

Государственное бюджетное
профессиональное образовательное учреждение
«Кунгурский колледж агротехнологий и управления»

МЕТОДИЧЕСКИЕ УКАЗАНИЯ
ПО ВЫПОЛНЕНИЮ ПРАКТИЧЕСКИХ И ЛАБОРАТОРНЫХ РАБОТ
ПО ПРОФЕССИОНАЛЬНОМУ МОДУЛЮ
ПМ.06 СОПРОВОЖДЕНИЕ ИНФОРМАЦИОННЫХ СИСТЕМ
09.02.07 «Информационные системы и программирование»

Организация-разработчик: ГБПОУ «ККАТУ»

Разработчик:

ГБПОУ «ККАТУ»
(место работы)

преподаватель
(занимаемая должность)

А.В. Атушкина
(инициалы, фамилия)

Пояснительная записка

Методические указания по выполнению лабораторных и практических работ по ПМ.06 Сопровождение информационных систем разработаны в соответствии с рабочей программой профессионального модуля и предназначены для приобретения необходимых практических навыков и закрепления теоретических знаний, полученных обучающимися при изучении профессионального модуля, обобщения и систематизации знаний перед экзаменом.

Методические указания предназначены для обучающихся специальности 09.02.07 Информационные системы и программирование.

ПМ.06 Сопровождение информационных систем относится к профессиональному циклу, изучается на 4 курсе и при его изучении отводится значительное место выполнению практических работ.

Освоение содержания ПМ.06 Сопровождение информационных систем во время выполнения практических работ обеспечивает достижение обучающимися следующих **результатов:**

Результаты (освоенные проф. компетенции)	Основные показатели оценки результата	Формы и методы контроля результатов обучения	Оценка результатов обучения
ПК 6.1	Уметь разрабатывать техническое задание на сопровождение информационной системы	Наблюдение при выполнении практических заданий. Защита отчетов по практическим работам. Экспертное наблюдение за выполнением различных видов работ во время учебной и производственной практики	Оценка «отлично» - проанализирована предметная область функционирования системы; выделены и определены признаки системы по нескольким основаниям классификации; указаны все функции предложенной информационной системы; сформировано и обосновано несколько предложений по расширению перечня выполняемых функций. Сформированы и обоснованы предложения по реинжинирингу системы Оценка «хорошо» - проанализирована предметная область функционирования системы; выделены и определены признаки системы и указана ее принадлежность по классификации; указаны основные функции предложенной информационной системы; сформированы и обоснованы предложения по расширению перечня выполняемых функций. Сформированы предложения по реинжинирингу системы Оценка «удовлетворительно» - проанализирована предметная область функционирования системы; указана ее принадлежность по классификации; указаны функции предложенной информационной системы; сформированы предложения по расширению перечня выполняемых функций. Внесено хотя бы одно предложение по реинжинирингу системы
ПК 6.2	Уметь выполнять исправление ошибок в программном коде информационной системы		
ПК 6.3	Уметь разрабатывать обучающую документацию для пользователей информационной системы		
ПК 6.4	Уметь оценивать качество и надёжность функционирования информационной системы в соответствии с критериями технического задания		
ПК 6.5	Уметь осуществлять техническое сопровождение, обновление и восстановление данных ИС в соответствии с техническим заданием		

Результаты (освоенные общие компетенции)	Основные показатели оценки результата	Формы и методы контроля результатов обучения	Оценка результатов обучения
ОК 1	Умение понимать выбор способа решения задач профессиональной применительно к различным контекстам	Наблюдение при собеседовании с преподавателем, наблюдение за организацией деятельности в процессе промежуточной аттестации, наблюдение за организацией работы с информацией	Экспертное наблюдение за выполнением работ.
ОК 2	Демонстрирует навыки использования современные средства поиска, анализа и интерпретации информации, и информационные технологии для выполнения задач профессиональной деятельности анализ и		
ОК 3	Планирует и реализует собственное профессиональное и личностное развитие, предпринимательскую деятельность в профессиональной сфере, использует знания по финансовой грамотности в различных жизненных ситуациях		
ОК 4	Демонстрирует работу в команде, эффективно взаимодействует с коллективом и коллегами, руководством, клиентами		
ОК 5	Осуществляет устную и письменную коммуникацию на государственном языке с учетом особенностей социального и культурного контекста		
ОК 6	Проявляет гражданско-патриотическую позицию, демонстрировать осознанное поведение на основе традиционных общечеловеческих ценностей, в том числе с учетом гармонизации межнациональных и межрелигиозных отношений, применять стандарты антикоррупционного поведения;		
ОК 7	Содействует сохранению окружающей среды, ресурсосбережению, применять знания об изменении климата, принципы бережливого производства, эффективно действовать в чрезвычайных ситуациях		
ОК 8	Использует средства физической культуры для сохранения и укрепления здоровья в процессе профессиональной деятельности и поддержания необходимого уровня физической подготовленности		
ОК 9	Демонстрирует использование информационных технологий в профессиональной деятельности		

Результаты	Основные показатели оценки результата	Формы и методы контроля результатов обучения	Оценка результатов обучения
A/01.4	Может осуществлять сбор данных для выявления требований к типовой ИС в соответствии с трудовым заданием	Наблюдение при выполнении практических заданий. Защита отчетов по практическим работам. Экспертное наблюдение за выполнением различных видов работ во время учебной и производственной практики	Оценка «отлично» - проанализирована предметная область функционирования системы; выделены и определены признаки системы по нескольким основаниям классификации; указаны все функции предложенной информационной системы; сформировано и обосновано несколько предложений по расширению перечня выполняемых функций. Сформированы и обоснованы предложения по реинжинирингу системы Оценка «хорошо» - проанализирована предметная область функционирования системы; выделены и определены признаки системы и указана ее принадлежность по классификации; указаны основные функции предложенной информационной системы; сформированы и обоснованы предложения по расширению перечня выполняемых функций. Сформированы предложения по реинжинирингу системы Оценка «удовлетворительно» - проанализирована предметная область функционирования системы; указана ее принадлежность по классификации; указаны функции предложенной информационной системы; сформированы предложения по расширению перечня выполняемых функций. Внесено хотя бы одно предложение по реинжинирингу системы
A/04.4	Может осуществлять модульное тестирование ИС (верификация) в соответствии с трудовым заданием		
A/14.4	Может осуществлять идентификацию конфигурации ИС в соответствии с трудовым заданием		
B/02.5	Может осуществлять инженерно-техническую поддержку подготовки коммерческого предложения заказчику о создании (модификации) и вводе в эксплуатацию типовой ИС на этапе предконтрактных работ		
B/11.5	Может осуществлять модульное тестирование ИС (верификация) в рамках выполнения работ по созданию (модификации) и сопровождению ИС		
B/14.5	Может осуществлять создание пользовательской документации к модифицированным элементам типовой ИС		

Рабочая программа профессионального модуля предусматривает проведение практических работ в объеме 199 часов.

Порядок выполнения практической работы

- записать название работы, ее цель в тетрадь;
- выполнить основные задания в соответствии с ходом работы;
- выполнить индивидуальные задания.

Рекомендации по оформлению практической работы

Задания выполняются обучающимися по шагам. Необходимо строго придерживаться порядка действий, описанного в практической работе.

Результаты выполнения практических работ необходимо сохранять в своей папке на компьютере или USB – накопителе.

В случае пропуска занятий обучающийся осваивает материал самостоятельно в свободное от занятий время и сдает практическую работу с пояснениями о выполнении.

Требования к технике безопасности при выполнении лабораторных /практических работ

Вводный инструктаж

Инструкция №1 ПРАВИЛА ПОВЕДЕНИЯ СТУДЕНТОВ В КАБИНЕТЕ ИНФОРМАТИКИ

Кабинет информатики относится к кабинетам повышенной опасности, находясь в нем необходимо соблюдать требования настоящей инструкции.

1. Не заходите в кабинет без разрешения преподавателя.
2. Во время перемены все студенты выходят в коридор. В кабинете кроме преподавателя могут находиться только дежурные.
3. Запрещается находиться в кабинете в верхней одежде, грязной обуви.
4. Не бегайте по кабинет - можете получить травму или повредить оборудование.
5. Не бросайте мусор в кабинете, этим вы создаете пожарную опасность.
6. Будьте внимательны и дисциплинированы, точно выполняйте указания преподавателя.
7. Не приступайте к выполнению работы без разрешения преподавателя.
8. Не пользуйтесь электрическими розетками для шалости, это опасно для вашей жизни.

Инструктаж №2 ПРАВИЛА РАБОТЫ СТУДЕНТОВ ЗА

КОМПЬЮТЕРОМ

ПЕРЕД ВЫПОЛНЕНИЕМ РАБОТЫ необходимо выслушать преподавателя о ее содержании и ходе выполнения.

ВО ВРЕМЯ РАБОТЫ

1. Компьютер можно включать только после разрешения преподавателя.

2. ЗАПРЕЩАЕТСЯ:

- прикасаться к проводам, лишенных изоляции;
- включать компьютер со снятым корпусом;
- производить подключение устройств к включенному компьютеру;
- прикасаться руками и острыми предметами к экрану монитора, внутренним частям компьютера;
- есть за компьютером;
- включать компьютер сразу же после его выключения (необходимо подождать 10-15 секунд).

3. Обнаружив неисправность в электрических устройствах, находящихся под напряжением, необходимо немедленно отключить источник электропитания и сообщить об этом преподавателю.

4. Не оставляйте рабочего места без разрешения преподавателя.

ПОСЛЕ ОКОНЧАНИЯ РАБОТЫ

1. Корректно завершите работу компьютера.

2. Сдай рабочее место преподавателю.

Инструктаж №3 ПРАВИЛА ПОЖАРНОЙ БЕЗОПАСНОСТИ

1. В кабинете должны быть средства для тушения пожара: огнетушитель.

2. Кабинет должен содержаться в чистоте. Весь сгораемый мусор следует систематически выносить.

3. В кабинете не допускается хранение легковоспламеняющихся жидкостей.

4. ЗАПРЕЩАЕТСЯ:

- допускать к работе студентов, не ознакомленных с правилами техники безопасности;
- оставлять без присмотра включенное в электрическую сеть оборудование;
- подвешивать электропроводку на гвоздях, использовать электропровода с поврежденной изоляцией, некалиброванные предохранители, обертывать электрические лампы бумагой или тканью, подвешивать стенды, таблицы и пр. на электропровода;
- работа в кабинете с нагревательными приборами;
- производить сложный ремонт компьютерной техники.

5. По окончании работы необходимо тщательно осмотреть помещение, устранить все недочеты, отключить напряжение электросети с

помощью рубильника.

6. В случае возникновения пожара необходимо:

- отключить напряжение электрической сети;
- немедленно эвакуировать студентов;
- принять меры по тушению пожара;
- сообщить о пожаре по телефону 01 или 112.

Критерии оценки выполнения практических работ

- наличие Цель выполняемой работы, выполнение более половины основных заданий (удовлетворительно);
- наличие Цель выполняемой работы, выполнение всех основных и более половины дополнительных заданий (хорошо);
- наличие Цель выполняемой работы, выполнение всех основных и индивидуальных заданий (отлично).

Перечень практических работ

МДК 06.01 Внедрение информационных систем

1. Применение мер по поддержанию документации в актуальном состоянии
2. Применение мер по поддержанию документации в защищенном состоянии
3. Анализ структурной схемы управления предприятия
4. Формирование предложения о расширении функциональности ИС
5. Формировать предложения о прекращении эксплуатации ИС
6. Модернизация информационной системы
7. Выявление и документирование проблем установки программного обеспечения
8. Функциональное тестирование ИС
9. Нефункциональное тестирование ИС
10. Приемочное тестирование ИС
11. Разработка (подготовка) документации и отчетных форм для внедрения программных средств

МДК 06.02 Инженерно - техническая поддержка сопровождения информационных систем

1. Тестирование программных продуктов
2. Соотнесение характеристик и атрибутов качества ИС
3. Разработка требований к аппаратной составляющей ИС
4. Подбор конфигурации сервера в соответствии с решаемыми задачами
5. Расчет стоимости технического оборудования для ИС
6. Настройка политики безопасности
7. Работа с программой восстановления файлов и очистки дисков
8. Тестирование защиты программного обеспечения

9. Составление плана резервного копирования
10. Определение интервала резервного копирования
11. Восстановление базы данных информационной системы
12. Техническое сопровождение базы данных информационной системы

МДК 06.03 Устройство и функционирование информационной системы

1. Применение технической документации при реализации информационных процессов в ИС
2. Составление ТЗ
3. Разработка инструкции пользователя
4. Разработка инструкции оператора
5. Построение прогнозов. Моделирование прогноза
6. Анализ рисков
7. Разработка мероприятий по снижению рисков при эксплуатации ИС
8. Анализ отечественных стандартов по эксплуатации ИС
9. Анализ международных стандартов по эксплуатации ИС

МДК 06.04 Интеллектуальные системы и технологии

1. Подготовка процедуры экспертного анализа
2. Проведение процедуры экспертного анализа ИС
3. Использование экспертных систем при решении практических задач
4. Модернизация экспертных систем
5. Внедрение экспертных систем

Практическая работа «Разработка и оформление предложений по расширению функциональности информационной системы»

Цель работы – описать и проанализировать информационную систему, распределить роли в группе разработчиков.

Теоретические положения

Проблемы управления программными проектами впервые проявились в 60-х – начале 70-х годов, когда провалились многие большие проекты по разработке программных продуктов. Были зафиксированы задержки в создании программного обеспечения (ПО), оно было ненадежным, затраты на разработку в несколько раз превосходили первоначальные оценки, созданные программные системы часто имели низкие показатели производительности. Причины провалов коренились в тех подходах, которые использовались в управлении проектами. Применяемая методика была основана на опыте управления техническими проектами и оказалась неэффективной при разработке программного обеспечения.

Важно понимать разницу между профессиональной разработкой ПО и любительским программированием. Необходимость управления программными проектами вытекает из того факта, что процесс создания профессионального ПО всегда является субъектом бюджетной политики организации, где оно разрабатывается, и имеет временные ограничения. Работа руководителя программного проекта по большому счету заключается в том, чтобы гарантировать выполнение этих бюджетных и временных ограничений с учетом бизнес-целей организации относительно разрабатываемого ПО.

Руководители проектов призваны спланировать все этапы разработки программного продукта. Они также должны контролировать ход выполнения работ и соблюдения всех требуемых стандартов. Постоянный контроль за ходом выполнения работ необходим для того, чтобы процесс разработки не выходил за временные и бюджетные ограничения. Хорошее управление не гарантирует успешного завершения проекта, но плохое управление обязательно приведет к его провалу. Это может выразиться в задержке сроков сдачи готового ПО, в превышении сметной стоимости проекта и в несоответствии готового ПО спецификации требований.

Процесс разработки ПО существенно отличается от процессов реализации технических проектов, что порождает определенные сложности в управлении программными проектами:

Программный продукт нематериален. Программное обеспечение нематериально, его нельзя увидеть или потрогать. Руководитель программного проекта не видит процесс «роста» разрабатываемого ПО. Он может полагаться только на документацию, которая фиксирует процесс разработки программного продукта.

Не существует стандартных процессов разработки ПО. На сегодняшний день не существует четкой зависимости между процессом создания ПО и типом создаваемого программного продукта. Другие технические дисциплины имеют длительную историю, процессы разработки технических изделий многократно опробованы и проверены. Процессы создания большинства технических систем хорошо изучены. Изучением же процессов создания ПО специалисты занимаются только последнее время. Поэтому пока нельзя точно предсказать,

на каком этапе процесса разработки ПО могут возникнуть проблемы, угрожающие всему программному проекту.

Большие программные проекты – это часто «одноразовые» проекты. Большие программные проекты, как правило, значительно отличаются от проектов, реализованных ранее. Поэтому, чтобы уменьшить неопределенность в планировании проекта, руководители проектов должны обладать очень большим практическим опытом. Но постоянные технологические изменения в компьютерной технике и коммуникационном оборудовании обесценивают предыдущий опыт. Знания и навыки, накопленные опытом, могут не востребоваться в новом проекте.

Перечисленные отличия могут привести к тому, что реализация проекта выйдет из временного графика или превысит бюджетные ассигнования. Программные системы зачастую оказываются новинками, как в «идеологическом», так и в техническом плане. Поэтому, предвидя возможные проблемы в реализации программного проекта, следует всегда помнить, что многим из них свойственно выходить за рамки временных и бюджетных ограничений.

Процесс управления разработкой программного обеспечения

Невозможно описать и стандартизировать все работы, выполняемые в проекте по созданию ПО. Эти работы весьма существенно зависят от организации, где выполняется разработка ПО, от типа создаваемого программного продукта. Но всегда можно выделить следующие:

1. Написание предложений по созданию ПО.
2. Планирование и составление графика работ по созданию ПО.
3. Оценивание стоимости проекта.
4. Подбор персонала.
5. Контроль за ходом выполнения работ.
6. Написание отчетов и представлений.

Первая стадия программного проекта может состоять из написания предложений по реализации этого проекта. Предложения должны содержать описание целей проектов и способов их достижения. Они также обычно включают в себя оценки финансовых и временных затрат на выполнение проекта. При необходимости здесь могут приводиться обоснования для передачи проекта на выполнение сторонней организации или команде разработчиков.

Написание предложений – очень ответственная работа, так как для многих организаций вопрос о том, будет ли проект выполняться самой организацией или разрабатываться по контракту сторонней компанией, является критическим. Не существует каких-либо рекомендаций по написанию предложений, многое здесь зависит от опыта.

На этапе планирования проекта определяются процессы, этапы и полученные на каждом из них результаты, которые должны привести к выполнению проекта. Реализация этого плана приведет к достижению целей проекта. Определение стоимости проекта напрямую связано с его планированием, поскольку здесь оцениваются ресурсы, требующиеся для выполнения плана.

Контроль за ходом выполнения работ (мониторинг проекта) – это непрерывный процесс, продолжающийся в течение всего срока реализации проекта. Руководитель должен постоянно отслеживать ход реализации проекта и сравнивать фактические и плановые показатели выполнения работ с их стоимостью. Хотя многие организации имеют механизмы формального мониторинга работ, опытный руководитель может составить ясную картину о стадии развития проекта просто путем неформального общения с разработчиками.

Неформальный мониторинг часто помогает обнаружить потенциальные проблемы, которые в явном виде могут обнаружиться позднее. Например, ежедневное обсуждение хода выполнения работ может выявить отдельные недоработки в создаваемом программном продукте. Вместо ожидания отчетов, в которых будет отражен факт «пробуксовки» графика работ, можно обсудить со специалистами намечающиеся программистские проблемы и не допустить срыва графика работ.

В течение реализации проекта обычно происходит несколько формальных контрольных проверок хода выполнения работ по созданию ПО. Такие проверки должны дать общую картину хода реализации проекта в целом и показать, насколько уже разработанная часть ПО соответствует целям проекта.

Время выполнения больших программных проектов может занимать несколько лет. В течение этого времени цели и намерения организации, заказавшей программный проект, могут существенно измениться. Может оказаться, что разрабатываемый программный продукт стал уже ненужным либо исходные требования к создаваемому ПО, просто устарели и их необходимо кардинально менять. В такой ситуации руководство организации-разработчика может принять решение о прекращении разработки ПО или об изменении проекта в целом с тем, чтобы учесть изменившиеся цели и намерения организации-заказчика.

Руководители проектов обычно обязаны сами подбирать исполнителей для своих проектов. В идеальном случае профессиональный уровень исполнителей должен соответствовать той работе, которую они будут выполнять в ходе реализации проекта. Однако во многих случаях руководители должны полагаться на команду разработчиков, которая далека от идеальной. Такая ситуация может быть вызвана следующими причинами:

Бюджет проекта не позволяет привлечь высококвалифицированный персонал. В таком случае за меньшую плату привлекаются менее квалифицированные специалисты.

Бывают ситуации, когда невозможно найти специалистов необходимой квалификации, как в самой организации-разработчике, так и вне ее. Например, в организации «лучшие люди» могут быть уже заняты в других проектах.

Организация хочет повысить профессиональный уровень своих работников. В этом случае она может привлечь к участию в проекте неопытных или недостаточно квалифицированных работников, чтобы они приобрели необходимый опыт и поучились у более опытных специалистов.

Таким образом, почти всегда подбор специалистов для выполнения проекта имеет определенные ограничения и не является свободным. Вместе с тем необходимо, чтобы хотя бы несколько членов группы разработчиков имели

квалификацию и опыт, достаточные для работы над данным проектом. В противном случае невозможно избежать ошибок в разработке ПО.

Руководитель проекта обычно обязан посылать отчеты о ходе его выполнения, как заказчику, так и подрядным организациям. Это должны быть краткие документы, основанные на информации, извлекаемой из подробных отчетов о проекте. В этих отчетах должна быть та информация, которая позволяет четко оценить степень готовности создаваемого программного продукта.

Выделим следующие роли в группе по разработке ПО:

1. Руководитель – общее руководство проектом, написание документации, общение с заказчиком ПО.
2. Системный аналитик – разработка требований (составление технического задания, проекта программного обеспечения).
3. Тестер – составление плана тестирования и аттестации готового ПО (продукта), составление сценария тестирования, базовый пример, проведение мероприятий по плану тестирования.
4. Разработчик – моделирование компонент программного обеспечения, кодирование.

Планирование проекта разработки программного обеспечения

Эффективное управление программным проектом напрямую зависит от правильного планирования работ, необходимых для его выполнения. План помогает руководителю предвидеть проблемы, которые могут возникнуть на каких-либо этапах создания ПО, и разработать превентивные меры для их предупреждения или решения. План, разработанный на начальном этапе проекта, рассматривается всеми его участниками как руководящий документ, выполнение которого должно привести к успешному завершению проекта. Этот первоначальный план должен максимально подробно описывать все этапы реализации проекта.

Процесс планирования начинается, исходя из описания системы, с определения проектных ограничений (временные ограничения, возможности наличного персонала, бюджетные ограничения и т. д.). Эти ограничения должны определяться параллельно с оцениванием проектных параметров, таких как структура и размер проекта, а также распределением функций среди исполнителей. Затем определяются этапы разработки и то, какие результаты документация, прототипы, подсистемы или версии программного продукта) должны быть получены по окончании этих этапов. Далее начинается циклическая часть планирования. Сначала разрабатывается график работ по выполнению проекта или дается разрешение на продолжение использования ранее созданного графика. После этого проводится контроль выполнения работ и отмечаются расхождения между реальным и плановым ходом работ.

Далее, по мере поступления новой информации о ходе выполнения проекта, возможен пересмотр первоначальных оценок параметров проекта. Это, в свою очередь, может привести к изменению графика работ. Если в результате этих изменений нарушаются сроки завершения проекта, должны быть пересмотрены (и согласованы с заказчиком ПО) проектные ограничения.

Конечно, большинство руководителей проектов не думают, что реализация их проектов пройдет гладко, без всяких проблем. Желательно описать возможные проблемы еще до того, как они проявят себя в ходе

выполнения проекта. Поэтому лучше составлять «пессимистические» графики работ, чем «оптимистические». Но, конечно, невозможно построить план, учитывающий все, в том числе случайные, проблемы и задержки выполнения проекта, поэтому и возникает необходимость периодического пересмотра проектных ограничений и этапов создания программного продукта.

План проекта должен четко показать ресурсы, необходимые для реализации проекта, разделение работ на этапы и временной график выполнения этих этапов. В некоторых организациях план проекта составляется как единый документ, содержащий все виды планов, описанных выше. В других случаях план проекта описывает только технологический процесс создания ПО. В таком плане обязательно присутствуют ссылки на планы других видов, но они разрабатываются отдельно от плана проекта.

Детализация планов проектов очень разнится в зависимости от типа разрабатываемого программного продукта и организации разработчика. Но в любом случае большинство планов содержат следующие разделы.

1. Введение. Краткое описание целей проекта и проектных ограничений (бюджетных, временных и т. д.), которые важны для управления проектом.

2. Организация выполнения проекта. Описание способа подбора команды разработчиков и распределение обязанностей между членами команды.

3. Анализ рисков. Описание возможных проектных рисков, вероятности их проявления и стратегий, направленных на их уменьшение.

4. Аппаратные и программные ресурсы, необходимые для реализации проекта. Перечень аппаратных средств и программного обеспечения, необходимого для разработки программного продукта. Если аппаратные средства требуется закупать, приводится их стоимость совместно с графиком закупки и поставки.

5. Разбиение работ на этапы. Процесс реализации проекта разбивается на отдельные процессы, определяются этапы выполнения проекта, приводится описание результатов («выходов») каждого этапа и контрольные отметки.

6. График работ. В этом графике отображаются зависимости между отдельными процессами (этапами) разработки ПО, оценки времени их выполнения и распределение членов команды разработчиков по отдельным этапам.

7. Механизмы мониторинга и контроля за ходом выполнения проекта. Описываются предоставляемые руководителем отчеты о ходе выполнения работ, сроки их предоставления, а также механизмы мониторинга всего проекта.

План должен регулярно пересматриваться в процессе реализации проекта. Одни части плана, например, график работ, изменяются часто, другие более стабильны. Для внесения изменений в план требуется специальная организация документопотока, позволяющая отслеживать эти изменения.

Общие сведения о требованиях к информационным системам

Проблемы, которые приходится решать специалистам в процессе создания программного обеспечения, очень сложны. Природа этих проблем не всегда ясна, особенно если разрабатываемая программная система инновационная. В частности, трудно чётко описать те действия, которые должна выполнять система. Описание функциональных возможностей и

ограничений, накладываемых на систему, называется требованиями к этой системе, а сам процесс формирования, анализа, документирования и проверки этих функциональных возможностей и ограничений – разработкой требований.

Требования подразделяются на пользовательские и системные. Пользовательские требования – это описание на естественном языке (плюс поясняющие диаграммы) функций, выполняемых системой, и ограничений, накладываемых на неё. Системные требования – это описание особенностей системы (архитектура системы, требования к параметрам оборудования и т. д.), необходимых для эффективной реализации требований пользователя.

Первые шаги по разработке требований к информационным системам – анализ осуществимости.

Разработка требований – это процесс, включающий мероприятия, необходимые для создания и утверждения документа, содержащего спецификацию системных требований. Для новых программных систем процесс разработки требований должен начинаться с анализа осуществимости. Началом такого анализа является общее описание системы и ее назначения, а результатом анализа – отчет, в котором должна быть четкая рекомендация, продолжать или нет процесс разработки требований проектируемой системы. Другими словами, анализ осуществимости должен осветить следующие вопросы.

1. Отвечает ли система общим и бизнес-целям организации-заказчика и организации-разработчика?
2. Можно ли реализовать систему, используя существующие на данный момент технологии и не выходя за пределы заданной стоимости?
3. Можно ли объединить систему с другими системами, которые уже эксплуатируются?

Критическим является вопрос, будет ли система соответствовать целям организации. Если система не соответствует этим целям, она не представляет никакой ценности для организации. В то же время многие организации разрабатывают системы, не соответствующие их целям, либо не совсем ясно понимая эти цели, либо под влиянием политических или общественных факторов.

Выполнение анализа осуществимости включает сбор и анализ информации о будущей системе и написание соответствующего отчета. Сначала следует определить, какая именно информация необходима, чтобы ответить на поставленные выше вопросы. Например, эту информацию можно получить, ответив на следующее:

4. Что произойдет с организацией, если система не будет введена в эксплуатацию?
5. Какие текущие проблемы существуют в организации и как новая система поможет их решить?
6. Каким образом система будет способствовать целям бизнеса?
7. Требуется ли разработка системы технологии, которая до этого не использовалась в организации?

Далее необходимо определить источники информации. Это могут быть менеджеры отделов, где система будет использоваться, разработчики программного обеспечения, знакомые с типом будущей системы, технологи, конечные пользователи и т. д.

После обработки собранной информации готовится отчет по анализу осуществимости создания системы. В нем должны быть даны рекомендации относительно продолжения разработки системы. Могут быть предложены изменения бюджета и графика работ по созданию системы или предъявлены более высокие требования к системе.

Порядок выполнения работы

Данное практическое занятие предполагает выполнение следующих этапов:

- изучить методические указания;
- составить подробное описание информационной системы;
- провести анализ осуществимости;
- распределить роли в группе;
- заполнить разделы плана;
- ответить на контрольные вопросы.

Контрольные вопросы

1. Перечислите сложности при управлении программными проектами.
2. В чем заключается этап написания предложений?
3. Поясните этап мониторинг проекта.
4. Для чего используется неформальный мониторинг?
5. Поясните анализ осуществимости.
6. Что понимается под пользовательскими требованиями?
7. Что понимается под системными требованиями?

Практическое занятие № 9 «Создание плана резервного копирования»

Целью работы является изучение порядка составления плана резервного копирования.

Результатом практической работы является отчет, в котором должны быть отражены ход работ и подготовленный план резервного копирования.

Для выполнения практической работы № 9 студент должен изучить приведенный ниже теоретический материал. Отчет сдается в распечатанном и электронном (файл Word) видах.

План резервного копирования

План резервного копирования это комплекс мер и последовательность действий для создания актуальной копии защищаемых данных на резервном носителе. План может состоять из одного или нескольких заданий, включающих в себя следующие параметры:

- Объекты копирования: сервера, виртуальные машины, диски, тома, папки, файлы, приложения, базы данных.
- Способ копирования: полное, инкрементальное, дифференциальное.
- Расписание: дата, время, период или событие для запуска копирования данных.
- Хранилище: место для хранения резервной копии (облако, локальные или сетевые папки и диски, устройства хранения).

- Параметры хранения: жизненный цикл резервной копии в хранилище и действия, выполняемые после окончания срока хранения.
- Дополнительные параметры: уровень компрессии (сжатия), шифрование потока, проверка целостности резервной копии и др.

Разработка и реализация плана резервного копирования

При создании плана ответьте на следующие вопросы:

- **Насколько важны данные?** Этот критерий поможет решить, как, когда и какую информацию архивировать. Для критичной информации, например, баз данных, следует создавать избыточные архивные наборы, охватывающие несколько периодов архивации. Для менее важной информации, например, для текущих пользовательских файлов, сложный план архивации не нужен, достаточно регулярно сохранять их и уметь легко восстанавливать.

К какому типу относится архивируемая информация?

Тип информации поможет определить необходимость архивации данных: как и когда данные должны быть сохранены.

- **Как часто изменяются данные?** Частота изменения влияет на выбор частоты архивирования. Например, ежедневно меняющиеся данные необходимо сохранять каждый день.

Нужно ли дополнить архивацию созданием теневых копий?

При этом следует помнить, что теневая копия – это дополнение к архивации, но ни в коем случае не ее замена.

- **Как быстро нужно восстанавливать данные?** Время – важный фактор при создании плана архивации. В критичных к скорости системах нужно проводить восстановление очень быстро.

- **Какое оборудование оптимально для архивации и есть ли оно у вас?** Для своевременной архивации вам понадобится несколько архивирующих устройств и несколько наборов носителей. Аппаратные средства архивации включают ленточные накопители (это наименее дорогой, но и самый медленный тип носителя), оптические диски и съемные дисковые накопители.

- **Кто отвечает за выполнение плана архивации и восстановления данных?** В идеале и за разработку плана, и, собственно, за архивацию и восстановление должен отвечать один человек.

- **Какое время оптимально для архивации?** Архивация в период наименьшей загрузки системы пройдет быстрее, но не всегда возможно провести ее в удобные часы. Поэтому с особой тщательностью архивируйте ключевые данные.

- **Нужно ли сохранять архивы вне офиса?** хранение архивов вне офиса – важный фактор на случай стихийного бедствия. Вместе с архивами сохраните и копии ПО для установки или переустановки ОС.

Для построения правильной и эффективной системы резервного копирования необходимо детально изучить и задокументировать все файловые ресурсы, используемые в компании, а затем тщательно спланировать стратегию резервного копирования и реализовать на практике

Контрольные вопросы

1. Каковы причины резервирования данных?
2. Какие типы резервного копирования вы знаете? В чем их особенности?
3. Кто планирует, какие данные нужно резервировать?

4. Какие данные необходимо резервировать?
5. Из каких разделов состоит план резервного копирования?

Практическая работа «Разработка руководства оператора»

Цель работы – изучение структуры разделов руководства оператора по ГОСТ 19.505-79.

Теоретические положения

В 2004-2005 годах был опубликован минимально необходимый набор «учебно-тренировочных» документов на программы, включающий техническое задание на программу по ГОСТ 19.201-78, программу и методику испытаний по ГОСТ 19.301-79, руководство оператора по ГОСТ 19.505-79. Этого достаточно для разработки программы, проведения испытаний и сдачи ее заказчику.

Рассмотрим структуру разделов руководства оператора по ГОСТ 19.505-79.

Структуру и оформление документа устанавливают в соответствии с ГОСТ 19.105-78.

Составление информационной части (аннотации и содержания) является обязательным.

В аннотации целесообразно привести следующую фразу: «Настоящее руководство распространяется исключительно на программу и не заменяет учебную, справочную литературу, руководства от производителя ОС и прочие источники информации, освещающие работу с графическим пользовательским интерфейсом операционной системы». Допустимо создание подраздела «Назначение руководства» или «Рекомендации по освоению».

Руководство оператора должно содержать следующие разделы:

- назначение программы;
- условия выполнения программы;
- выполнение программы;
- сообщения оператору.

В зависимости от особенностей документы допускается объединять отдельные разделы или вводить новые.

Последняя фраза предоставляет разработчикам программной документации пространство для маневра.

В разделе «Назначение программы» должны быть указаны сведения о назначении программы и информация, достаточная для понимания функций программы и ее эксплуатации «...должны быть указаны сведения о назначении программы». Сведения о назначении программы изложены в основополагающем документе – в техническом задании.

Функциональным назначением программы является предоставление пользователю возможности работы с текстовыми документами в формате rtf.

Эксплуатационное назначение. Программа должна эксплуатироваться в профильных подразделениях на объектах заказчика.

Пользователями программы должны являться сотрудники профильных подразделений объектов заказчика.

Состав функций. Программа обеспечивает возможность выполнения

перечисленных ниже функций:

- функции создания нового (пустого) файла;
- функции открытия (загрузки) существующего файла;
- функции редактирования открытого (далее – текущего) файла путем ввода, замены, удаления содержимого файла с применением стандартных устройств ввода;
- функции редактирования текущего файла с применением буфера обмена операционной системы;
- функции сохранения файла с исходным именем;
- функции сохранения файла с именем, отличным от исходного;
- функции отправки содержимого текущего файла электронной почтой с помощью внешней клиентской почтовой программы;
- функции вывода оперативных справок в строковом формате (подсказок);
- функции интерактивной справочной системы;
- функции отображения названия программы, версии программы, копирайта и комментариев разработчика.

Условия выполнения программы. В разделе «Условия выполнения программы» должны быть указаны условия, необходимые для выполнения программы (минимальный и (или) максимальный состав аппаратурных и программных средств и т.п.).

Климатические условия эксплуатации, при которых должны обеспечиваться заданные характеристики, должны удовлетворять требованиям, предъявляемым к техническим средствам в части условий их эксплуатации.

Минимальный состав технических средств. В состав технических средств должен входить IBM-совместимый персональный компьютер (ПЭВМ), включающий в себя:

- процессор Pentium-1000 с тактовой частотой, ГГц – 10, не менее;
- материнскую плату с FSB, ГГц – 5, не менее;
- оперативную память объемом, Тб – 10, не менее;
- и т. д.

Минимальный состав программных средств. Системные программные средства, используемые программой, должны быть представлены лицензионной локализованной версией операционной системы. Допускается использование пакета обновления.

Требования к персоналу (пользователю). Минимальное количество персонала, требуемого для работы программы, должно составлять не менее 2 штатных единиц – системный администратор и пользователь программы – оператор.

Системный администратор должен иметь высшее профильное образование и сертификаты компании-производителя операционной системы. В перечень задач, выполняемых системным администратором, должны входить:

- задача поддержания работоспособности технических средств;
- задачи установки (инсталляции) и поддержания работоспособности системных программных средств – операционной системы;
- задача установки (инсталляции) программы. Пользователь программы (оператор) должен обладать практическими навыками работы с графическим пользовательским интерфейсом операционной системы.

Персонал должен быть аттестован на II квалификационную группу по электробезопасности (для работы с конторским оборудованием).

Выполнение программы. В разделе «Выполнение программы» должна быть указана последовательность действий оператора, обеспечивающих загрузку, запуск, выполнение и завершение программы, приведено описание функций, формата и возможных вариантов команд, с помощью которых оператор осуществляет загрузки и управляет выполнением программы, а также ответы программы на эти команды.

Сообщения оператору. В данном разделе должны быть приведены тексты сообщений, выдаваемых в ходе выполнения программы, описание их содержания и соответствующие действия оператора (действия оператора в случае сбоя, возможности повторного запуска программы и т. п.).

Завершение работы программы. Ключевая фраза подраздела «Требования к количеству и квалификации персонала» технического задания – «пользователь программы (оператор) должен обладать практическими навыками работы с графическим пользовательским интерфейсом операционной системы» снимает с автора обязанность подробно расписывать способы загрузки и запуска программы... Не обязан разработчик разжевывать оператору приемы работы с графическим пользовательским интерфейсом операционной системы. За исключением случаев применения в программе элементов интерфейса, не свойственных операционной системе.

Порядок выполнения работы

Данное практическое занятие предполагает выполнение следующих этапов:

- изучить методические указания;
- ответить на контрольные вопросы.

Контрольные вопросы

1. Какой стандарт устанавливает требования к содержанию и оформлению руководства оператора?
2. Перечислите разделы руководства оператора.
3. Что указывается в разделе «Назначение программы»?
4. Что указывается в разделе «Условия выполнения программы»?
5. Что указывается в разделе «Выполнение программы»?
6. Что указывается в разделе «Сообщения оператору»?

Практическое занятие «Использование экспертных систем при решении практических задач»

Цель работы – изучить основы алгебры логики и логического вывода.

Результатом практической работы является отчет. Отчет должен содержать формулы и их решения с помощью таблиц истинности и эквивалентных преобразований

Для выполнения практической работы студент должен изучить

приведенный ниже теоретический материал

Логические операции над логическими переменными

При изучении информатики (языков программирования) возникает понятие *логической* или *булевой переменной*, т. е. переменной, которая принимает одно из двух возможных значений true или false (истина или ложь, 1 или 0), называемых логическими константами.

Основные логические операции над логическими переменными:

- отрицание;
- конъюнкция (логическое «и»);
- дизъюнкция (логическое «или»);
- импликация («если...то»).

Логические операции называют также логическими (булевыми) функциями.

Знаки логических операций называют *логическими связками*, а выражения, которые получаются при использовании логических переменных и логических связок – *логическими* или *пропозициональными* (от proposition – высказывание) *формулами*.

Отрицание логической переменной A , обозначается $\neg A$, или подчеркиванием над буквой $\neg A$, или $\sim A$ (этот знак, правда, используется иногда для другой логической операции – эквивалентности): читается «*отрицание A* » или «*не A* » и означает противоположное A значение.

В случае, когда A принимает значение истина (И), $\neg A$ принимает значение ложь (Л), и если $A = \text{false}$, то $\neg A = \text{true}$.

То же самое выражают так называемой *таблицей истинности* (табл. 1).

Таблица 1. Таблица истинности для отрицания

A	$\neg A$
И	Л
Л	И

Двойное отрицание переменной равно самой переменной, что может быть выражено следующей формулой: $\neg \neg A = A$. Знак равенства означает то, что таблицы истинности для выражения слева и выражения справа от знака равенства совпадают.

Конъюнкция обозначается $A \& B$, или $A \sqcap B$, или $A B$, по аналогии с умножением чисел, читается как «*конъюнкция A и B* » или « *A и B* ». Конъюнкция двух логических переменных (табл. 2) истинна тогда

и только тогда, когда истинны обе переменные.

Таблица 2. Таблица истинности для конъюнкции

A	B	$A \& B$
Л	Л	Л
Л	И	Л
И	Л	Л

И	И	И
---	---	---

Дизъюнкция двух логических переменных обозначается $A \sqcup B$ или $A + B$ (читается «дизъюнкция A и B » или « A или B ») ложна тогда и только тогда, когда ложны обе переменные (табл. 3).

Таблица 3. Таблица истинности для дизъюнкции

A	B	$A \sqcup B$
Л	Л	Л
Л	И	И
И	Л	И
И	И	И

Импликация обозначается $A \rightarrow B$, читается «из A следует B », или « B следует из A », или «если A , то B », или « A достаточное условие (достаточно) для B », или « B необходимое условие (необходимо) для A ».

A называют *посылкой* (или *антецедентом*); B называют *заключением* (консеквентом).

$A \rightarrow B$ ложна в одном только случае, когда $A = \text{true}$, $B = \text{false}$, в остальных случаях она истинна (табл. 4).

Таблица 4. Таблица истинности для импликации

A	B	$A \rightarrow B$
Л	Л	И
Л	И	И
И	Л	Л
И	И	И

Относительно интерпретации логических операций нужно сказать следующее. Интерпретация первых трех логических операций не вызывает сомнений, т. е. они вполне соответствуют здравому смыслу. Правда, мы иногда используем *разделяющее или*: высказывание истинно только в тех двух случаях, когда одно высказывание истинно, а другое ложно. Причем, операцию импликации можно было бы не вводить, а пользоваться вместо нее операциями отрицания и дизъюнкции.

Таблица истинности для импликации определяется исходя из следующих соображений. То, что из истинного значения посылки следует истинное значение заключения – истинно, а из истинного значения посылки следует ложное значение заключения – ложно, не вызывает интуитивного протеста. Непонятной обычно кажется истинность того, что из ложной посылки следует ложное заключение и из ложной посылки – истинное заключение (первые две строчки таблицы истинности). Это можно оправдать тем соображением, что

нельзя делать неправильные выводы из правильных посылок.

Еще одна логическая операция – эквивалентность: $A \equiv B$ (иногда для этого используют знак $\sim$): A тождественно (или эквивалентно) B принимает значение истина тогда и только тогда, когда значения истинности A и B совпадают. Значение ложь, когда они различные.

Алгебра логики

Под *алгеброй* понимают множество с заданными на нем операциями. Например, в качестве множества можно взять множество вещественных чисел, а в качестве операций – арифметические операции.

Пусть $B = \{0;1\}$ – двухэлементное множество, и L – множество всех возможных операций на этом множестве, алгебра $A = \langle B, L \rangle$ называется *алгеброй логики*. Эти операции называют булевыми операциями или функциями алгебры логики.

Элементы двухэлементного множества можно обозначать по-разному: 0, 1 или ложь, истина или true, false. Операции, которые были рассмотрены в предыдущем пункте, принадлежат множеству L .

Любая функция алгебры логики может быть выражена с помощью отрицания, конъюнкции и дизъюнкции. Проверим истинность этого утверждения. Рассмотрим в качестве примера функцию (операцию) «*исключающее или*». Обозначим ее знаком $\oplus$. Таблица истинности для этой функции представлена в табл. 5.

Таблица 5. Таблица истинности для «исключающего или»

A	B	$A \oplus B$
Л	Л	Л
Л	И	И
И	Л	И
И	И	Л

Формула выражающая «*разделяющее или*» через дизъюнкцию конъюнкций, причем каждая конъюнкция состоит из переменных и отрицаний логических переменных A и B выглядит так:

$$A \oplus B = (\neg A \& B) \vee (A \& \neg B)$$

Причем формула $((\neg A \oplus \neg B) \& (A \oplus B)) \oplus ((\neg A \& B) \vee (A \& \neg B))$ будет истинна при любых значениях A и B , т. е. будет тавтологией. Справедливость равенства может быть проверена с помощью построения таблиц истинности.

В дальнейшем нам потребуются некоторые логические эквивалентности (функции алгебры логики, стоящие справа и слева от знака эквивалентности, имеют одинаковые таблицы истинности).

$A \rightarrow B \equiv \neg A \vee B$	(1)
$(A \& B) \vee C \equiv (A \vee C) \& (B \vee C)$	(2)
$(A \vee B) \& C \equiv (A \& C) \vee (B \& C)$	(3)
$\neg(A \& B) \equiv \neg A \vee \neg B$	(4)
$\neg(A \vee B) \equiv \neg A \& \neg B$	(5)
$A \& \neg A \equiv \text{false}$	(6)
$A \vee \neg A \equiv \text{true}$	(7)

Пропозициональные формулы

Под высказыванием (proposition) понимается утверждение, относительно которого можно сказать истинно оно или ложно, при этом оно не может быть и истинным, и ложным одновременно. Из простых высказываний с помощью логических операций, рассмотренных выше, строятся более сложные высказывания.

В исчислении высказываний рассмотренные выше символы логических (булевских) операций (функций) $\neg$, $\&$, $\vee$, $\rightarrow$, $\equiv$ называют *пропозициональными или логическими связками*.

Логические (булевские) переменные, обозначаемые буквами латинского алфавита (иногда с индексами), называют *пропозициональными переменными* (пропозициональными буквами).

Алфавит языка исчисления высказываний состоит из пропозициональных переменных (букв), логических связок и скобок: (,).

Правильно построенные выражения языка исчисления высказываний называются *пропозициональными формулами* или просто формулами.

Все пропозициональные переменные являются формулами. *Правила построения языка* исчисления высказываний задают его *синтаксис*.

Так как буквы, входящие в формулу, сами могут быть формулами, то каждую формулу можно рассматривать как *схему* бесконечного множества формул, получаемых заменой пропозициональных букв на соответствующие им формулы. Для того чтобы указать это, используют вместо термина пропозициональная формула термин *пропозициональная форма*.

Скобки (определяющие структуру формулы) могут опускаться в соответствии со следующим приоритетом логических связок: $\neg$, $\&$, $\vee$, $\rightarrow$, $\equiv$. Любое вхождение знака $\neg$ относится к наименьшей формуле (пропозициональной форме), следующей за ним; после расстановки всех скобок для формул, содержащих знак $\neg$, каждое вхождение знака

$\&$ связывает наименьшие формулы, стоящие рядом с этим знаком; затем то же самое относится к знаку $\vee$, далее к $\rightarrow$ — импликации и потом к знаку $\equiv$ — эквивалентности. В случае равного приоритета скобки расставляются слева направо.

Так, например, пропозициональная формула $A \vee B \rightarrow C \vee D \& E$ должна пониматься как $((A \vee B) \rightarrow (C \vee (D \& E)))$, поскольку сначала выполняется (применяется) операция $\&$, которая «старше», чем операция $\vee$, затем левая дизъюнкция, т. е. направление чтения формул слева направо, потом правая

дизъюнкция и далее импликация.

Приписывание значений истинности (true или false) пропозициональным переменным называется *интерпретацией* этих переменных. Под *интерпретацией формулы* понимается приписывание значений истинности переменным, входящим в эту формулу.

Пропозициональная формула, истинная при любых значениях истинности входящих в нее пропозициональных переменных, т. е. истинная при любых интерпретациях, называется *тождественно истинной*, или *общезначащей пропозициональной формулой*, или *тавтологией*.

Пропозициональная формула, ложная при любых значениях истинности входящих в нее пропозициональных переменных, называется *тождественно ложной*, или *невыполнимой пропозициональной формулой*, или *противоречием*.

Пропозициональная формула, которая может принимать истинные значения при некоторых интерпретациях, называется *выполнимой*. Интерпретации, для которых пропозициональная формула принимает истинное значение, называются ее *моделями*.

Логическое следствие и логический вывод

Выявление того факта, что из множества высказываний (формул исчисления) логически следует некоторое другое высказывание (формула) и является, по существу, одной из основных задач исчисления.

U логически влечет W (W логически следует из U) в исчислении высказываний, что обозначим как $U \sqsubset W$, если пропозициональная форма $(U \rightarrow W)$ является тавтологией.

Говорят, что множество формул $\{F_1, F_2, \dots, F_n\}$ логически влечет F в исчислении высказываний, что будем записывать как $\{F_1, F_2, \dots, F_n\} \sqsubset F$, если формула $\{F_1 \sqsubset F_2 \sqsubset \dots \sqsubset F_n\} \rightarrow F$ является тавтологией.

Для обозначения логического следования здесь используется знак $\sqsubset$. Надо отметить, что в литературе иногда для обозначения логического следования используется знак $\models$, а знак $\sqsubset$ используется для обозначения импликации.

U и W логически эквивалентны, что будем записывать в виде $U \sqsubset W$, если пропозициональная форма $(U \sqsubset W)$ является тавтологией. Знак $\sqsubset$ также используется для указания эквивалентных преобразований формул.

Вместо каждой буквы пропозициональной формы может быть подставлено любое высказывание естественного языка, причем одни и те же буквы должны заменяться одними и теми же высказываниями. Если сделать такую подстановку в тавтологии, то независимо от высказываний естественного языка мы получим логически истинное высказывание. Если пропозициональная формула ложна при любой интерпретации (является противоречием), то при такой подстановке будет получаться высказывание, являющееся логически ложным.

Истинность или ложность получаемых высказываний не будет зависеть от высказываний-подстановок, т. к. истинность или ложность результирующего высказывания зависит только от логической структуры пропозициональной формы, в которую они подставлялись.

В качестве примера рассмотрим высказывание: если верно, что *после снегопада крыша становится белой*, то справедливо также и следующее утверждение: *если крыша не белая, то снегопада не было*.

Обозначим буквой S утверждение «*снегопад был*», буквой B – «*крыша белая*». Тогда утверждение о том, что «*после снегопада крыша становится белой*», запишется в виде формулы $S \rightarrow B$. Утверждение

«*если крыша не белая, то снегопада не было*» запишется так: $\neg B \rightarrow \neg S$.

То, что из первого утверждения следует второе (т. е. всегда, когда истинно первое утверждение, истинным является и второе), записывается как логическое следование: $(S \rightarrow B) \vdash (\neg B \rightarrow \neg S)$, что эквивалентно тому, что формула $(S \rightarrow B) \rightarrow (\neg B \rightarrow \neg S)$ является тавтологией.

Докажем, что эта формула тождественно истина (тавтология). Это можно сделать с помощью таблицы истинности или с помощью эквивалентных преобразований (1)–(7):

$$\begin{aligned}(S \rightarrow B) \rightarrow (\neg B \rightarrow \neg S) &\vdash && \text{по формуле (1)} \\ \neg (\neg S \wedge B) &\vdash (\neg \neg B \wedge \neg \neg S) &\vdash & \text{по формуле (5)} \\ (S \wedge \neg B) &\vdash (B \wedge \neg S) &\vdash & \text{по формуле (2)} \\ (S \wedge B \wedge \neg S) \wedge (\neg B \wedge B \wedge \neg S) &\vdash && \text{по формуле (7)} \\ \text{true} \wedge \text{true} &\vdash && \\ \text{true} &&& \end{aligned}$$

Метод резолюций

Другое рассуждение (дающее ключ к пониманию метода логического вывода, называемого методом резолюций) заключается в следующем.

Чтобы доказать, что $\{F_1, F_2, \dots, F_n\} \vdash F$, необходимо доказать, что формула $(F_1 \wedge F_2 \wedge \dots \wedge F_n) \rightarrow F$ тавтология. Для того чтобы доказать, что данная формула является тавтологией, для этого достаточно доказать, что ее отрицание является противоречием, т. е. что формула $\neg(F_1 \wedge F_2 \wedge \dots \wedge F_n \rightarrow F)$ – противоречие. То есть противоречием должна являться $\neg(\neg(F_1 \wedge F_2 \wedge \dots \wedge F_n) \wedge F)$ или формула $(F_1 \wedge F_2 \wedge \dots \wedge F_n \wedge \neg F)$ т. е. противоречивым является множество формул $\neg$, состоящее из множества посылок $F_1, F_2, \dots, F_n$ и отрицания заключения $\neg F$.

Первым шагом для применения метода резолюций служит приведение пропозициональной формулы к логически эквивалентной ей конъюнктивной нормальной форме.

Конъюнктивной нормальной формой (КНФ) называется конъюнкция (логическое «и») конечного числа дизъюнктов. Любая пропозициональная формула имеет логически эквивалентную ей КНФ.

Дизъюнктом (в англоязычной литературе называют *clause*, что означает предложение в сложносочиненном предложении) называется дизъюнкция конечного числа *литер*, т. е. пропозициональных символов или их отрицаний:

$$D = (p_1 \vee p_2 \vee \dots \vee p_n).$$

Дизъюнкты, содержащие противоположные литеры ($A \vee \neg A$) являются тавтологиями и их можно отбросить, т. к. они не влияют на значение формы. Если в пределах одного и того же дизъюнкта встречается несколько раз одна и та же литера (например, $A \vee A$), то достаточно записать ее лишь один раз. В результате таких упрощений будет получена *приведенная* или *совершенная* КНФ.

Пустым дизъюнктом называется константа false. Если КНФ не содержит ни одного дизъюнкта (иначе *пуста*), то она эквивалентна true.

КНФ общезначима (является тавтологией), если все ее дизъюнкты общезначимы.

Дизъюнкт невыполним в том и только том случае, если он пустой. Понятие

дизъюнкта важно в логическом программировании и, в частности, в языке Пролог, т. к. описание задачи осуществляется в терминах дизъюнктов, часто называемых (в переводах на русский) *клаузами* или *клозами*.

Исчисление высказываний как формальная теория

Рассмотренные нами методы логического вывода, доказательства истинности или ложности формул являются семантическими, т. к. используют интерпретацию формул: придание значений истинности пропозициональным переменным. Один из этих способов – построение таблицы истинности. Другой способ – доказательство того, что данная формула логически (на основе понятия логического следования) следует из некоторого множества формул.

Если пропозициональная формула логически следует из пустого множества формул, то она является тождественно истинной или тавтологией.

Еще один способ, рассмотренный нами – метод резолюций. Другой способ доказательства, формально-логический или синтаксический, не требует для своего определения понятия интерпретации формул, он основан на построении формальной теории (формальной или дедуктивной системы).

Именно построение такой формальной системы или теории и может являться способом моделирования наших рассуждений во многих практически важных случаях.

Формальная система представляет собой набор начальных состояний или аксиом, которые могут быть описаны на некотором формальном языке, и правил вывода, позволяющих на основе аксиом и их следствий, получаемых с помощью правил вывода, строить новые следствия, называемые теоремами формальной теории. Процесс построения теорем называется формальным выводом или доказательством в формальной теории. Из школьного курса геометрии вспомним, что представляют собой аксиомы и теоремы формальной теории.

Варианты для выполнения практической работы

Решить заданные пропозициональные формы при условии, что:

$A = \text{False}; B = \text{True}$

1. $A \& \neg B \vee \neg A \& \neg B \vee A$
2. $\neg A \& \neg B \vee (A \vee \neg B) \vee A$
3. $A \& \neg B \vee A \vee \neg B \& A$
4. $A \& \neg B \vee (A \vee \neg B \vee A)$
5. $A \& \neg B \vee A \& \neg B \vee A$
6. $A \& \neg B \vee A \vee \neg B \& A$
7. $A \vee \neg B \& A \& \neg B \vee A$
8. $A \vee \neg B \& A \vee \neg B \vee A$
9. $A \vee \neg B \& A \vee \neg B \& A$
10. $A \vee \neg B \& A \vee \neg B \vee A$
11. $A \vee \neg B \vee A \& \neg B \vee A$
12. $A \vee \neg B \vee A \vee \neg B \& A$
13. $A \vee \neg B \& A \& \neg B \vee A$
14. $A \vee \neg B \& A \vee \neg B \& A$

Доказать или опровергнуть заданные равенства.

1. $A \& \neg B \vee A = A \vee B \& \neg A$

2. $A \vee B \vee \neg A = (\neg A \wedge \neg B) \vee A$
3. $A \vee \neg B \vee A = A \wedge B \wedge \neg A$
4. $\neg B \vee A \wedge B = \neg A \vee B \vee A$
5. $\neg B \vee A \wedge B \vee A = \neg A \vee \neg B \vee A$
6. $A \vee \neg B \wedge A = A \wedge \neg B \vee \neg A$
7. $A \wedge \neg B \wedge A = A \vee \neg B \vee \neg A$
8. $\neg A \wedge B \vee A = A \vee \neg B \wedge \neg A$
9. $\neg A \vee B \vee A = A \wedge \neg B \wedge \neg A$
10. $\neg A \vee B \wedge A = \neg A \vee B \wedge A$
11. $\neg A \wedge B \wedge A = \neg A \wedge \neg B \vee \neg A$
12. $A \wedge B \vee \neg A = \neg A \vee \neg B \vee \neg A$
13. $A \vee B \vee \neg A = \neg A \wedge \neg B \wedge \neg A$
14. $A \wedge \neg B \vee A = A \vee B \wedge \neg A$

Контрольные вопросы

1. Что такое алгебра логики?
2. Какая формула является пропозициональной?
3. Какая формула является пропозициональной?
4. В чем суть метода резолюций?

Перечень учебных изданий, Интернет-ресурсов, дополнительной литературы

Основные источники:

1. Чистов Д.В., Мельников П.П., Золотарюк А.В. Проектирование информационных систем: учебник и практикум для среднего профессионального образования. – М.: – Издательство Юрайт, 2022. – 258 с.
2. Федорова Г.И. Разработка, внедрение и адаптация программного обеспечения отраслевой направленности. Учебное пособие. – М.: Изд.: КУРС, Инфра-М. Среднее профессиональное образование. 2018. – 336 стр.
3. Рудаков, А.В. Технология разработки программных продуктов: практикум Учебное пособие для ССУЗов / А.В. Рудаков. - М.: Академия, 2014 [электронный ресурс]