МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное бюджетное образовательное
учреждение высшего образования
«ДАГЕСТАНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»
Факультет информатики и информационных технологий

МЕТОДИЧЕСКИЕ УКАЗАНИЯ

по организации и проведению учебной (ознакомительной) практики

для студентов направления подготовки
09.03.03 Прикладная информатика

Профиль подготовки:
Информационные системы и программирование

Махачкала 2026

Содержание

I. Общие положения 3

1.1. Цель учебной ознакомительной практики 3

1.2. Место проведения практики 4

1.3. Руководство практикой 4

1.4. Права и обязанности студентов в период практики 5

1.6. Требования к оформлению отчета по учебной ознакомительной практике 6

1.7. Подготовка презентационного материала 10

1.8. Аттестация по итогам практики 11

II. Примерный перечень заданий по учебной практике 12

2.2. Индивидуальные задания по учебной практике 27

III. Примерный перечень контрольных вопросов при приеме материалов учебной практики 38

I. Общие положения

Учебная практика является обязательным компонентом учебного процесса в высших учебных заведениях.

Учебные практики проводятся в соответствии с программами для студентов очной формы обучения по направлению подготовки 09.03.03 Прикладная информатика , профиль – «Информационные системы и программирование».

Методические указания содержат требования к организации и порядку проведения учебной практики студентов, её основное содержание.

В методических указаниях сформулированы права и обязанности практикантов, рассмотрен перечень вопросов для изучения студентами, а также требования к формированию и защите отчета по практике.

Методические указания предназначены для студентов указанного направления подготовки, профессорско-преподавательского состава выпускающих кафедр, руководителей практики от кафедры.

1.1. Цель учебной ознакомительной практики

Целями учебной (ознакомительной) практики по направлению подготовки 09.03.03 «Прикладная информатика», направленность (профиль) «Информационные системы и программирование» являются закрепление и углубление теоретической подготовки обучающегося и приобретение ими практических навыков и компетенций в сфере профессиональной деятельности.

Кроме того, целями учебной ознакомительной практики по направлению подготовки 09.03.03 Прикладная информатика являются также обеспечение непрерывности и последовательности овладения студентами умениями и навыками профессиональной деятельности, закрепление и углубление универсальных и общепрофессиональных компетенций в сфере программной инженерии, сформированных у студентов в процессе обучения и в результате самостоятельной работы по исследованию предметной области, исследованию информационно-вычислительной системы.

Основные задачи учебной практики:

Задачами учебной ознакомительной практики по направлению подготовки 09.03.03 Прикладная информатика, профиль - «Информационные системы и программирование»:

– изучение обучающимися опыта создания и применения информационных технологий в структурных подразделениях вуза;

– изучение обучающимися опыта применения технологий разработки программного обеспечения в структурных подразделениях вуза;

– углубление знаний по дисциплинам, полученным за время обучения на курсе, таких как «Web-программирование и разработка сайтов», «Реинжиниринг и управление информационными процессами», «Инженерия знаний»;

– формирование навыков подготовки и систематизации необходимых материалов и научно-технической информации для

выполнения задания;

– создание условий для практического применения знаний в области общепрофессиональных, специализированных компьютерных и математических дисциплин;

– формирование и совершенствование базовых профессиональных навыков и умений в области применения современных информационных технологий;

– формирование информационной компетентности с целью успешной работы в профессиональной сфере деятельности;

– анализ результатов проведенного исследования, подготовка и составление отчетов;

– приобретение навыков создания отчетов, в том числе и научно-технических, обеспечение успеха дальнейшей профессиональной карьеры.

1.2. Место проведения практики

Учебная (ознакомительная) практика проводится на 3 курсе во 2 семестре, срок проведения – 4 недели.

Местом проведения практики является факультет информатики и информационных технологий, кафедра информационных систем и технологий программирования, а также иные подразделения университета.

План-график прохождения практики разрабатывается руководителями от университета на основе баланса времени и с учетом особенностей базы практики и ее вида.

При наличии обучающихся по данной образовательной программе с ограниченными возможностями здоровья и инвалидов выбор мест прохождения практики осуществляется с учетом состояния здоровья и требований их доступности для данной категории.

1.3. Руководство практикой

Руководство практикой осуществляют факультет информатики и информационных технологий и кафедра информационных систем и технологий программирования (ИСиТП). Кафедра отвечает за выполнение программы и качество проведения учебной практики.

Руководитель практики от кафедры:

– осуществляет методическое руководство учебной (ознакомительной) практикой студентов;

– обеспечивает выполнение подготовительной и текущей работы по организации практики;

– разрабатывает программу учебной практики, индивидуальные задания студентам с указанием основной и дополнительной литературы;

– готовит и проводит совместно с деканатом организационные собрания студентов перед началом практики;

– при необходимости проводит консультации студентов в период учебной практики и перед защитой отчетов;

– совместно с факультетом в назначенный срок после окончания практики организует и принимает защиту отчетов по практике;

– представляет на кафедру итоги учебной практики;

– подводит итоги практики студентов кафедры;

– организует на кафедре хранение отчетов студентов по практике в течение установленного срока.

1.4. Права и обязанности студентов в период практики

С целью наилучшей подготовки к практике студент обязан ознакомиться с программой и содержанием предстоящих работ, собрать и изучить рекомендованную литературу.

В процессе прохождения практики студенты должны выполнять задания, предусмотренные планом, правила внутреннего распорядка организации, следовать указаниям руководителей практики, собрать материал для отчёта и оформить его. Порядок сбора и обработки материалов согласовывается с руководителем практики.

Перед завершением практики студент должен обсудить результаты своей работы над отчётом с руководителей практики.

Студент при прохождении практики обязан:

- в случае болезни обязательно поставить в известность руководителя практики и предоставить медицинскую справку;

- в течении последней недели практики представить отчёт руководителю практики для проверки и отзыва;

- в последние два дня практики представить руководителю практики от Университета отчёт и после проверки защитить его.

В случае затруднений с выполнением задания практики, связанных с характером работы, студент должен сообщить об этом руководителю практики от Университета.

1.5. Структура и содержание учебной практики»

При формировании отчёта по практике следует соблюдать определённую последовательность расположения материала:

– Титульный лист (см. Приложение 1)

– Содержание (перечень разделов и подразделов отчёта).

– Введение (цель учебной практики, особенности ее прохождения).

– Перечень, особенности и описание практических задач, выполненных студентом во время практики (пункты 2.1.1. и 2.1.2.)

– Выполнение индивидуальной работы (написание реферата).

– Заключение (итоги практики, выводы и предложения студента по практике).

– Список использованной литературы, технической и нормативной документации.

Руководитель практики должен проверить содержание отчета по практике с целью проверки грамотности его выполнения, допустимости

разглашения отдельной информации.

1.6. Требования к оформлению отчета по учебной ознакомительной практике

Основные принципы оформления отчетов по практике приведены в Стандарте организации «Порядок оформления методических разработок, курсовых и выпускных квалифицированных работ». Порядок разработан на основе Государственных стандартов оформления документов (ГОСТ 1.5-2001, ГОСТ 7.1-2003).

При оформлении отчёта по практике необходимо соблюдать следующие требования:

– отчёт оформляется на одной стороне листа А4 (210х297 мм), белой бумаги с соблюдением следующих размеров полей: верхнее и нижнее - по 20 мм, левое поле – 30 мм, правое поле – 10 мм;

– таблицы и иллюстрации допускается представлять на листах формата А3 (297х420 мм);

– текст готовится к распечатке на одной стороне листа на компьютере;

– шрифт «Times New Roman» 14 pt, межстрочный интервал – полуторный;

– наименования разделов располагаются по центру строки без точки в конце и набирается прописными буквами без подчёркиваний;

– заголовки подразделов набираются с абзацного отступа строчными буквами (кроме первой) без точки в конце;

– после номера раздела и подраздела в тексте точка не ставится;

– не допускаются переносы в словах разделов и подразделов;

– расстояние между заголовками и текстом должен составлять два интервала;

– страницы следует нумеровать цифрами;

– номер проставляется внизу страницы по центру в пределах рабочего поля; на титульном листе, номер страницы не ставится. Формулы следует выделять в отдельную строку, отделяя её от текста свободной строкой (сверху и снизу). Оформление формул производится при помощи любого редактора формул. Номера формул указываются арабскими цифрами в круглых скобках и имеют сквозную нумерацию;

– при оформлении таблиц и иллюстраций следует руководствоваться следующими правилами.

Все иллюстрации (кроме таблиц) в тексте работы называются **рисунками,** их представляют сразу же после упоминания или на следующей странице.

Выполняют рисунки черными чернилами (пастой) или тушью, если они не могут быть представлены с помощью компьютерной графики или в виде фотографий.

Под изображением рисунка приводится слово «Рисунок» с указанием после него через интервал цифры (номера) далее (через интервал) указывается название рисунка с заглавной буквы. При этом номер и название располагаются на одной и той же строке.

Перед рисунком в тексте оставляют одну свободную строку. Рисунок

должен быть расположен по центру страницы, в абзаце, содержащем рисунок, не должно быть заказано отступа или выступа первой строки. Выравнивание названия рисунка производится по левому краю. Нумерация рисунков в тексте должна быть сквозной.

Пример оформления рисунка:



Рис. 1. Функциональная схема котлоагрегата на экране монитора

Уравнения и формулы следует выделять из текста в отдельную строку. В формулах в качестве символов следует применять обозначения, установленные соответствующими государственными стандартами, и отраженные в формате формул. Образец оформления формулы представлен на рисунке 2.

Образец оформления формулы представлен на рисунке 2.

$$S = P \times (1 + \frac{I \times j}{100 \times K})^n$$

Рис. 2. Образец оформления формулы

где S – общая сумма («тело» вклада + проценты), причитающаяся к возврату вкладчику по истечении срока действия вклада;

P – первоначальная величина вклада;

n - общее количество операций по капитализации процентов за весь срок привлечения денежных средств (в данном случае оно соответствует количеству лет);

I – годовая процентная ставка.

K – количество дней в текущем году,

j – количество дней в периоде, по итогам которого банком производится капитализация начисленных процентов (остальные обозначения – как и в предыдущей формуле).

Цифровой материал рекомендуется помещать в тексте в виде **таблиц.** Каждая из них должна иметь заголовок, точно отражающий её содержание и расположенный по центру. Помещается заголовок непосредственно над таблицей по центру. После слова таблица через интервал цифры (номера) далее (через интервал) указывается название таблицы с заглавной буквы. При этом номер и название располагаются на одной и той же строке.

Пример оформления таблицы:

Таблица 1

Доля государственного долга в ВВП в развитых странах G-20
в 2017–2025 гг., %

Страны	2017 (факт)	2019 (факт)	2020 (оценка)	2021 (прогноз)	2025 (прогноз)
G-20 – развитые экономики	77,9	96,9	104,4	108,8	117,1
Япония	187,7	217,7	227,1	234,6	250,0
Италия	103,4	115,8	118,6	120,5	124,7
США	62,1	83,2	92,6	97,4	109,7
Франция	63,8	77,4	84,2	88,6	94,8
Великобритания	44,1	68,2	78,2	84,9	90,6

Источник: IMF World Economic Outlook. 2020. April. [Электронный ресурс]. – Электр. данн. – Режим доступа: http://www.imf.org/external/pubs/ft/weo/

Располагаются таблицы в тексте сразу после первой ссылки на них или на следующей (за первой ссылкой) странице. Оформляются таблицы так, как показано на примере приведенной Таблицы 1. При переносе части таблицы на другую страницу название помещают только над первой её частью, а над второй справа пишут «Продолжение таблицы» с указанием её номера. Допускается располагать таблицу вдоль длинной стороны листа. Графу «Номер по порядку» в таблице не делают.

Если повторяющийся в разных строках таблицы текст состоит из одного слова, то после его первого написания далее допускается заменять его кавычками. Если текст состоит из двух или более слов, то его можно заменять словами «то же», а далее заменять кавычками. Ставить кавычки вместо повторяющихся цифр не допускается.

Если таблица заимствована или рассчитана по данным статистического ежегодника или другого литературного источника, надо обязательно делать ссылку на первоисточник (см. пример Таблица 1).

Если в тексте необходимо сослаться на таблицу, то следует указать номер таблицы и страницу, на которой она расположена.

Приложения помещают за списком литературы в порядке упоминания их в основном тексте. Каждое приложение следует начинать с новой страницы с указанием наверху справа слова «Приложение».

Приложения обозначаются заглавными буквами русского алфавита, начиная с А. После слова «Приложение» должна следовать буква, обозначающая его последовательность.

Разработанные Обучающимся документы, имеющие самостоятельное значение и подлежащие исполнению в соответствии с заданием и нормативно-закрепленными требованиями, помещают в прилагаемую к текстовой части папку с соответствующей описью. При наличии аудио - или видеоматериалов и материалов в электронном виде их носители размещают в кармане этой же

папки.

Наличие графического материала является необходимым условием её написания. Этот материал может быть представлен в виде схем, рисунков, таблиц, графиков и диаграмм, которые должны наглядно иллюстрировать и подтверждать основную часть работы. Графический материал может помещаться или в основном тексте, или (если он представлен в формате, большем, чем А4) выноситься в приложения к работе. Перечень графического материала устанавливается в задании на практику.

Список использованной литературы оформляется в соответствии с требованиями ГОСТ Р 7.0.100-2018, ГОСТ 7.32-2017.

Рекомендуется представлять единый список литературы к работе в целом. Список обязательно должен быть пронумерован. Каждый источник упоминается в списке один раз, вне зависимости от того, как часто на него делается ссылка в тексте работы.

Наиболее удобным является алфавитное расположение материала, так как в этом случае произведения собираются в авторских комплексах. Произведения одного автора расставляются в списке по алфавиту заглавий.
Официальные документы ставятся в начале списка в определенном порядке
Конституции; Кодексы; Законы; Указы Президента; Постановления Правительства; другие нормативные акты (письма, приказы и т.д.). Внутри каждой группы документы располагаются в хронологическом порядке.

Литература на иностранных языках ставится в конце списка после литературы на русском языке, образуя дополнительный алфавитный ряд. Для каждого документа предусмотрены следующие элементы библиографической характеристики: фамилия автора, инициалы; название; подзаголовочные сведения (учебник, учебное пособие, словарь и т.д.); выходные сведения (место издания, издательство, год издания); количественная характеристика (общее количество страниц в книге).

В каждой группе библиографические записи должны располагаться в алфавитном порядке. Нумерация всего **списка использованных источников** сквозная. Библиографическая запись должна выполняться согласно ГОСТ Р 7.0.100-2018, ГОСТ 7.32-2017. Как правило, на языке выходных сведений: Автор (ФИО). Название источника. – Место издания: Издательство, год издания, количество страниц (общее или же страницы того раздела, который был использован при написании работы). Сведения об используемом издании находятся на обороте титульного листа книги;

– ссылки формируются в квадратных скобках, внутри которых указывается номер источника литературы, который соответствует его порядковому номеру в списке использованных источников. Например, ссылка [5] означает, что ссылка по тексту работы приведена на литературный источник, расположенный под порядковым номером 5 в Списке использованных источников;

– в случае необходимости дословного цитирования фрагмента авторского произведения заимствованный текст должен быть взят в кавычки

и снабжён ссылкой на источник, содержащий данный текст. Если в тексте присутствует заключенная в кавычки цитата, то ссылка на источник должна приводиться с указанием не только номера источника в Списке использованных источников, но и номера страницы источника, где расположена цитата, например, [10, с.37];

– указанные в Списке использованных источников электронные источники должны быть актуальными. Это означает, что дата обращения к электронному ресурсу должна соответствовать текущему месяцу, в котором осуществляются написание и проверка работы;

– запрещается использование в работе множественных ссылок на одно смысловое предложение, что наводит на мысль об их бессистемном, произвольном использовании.

В раздел Приложение включаются материалы, не вошедшие в текст основной части работы. Каждое приложение должно быть пронумеровано в порядке расположения приложений в тексте отчета практики. Заголовок «Приложение» располагается в правом верхнем углу с номером. Каждое приложение начинается с новой страницы.

– каждое Приложение имеет своё название, заголовок которого печатается в центре полужирным шрифтом, размер №16. Слово «Приложение» с номером (без знака № и точки на конце) печатается заглавными буквами в правом верхнем углу страницы, над названием. Каждое приложение начинается с новой страницы. Объем **Приложения** не входит в обязательное количество страниц работы и не должен превышать 1/3 всего текста работы;

– рисунки и таблицы имеют сквозную нумерацию в каждом отдельном Приложении. Например, рисунок 5, расположенный в Приложении В, обозначается: Рисунок В.5, таблица 3, расположенная в Приложении А, обозначается: Таблица А.3;

– работа должна быть написана от третьего лица. Использование в работе личных местоимений запрещается. Необходимо использовать выражения, типа: «известно, что», «существует мнение», «учёные придерживаются следующей точки зрения», «необходимо заметить», «представляет интерес» и т.п. Не допускается сокращение слов, все используемые аббревиатуры необходимо расшифровывать;

В тексте рекомендуется чаще применять красную строку, выделяя законченную мысль в самостоятельный абзац. Не следует приводить слишком много цитат, которые используются как приём аргументации. Текст не должен представлять собой сплошные перечисления. Необходимо чётко и ясно излагать материал, применяя принятую научную терминологию.

1.7. Подготовка презентационного материала

Обучающийся должен подготовить презентационный материал для защиты отчета по практике перед членами кафедры, используя средства Power Point. Структура презентационного материала (5– 10 слайдов):

– титульный лист (фирменный логотип ДГУ; название практики; выполнил обучающийся Ф.И.О., руководитель практики Ф.И.О., с указанием

научного звания и степени, Махачкала 202_г»;

- Введение
- результаты практики (реферат)

1.8. Аттестация по итогам практики

Защита учебной практики проводится после окончания практики в сроки, определяемые деканатом.

К защите допускаются студенты, у которых материалы по практике оформлены надлежащим образом и собраны все необходимые документы, указанные в предыдущем подразделе «Требования к оформлению отчета по учебной ознакомительной практике».

Защита студентом учебной практики оценивается комиссией по модульно-рейтинговой системе (отлично, хорошо, удовлетворительно, неудовлетворительно).

Сроки прохождения практики по согласованию с деканатом факультета могут быть продлены студентам, нарушающим трудовую дисциплину.

Полностью оформленный отчет представляется руководителю практики от кафедры «Информационные системы и технологии программирования» для проверки и защиты.

На основании полученного отчета, руководитель практики от кафедры принимает решение о допуске студента к защите отчета. Допуск студента к защите указывается на отчете, который вместе с рецензией руководителя практики от кафедры передается студенту для защиты. При отсутствии отчета с соответствующими рецензиями комиссия вправе не допустить студента к защите.

Защита отчетов проводится на кафедре ИСиТП комиссией, в состав которой входят руководители практик от кафедры, а также другие преподаватели кафедры.

Отчет с указанием даты защиты передается руководителем практики от кафедры ИСиТП ответственному лицу (методисту) по кафедре ИСиТП. Отметки о защите отчета по практике проставляются руководителем практики от кафедры ИСиТП в зачетной книжке и экзаменационной ведомости. Студенты, не прошедшие практику в установленные учебным планом сроки, допускаются к прохождению практики только по решению деканата.

II. Примерный перечень заданий по учебной практике
Задания для развития профессиональных навыков по дисциплине «Web-программирование и разработка сайтов»

Каждое задание содержит:
Название и цель (акцент на профессиональный навык).
Техническое задание (ТЗ).
Требования к реализации (кодстайл, безопасность, работа с памятью).

Задание 1. Адаптивный интернет-магазин (сетки и медиазапросы)

Цель: развить навык кроссбраузерной адаптивной вёрстки с использованием современных CSS-технологий (Grid, Flexbox, медиазапросы).

Техническое задание:

Разработать страницу каталога интернет-магазина (карточки товаров). На десктопе (≥1024px) карточки выводятся в 4 колонки, на планшете (768–1023px) – в 2 колонки, на мобильных (<768px) – в 1 колонку. Шапка и подвал должны адаптироваться (на мобильных меню превращается в «бургер»). Все изображения должны быть отзывчивыми (srcset, aspect-ratio).

Требования к реализации:

Семантическая вёрстка (header, main, section, article, footer).

Использовать CSS Grid для сетки каталога, Flexbox – для внутреннего выравнивания.

Медиазапросы в отдельных блоках или с помощью @container.

Кодстайл: BEM-именование классов, форматирование по Prettier.

Проверить в Chrome, Firefox, Safari последних версий.

Безопасность: нет встроенных стилей и скриптов, внешние ресурсы через HTTPS.

Задание 2. SPA с роутингом на чистом JavaScript

Цель: освоить управление состоянием и роутинг без фреймворков, понять принципы работы History API.

Техническое задание:

Создать одностраничное приложение – список задач (ToDo). Страницы: «Все задачи», «Активные», «Выполненные». Переходы должны происходить без перезагрузки, с изменением URL (pushState). При обновлении страницы открывается тот же роут. Данные хранятся в localStorage.

Требования к реализации:

Использовать модули ES6 (import/export).

Реализовать класс Router с методами addRoute, navigate, start.

Все обработчики событий должны удаляться при смене роута (предотвращение утечек памяти).

Кодстайл: ESLint (Airbnb config), комментарии к сложным участкам.

Безопасность: санитизация вывода задач (защита от XSS через textContent).

Задание 3. RESTful API с авторизацией JWT (Node.js/Express)

Цель: получить навык разработки защищённого бэкенда с аутентификацией и авторизацией.

Техническое задание:

Реализовать API для блога: регистрация, логин, создание/редактирование/удаление постов (только автором). Эндпоинты:

POST /register – хеширование пароля (bcrypt), выдача JWT.

POST /login – проверка, выдача JWT.

GET /posts – публичный список постов.

POST /posts – создание поста (требуется JWT).

PUT /posts/:id – обновление (только автор).

DELETE /posts/:id – удаление (только автор).

Требования к реализации:

Валидация входных данных (Joi или express-validator).

Защита от NoSQL-инъекций (если MongoDB) или SQL-инъекций (ORM/параметризованные запросы).

Пароли не хранить в открытом виде, использовать bcrypt.

JWT хранить в httpOnly-куках (безопаснее, чем localStorage).

Логирование ошибок (winston).

Кодстайл: ESLint, Prettier, разделение на маршруты, контроллеры, сервисы.

Задание 4. Оптимизация производительности и работа с памятью

Цель: научиться выявлять и устранять утечки памяти, оптимизировать рендеринг больших объёмов данных.

Техническое задание:

Разработать приложение для отображения большого списка (10 000 элементов) с возможностью фильтрации и удаления. Реализовать:

Виртуальный скроллинг (отображать только видимые элементы).

Ленивую загрузку изображений (Intersection Observer).

Кнопку «Очистить неиспользуемые обработчики» (демонстрация удаления слушателей).

Требования к реализации:

Использовать requestAnimationFrame для плавного скролла.

В DevTools (Performance, Memory) доказать отсутствие утечек (сделать скриншоты до/после).

Все обработчики событий, добавленные динамически, должны удаляться при удалении элементов.

Кодстайл: избегать глобальных переменных, использовать WeakMap/WeakSet при необходимости.

Безопасность: ограничение частоты событий (throttle для скролла).

Задание 5. Интеграция с внешним API (погода) и кэширование

Цель: освоить асинхронные запросы, обработку ошибок, стратегии кэширования на клиенте.

Техническое задание:

Веб-приложение для отображения погоды в любом городе. При вводе названия отправляется запрос к OpenWeatherMap API. Данные кэшируются на 10 минут (localStorage или IndexedDB). Реализовать:

Индикатор загрузки.

Обработку ошибок (нет города, лимиты API, отсутствие сети).

Автодополнение названий городов (debounce).

Требования к реализации:

Использовать Fetch API с AbortController (отмена предыдущих запросов).

Кэш должен инвалидироваться по времени и по ключу (город).

Кодстайл: async/await, try/catch, отдельный модуль API.

Безопасность: API-ключ хранить на бэкенде (проксировать запросы) или использовать переменные окружения в сборщике.

Работа с памятью: очищать таймеры (debounce) при размонтировании компонента.

Задание 6. Компонент карусели (слайдера) с нуля на классах

Цель: развить навыки ООП в JavaScript и эффективной работы с DOM-элементами.

Техническое задание:

Создать переиспользуемый класс Carousel, который принимает селектор контейнера и массив изображений. Функциональность:

Автоматическое переключение слайдов (интервал).

Кнопки «вперёд/назад», индикаторы (точки).

Поддержка свайпов на тач-устройствах.

Пауза при наведении.

Требования к реализации:

Карусель не должна создавать утечек – метод destroy для удаления всех обработчиков и остановки таймера.

Использовать делегирование событий для кнопок/индикаторов.

Кодстайл: приватные поля (#currentIndex), геттеры/сеттеры.

Анимация с использованием transform и transition (аппаратное ускорение).

Безопасность: проверять существование элементов перед доступом.

Задание 7. Форма с валидацией и защитой от XSS/CSRF

Цель: применить на практике основные меры веб-безопасности при работе с пользовательским вводом.

Техническое задание:

Форма обратной связи (имя, email, сообщение). На клиенте – валидация (регулярные выражения, подсказки). На сервере (Node.js) – повторная валидация, санитизация HTML (DOMPurify или аналоги). Реализовать защиту от CSRF (токен в форме). Данные отправляются через fetch, сохраняются в базу (SQLite/PostgreSQL). После отправки – вывод сообщения без перезагрузки.

Требования к реализации:

Клиент: запрещено отправлять форму с пустыми или некорректными полями.

Сервер: все входящие данные экранируются, HTML-теги удаляются (кроме безопасных).

CSRF-токен: генерируется на сервере, хранится в сессии, проверяется при POST-запросе.

Заголовки: Content-Security-Policy (базовая политика), X-Content-Type-Options.

Кодстайл: использование prepared statements для SQL.

Работа с памятью: очистка таймеров для подсказок, удаление обработчиков после размонтирования (если SPA).

Задание 8. Бесконечная прокрутка (infinite scroll) с виртуализацией

Цель: научиться эффективно подгружать и отображать большие объёмы данных с сервера.

Техническое задание:

Лента постов (заголовок + текст), данные подгружаются порциями по 20 штук с бэкенда (JSON-файл или простая имитация с задержкой). При скролле до низа – загружается следующая страница. Реализовать виртуальный скроллинг (отображаются только посты, попадающие во вьюпорт + буфер). Общее количество постов может быть любым (до 5000).

Требования к реализации:

Высота каждого поста может быть разной (динамическая) – использовать библиотеку или ручной расчёт (ResizeObserver).

Отмена предыдущих запросов при быстром скролле.

Обработка состояния «идёт загрузка», «ошибка», «конец данных».

Кодстайл: модульная структура, отсутствие глобальных переменных.

Работа с памятью: удаление DOM-элементов, вышедших из буфера, отписка от ResizeObserver.

Задание 9. Многоязычный сайт (i18n) с динамической подгрузкой переводов

Цель: освоить интернационализацию веб-приложений и управление языковыми ресурсами.

Техническое задание:

Сайт-визитка (главная, о нас, контакты) с поддержкой русского и английского языков. Переключение языка без перезагрузки (изменение текстов на странице). Переводы хранятся в JSON-файлах, которые подгружаются динамически при смене языка. В URL добавляется префикс /ru/ или /en/ (роутинг на клиенте). Сохранять выбранный язык в localStorage.

Требования к реализации:

Использовать атрибут data-i18n для элементов, которые нужно переводить.

Реализовать класс I18n с методами setLanguage, translatePage, loadLocale.

Кэшировать загруженные переводы в памяти (Map).

Кодстайл: обработка отсутствующих ключей (fallback на язык по умолчанию).

Безопасность: не доверять содержимому JSON-файлов, экранировать при вставке.

Работа с памятью: очищать кэш при долгом бездействии (опционально), удалять старые обработчики.

Задание 10. WebSocket-чат с комнатами и уведомлениями о наборе текста

Цель: получить опыт разработки real-time приложений с двунаправленной связью (WebSocket).

Техническое задание:

Чат, в котором пользователь может создать комнату или присоединиться к существующей (по ID). Возможности:

Отправка и получение сообщений.

Уведомления «Пользователь печатает...» (typing indicator).

Список участников в комнате (онлайн).

История сообщений (последние 50, хранятся на сервере в памяти/БД).

Требования к реализации:

Сервер: Node.js + ws или Socket.IO.

Клиент: подключение, переподключение при обрыве соединения.

Обработка ошибок: разрыв соединения, неверный ID комнаты.

Кодстайл: использование бинарных фреймов для оптимизации (опционально).

Безопасность: валидация входных сообщений (длина, запрещённые символы), rate limiting.

Работа с памятью: на сервере – удалять закрытые соединения, очищать историю комнат при пустоте. На клиенте – удалять обработчики WebSocket перед пересозданием.

Задания для развития профессиональных навыков по дисциплине «Инженерия знаний»

Задание 1. Разработка экспертной системы на продукционных правилах

Название: «Продукционная экспертная система для диагностики неисправностей ПК»

Цель: формирование навыка представления знаний в виде правил «ЕСЛИ–ТО», реализации механизма прямого логического вывода (forward chaining) и работы с конфликтным набором.

Техническое задание (ТЗ):

Разработать базу знаний, содержащую не менее 10 правил диагностики неисправностей персонального компьютера (например, по симптомам: не включается, шумит кулер, нет изображения и т.д.).

Реализовать интерпретатор правил, который на основе введённых пользователем фактов (симптомов) строит цепочку вывода и выдаёт возможную причину неисправности.

Предусмотреть возможность добавления/удаления правил без перекомпиляции.

Вывод должен быть объяснимым (показывать, какие правила сработали и в каком порядке).

Требования к реализации:

Язык: Python 3.10+ или Prolog (SWI-Prolog).

Кодстайл: PEP 8 для Python; для Prolog – единый стиль именования предикатов.

Безопасность: валидация вводимых фактов (защита от инъекций, если используется eval – запрещён); использование только безопасных методов парсинга.

Работа с памятью: избегать рекурсии с неограниченной глубиной (использовать итеративный алгоритм или ограничить глубину вывода); освобождать ресурсы (закрывать файлы с правилами).

Задание 2. Построение онтологии предметной области

Название: «Онтология “Университет” на языке OWL и работа с ней через RDFlib»

Цель: развитие навыка инженерии онтологий – выделение классов, свойств, экземпляров; практическое использование стандартов семантического веба (RDF, OWL).

Техническое задание (ТЗ):

Спроектировать онтологию предметной области «Университет» (классы: Студент, Преподаватель, Кафедра, Дисциплина, Зачётная книжка и т.д.).

Задать свойства (data properties, object properties), например «преподаёт», «изучает», «имеет оценку».

Создать не менее 5 экземпляров каждого класса.

Реализовать на Python с использованием библиотеки RDFlib:

загрузку/сохранение онтологии в формате Turtle;

выполнение SPARQL-запроса «Найти всех студентов, сдавших дисциплину “Инженерия знаний” на отлично».

Предоставить визуализацию иерархии классов (например, с помощью graphviz).

Требования к реализации:

Язык: Python 3.10+; библиотеки rdflib, pydot (для визуализации).

Кодстайл: аннотации типов, docstring для всех функций.

Безопасность: экранирование строк при построении SPARQL-запросов (использовать параметризованные запросы RDFlib).

Работа с памятью: не загружать всю онтологию в память, если она большая (в учебных целях допустимо, но использовать генераторы при итерации по экземплярам).

Задание 3. Реализация обратного логического вывода (backward chaining)

Название: «Алгоритм обратного вывода для игры “Одиннадцать палочек”»

Цель: освоение механизма целеориентированного вывода, рекурсивного поиска в пространстве состояний и организации базы правил.

Техническое задание (ТЗ):

Описать правила игры: из кучки из 11 палочек два игрока по очереди берут 1, 2 или 3 палочки. Проигрывает тот, кто берёт последнюю палочку.

Реализовать предикаты (на Prolog или на Python с использованием рекурсии), которые с помощью обратного вывода определяют выигрышную стратегию для первого игрока.

На запрос win(11, first) программа должна выдать последовательность ходов, гарантирующих победу.

База знаний должна содержать правила, описывающие выигрышные и проигрышные позиции.

Требования к реализации:

Язык: Prolog (SWI-Prolog) или Python (с рекурсивным поиском в глубину).

Кодстайл: для Prolog – атомы и переменные в разных регистрах; для Python – именование функций в snake_case.

Безопасность: ограничение глубины рекурсии (sys.setrecursionlimit) и защита от бесконечного зацикливания путём мемоизации.

Работа с памятью: использование кэша (таблицы) для вычисленных состояний, чтобы избежать экспоненциального роста числа рекурсивных вызовов.

Задание 4. Извлечение знаний из текста (именованные сущности)

Название: «Извлечение фактов из новостной ленты с помощью NER и регулярных правил»

Цель: формирование навыка извлечения структурированных знаний из неструктурированных текстов, настройка NLP-пайплайна.

Техническое задание (ТЗ):

Используя библиотеку spaCy (модель ru_core_news_sm или en_core_web_sm), разработать конвейер, который из новостного текста выделяет:

организации (ORG),

персоны (PER),

даты (DATE),

суммы денег (MONEY).

Для каждого найденного факта построить RDF-триплеты (субъект–предикат–объект), например: ("Роснефть", "заключила контракт с", "РЖД"), ("Сергей Иванов", "назначен", "директором").

Результат сохранить в CSV-файл и в Turtle-файл (RDF).

Предусмотреть обработку нескольких текстов (не менее 5) из файлового

каталога.

Требования к реализации:

Язык: Python 3.10+; библиотеки spaCy, re, rdflib.

Кодстайл: логическое разделение на модули (извлечение, сериализация).

Безопасность: не использовать eval; экранировать спецсимволы при записи в CSV; проверять размер загружаемых текстов.

Работа с памятью: обрабатывать тексты потоково (по одному), не накапливать все результаты в одном огромном списке.

Задание 5. Семантическая сеть и поиск путей в графе знаний

Название: «Построение семантической сети по сказке “Колобок” и поиск связей между персонажами»

Цель: развитие навыка моделирования знаний в виде ориентированного графа, реализация алгоритмов поиска путей и распространения активации.

Техническое задание (ТЗ):

На основе сюжета русской народной сказки «Колобок» построить семантическую сеть:

вершины – персонажи (Колобок, Заяц, Волк, Медведь, Лиса) и события (встретил, укатился, съел).

рёбра с метками (например, «Колобок» –[встретил]– «Заяц»).

Используя библиотеку NetworkX, реализовать:

добавление/удаление вершин и рёбер;

визуализацию графа;

поиск всех путей между двумя заданными вершинами (например, от «Колобок» до «Лиса»).

Написать функцию, которая по запросу «Кто встретился Колобку после Волка?» возвращает ответ, используя обход графа.

Требования к реализации:

Язык: Python 3.10+; networkx, matplotlib.

Кодстайл: разделение на функции создания графа, визуализации, поиска.

Безопасность: проверять существование вершин перед операциями; избегать инъекций при динамическом формировании запросов (если реализован текстовый парсер).

Работа с памятью: для больших графов не хранить все пути сразу (использовать генераторы), но в учебном проекте допустим список.

Задание 6. Фреймовая модель представления знаний

Название: «Фреймовая система для описания библиотечного каталога»

Цель: освоение фреймового подхода – слотов, демонов (процедур-присоединений), наследования.

Техническое задание (ТЗ):

Спроектировать фреймы «Книга», «Читатель», «Библиотекарь».

Каждый фрейм должен содержать слоты (атрибуты) с типом данных и значениями по умолчанию.

Реализовать на Python классы, имитирующие фреймы, и добавить «демонов»: при изменении слота статус_книги (выдана/в наличии) должен автоматически обновляться слот доступность и создаваться запись в журнале.

Написать простой интерфейс командной строки для создания экземпляров фреймов (конкретных книг и читателей) и выполнения операций выдачи/возврата.

Требования к реализации:

Язык: Python 3.10+ (ООП).

Кодстайл: использование свойств (@property, @setter) для реализации демонов; наследование для общих слотов.

Безопасность: контроль типов при установке слотов (нельзя присвоить строку в числовой слот); валидация имени читателя.

Работа с памятью: удаление экземпляров фреймов через del или слабые ссылки (при необходимости), освобождение ресурсов при выходе.

Задание 7. Работа с публичной базой знаний через SPARQL

Название: «Извлечение и анализ данных о российских учёных из Wikidata»

Цель: приобретение навыка использования распределённых знаний через SPARQL-конечные точки, фильтрации и агрегации.

Техническое задание (ТЗ):

Используя общедоступный SPARQL-эндпоинт Wikidata (https://query.wikidata.org/sparql), написать запросы для получения:

списка российских учёных в области информатики, у которых есть дата рождения;

для каждого – количества публикаций (если данные есть);

среднего количества публикаций по десятилетиям рождения.

Результаты сохранить в формате JSON и CSV.

Реализовать на Python автоматическую отправку запросов (библиотека SPARQLWrapper) и обработку возможных ошибок (таймаут, пустой результат).

Требования к реализации:

Язык: Python 3.10+; SPARQLWrapper, pandas (опционально).

Кодстайл: вынос SPARQL-запросов в отдельные строки-константы; логирование процесса.

Безопасность: не подставлять пользовательский ввод напрямую в запрос (использовать параметры через SPARQLWrapper.setParameter).

Работа с памятью: обрабатывать постраничные результаты (OFFSET/LIMIT), чтобы не загружать сразу миллион записей.

Задание 8. Реализация алгоритма классификации (дерево решений)

Название: «Извлечение правил из данных: построение дерева решений ID3»

Цель: развитие навыка автоматического извлечения эмпирических знаний из табличных данных, понимание принципов индуктивного вывода.

Техническое задание (ТЗ):

Взять датасет «Погода» (прогноз игры в теннис) или любой другой из UCI (не менее 10 записей, 4 атрибута).

Реализовать алгоритм ID3 (или C4.5 упрощённый) на Python без использования готовых библиотек машинного обучения.

Построить дерево решений, вывести его в текстовом виде (отступы, правила).

Извлечь из дерева продукционные правила (ЕСЛИ … ТО …).

Проверить точность на тестовой выборке (2–3 примера).

Требования к реализации:

Язык: Python 3.10+; pandas (для чтения данных), без sklearn.tree.

Кодстайл: рекурсивная функция build_tree, расчёт энтропии, информационного выигрыша.

Безопасность: обработка пропущенных значений (замена на наиболее частый класс); предотвращение деления на ноль.

Работа с памятью: ограничение глубины дерева (параметр max_depth) во избежание переполнения стека рекурсии.

Задание 9. База знаний на основе Datalog и SQL

Название: «Реализация дедуктивной базы данных на SQLite с поддержкой рекурсивных запросов»

Цель: формирование навыка интеграции реляционных данных и логического вывода, использование рекурсивных SQL-запросов (WITH RECURSIVE) для вывода новых фактов.

Техническое задание (ТЗ):

Спроектировать схему БД (SQLite), хранящую факты: Родитель(ребёнок, родитель), Мужчина(человек), Женщина(человек).

Заполнить не менее 6 фактов (семейное дерево).

Написать рекурсивный запрос для вывода всех предков заданного человека.

Добавить правила (представления), определяющие «Брата», «Сестру», «Дядю».

На основе этих правил написать запрос: «Найти всех дядьев Ивана».

Реализовать на Python скрипт, создающий БД, вставляющий данные и выполняющий запросы, выводящий результат.

Требования к реализации:

Язык: Python 3.10+; sqlite3.

Кодстайл: использование контекстных менеджеров для соединений; именование констант.

Безопасность: параметризованные запросы (не конкатенировать строки); ограничение глубины рекурсии в SQLite (установка max_recursion).

Работа с памятью: закрытие курсора и соединения; удаление временных представлений после использования.

Задание 10. Нечёткая экспертная система

Название: «Нечёткое управление скоростью вентилятора на основе

температуры и влажности»

Цель: приобретение навыка моделирования неопределённых знаний с использованием нечёткой логики (фаззификация, база нечётких правил, дефаззификация).

Техническое задание (ТЗ):

Разработать нечёткую экспертную систему, которая по входным параметрам «температура (от 0 до 40°C)» и «влажность (0–100%)» определяет скорость вращения вентилятора (0–100%).

Задать терм-множества: температура («низкая», «средняя», «высокая»), влажность («сухо», «норма», «влажно»), скорость («минимальная», «умеренная», «максимальная»).

Сформулировать не менее 5 нечётких правил (например, «ЕСЛИ температура высокая И влажность норма, ТО скорость максимальная»).

Реализовать на Python (библиотека scikit-fuzzy или самодельные треугольные функции) полный цикл: фаззификация -> агрегирование -> дефаззификация (центр тяжести).

Предоставить графики функций принадлежности и поверхности управления.

Требования к реализации:

Язык: Python 3.10+; numpy, matplotlib, scikit-fuzzy (разрешено).

Кодстайл: модульная структура (фаззификатор, база правил, дефаззификатор).

Безопасность: проверка входных параметров на диапазон; обработка нечислового ввода.

Работа с памятью: предварительное вычисление и кэширование функций принадлежности для ускорения; очистка графиков после сохранения.

Задания для развития профессиональных навыков по дисциплине «Реинжиниринг и управление информационными процессами»

Задание 1. «Выбор процесса для реинжиниринга»

Название: Оценка процессов и обоснование выбора кандидата на реинжиниринг

Цель: Развить навык критического отбора бизнес-процессов, наиболее нуждающихся в реинжиниринге (BPR), на основе количественных и качественных критериев.

Техническое задание (ТЗ):

Выберите одну организацию (учебную, фиктивную или реальную из открытых источников).

Составьте список из 5–7 её сквозных процессов.

Определите критерии оценки: длительность, стоимость, количество ошибок, степень автоматизации, критичность для клиента.

Проведите приоритизацию процессов (матрица «эффективность – важность») и выберите один процесс для дальнейшего реинжиниринга.

Требования к реализации:

Отчёт 3–5 страниц, включая таблицу критериев и матрицу выбора.

Обоснование должно содержать не менее 3 числовых метрик (например, «время выполнения 5 дней, что в 3 раза выше отраслевого среднего»).

Допускается использование любых инструментов (Excel, Miro, Word).

Задание 2. «As-Is модель процесса в нотации BPMN 2.0»

Название: Детальное картирование существующего процесса

Цель: Сформировать навык формализации и визуализации текущего состояния процесса с выявлением «узких мест».

Техническое задание:

Возьмите процесс, выбранный в Задании 1.

Соберите информацию о шагах, участниках, входах/выходах, используемых ИС.

Постройте модель As-Is в нотации BPMN 2.0 (уровень детализации – до ролей и элементарных операций).

Отметьте на схеме проблемные зоны: задержки, дублирование, ручные операции, разрывы в данных.

Рассчитайте текущее время выполнения (Takt time) и долю ручного труда.

Требования к реализации:

Модель в одном из инструментов: Camunda Modeler, Draw.io, Bizagi Modeler.

Таблица с метриками (время, стоимость операции, частота ошибок, если данные есть).

Задание 3. «Анализ узких мест с помощью симуляции»

Название: Имитационное моделирование процесса для прогноза пропускной способности

Цель: Освоить метод количественного анализа процесса через дискретно-событийную симуляцию.

Техническое задание:

На основе As-Is модели (Задание 2) создайте имитационную модель в среде (AnyLogic, GPSS World или BPM-симулятор в Bizagi/Camunda).

Задайте входные потоки заявок (интенсивность, распределение), время выполнения операций, ресурсы.

Запустите симуляцию на 100–1000 событий.

Определите: среднее время ожидания, загрузку ресурсов, длину очередей.

Укажите 2–3 узких места, подтверждённых симуляцией.

Требования к реализации:

Скриншоты модели и графиков результатов.

Таблица сравнения расчётных и симуляционных показателей.

Вывод: какие операции лимитируют производительность.

Задание 4. «To-Be модель и идеи реинжиниринга»

Название: Проектирование целевого процесса (To-Be) с разрывом в 50%

улучшений

Цель: Научиться генерировать и формализовать радикальные изменения процесса (BPR-подход).

Техническое задание:

Предложите 3–4 изменения, которые сократят время выполнения процесса минимум на 50% (удаление шагов, объединение ролей, внедрение ИТ-решений).

Постройте модель To-Be в BPMN 2.0.

Опишите изменения по каждому элементу модели (было → стало).

Оцените ожидаемый эффект: время, стоимость, качество (SLA).

Требования к реализации:

Сравнительная таблица As-Is vs To-Be по метрикам.

Обоснование, почему изменения являются реинжинирингом, а не улучшением (пункт Хаммера и Чампи).

Модель To-Be в том же инструменте, что и As-Is.

Задание 5. «Информационная архитектура нового процесса»

Название: Проектирование потоков данных и требований к информационной системе

Цель: Развить навык согласования бизнес-процесса и архитектуры данных.

Техническое задание:

Для To-Be модели (Задание 4) выделите все информационные объекты (заявка, договор, счёт и т.д.).

Постройте диаграмму потоков данных (DFD) уровня 0 и 1 (нотация Гейна-Сарсона или Yourdon).

Определите, какие данные должны храниться, какие – передаваться в реальном времени.

Сформулируйте 5–7 функциональных требований к ИС, поддерживающей новый процесс.

Укажите необходимость интеграции с существующими системами (ERP, CRM, БД).

Требования к реализации:

DFD в Draw.io или Lucidchart.

Список требований в формате «пользовательская история» или Use Case.

Матрица «информационный объект → система-источник».

Задание 6. «Расчёт ROI реинжиниринга»

Название: Экономическое обоснование проекта BPR

Цель: Сформировать навык оценки затрат и выгод при внедрении реинжиниринга.

Техническое задание:

Оцените затраты на внедрение To-Be модели: лицензии ПО, разработка/доработка ИС, обучение, управление изменениями.

Рассчитайте операционные выгоды за 3 года (снижение времени, уменьшение ошибок, сокращение ФОТ).

Вычислите NPV, ROI и срок окупаемости (Payback Period). Ставка дисконтирования – 12%.

Проведите анализ чувствительности: изменение выгод на ±20%.

Дайте заключение – стоит ли реализовывать проект.

Требования к реализации:

Таблицы в Excel (приложить файл или подробные расчёты в отчёте).

Формулы и допущения явно прописаны.

Вывод: приемлемый или нет ROI (порог >30% для BPR-проектов).

Задание 7. «Управление изменениями и рисками при BPR»

Название: План сопротивления и митигации рисков

Цель: Развить навыки организационного проектирования и управления человеческим фактором при реинжиниринге.

Техническое задание:

Составьте карту стейкхолдеров процесса (роли, их выгоды/потери от изменений).

Выявите 5 основных рисков внедрения To-Be (технических, культурных, ресурсных).

Для каждого риска предложите меру митигации и ответственного.

Разработайте план коммуникаций на 3 месяца (ключевые сообщения, каналы, частота).

Опишите, как вы будете измерять готовность к изменениям (опросы, чек-листы).

Требования к реализации:

Матрица рисков (вероятность × влияние).

Календарный план-график коммуникаций в виде таблицы.

Пример анкеты для оценки готовности персонала.

Задание 8. «Разработка метрик и панели мониторинга процесса»

Название: Построение BPM-дашборда для контроля нового процесса

Цель: Научиться определять KPI и визуализировать их для оперативного управления.

Техническое задание:

Определите 5–7 ключевых показателей для To-Be процесса (время цикла, качество, загрузка, удовлетворённость клиента, процент автоматизации).

Разработайте макет дашборда в Tableau Public, Power BI или даже Excel (с активными элементами).

Укажите источники данных для каждого KPI (логи ИС, опросы, ручной ввод).

Добавьте на дашборд пороговые значения («красная/жёлтая/зелёная» зоны).

Напишите инструкцию для менеджера по использованию дашборда для

принятия решений.

Требования к реализации:

Скриншот или ссылка на интерактивный дашборд.

Таблица KPI с формулами расчёта.

Пример сценария: «Если KPI X становится красным, менеджер должен сделать Y».

Задание 9. «Автоматизация процесса с помощью RPA или low-code»

Название: Прототип программного робота для рутинной операции

Цель: Получить практический навык автоматизации повторяющихся шагов процесса.

Техническое задание:

Выберите одну ручную операцию из As-Is модели (например, перенос данных из Excel в CRM, формирование отчёта).

Разработайте сценарий автоматизации с использованием RPA-среды (UiPath Community, Power Automate Desktop) или low-code (n8n, Make).

Реализуйте работающий прототип (можно на тестовых данных).

Замерьте время выполнения до и после автоматизации.

Оцените, на сколько процентов сократится длительность процесса после внедрения робота.

Требования к реализации:

Видео (1–2 минуты) или пошаговые скриншоты работы робота.

Исходный код/файл автоматизации (в архиве).

Сравнение: «было 5 мин → стало 15 сек».

Задание 10. «Презентация проекта реинжиниринга перед "советом директоров"»

Название: Защита бизнес-кейса BPR в формате питча

Цель: Сформировать навык коммуникации результатов реинжиниринга для принятия инвестиционного решения.

Техническое задание:

Подготовьте презентацию (6–8 слайдов) по итогам Заданий 1–9:

Проблема и выбор процесса (As-Is)

Идеи реинжиниринга (To-Be)

Ожидаемые метрики и ROI

План внедрения и риски

Дашборд контроля

Требования к реализации:

Файл презентации в PDF/PPTX.

Видео (ссылка на облако или файл) с чёткой дикцией и визуальным контактом.

2.2. Индивидуальные задания по учебной практике
Инструкция по оформлению тестовых заданий для электронной базы ДГУ.

Формат файла заданий

Задания набираются в редакторе **Word** и сохраняются в формате **DOC.** Шрифт Times New Roman, размер шрифта 14.

Написание ключевых слов

Ключевое слово – это набор символов, расположенный в начале каждого файла заданий и каждого раздела тестового задания, необходимый для автоматической обработки заданий. По ключевым словам программа распознает и обрабатывает каждое тестовое задание и весь файл заданий в целом.

***Внимание*!** Если Вы не отследите правильность написания ключевых слов, то Ваш файл заданий будет обработан с ошибками.

Правила написания ключевых слов:

- каждое ключевое слово пишется на новой строке
- ключевое слово начинается с символа № в начале строки без пробела (набирается одновременным нажатием клавиши SHIFT и цифры 3 в кириллической раскладке клавиатуры)
- ключевые слова пишутся исключительно русскими буквами

При оформлении комплектов тестовых заданий используйте следующие ключевые слова:

для заголовка файла

№автор=
№дисциплина=
№блок=
№модуль=
№тема=

для оформления отдельного задания

№вопрос<форма>
№да
№нет

Внутри ключевого слова не должно быть пробелов!

Заголовок файла

Обязательный набор ключевых слов в заголовке файла:
№дисциплина= <название дисциплины>
№блок= <1, 2, 3>

Формы тестовых заданий и их кодировка

Предусмотрены следующие формы тестовых заданий:

- 1 – с выбором одного правильного ответа из предложенных и развернутым обоснованным ответом (1-5 предложений).
- 2 – с выбором нескольких правильных ответов из предложенных
- 3 – на установление правильной последовательности
- 4 – на установление соответствия
- 5 – с кратким ответом и обоснованием ответа (1-5 предложений)

Форма задания указывается в ключевом слове №вопрос<1, 2, 3, 4, 5>.

Например: №вопрос1 - означает, что за ней следует вопрос с выбором одного правильного ответа из нескольких предложенных.

***Внимание*!** Код формы задания **не отделяется** от ключевого слова пробелом.

Обязательно! После №вопрос<1, 2, 3, 4, 5> указать инструкцию по выполнению задания.

Для каждого типа заданий имеется своя типовая инструкция по выполнению

Варианты оформления тестовых заданий.

№вопрос1 – задания с выбором одного правильного ответа из списка

Перед каждым тестовым заданием должно быть ключевое слово **№вопрос1**. На следующей строке после текста задания должно находиться ключевое слово **№да**, а сточкой ниже – правильный ответ. На следующей строке – ключевое слово **№нет**, затем неправильный ответ, далее – опять ключевое слово **№нет** и следующий неправильный ответ и так далее. В конце тестового задания с новой строки ключевое слово **№ключ,** строкой ниже обоснование выбранного ответа (3-7 предложений)**.**

Количество предложенных ответов может быть в разных заданиях от 4 до 6.

№вопрос1
<Текст задания>
№да
<правильный ответ>
№нет
<дистрактор>
№нет
<дистрактор>
№нет
<дистрактор>
№ключ
<Обоснование выбора ответа. От 3 до 7 предложений.>

Примеры заданий формы 1:

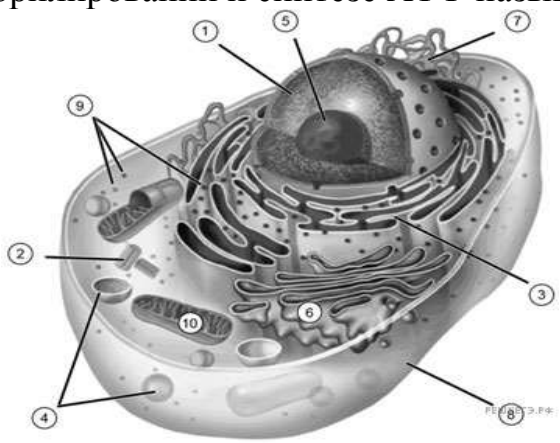
№вопрос1

В модели кругооборота продуктов и доходов в состав денежных утечек включаются

№да

чистые налоги
№нет
государственные закупки
№нет
валовые внутренние инвестиции
№нет
экспортная выручка
№ключ
Совокупные денежные утечки в модели кругооборота продуктов и доходов включают: сбережения, чистые налоги, расходы на импорт.

№вопрос1
Рассмотрите рисунок. Органоид, участвующий в окислительном фосфорилировании и синтезе АТФ называется



№да
митохондрии
№нет
лизосома
№нет
рибосома
№нет
ядрышко
№ключ
Органоид, который участвует в окислительном фосфорилировании и синтезе АТФ, называется: ядро, аппарат Гольджи, вакуоль, митохондрии.

№вопрос2 – задания с выбором нескольких правильных ответов из предложенных

Перед каждым тестовым заданием данной формы должно быть ключевое слово **№вопрос2.**

Количество ответов должно быть от 4 до 6, также обязателен хотя бы один неправильный ответ. Каждый правильный ответ предваряется ключевым словом **№да**, все они последовательно располагаются после текста задания, каждый неправильный ответ предваряется ключевым словом **№нет,** все неправильные ответы располагаются после правильных.

№вопрос2
<Текст задания>
№да
<правильный ответ>
№да
<правильный ответ>
№да
<правильный ответ>
№нет
<дистрактор>
№нет
<дистрактор>
№нет
<дистрактор>

Примеры задания формы 2:

№вопрос2

Мероприятиями Петра I, направленными на «европеизацию» страны, являлись

№да

упразднение патриаршества и создание Синода

№да

создание регулярной армии

№нет

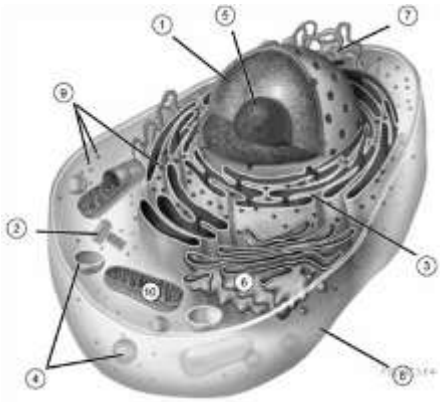
введение общерусского свода законов – Судебника

№нет

освобождение дворянства от обязательной службы

№вопрос2

Перечислить основные признаки подтверждающие, что на рисунке изображена животная клетка.



№да

отсутствие центральной крупной вакуоли

№да

отсутствие пластид

№да

клеточный центр с центриолями
№да
отсутствие толстой клеточной стенки
№нет
наличие ядрышка
№нет
наличие рибосом

№вопрос3 – задания на установление правильной последовательности

Перед каждым тестовым заданием данной формы должно быть ключевое слово **№вопрос3**. Для этой формы допустимы только правильные ответы, при этом последовательность расположения ответов соответствует правильной последовательности (последовательность должна содержать от 4 до 6 ответов).

№вопрос3
<Текст задания>
№да
<правильный ответ>
№да
<правильный ответ>
№да
<правильный ответ>
№да
<правильный ответ>

Пример задания формы 3:

№вопрос3
Расположите туристские районы России с запада на восток
№да
Республика Крым
№да
Верхневолжский
№да
Кольско-Карельский
№да
Уральский
№да
Саянский

№вопрос3
Установите, в какой последовательности происходят процессы митоза:
№да
спирализация хромосом
№да

образование метафазной пластинки
№да
расхождение сестринских хроматид - анафаза
№да
деление цитоплазмы - телофаза

№вопрос4 – задания на установление соответствия

Перед тестовым заданием данной формы обязательно наличие ключевого слова **№вопрос4**.

В данном типе заданий необходимо установить соответствие между двумя списками. Первый список входит в текст самого задания. Второй список относится к ответам, при этом последовательность расположения ответов соответствует правильному соответствию по порядку перечисления.

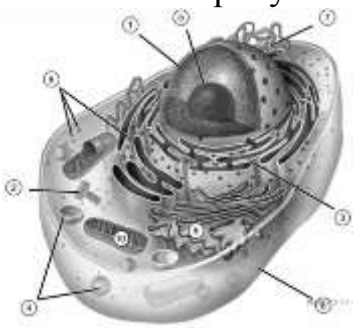
№вопрос4
<Текст задания>
№да
<правильный ответ>
№да
<правильный ответ>
№да
<правильный ответ>
№да
<правильный ответ>

Примеры задания формы 4:

№вопрос4
Установите соответствие между музеем и его местонахождением
1. Галерея Уффици
2. Центр Помпиду
3. Музей Шерлока Холмса
4. Прадо
5. Бельведер
№да
Флоренция
№да
Париж
№да
Лондон
№да
Мадрид
№да
Вена

№вопрос4
Рассмотрите рисунок и установите соответствие между цифрами,

обозначенными на рисунке и органоидами клетки:



1. 2.
2. 3.
3. 4.
4. 5.
5. 7.
6. 9.

№да
центриоли
№да
гранулярный ЭПС
№да
лизосомы
№да
ядрышко
№да
гладкий ЭПС
№да
Рибосомы

№вопрос5 – задания с кратким ответом

В данном задании предполагается только правильный ответ. Ответ должен быть представлен одним словом или числом.

№вопрос5
<Текст задания>
№да
<правильный ответ>
№ключ
<Обоснование выбора ответа. От 3 до 7 предложений>

Примеры задания формы 5

№вопрос5

Цена товара А равна 22,5, цена товара В равна 18 доход потребителя равен 400 у.е. если потребитель приобрел товаров А в 5 раз меньше, чем товаров В, то объем приобретенных товаров А составит _____. (Ответ округлите с точностью до десятых.)

№да

3,5

№ключ

Так как покупатель приобрел товаров А в 5 раз меньше, чем товара В, то $Q_b=5Q_a$.

Подставим данные значения в формулу дохода $M = Q_aP_a + Q_bP_b$. Получим $400 = 22,5Q_a + 18*5Q_a$. $Q_a = 3,5$

№вопрос5

*Моносахарид, входящий в состав ДНК, называется*_____

№да

дезоксирибоза

№ключ

Нуклеиновые кислоты это – высокомолекулярные соединения, которые являются хранителями и носителями наследственной информации. Известно два вида нуклеиновых кислот – ДНК и РНК. По своей структуре нуклеиновые кислоты являются полимерами, мономерами которых являются нуклеотиды. В состав нуклеотида входит: азотистое основание, моносахарид и остаток фосфорной кислоты. В состав ДНК входят азотистые основания: аденин, гуанин, цитозин и тимин. В состав РНК входят аденин, гуанин, цитозин и урацил, вместо тимина. Моносахаридом, входящим в состав ДНК, является дезоксирибоза, в составе РНК – рибоза. Обязательным компонентом всех нуклеиновых кислот является фосфорная кислота, через которую происходит соединение нуклеотидов в длинные линейные цепи.

№вопрос5

Мера государственного принуждения, назначаемая по приговору суда к лицу, признанному виновным в совершении преступления, заключающаяся в лишении или ограничении прав и свобод этого лица, является _____ (Введите слово в форме соответствующего падежа.)

№да

наказание

№ключ

УК РФ Статья 43. Понятие и цели наказания
1. Наказание есть мера государственного принуждения, назначаемая по приговору суда. Наказание применяется к лицу, признанному виновным в совершении преступления, и заключается в предусмотренных настоящим Кодексом лишении или ограничении прав и свобод этого лица.
2. Наказание применяется в целях восстановления социальной справедливости, а также в целях исправления осужденного и предупреждения совершения новых преступлений.

Кейс – задание обозначается с помощью ключевого слова **№вопроскейс**.

№вопроскейс

<Текст задания>

№ключ
<Ответ. ½ листа формата А4.>

Примеры оформления кейс – задания:

№вопроскейс

Прочитайте текст и запишите развернутый обоснованный ответ.

Министерство внутренних дел Российской Федерации (МВД России) является

№ключ

федеральным органом исполнительной власти, осуществляющим функции по выработке и реализации государственной политики и нормативно-правовому регулированию в сфере внутренних дел, в сфере контроля за оборотом наркотических средств, психотропных веществ и их прекурсоров, в сфере миграции (далее - сфера внутренних дел), а также правоприменительные функции по федеральному государственному контролю (надзору) в сфере внутренних дел

№вопроскейс

Прочитайте текст и запишите развернутый обоснованный ответ.

Одним из самых опасных для человека червей является эхинококк, заражение которым часто приводит к смертельному исходу.



Рассмотрите схему жизненного цикла эхинококка и ответьте, каким из хозяев является человек. Ответ поясните.

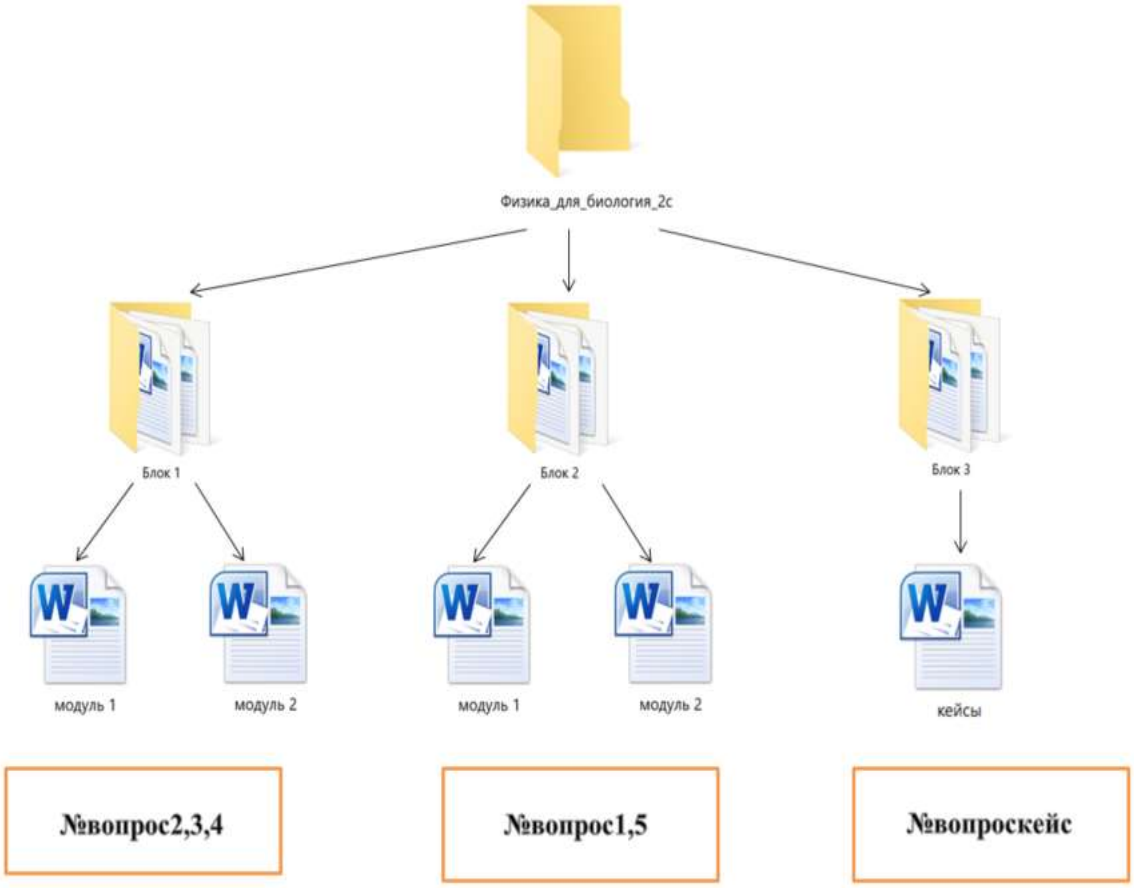
№ключ

Эхинококк - опасный паразит, не щадит человека - использует его только как промежуточного хозяина. В роли промежуточного хозяина эхинококка может выступать домашний скот (коровы, овцы, свиньи) или человек. Заражение происходит перорально. В кишечнике из яйца выходит личинка с шестью подвижными крючьями на заднем конце тела (онкосфера). С помощью крючьев она проникает через стенку кишечника в систему воротной вены и с кровью заносится в печень, реже в лёгкие, мышцы, кости или другие органы.

Человек может заразиться эхинококкозом от собак при несоблюдении правил личной гигиены, при употреблении немытых овощей, которые контаминированы яйцами эхинококка. Окончательный хозяин заражается, поедая органы (финнозное мясо) пораженных животных. Профилактика заключается в соблюдении правил гигиены, мытье рук после соприкосновения с собакой, дегельминтизации домашних животных (выведение червей из

организма).

Структура электронной базы тестовых заданий по дисциплине



Тест состоит из 16 ТЗ и 1 кейса

	Кол-во ТЗ по формам	**Время, мин**	**Кол-во баллов за 1 ТЗ**	**16 ТЗ+ 1 кейс**	**Из 100 баллов**	**За 1 ТЗ по 100 балльной шкале**
Блок 1						
№вопрос2	**4**	1-3	1	12	50	4,2
№вопрос3	**4**	1-3	1			4,2
№вопрос4	**4**	1-3	1			4,2
Блок 2						
№вопрос1	**2**	5-7	1	4	35	8,75
№вопрос5	**2**	5-7	1			8,75
Блок 3						
№вопроскейс	**1**	15	3	1	15	15,00



Требования к количеству тестовых заданий

№	*Кол-во л/ч*	*Минимальное кол-во ТЗ*	*Количество Т/З по формам**					
			Блок 1			**Блок 2**		**Блок 3**
			№вопрос2	*№вопрос3*	*№вопрос4*	*№вопрос1*	*№вопрос5*	*№вопроскейс*
1	**34-36**	**500**	**125**	**75**	**75**	**125**	**100**	**50**
2	**28-32**	**400**	**100**	**60**	**60**	**100**	**80**	**40**
3	**<26**	**300**	**75**	**45**	**45**	**75**	**60**	**30**

*№вопрос2 - 25% от общего количества ТЗ
№вопрос3 - 15% от общего количества ТЗ
№вопрос4 - 15% от общего количества ТЗ
№вопрос1 - 25% от общего количества ТЗ
№вопрос5 - 20% от общего количества ТЗ
№вопроскейс – фиксированное число

III. Примерный перечень контрольных вопросов при приеме материалов учебной практики

1. Определение реинжиниринга бизнес-процессов (РБП). Отличие от постепенных улучшений и автоматизации.
2. Ключевые принципы реинжиниринга по М. Хаммеру и Дж. Чампи.
3. Понятие «фундаментальное переосмысление» и «радикальное перепроектирование» процессов.
4. Цели и задачи реинжиниринга в эпоху цифровой трансформации.
5. Роль информационных технологий в реинжиниринге: от «автоматизации существующего» к «инновации процессов».
6. Понятие бизнес-процесса, его атрибуты (входы, выходы, владельцы, ресурсы).
7. Классификация процессов: основные, обеспечивающие, управляющие и процессы развития.
8. Нотации описания процессов: IDEF0, BPMN 2.0, EPC (ARIS). Сравнительный анализ.
9. Методология SADT (Structured Analysis and Design Technique) для информационных процессов.
10. Методы сбора информации о процессах (интервью, наблюдение, анализ журналов событий).
11. Понятие «узких мест» (bottlenecks) и потерь (waste) в информационных потоках.
12. Классическая методология проведения проекта РБП (5-6 стадий).
13. Стратегическое выравнивание: согласование целей РБП с ИТ-стратегией предприятия.
14. Подходы к реинжинирингу: «чистый лист» (clean slate) vs. целенаправленный редизайн.
15. Роль бенчмаркинга при выборе целевых показателей процессов.
16. Функционально-стоимостной анализ (ABC/ABM) в задачах реинжиниринга.
17. Критические факторы успеха (CSF) и типичные ошибки проектов РБП.
18. Понятие управления информационными процессами (УИП) и его отличие от управления ИТ-инфраструктурой.
19. Библиотека ITIL 4: ключевые концепции и «система ценности обслуживания» (SVS).
20. Управление инцидентами (Incident Management): цель, процессы, эскалация.
21. Управление проблемами (Problem Management): проактивный и реактивный подходы. Связь с инцидентами.
22. Управление изменениями (Change Management) в информационных системах: стандартный, нормальный, экстренный изменения.
23. Управление конфигурациями (CMDB): роль в отслеживании зависимостей информационных процессов.
24. Управление уровнями услуг (SLA, OLA, UC). Показатели качества ИТ-процессов.

25. Автоматизация процессов (RPA – Robotic Process Automation): сценарии применения при реинжиниринге.
26. Low-code/No-code платформы для быстрой трансформации информационных процессов.
27. Роль ERP-систем (SAP, Oracle, 1C) в реализации реинжиниринга.
28. Использование систем BPM (BPMS) для управления жизненным циклом процессов.
29. Мониторинг процессов: Process Mining (Celonis, ARIS) как инструмент выявления реальных информационных потоков.
30. Влияние искусственного интеллекта и предиктивной аналитики на управление информационными процессами.
31. Сопротивление изменениям: причины и методы преодоления при внедрении РБП.
32. Роль владельца процесса (Process Owner) и комитета по управлению изменениями.
33. Управление проектами реинжиниринга: отличия от классических ИТ-проектов (гибкость, масштаб).
34. Оценка эффективности реинжиниринга: KPI процессов (Throughput, TAT, Cost per transaction).
35. Риски реинжиниринга информационных процессов: сбои в непрерывности, утечка данных, недостижимость ROI.
36. Влияние DevOps-культуры на управление информационными процессами (CI/CD, автоматизация развертывания).
37. Реинжиниринг процессов обеспечения информационной безопасности (Security BPR).
38. Кейсы цифрового реинжиниринга в государственном секторе (Госуслуги, МФЦ).
39. Построение сквозной аналитики на базе Data Lakehouse как элемент реинжиниринга.
40. Будущее РБП: гипер-автоматизация и управление процессами в режиме реального времени.
41. Разработать план реинжиниринга процесса «Согласование заявки на доступ к данным» (выделить этапы, метрики, риски).
42. Построить диаграмму «AS-IS» (как есть) и предложить диаграмму «TO-BE» (как будет) для заданного процесса (например, обработка входящей документации).
43. Рассчитать эффект от внедрения RPA-бота для процесса выписки счетов: экономия времени, FTE, ошибки.
44. Определить тип изменения по ITIL (стандартное, нормальное, экстренное) для различных сценариев (апдейт антивируса, замена сервера, патч безопасности).
45. Проанализировать кейс провала реинжиниринга (например, Hertz или National Grid) и назвать 3 причины неудачи.

Приложение 1

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«ДАГЕСТАНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»
Факультет информатики и информационных технологий

Отчет

по учебной (ознакомительной) практике

Выполнил: студент 3 курса,
очной формы обучения
направления 09.03.03.
Прикладная информатика
Магомедов А.А.

Руководитель: к.э.н.,
доцент кафедры ИСиТП
Баммаева Г.А.

Махачкала 2026