \dots \\ x_n \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ \dots \\ b_n \end{pmatrix}. \quad (3.6)$$

Система уравнений (3.6) называется *совместной*, если она имеет хотя бы одно решение, и *несовместной* – в противном случае. Совместная система (3.6) называется *определенной*, если она имеет единственное решение, и *неопределенной*, если она имеет больше одного решения.

РЕШЕНИЕ ЛИНЕЙНЫХ АЛГЕБРАИЧЕСКИХ СИСТЕМ

Все же методы решения линейных алгебраических систем принято делить на два класса: *прямые* и *итерационные*. Прямыми называются методы решения СЛАУ, которые приводят к решению за конечное число арифметических операций. В случае точной реализации операций реше-

ние будет точным, поэтому и прямые методы иногда называют *точными*. Итерационными методами называют такие методы, в которых точное решение может быть получено за счет реализации многократно повторяющихся действий (итераций). Эффективность способов решения системы (3.6) во многом определяется свойствами матрицы (3.3) A (размерностью, симметричностью, заполненностью и т.д.). MS Excel располагает рядом возможностей для работы с матрицами.

Табличные формулы и операции с матрицами

Табличные формулы или *формулы массива* – очень мощное вычислительное средство Excel, позволяющее работать с блоками рабочего листа как с отдельными ячейками. Табличные формулы в качестве результата возвращают массив значений. Поэтому перед вводом такой формулы необходимо выделить диапазон ячеек, куда будут помещены результаты. Потом набирается сама формула. Ввод ее в выделенный диапазон ячеек осуществляется нажатием комбинации клавиш Ctrl+Shift+Enter. Формула вводится во все ячейки выделенного интервала. При активизации любой ячейки из интервала, содержащего формулу массива, в строке формул отображается введенная формула, заключенная в *фигурные скобки*. Именно фигурные скобки являются признаком табличной формулы. Для выделения всего блока, содержащего табличную формулу, необходимо выделить одну из его ячеек, после чего нажать комбинацию клавиш Ctrl+/. Невозможно редактировать содержимое только одной ячейки из интервала с табличной формулой. Изменить можно только весь блок целиком, для чего он и должен быть предварительно выделен.

Например, пусть необходимо сложить две матрицы размера 3×3. Элементы первой матрицы (9 элементов) разместим в интервале A1:C3, второй – в диапазоне E1:G3. Под результат выделим интервал A5:C7. После чего, не снимая выделения, введем формулу =A1:C3+E1:G3, нажав комбинацию клавиш Ctrl+Shift+Enter. В ячейках интервала A5:C7 отобразится результат – сумма соответствующих элементов матриц, а в строке формул мы увидим {=A1:C3+E1:G3}. Пусть вместо сложения нам надо умножить первую матрицу на число 2. Для этого перемещаемся внутрь интервала A5:C7, выделяем его, нажав комбинацию Ctrl+/, вносим в формулу исправления =A1:C3*2, вводим ее нажатием Ctrl+Shift+Enter. В интервале A5:C7 увидим результат умножения, а в строке формул – табличную формулу {=A1:C3*2}.

К простейшим операциям с матрицами принято относить следующие: сложение и вычитание матриц, умножение и деление матрицы на число, перемножение матриц, транспонирование, вычисление обратной матрицы. Умножение (деление) матрицы на число, сложение (вычитание) матриц в Excel реализуются достаточно просто: с помощью обычных формул (поэлементное сложение или вычитание, умножение или деление на число), либо с использованием табличных формул, как это было описано выше. Для остальных матричных операций в Excel предусмотрены функции рабочего листа из категории «Арифметические и тригонометрические функции»:

- 1) МОПРЕД(матрица) – вычисление определителя матрицы,
- 2) МОБР(матрица) – вычисление обратной матрицы,
- 3) МУМНОЖ(матрица1;матрица2) – произведение матриц,
- 4) ТРАНСП(матрица) – транспонирование матрицы.

Первая из этих функций в качестве результата возвращает число (определитель матрицы), поэтому вводится как обычная формула (Enter). Последние три возвращают блок ячеек, поэтому должны вводиться как табличные формулы (Ctrl+Shift+Enter).

Группировка рабочих листов

Рассмотрим задачу решения СЛАУ на следующем примере

$$\begin{cases} 8x_1 + 2x_2 - 8x_3 = -24, \\ -2x_1 - 2x_2 - 10x_3 = -48, \\ -2x_1 + 4x_2 + 8x_3 = 18. \end{cases} \quad (3.7)$$

Т.е. будем решать систему из трех алгебраических уравнений относительно трех неизвестных. Размерность системы (3.7) $n=3$, матрица системы A (3.3) размерности 3×3 имеет вид

$$A = \begin{pmatrix} 8 & 2 & -8 \\ -2 & -2 & -10 \\ -2 & 4 & 8 \end{pmatrix}, \quad (3.8)$$

а вектор-столбец свободных членов (5) $B = (-24, -48, 18)^T$.

Решим СЛАУ (3.7) в среде MS Excel тремя различными способами. Для чего создадим рабочую книгу из трех листов и назовем ее Решение СЛАУ.xls. Поскольку исходные данные для трех различных способов решения (а значит и трех рабочих листов книги) одни и те же (матрица системы A (3.8) и вектор-столбец свободных членов B), то неплохо было бы их одновременно ввести в эти рабочие листы. Excel предоставляет такую возможность. Этот инструмент называется *группировкой* рабочих листов. Для того, чтобы применить средство Группа, необходимо выделить группируемые рабочие листы, щелкнув первый рабочий лист (Лист1), на котором будут вводиться данные, а затем, удерживая клавишу Ctrl, щелкнуть ярлычки листов (Лист2 и Лист3), куда одновременно должны вводиться те же самые данные. Либо, если группируемые рабочие листы расположены подряд, как в нашем случае, при выделенном первом (Лист1) щелкнуть, удерживая нажатой клавишу Shift, на ярлычке последнего (Лист3). После этого можно вводить данные на текущем рабочем листе, они автоматически появятся в одноименных ячейках на всех остальных сгруппированных листах. Признаком группировки нескольких листов является появившееся в строке заголовка слово [Группа] ([Group]), заключенное в квадратные скобки (рис. 1). После ввода группировку необходимо отменить. Для отмены необходимо выбрать любой из листов, не входящих в группу, либо щелкнуть правой кнопкой мыши на любом ярлычке листа из группы и выполнить команду Разгруппировать листы.

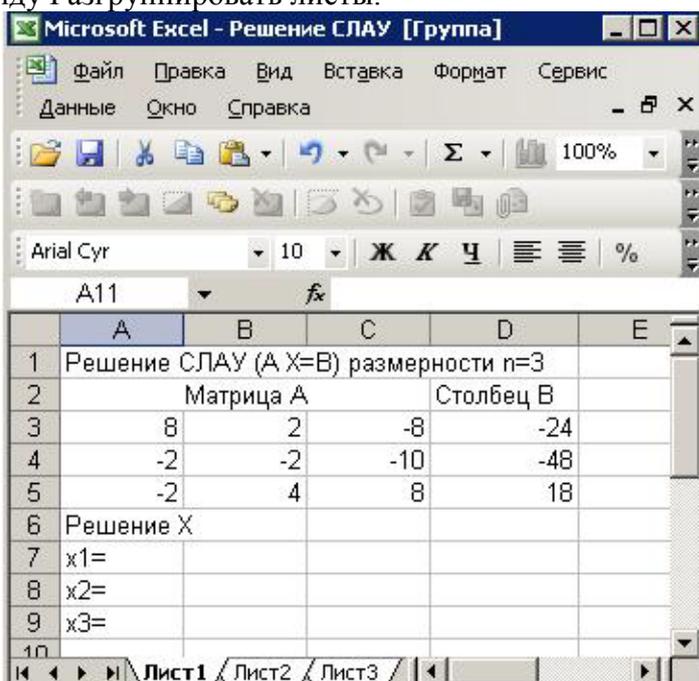


Рис. 1

Для решения рассматриваемой СЛАУ (3.7) сгруппируем листы (Лист1:Лист3), разместим в ячейках текущего листа (Лист1) A1, B2, D2, A6:A9 соответствующие поясняющие тексты (заголовки), в интервале A3:C5 – элементы матрицы A (8), а в интервале D3:D5 – элементы вектора B . Интервал B7:B9 зарезервируем под искомое решение – вектор X (3.4). После этих манипуляций все три рабочих листа примут одинаковый вид (рис. 1). Перед дальнейшей работой не забудьте разгруппировать рабочие листы.

Метод Крамера

Решение СЛАУ (6) находится по формулам Крамера

$$X = \begin{pmatrix} \frac{\det A_1}{\det A} \\ \frac{\det A_2}{\det A} \\ \dots\dots\dots \\ \frac{\det A_n}{\det A} \end{pmatrix}, \quad (3.9)$$

где $\det A = |A|$ – определитель матрицы (3.3) системы (главный определитель), $\det A_i = |A_i|$ ($i = 1, 2, \dots, n$) – определители матриц A_i (вспомогательные определители), которые получаются из A заменой i -го столбца на столбец свободных членов B (3.5). Линейная алгебраическая система несовместна (не имеет решений), если $\det A=0$. Для рассматриваемой СЛАУ (3.7) вспомогательные матрицы имеют следующий вид

$$A_1 = \begin{pmatrix} -24 & 2 & -8 \\ -48 & -2 & -10 \\ 18 & 4 & 8 \end{pmatrix}, A_2 = \begin{pmatrix} 8 & -24 & -8 \\ -2 & -48 & -10 \\ -2 & 18 & 8 \end{pmatrix}, A_3 = \begin{pmatrix} 8 & 2 & -24 \\ -2 & -2 & -48 \\ -2 & 4 & 18 \end{pmatrix}. \quad (3.10)$$

Разместим их на рабочем листе (рис. 2). Причем сделаем это не путем простого копирования соответствующих значений, а вводом формул с использованием абсолютных ссылок (рис. 3) на элементы матрицы A из интервала A3:C5 и элементы вектора B из интервала D3:D5 (рис. 1). Во-первых, это ускорит процесс ввода матриц A_i ($i = 1, 2, 3$) (формулы введем только в интервал A11:C13 матрицы A_1 и в интервал E11:E13 первого столбца матрицы A_2 , далее же будем их блоками только копировать: A11:A13 в F11:F13 и в K11:K13, B11:B13 в J11:J13, C11:C13 в G11:G13, E11:E13 в I11:I13). Во-вторых, это сделает проектируемую таблицу универсальной в том смысле, что можно будет изменять только исходные данные (матрицу системы A в интервале A3:C5 и вектор-столбец свободных членов B в D3:D5), а все остальное (в том числе и решение СЛАУ) будет автоматически вычисляться.

10	Матрица A1			Матрица A2			Матрица A3		
11	-24	2	-8	8	-24	-8	8	2	-24
12	-48	-2	-10	-2	-48	-10	-2	-2	-48
13	18	4	8	-2	18	8	-2	4	18

Рис. 2

10	Матрица A1			Матрица A2			Матрица A3		
11	=D\$3	=B\$3	=C\$3	=A\$3	=D\$3	=C\$3	=A\$3	=B\$3	=D\$3
12	=D\$4	=B\$4	=C\$4	=A\$4	=D\$4	=C\$4	=A\$4	=B\$4	=D\$4
13	=D\$5	=B\$5	=C\$5	=A\$5	=D\$5	=C\$5	=A\$5	=B\$5	=D\$5

Рис. 3

Далее, воспользовавшись функцией МОПРЕД(матрица), вычислим определители всех матриц (рис. 4). Аналогичная формула (=МОПРЕД(A3:C5)) для вычисления определителя матрицы A записана в ячейку C18. Осталось по формулам Крамера (3.9) найти решение системы (3.7). Соответствующие формулы Excel запишем в интервал решения B7:B9 (рис. 5), в котором и увидим результат (рис. 6). Обратите внимание на то (рис. 5), что при вычислении x_i ($i = 1, 2, 3$) анализируется значение определителя матрицы системы A , вычисленное в ячейке E8, и, если оно равно нулю (система несовместна), то в B7 помещается текст «Решения нет», а в ячейки B8 и B9 – пустые строки.

15	det A1=	=МОПРЕД(A11:C13)
16	det A2=	=МОПРЕД(E11:G13)
17	det A3=	=МОПРЕД(I11:K13)

Рис. 4 а

(установить целевую ячейку C7 равной нулю, решение в изменяемых ячейках B7:B9, ограничения заданы формулами в ячейках C8 и C9). После щелчка по кнопке Выполнить в интервале B7:B9 получим результат (рис. 8) – решение СЛАУ (3.7).

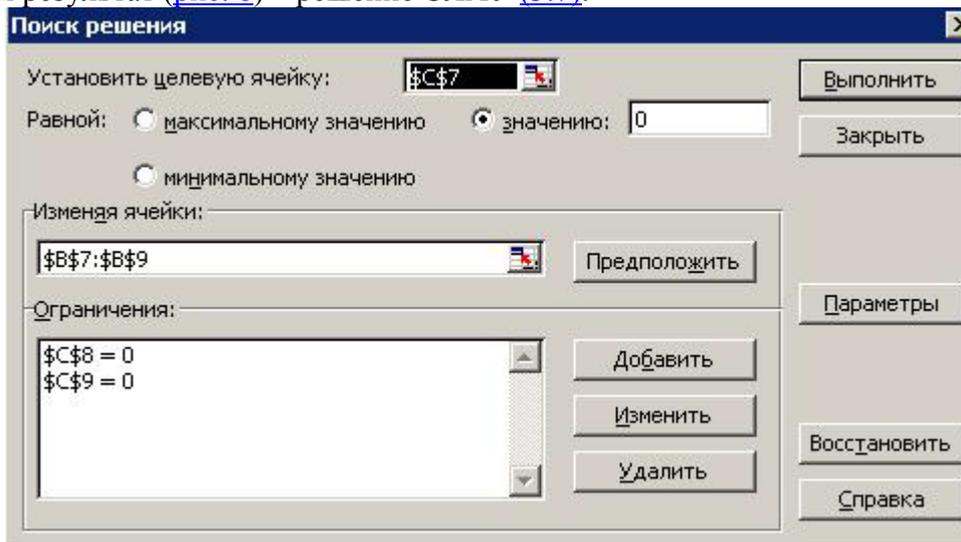


Рис. 7

Б	Решение X			
7	x1=	3	0	<< целевая функция
8	x2=	-4	0	<< первое ограничение
9	x3=	5	0	<< второе ограничение

Рис. 8

В завершение работы можно защитить ячейки созданных таблиц от несанкционированного, часто случайного, изменения и скрыть формулы, по которым находится решение СЛАУ. Для этого существует стандартное средство Excel – пункт меню Сервис/Защита/Защитить лист. Перед этим необходимо снять защиту с ячеек, содержащих исходные данные (A3:C5 – элементы матрицы A (8), и D3:D5 – элементы вектора B), выделив эти интервалы, выбрав меню Формат/Ячейки вкладка Защита и сбросив флажок Защищаемая ячейка. Для ячеек же, содержащих формулы, надо в этом диалоге (Формат ячеек) установить флажок Скрыть формулы. Надо знать, что после такой защиты невозможно будет воспользоваться средством Поиск решения. Поэтому защитить ячейки и скрыть формулы можно на первом и втором листах. В случае необходимости можно скрыть и отображаемую в ячейках информацию, поставив в соответствие этим ячейкам пользовательский формат ;;; (три точки с запятой).

Контрольные вопросы

1. Что принято называть системой линейных уравнений
2. Что такое коэффициенты системы
3. Что такое свободные члены
4. Как записать систему линейных уравнений в матричной форме
5. Какая система называется совместной, несовместной
6. Какая система называется определенной, неопределенной
7. Какие существуют методы решения линейных алгебраических систем
8. Какие матричные операции предусмотрены в Excel
9. Как осуществляется группировка рабочих листов
10. В чем заключается метод Крамера
11. Как осуществляется поиск решения

**Математический пакет MathCAD. Решение вычислительных задач.
Статистический анализ данных. Временные ряды.**

Методы решения систем линейных алгебраических уравнений можно разделить на точные и приближенные.

Метод решения задачи относят к классу точных, если в предположении отсутствия округлений с его помощью можно найти решение в результате конечного числа арифметических и логических операций.

Метод Гаусса – один из точных методов решения невырожденной системы линейных алгебраических уравнений. Метод Гаусса, его еще называют методом гауссовых исключений, состоит в том, что систему n линейных алгебраических уравнений относительно n неизвестных x_1, x_2, \dots, x_n

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n = b_1, \\ a_{21}x_1 + a_{22}x_2 + \dots + a_{2n}x_n = b_2, \\ \dots \\ a_{n1}x_1 + a_{n2}x_2 + \dots + a_{nn}x_n = b_n. \end{cases}$$

Приводят последовательным исключением неизвестных к эквивалентной системе с треугольной матрицей

$$\begin{cases} x_1 + c_{12}x_2 + \dots + c_{1n}x_n = d_1, \\ x_2 + \dots + c_{2n}x_n = d_2, \\ \dots \\ x_n = d_n, \end{cases}$$

Решение которой находят по рекуррентным формулам

$$x_n = d_n, \quad x_i = d_i - \sum_{k=i+1}^n c_{ik}x_k, \quad i = n-1, n-2, \dots, 1.$$

В матричной записи это означает, что сначала (прямой ход метода Гаусса) элементарными операциями над строками приводят расширенную матрицу системы к ступенчатому виду:

$$A_p = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} & b_1 \\ a_{21} & a_{22} & \dots & a_{2n} & b_2 \\ \dots & \dots & \dots & \dots & \dots \\ a_{n1} & a_{n2} & \dots & a_{nn} & b_n \end{pmatrix} \Rightarrow \begin{pmatrix} 1 & c_{12} & \dots & c_{1n} & d_1 \\ 0 & 1 & \dots & c_{2n} & d_2 \\ \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & 1 & d_n \end{pmatrix}$$

а затем (обратный ход метода Гаусса) эту ступенчатую матрицу преобразуют так, что в первых n столбцах получилась единичная матрица:

$$\begin{pmatrix} 1 & 0 & \dots & 0 & x_1 \\ 0 & 1 & \dots & 0 & x_2 \\ \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & 1 & x_n \end{pmatrix}$$

Последний, $(n+1)$ -й столбец этой матрицы содержит решение системы.

В MathCad прямой и обратный ходы метода Гаусса выполняет функция `rref(A)`.

Решение системы линейных алгебраических уравнений методом простых итераций

Точные методы решения линейных систем применяют для решения линейных систем относительно небольшой размерности (до 10^3). Для решения систем большей размерности ($10^3 - 10^6$) используют итерационные методы. Итерационные методы хороши для систем с разреженными матрицами. Рассмотрим простейший итерационный метод решения линейной системы – метод простых итераций.

Метод состоит в том, что система уравнений $Cx=d$ преобразуется к виду $x=b+Ax$ и ее решение вычисляется как предел последовательности

$$x^{(k)} = b + Ax^{(k-1)}, \quad k = 1, 2, \dots$$

Преобразовать систему $Cx=d$ к виду $x=b+Ax$ можно, выделив диагональные элементы:

$$x_i = \frac{1}{c_{ii}} \left(d_i - \sum_{i \neq j} c_{ij} x_j \right), \quad i = 1, 2, \dots, n.$$

Для того чтобы сформулировать достаточное условие сходимости метода, напомним определения норм, наиболее часто употребляемых при исследовании линейных систем. Понятие нормы позволяет оценить степень близости двух векторов. В частности, если норма разности точного и приближенного решений системы мала, то, по-видимому, приближенное решение хорошо аппроксимирует точное решение.

Существует много способов введения нормы вектора. Чаще всего используются следующие три нормы:

$$\|x\|_1 = \sum_{i=1}^n |x_i|; \quad \|x\|_2 = \sqrt{\sum_{i=1}^n |x_i|^2}; \quad \|x\|_i = \max_i |x_i|,$$

$$\text{где } x = (x_1, x_2, \dots, x_n).$$

Внешне столь различные, эти нормы эквивалентны: когда некоторая последовательность векторов по одной из этих норм стремится к нулю, то она стремится к нулю и по другой норме.

Если для векторов $x = (x_1, x_2, \dots, x_n)$ введена норма $\|x\|$, то согласованной с ней нормой матриц называют величину

$$\|A\| = \sup_{x \neq 0} \frac{\|Ax\|}{\|x\|}.$$

Так, в случае нормы $\|x\|_1$ согласованная норма матрицы равна $\|A\|_1 = \max_j \sum_{i=1}^n |a_{ij}|$, а в случае нормы $\|x\|_i$ согласованная норма матрицы равна $\|A\|_i = \max_i \sum_{j=1}^n |a_{ij}|$. Обе эти нормы легко вычислить.

Сложнее дело обстоит с вычислением нормы $\|A\|_2$, согласованной с нормой векторов $\|x\|_2$. Точная формула для нее имеет вид $\|A\|_2 = \sqrt{\lambda_{\max}(AA^T)}$, где $\lambda_{\max}(AA^T)$ - максимальное собственное значение матрицы AA^T .

Вычисление нормы $\|A\|_2$ - трудоемкая задача. Однако, поскольку справедливо неравенство $\|A\|_2 \leq \|A\|_e$, где норма $\|A\|_e = \sqrt{\sum_{i=1}^n \sum_{j=1}^n |a_{ij}|^2}$ вычисляется просто, в оценках вместо $\|A\|_2$ можно использовать $\|A\|_e$. Норму $\|A\|_e$ называют евклидовой нормой матрицы.

Для сходимости метода простых итераций $x^{(k)} = b + Ax^{(k-1)}$ достаточно, чтобы выполнялось условие $\|A\| < 1$ по какой-либо норме матрицы, согласованной с нормой векторов. В качестве условия окончания итерационного процесса можно взять условие

$$\frac{\|x^{(k)} - x^{(k-1)}\|}{\|x^{(k)}\|} \leq \varepsilon,$$

где ε - заданная погрешность приближенного решения $x \approx x^{(k)}$.

Задание 1

Решите заданную систему линейных алгебраических уравнений как матричное уравнение $Ax = b$.

Порядок выполнения задания

1. Введите матрицу системы и матрицу-столбец правых частей.
2. Вычислите решение системы по формуле $x = A^{-1}b$.
3. Проверьте правильность решения умножением матрицы системы на вектор-столбец решения.
4. Найдите решение системы с помощью функции `lsolve` и сравните результаты вычислений.

Пример выполнения задания

Фрагмент рабочего документа Mathcad, содержащий решение системы

$$\begin{cases} x + 2y + 3z = 7, \\ x - 3y + 2z = 5, \\ x + y + z = 3. \end{cases}$$

Зададим матрицу системы A и правую часть b

$$A := \begin{pmatrix} 1 & 2 & 3 \\ 1 & -3 & 2 \\ 1 & 1 & 1 \end{pmatrix} \quad b := \begin{pmatrix} 7 \\ 5 \\ 3 \end{pmatrix}$$

Вычислим решение системы по формуле $x = A^{-1}b$ и проверим правильность решения

$$x := A^{-1}b \quad x = \begin{pmatrix} 1 \\ 0 \\ 2 \end{pmatrix} \quad A \cdot x - b = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}$$

Решим систему с помощью функции `lsolve` и сравним результат с решением $x = A^{-1}b$

$$\text{lsolve}(A, b) = \begin{pmatrix} 1 \\ 0 \\ 2 \end{pmatrix} \quad x - \text{lsolve}(A, b) = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}$$

Задание 2

Найдите решение заданной в задании 1 системы линейных алгебраических уравнений методом Гаусса.

Порядок выполнения задания

1. Введите матрицу системы и матрицу-столбец правых частей
2. Сформируйте расширенную матрицу системы
3. Приведите расширенную матрицу системы к ступенчатому виду
4. Сформируйте столбец решения системы
5. Проверьте правильность решения

Пример выполнения задания

Ниже приведен фрагмент рабочего документа MathCad, содержащий решение методом Гаусса системы трех линейных уравнений относительно трех неизвестных.

Зададим матрицу системы A и правую часть b

$$A := \begin{pmatrix} 1 & 2 & 3 \\ 1 & -3 & 2 \\ 1 & 1 & 1 \end{pmatrix} \quad b := \begin{pmatrix} 7 \\ 5 \\ 3 \end{pmatrix}$$

Сформируем расширенную матрицу системы Ag добавлением к матрице системы справа столбца правых частей

$$Ar := \text{augment}(A, b) \quad Ar = \begin{pmatrix} 1 & 2 & 3 & 7 \\ 1 & -3 & 2 & 5 \\ 1 & 1 & 1 & 3 \end{pmatrix}$$

Приведем расширенную матрицу системы Ag к ступенчатому виду. Функция `rref` приводит расширенную матрицу, системы к ступенчатому виду, выполняя прямой и обратный ходы гауссова исключения

$$Ag := \text{rref}(Ar) \quad Ag = \begin{pmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 2 \end{pmatrix}$$

Выделим блок матрицы Ag – ее последний столбец, содержащий решение системы

$$x := \text{submatrix}(Ag, 0, 2, 3, 3) \quad x = \begin{pmatrix} 1 \\ 0 \\ 2 \end{pmatrix}$$

Проверим правильность решения

$$A \cdot x - b = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}$$

Решим систему с помощью функции `lsolve` и сравним с решением, полученным методом Гаусса

$$\text{lsolve}(A, b) = \begin{pmatrix} 1 \\ 0 \\ 2 \end{pmatrix} \quad x - \text{lsolve}(A, b) = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}$$

Задание 3

Найдите методом простых итераций приближенное решение заданной линейной системы $Cx = d$.

Порядок выполнения задания

1. Введите матрицы C и d .
2. Преобразуйте исходную систему $Cx = d$ к виду $x = b + Ax$.
3. Проверьте достаточное условие сходимости.
4. Определите нулевое (начальное) приближение решения.
5. Задайте количество итераций.
6. Вычислите последовательные приближения.
7. Вычислите погрешности найденных приближений.
8. Сформулируйте выводы.

Пример выполнения задания

Приведен фрагмент рабочего документа MathCad, содержащий решение методом простых итераций линейной системы

$$\begin{cases} 100x_1 + 6x_2 - 2x_3 = 200, \\ 6x_1 + 200x_2 - 10x_3 = 600, \\ x_1 + 2x_2 + 100x_3 = 500. \end{cases}$$

ORIGIN := 1

Нумерация столбцов и строк матрицы начинается с 1.

Зададим матрицы C и d системы $Cx = d$

$$C := \begin{pmatrix} 100 & 6 & -2 \\ 6 & 200 & -10 \\ 1 & 2 & 100 \end{pmatrix} \quad d := \begin{pmatrix} 200 \\ 600 \\ 500 \end{pmatrix}$$

Сформируем матрицы A и b эквивалентной системы $x = b + Ax$

$$i := 1..3 \quad j := 1..3 \quad b_i := \frac{d_i}{C_{i,i}} \quad A_{i,j} := \frac{-C_{i,j}}{C_{i,i}} \quad A_{i,i} := 0$$

$$A = \begin{pmatrix} 0 & -0.06 & 0.02 \\ -0.03 & 0 & 0.05 \\ -0.01 & -0.02 & 0 \end{pmatrix} \quad b = \begin{pmatrix} 2 \\ 3 \\ 5 \end{pmatrix}$$

Проверим выполнение достаточного условия сходимости : $\|A\|_1 = 0,08 < 1$ и зададим начальное приближение

$$\text{norm1}(A) := 0.08$$

$$x^{(1)} := b$$

Вычислим 10 последовательных приближений и погрешности каждого из них

$$k := 2..10 \quad x^{(k)} := b + A \cdot x^{(k-1)}$$

$$e_k := \frac{|x^{(k)} - x^{(k-1)}|}{|x^{(k)}|}$$

	1	2	3	4	5	6	7	8	9
x =	2	1.92	1.907	1.90704	1.90702	1.90702	1.90702	1.90702	1.90702
	3	3.19	3.1884	3.18864	3.18865	3.18865	3.18865	3.18865	3.18865
	5	4.92	4.917	4.91716	4.91716	4.91716	4.91716	4.91716	4.91716

$e^T =$	1	2	3	4	5
	0	0.03584	0.00218	$4.7345 \cdot 10^{-5}$	$2.29727 \cdot 10^{-6}$

Выводы. Если положить приближенное решение равным $x^{(4)}$, то погрешность приближенного решения не превышает 10^{-4} . Если же положить приближенное решение равным $x^{(10)}$, то погрешность не превышает 10^{-12} .

Можно проверить вычисления подстановкой приближенного решения в исходную систему

$$X := x^{(4)} \quad X = \begin{pmatrix} 1.90704 \\ 3.18864 \\ 4.91716 \end{pmatrix} \quad C \cdot X - d = \begin{pmatrix} 0.00112 \\ -0.0014 \\ 0.00052 \end{pmatrix}$$

$$X := x^{(10)} \quad X = \begin{pmatrix} 1.90702 \\ 3.18865 \\ 4.91716 \end{pmatrix} \quad C \cdot X - d = \begin{pmatrix} 4.80327 \times 10^{-12} \\ -4.3201 \times 10^{-12} \\ 9.66338 \times 10^{-13} \end{pmatrix}$$

Контрольные вопросы

1. С помощью какой функции можно решить систему линейных уравнений

2. В чем заключается метод Гаусса
3. Для чего предназначены функции augment, rref, submatrix
4. В чем заключается метод простых операций
5. В чем заключается метод Зейделя

Математический пакет MathCAD. Регрессионный анализ. Рыночная модель.

Основными матричными операциями являются умножение матрицы на число, сложение и перемножение двух матриц.

По определению, чтобы умножить матрицу на число, нужно умножить на это число все элементы матрицы.

Суммой двух матриц одинаковой размерности называется матрица той же размерности, каждый элемент которой равен сумме соответствующих элементов слагаемых.

Операция умножения матрицы на матрицу определяется более сложным образом. Пусть заданы две матрицы A и B, причем число столбцов первой из них равно числу строк второй. Если

$$A = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \dots & \dots & \dots & \dots \\ a_{m1} & a_{m2} & \dots & a_{mn} \end{pmatrix}, B = \begin{pmatrix} b_{11} & b_{12} & \dots & b_{1k} \\ b_{21} & b_{22} & \dots & b_{2k} \\ \dots & \dots & \dots & \dots \\ b_{n1} & b_{n2} & \dots & b_{nk} \end{pmatrix},$$

то произведением матриц A и B называется матрица

$$C = \begin{pmatrix} c_{11} & c_{12} & \dots & c_{kn} \\ c_{21} & c_{22} & \dots & c_{2k} \\ \dots & \dots & \dots & \dots \\ c_{n1} & c_{n2} & \dots & c_{nk} \end{pmatrix}$$

Элементы которой вычисляются по формуле

$$c_{ij} = a_{i1}b_{1j} + a_{i2}b_{2j} + \dots + a_{in}b_{nj}, \quad i = 1, \dots, m, \quad j = 1, \dots, k.$$

Произведение матриц A и B обозначается AB, т.е. C=AB. Оно, вообще говоря, зависит от порядка сомножителей. Если AB=BA, то матрицы A и B называются перестановочными.

В множестве квадратных матриц определена единичная матрица – квадратная матрица, все диагональные элементы которой – единицы, а остальные – нули.:

$$E = \begin{pmatrix} 1 & 0 & \dots & 0 \\ 0 & 1 & \dots & 0 \\ \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & 1 \end{pmatrix}.$$

Единичная матрица чаще всего обозначается буквой E или E_n, где n – порядок матрицы. Непосредственным вычислением легко проверить основное свойство единичной матрицы: AE=EA=A.

Из приведенных ниже вычислений видно, что умножением на матрицы специального вида можно переставить в матрице столбцы или строки, вычислить сумму элементов любых строки или столбца, получить матрицу, равную строке или столбцу матрицы, реализовать операцию умножения матрицы на число, выполнить элементарное преобразование строк (столбцов), прибавив к элементам какой-либо строки матрицы элементы другой ее строки, умноженные на одно и то же число.

Зададим произвольную матрицу 3-го порядка D и единичную матрицу того же порядка.

$$D := \begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix} \quad E := \text{identity}(3) \quad E = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

Матрица не изменится, если ее умножить на единичную матрицу справа или слева.

$$E \cdot D = \begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix} \quad D \cdot E = \begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix}$$

Зададим вспомогательную матрицу Row – строку единиц.

$$\text{Row} := (1 \ 1 \ 1)$$

Результат умножения матрицы D на матрицу Row слева – суммы элементов матрицы D «по столбцам»

$$\text{Row} \cdot D = (12 \ 15 \ 18)$$

Зададим вспомогательные матрицы Col2 и Col3.

$$\text{Col2} := \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} \quad \text{Col3} := \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$$

В результате умножения D справа на матрицы Col2 и Col3 выделены соответственно 2-й и 3-й столбцы матрицы D.

$$D \cdot \text{Col2} = \begin{pmatrix} 2 \\ 5 \\ 8 \end{pmatrix} \quad D \cdot \text{Col3} = \begin{pmatrix} 3 \\ 6 \\ 9 \end{pmatrix}$$

Зададим вспомогательную матрицу Col – столбец единиц. Результат умножения D справа на матрицу Col – суммы элементов матрицы D «по строкам».

$$\text{Col} := \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} \quad D \cdot \text{Col} = \begin{pmatrix} 6 \\ 15 \\ 24 \end{pmatrix}$$

В результате умножения слева на матрицы Row2 и Row3 выделены соответственно 2-я и 3-я строки матрицы

$$\text{Row2} := (1 \ 0 \ 0) \quad \text{Row3} := (0 \ 1 \ 0)$$

$$\text{Row2} \cdot D = (1 \ 2 \ 3) \quad \text{Row3} \cdot D = (4 \ 5 \ 6)$$

Зададим матрицы перестановок P12 и P23

$$P12 := \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad P23 := \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix}$$

Умножение слева на P12 приводит к перестановке 1-й и 2-й строк, умножение справа – к перестановке соответствующих столбцов

$$P12 \cdot D = \begin{pmatrix} 4 & 5 & 6 \\ 1 & 2 & 3 \\ 7 & 8 & 9 \end{pmatrix} \quad D \cdot P12 = \begin{pmatrix} 2 & 1 & 3 \\ 5 & 4 & 6 \\ 8 & 7 & 9 \end{pmatrix}$$

Умножение слева на P23 приводит к перестановке 2-й и 3-й строк, умножение справа – к перестановке соответствующих столбцов.

$$P23 \cdot D = \begin{pmatrix} 1 & 2 & 3 \\ 7 & 8 & 9 \\ 4 & 5 & 6 \end{pmatrix} \quad D \cdot P23 = \begin{pmatrix} 1 & 3 & 2 \\ 4 & 6 & 5 \\ 7 & 9 & 8 \end{pmatrix}$$

Зададим элементарные матрицы E12 и E23

$$E21 := \begin{pmatrix} 1 & 0 & 0 \\ -4 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad E31 := \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -7 & 0 & 1 \end{pmatrix}$$

Результат умножения матрицы D слева на элементарные матрицы – сложение соответствующей строки матрицы D с другой строкой, умноженной на отличный от нуля внедиагональный элемент элементарной матрицы

$$E21 \cdot D = \begin{pmatrix} 1 & 2 & 3 \\ 0 & -3 & -6 \\ 7 & 8 & 9 \end{pmatrix} \quad E31 \cdot D = \begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 0 & -6 & -12 \end{pmatrix}$$

$$E31 \cdot E21 \cdot D = \begin{pmatrix} 1 & 2 & 3 \\ 0 & -3 & -6 \\ 0 & -6 & -12 \end{pmatrix}$$

$$E32 := \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & -2 & 1 \end{pmatrix}$$

Умножая матрицу D слева на произведение соответствующих элементарных матриц, можно привести матрицу D к ступенчатому виду

$$E_{32} \cdot E_{31} \cdot E_{21} \cdot D = \begin{pmatrix} 1 & 2 & 3 \\ 0 & -3 & -6 \\ 0 & 0 & 0 \end{pmatrix}$$

Транспонирование. Вычисление обратной матрицы.

Ортогональные матрицы.

Для прямоугольных матриц определена операция транспонирования. Рассмотрим произвольную матрицу A . Матрица, получающаяся из матрицы A заменой строк столбцами, называется транспонированной по отношению к матрице A и обозначается A^T :

$$A = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \dots & \dots & \dots & \dots \\ a_{m1} & a_{m2} & \dots & a_{mn} \end{pmatrix}, \quad A^T = \begin{pmatrix} a_{11} & a_{21} & \dots & a_{m1} \\ a_{12} & a_{22} & \dots & a_{m2} \\ \dots & \dots & \dots & \dots \\ a_{1n} & a_{2n} & \dots & a_{mn} \end{pmatrix}.$$

Квадратная матрица A называется обратимой, если существует квадратная матрица X , удовлетворяющая соотношениям $AX=XA=E$. Матрица X называется обратной к матрице A и обозначается A^{-1} , т.е. $AA^{-1}=A^{-1}A=E$.

Квадратная матрица A , для которой $A^T=A$, называется симметричной. Элементы такой матрицы, расположены симметрично относительно главной диагонали, равны.

Квадратная матрица U , для которой $U^{-1}=U^T$, называется ортогональной матрицей. Модуль определителя ортогональной матрицы равен единице; сумма квадратов элементов любого столбца ортогональной матрицы равна единице; сумма произведений элементов любого столбца ортогональной матрицы на соответствующие элементы другого столбца равна нулю. Такими же свойствами обладают строки ортогональной матрицы.

Фрагмент рабочего документа MathCad, содержащий символьные вычисления с ортогональной матрицей, приведен ниже:

Зададим квадратную матрицу 2-го порядка элементы которой – символы

$$\begin{pmatrix} \cos(\alpha) & \sin(\alpha) \\ -\sin(\alpha) & \cos(\alpha) \end{pmatrix}$$

Вычислим символьно определитель матрицы и обратную к ней матрицу. Покажем, что матрица ортогональна.

Введите ключевое слово `simplify` (щелчком по кнопке в панели `Symbolic Keyword`), введите в помеченной позиции знак определителя (щелчком по кнопке в панели `Matrix`), вставьте (введите или скопируйте) в помеченной позиции матрицу и щелкните по рабочему документу вне выделяющей рамки

$$\left| \begin{pmatrix} \cos(\alpha) & \sin(\alpha) \\ -\sin(\alpha) & \cos(\alpha) \end{pmatrix} \right| \text{ simplify } \rightarrow 1$$

Вычислим обратную матрицу символьно.

Введите ключевое слово `simplify` (щелчком по кнопке в панели `Symbolic Keyword`), введите в помеченной позиции выражение для обратной матрицы (матрица в степени -1) и щелкните по рабочему документу вне выделяющей рамки

$$\begin{pmatrix} \cos(\alpha) & \sin(\alpha) \\ -\sin(\alpha) & \cos(\alpha) \end{pmatrix}^{-1} \text{ simplify } \rightarrow \begin{pmatrix} \cos(\alpha) & -\sin(\alpha) \\ \sin(\alpha) & \cos(\alpha) \end{pmatrix}$$

Можно транспонировать, обратить и вычислить определитель матрицы символично, не используя simplify, а используя только знак символических вычислений (стрелка вправо), который вводит щелчком по кнопке в панели Symbolic Keyword:

$$\begin{pmatrix} \cos(\alpha) & \sin(\alpha) \\ -\sin(\alpha) & \cos(\alpha) \end{pmatrix}^T \rightarrow \begin{pmatrix} \cos(\alpha) & -\sin(\alpha) \\ \sin(\alpha) & \cos(\alpha) \end{pmatrix}$$

$$\left| \begin{pmatrix} \cos(\alpha) & \sin(\alpha) \\ -\sin(\alpha) & \cos(\alpha) \end{pmatrix} \right| \rightarrow \cos(\alpha)^2 + \sin(\alpha)^2$$

$$\begin{pmatrix} \cos(\alpha) & \sin(\alpha) \\ -\sin(\alpha) & \cos(\alpha) \end{pmatrix}^{-1} \rightarrow \begin{bmatrix} \frac{\cos(\alpha)}{\cos(\alpha)^2 + \sin(\alpha)^2} & \frac{-\sin(\alpha)}{\cos(\alpha)^2 + \sin(\alpha)^2} \\ \frac{\sin(\alpha)}{\cos(\alpha)^2 + \sin(\alpha)^2} & \frac{\cos(\alpha)}{\cos(\alpha)^2 + \sin(\alpha)^2} \end{bmatrix}$$

Можно выполнить те же вычисления, определив матрицу, как функцию α :

$$U(\alpha) := \begin{pmatrix} \cos(\alpha) & \sin(\alpha) \\ -\sin(\alpha) & \cos(\alpha) \end{pmatrix}$$

$$U(\alpha)^T \rightarrow \begin{pmatrix} \cos(\alpha) & -\sin(\alpha) \\ \sin(\alpha) & \cos(\alpha) \end{pmatrix}$$

$$U(\alpha)^{-1} \text{ simplify} \rightarrow \begin{pmatrix} \cos(\alpha) & -\sin(\alpha) \\ \sin(\alpha) & \cos(\alpha) \end{pmatrix}$$

$$U(\alpha)^{-1} \text{ simplify} \rightarrow \begin{pmatrix} \cos(\alpha) & -\sin(\alpha) \\ \sin(\alpha) & \cos(\alpha) \end{pmatrix}$$

$$U(\alpha)^{-1} \rightarrow \begin{bmatrix} \frac{\cos(\alpha)}{\cos(\alpha)^2 + \sin(\alpha)^2} & \frac{-\sin(\alpha)}{\cos(\alpha)^2 + \sin(\alpha)^2} \\ \frac{\sin(\alpha)}{\cos(\alpha)^2 + \sin(\alpha)^2} & \frac{\cos(\alpha)}{\cos(\alpha)^2 + \sin(\alpha)^2} \end{bmatrix}$$

Убедимся, что матрица ортогональная, т.е. $U^{-1}=U^T$, вычислив символично $U^{-1}=U^T$:

$$U(\alpha)^{-1} - U(\alpha)^T \text{ simplify} \rightarrow \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}$$

Убедимся, что матрица ортогональная, вычислив символично $U^T \times U$ и $U \times U^T$:

$$U(\alpha)^T U(\alpha) \text{ simplify} \rightarrow \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

$$U(\alpha) U(\alpha)^T \text{ simplify} \rightarrow \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

4. Методические указания к самостоятельной работе студентов

Самостоятельная работа по дисциплине «Информационные технологии обработки и анализа данных», направлена на углубление и закрепление знаний студента, на развитие практических умений и включает в себя следующие виды работ:

- работа с лекционным материалом, учебниками и учебными пособиями;
- изучение тем, вынесенных на самостоятельную проработку;
- подготовка к практическим занятиям;
- выполнение домашних индивидуальных заданий;
- подготовка к лабораторным работам;
- подготовка к текущему и итоговому контролю.