Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Максимов Алексей Борисович
Должность: директор департамента по образовательной политике
Дата подписания: 24.10.2023 11:54:08
Уникальный программный ключ:
8db180d1a3f02ac9e60521a5672742735c18b1d6

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное образовательное учреждение высшего образования

«МОСКОВСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

УТВЕРЖДАЮ

Декан факультета информационных технологий

/Д. Г. Демидов/

«28» 04 2022 г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

«Теоретические основы информатики»

Направление подготовки

09.03.01 Информатика и вычислительная техника

Образовательная программа (профиль подготовки)

«Программное обеспечение информационных систем»

Квалификация (степень) выпускника

бакалавр

Форма обучения

заочная

Москва 2022

Программа дисциплины «Теоретические основы информатики» составлена в соответствии с требованиями ФГОС ВО и учебным планом по направлению **09.03.01 Информатика и вычислительная техника** и профилю подготовки **«Программное обеспечение информационных систем».**

Программу составил



_____/А.В.Осипов /

Программа утверждена на заседании кафедры «Прикладная информатика»
«28» августа 2022 г. протокол № 1

Заведующий кафедрой
доцент, к.э.н.  /С. В. Суворов/

Программа согласована с руководителем образовательной программы по направлению подготовки **09.03.01 Информатика и вычислительная техника** по профилю подготовки **«Программное обеспечение информационных систем».**



_____ /С. В. Суворов/
«28» августа 2022 г.

Программа утверждена на заседании учебно-методической комиссии факультета Информационных технологий

Председатель комиссии  / Д. Г. Демидов/

«____»_____2022 г. Протокол:

1. Цели освоения дисциплины

Целями изучения дисциплины является:

- знакомство студентов с основными понятиями информатики и компьютерных технологий;
- формирование представления о сущности и значении информации в развитии современного общества;
- обучение методами обобщения, анализа и восприятия информации;
- изучение технологий получения, хранения и переработки информации;
- получение студентами навыков работы с компьютером как средством управления информацией;
- знакомство с методами работы в компьютерных сетях;
- овладение методиками использования программных средств для решения практических задач.

Задачами дисциплины являются:

- воспитать культуру мышления, способность к постановке цели и выбору путей ее достижения;
- сформировать навыки разработки моделей компонентов информационных систем, включая модели данных;
- освоить навыки разработки компонентов программных комплексов и баз данных;
- изучить современные инструментальные средства и технологии программирования;
- овладеть навыками обоснования принимаемых проектных решений, осуществления постановки и выполнения экспериментов по проверке их корректности и эффективности.

2. Место дисциплины в структуре ООП бакалавриата

Дисциплина «Информатика» находится в базовой части основной образовательной программы бакалавриата (Б1.1.10).

Логически и содержательно-методически дисциплина взаимосвязана с дисциплинами: «Методы хранения и обработки информации», «Программирование», «Операционные системы», «Структуры и алгоритмы компьютерной обработки данных».

3. Перечень планируемых результатов обучения по дисциплине, соотнесенные с планируемыми результатами освоения образовательной программы

В результате освоения дисциплины у обучающихся формируются следующие компетенции и должны быть достигнуты следующие результаты обучения как этап формирования соответствующих компетенций:

Код компетенции	В результате освоения образовательной программы обучающийся должен обладать	Перечень планируемых результатов обучения по дисциплине
ОПК-6	способность работать в коллективе, толерантно воспринимая социальные, этнические, конфессиональные и культурные различия	**знать:** способы обмена информацией между компонентами программного обеспечения при работе в команде **уметь:** создавать и редактировать документы в различных офисных пакетах; использовать в профессиональной деятельности средства поиска и обмена информацией **владеть:** офисным пакетом LibreOffice, навыками работы в облачных хранилищах, публиковать документы и предоставлять доступ к ним другим пользователям
ОПК-1	способность инсталлировать программное и аппаратное обеспечение для информационных и автоматизированных систем	**знать:** виды основных операционных систем, различать операционные системы по областям их применения, способы установки драйверов и другого специализированного программного обеспечения, собирать программное обеспечение из исходных кодов **уметь:** устанавливать операционные системы, пакеты программного обеспечения, пользоваться менеджерами приложений, настраивать работу операционных систем и отдельных приложений **владеть:** навыками использования утилит (поиск информации по названию и контексту, поиск документации, файловые утилиты)
ОПК-2	способность осваивать методики использования программных средств для решения практических задач	**знать:** понятие информации и методы её измерения, хранения и переработки; теоретические основы информатики; современные тенденции развития информатики и вычислительной техники, компьютерных технологий; основы логики высказываний и предикатов; введение в теорию алгоритмов и алгоритмических языков; элементы теории сложности **уметь:** применять методики разработки и использования программного обеспечения, создавать программное обеспечение для решения конкретных практических задач, реализовывать в программном коде известные алгоритмы, а также пользоваться готовыми библиотеками **владеть:** различными методами обработки информации; методами теории алгоритмов; методами оценки качества программ; навыками работы с командной строкой ОС Linux

4. Структура и содержание дисциплины

Общая трудоемкость дисциплины составляет **4** зачетные единицы, т.е. 144 академических часа (из них 130 часов – самостоятельная работа студентов). Все они осваиваются на первом курсе в 1 семестре.

Лекции – 1 час в неделю (4 часа), лабораторные занятия – 1 час в неделю (10 часов). Форма контроля в 1 семестре – экзамен.

Содержание разделов дисциплины

Тема 1. Теоретические основы информатики
Предмет науки информатики. Информация и методы ее хранения, переработки и передачи. Алгоритмы и их основные свойства. Высказывания и предикаты. Задание синтаксиса языка предикатов с помощью нормальной формы Бэкуса-Наура (НФБН). Цепочки и деревья вывода. Задание семантики предикатов с помощью таблиц истинности. Тавтологии и основные законы эквивалентности. Методы обработки информации и свойства программ. Слабейшее предусловие. Спецификация программы и понятие правильной программы. Доказательства правильности простейших программ. Понятие о временной и ёмкостной сложностях алгоритма. Элементы теории сложности. Методы оценки временной эффективности алгоритмов.

Тема 2. Основы работы в ОС Linux. Свободно–распространяемое программное обеспечение.
Операционные системы: понятие, назначение, разновидности. Классификация типов программного обеспечения. Правовая охрана программ и данных. Свободно распространяемое программное обеспечение. Техника безопасности при работе в компьютерных классах. ОС Linux и оконные менеджеры. Хранение информации: файлы и директории, права доступа к ним. Работа с командной строкой Linux. Текущая директория командной строки. Использование встроенной документации. Утилиты pwd, ls, mv, cp, rm, mkdir, touch, ln, chmod, chown, quota, du, df, wc, man.

Тема 3. Работа с файлами: редактирование, поиск, архивирование
Текстовые файлы и их назначение. Отличие простых текстовых файлов от форматированных документов. Просмотр и редактирование простых текстовых файлов. Редакторы с подсветкой синтаксиса. Понятие о способах кодирования текста и кодировках. Смена кодировки текстового файла. Архивирование файлов и директорий. Извлечение файлов из архивов. Поиск по файловой системе. Внутритекстовый поиск. Стандартные потоки ввода-вывода командной строки Linux, перенаправление ввода-вывода, конвейер. Утилиты echo, cat, more, less, nano, vim, enca, iconv, enconv, zip, rar, tar, find, grep.

Тема 4. Локальная сеть и Интернет
Структура и принципы работы локальных и глобальных сетей. Всемирная паутина (World Wide Web) и Интернет. Понятие об едином указателе ресурсов (URL). Понятие о гипертексте и языке HTML. Каскадные таблицы стилей и язык CSS. Использование сценариев для управления динамическими страницами. Браузеры и их назначение. Поиск информации в Интернет. Электронная почта (e-mail). Сетевые операционные системы. Работа на удаленном компьютере и передача файлов по сети. Понятие о локальных и удаленных процессах. Утилиты ftp, sftp, scp, ssh, date, cal, who, ps, kill, killall.

Тема 5. Работа в текстовых процессорах
Понятие о WYSIWYG-редакторах. Текстовые процессоры. Работа с национальными символами и Unicode. Кодирование русского текста и наиболее распространенные кодировки. Знакомство с текстовым процессором LibreOffice Writer. Структура текстового документа. Таблицы, списки, формулы и изображения. Форматирование символов и абзацев, страницы. Автоматическая генерация оглавления. Форматы файлов,

используемые для хранения документов, подготовленных текстовыми процессорами. Понятие о шаблонах и стилях документа.

Тема 6. Работа в табличных процессорах
Электронные таблицы и табличные процессоры. Общие принципы работы. Относительная и абсолютная адресация. Типы данных в ячейках. Копирование формул. Использование различных формул в таблицах. Сводные таблицы и работа с несколькими листами. Диаграммы и их виды. Знакомство с табличным процессором LibreOffice Calc. Оформление электронных таблиц. Объединение ячеек, границы и заливка. Стили и условное форматирование.

Тема 7. Работа в редакторах презентаций
Правила оформления презентаций. Знакомство с редактором презентаций LibreOffice Impress. Общие принципы работы. Слайды и макеты. Стили и шаблоны оформления. Анимированные мультимедийные презентации. Время показа слайдов и различные переходы между ними. Добавление аудио и видео. Создание векторных иллюстраций на слайдах. Операции над векторными объектами. Отличие векторной графики от растровой.

5. Образовательные технологии

Методика преподавания дисциплины «Информатика» и реализация компетентностного подхода в изложении и восприятии материала предусматривает использование следующих активных и интерактивных форм проведения групповых, индивидуальных, аудиторных занятий в сочетании с внеаудиторной работой с целью формирования и развития профессиональных навыков обучающихся:
– подготовка к выполнению практических работ;
– выполнение лабораторных работ;
– использование интерактивных форм проведения занятий,
– подготовка рефератов,
– выступление с докладом,
– работа в группе над поиском нетривиальных решений,
– проведение контрольных работ,
– сдача коллоквиумов.

Удельный вес занятий, проводимых в интерактивных формах, определен образовательной программой, особенностью контингента обучающихся и содержанием дисциплины «Информатика» и в целом по дисциплине составляет 25% аудиторных занятий. Занятия лекционного типа отсутствуют.

6. Оценочные средства для текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины и учебно-методическое обеспечение самостоятельной работы студентов

В процессе обучения используются следующие оценочные формы самостоятельной работы студентов, оценочные средства текущего контроля успеваемости и промежуточных аттестаций:
– Проверка выполнения и приём рефератов
– Слушание сообщений и докладов
– Проверка знаний при работе в группе (творческие и кейс-задачи)
– Проверка выполнения контрольных работ
– Проверка знаний на коллоквиумах
– Проверка готовности студентов

6.1. Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине

6.1.1. Перечень компетенций с указанием этапов их формирования в процессе освоения образовательной программы

В результате освоения дисциплины формируются следующие компетенции:

Код компетенции	В результате освоения образовательной программы обучающийся должен обладать
ОПК-6	способность работать в коллективе, толерантно воспринимая социальные, этнические, конфессиональные и культурные различия
ОПК-1	способность инсталлировать программное и аппаратное обеспечение для информационных и автоматизированных систем
ОПК-2	способность осваивать методики использования программных средств для решения практических задач

В процессе освоения образовательной программы данные компетенции, в том числе их отдельные компоненты, формируются поэтапно в ходе освоения обучающимися дисциплин, практик в соответствии с учебным планом и календарным графиком учебного процесса.

6.1.2. Описание показателей и критериев оценивания компетенций, формируемых по итогам освоения дисциплины, описание шкал оценивания

Показателем оценивания компетенций на различных этапах их формирования является достижение обучающимися планируемых результатов обучения по дисциплине.

ОПК-6 – способность работать в коллективе, толерантно воспринимая социальные, этнические, конфессиональные и культурные различия				
Показатель	**Критерии оценивания**			
	2	**3**	**4**	**5**
знать: способы обмена информацией между компонентами программного обеспечения при работе в команде	Обучающийся демонстрирует полное незнание особенностей работы в группах	Обучающийся способен работать в группе обсуждая правильность и сложность предложенных алгоритмов	Обучающийся способен как самостоятельно, так и в группе проектировать отдельные компоненты программного обеспечения	Обучающийся способен как самостоятельно, так и в группе проектировать отдельные компоненты программного обеспечения, знает основные способы обмена информацией

уметь: создавать и редактировать документы в различных офисных пакетах; использовать в профессиональной деятельности средства поиска и обмена информацией	Обучающийся демонстрирует полную неспособность создавать документы в офисных пакетах	Обучающийся способен создать и наполнить необходимым содержимым документ, используя офисный пакет	Обучающийся способен создать и наполнить необходимым содержимым документ, используя офисный пакет, умеет пользоваться большинством специализированных функций того или иного пакета	Обучающийся способен создать и наполнить необходимым содержимым документ, используя офисный пакет, умеет пользоваться большинством специализированных функций того или иного пакета, умеет пользоваться средствами поиска и обмена информацией
владеть: офисным пакетом LibreOffice, навыками работы в облачных хранилищах, публиковать документы и предоставлять доступ к ним другим пользователям	Обучающийся демонстрирует полное незнание офисного пакета LibreOffice	Обучающийся владеет навыками использования офисного пакета LibreOffice	Обучающийся владеет навыками использования офисного пакета LibreOffice, навыками публикации офисных документов в облаке	Обучающийся владеет навыками использования офисного пакета LibreOffice, навыками публикации офисных документов в облаке, а также способов предоставления доступа к этим документам и их содержимого для других пользователей облака
ОПК-1 - способность инсталлировать программное и аппаратное обеспечение для информационных и автоматизированных систем				
знать: виды основных операционных систем, различать операционные системы по областям их применения, способы установки драйверов и другого специализированного программного обеспечения, собирать программное обеспечение из исходных кодов	Обучающийся демонстрирует полное отсутствие знаний об операционных системах	Обучающийся демонстрирует знания о разновидностях и особенностях наиболее распространенных операционных систем	Обучающийся демонстрирует знания о разновидностях и особенностях наиболее распространенных операционных систем, понимает как устанавливать драйверы и стороннее программное обеспечение	Обучающийся демонстрирует знания о разновидностях и особенностях наиболее распространенных операционных систем, понимает как устанавливать драйверы и стороннее программное обеспечение, знает как собрать программное обеспечение из исходных кодов

уметь: устанавливать операционные системы, пакеты программного обеспечения, пользоваться менеджерами приложений, настраивать работу операционных систем и отдельных приложений	Обучающийся не в состоянии установить на компьютер операционную систему	Обучающийся в состоянии установить на компьютер одну из известных операционных систем	Обучающийся в состоянии установить на компьютер одну из известных операционных систем, уверенно устанавливает необходимое программное обеспечение и настаивает его работу	Обучающийся в состоянии установить на компьютер несколько операционных систем одновременно, настроить загрузчик, уверенно устанавливает необходимое программное обеспечение и настаивает его работу
владеть: навыками использования утилит (поиск информации по названию и контексту, поиск документации, файловые утилиты)	Обучающийся не имеет навыков использования утилит	Обучающийся имеет навыки использования выборочных утилит ОС Linux	Обучающийся имеет достаточные навыки использования основных утилит ОС Linux	Обучающийся имеет достаточные навыки использования основных утилит ОС Linux, умеет искать и читать manual pages
ОПК-2 - способность осваивать методики использования программных средств для решения практических задач				
знать: понятие информации и методы её измерения, хранения и переработки; теоретические основы информатики; современные тенденции развития информатики и вычислительной техники, компьютерных технологий; основы логики высказываний и предикатов; введение в теорию алгоритмов и алгоритмических языков; элементы теории сложности	Обучающийся демонстрирует полное незнание базовых знаний программирования	Обучающийся демонстрирует знание основ программировани я, знает основы логики высказываний и предикатов	Обучающийся демонстрирует знание основ программирования , знает основы логики высказываний и предикатов, знает теорию алгоритмов, знает хотя бы один язык программирования	Обучающийся демонстрирует знание основ программирования, знает основы логики высказываний и предикатов, знает теорию алгоритмов, знает хотя бы один язык программирования, знает способы оценки сложности реализуемых алгоритмов

уметь: применять методики разработки и использования программного обеспечения, создавать программное обеспечение для решения конкретных практических задач, реализовывать в программном коде известные алгоритмы, а также пользоваться готовыми библиотеками	Обучающийся не имеет навыков создания программного обеспечения	Обучающийся имеет навыки создания программного обеспечения, способен написать завершенную программу исходя из постановки задачи	Обучающийся имеет навыки создания программного обеспечения, способен написать завершенную программу исходя из постановки задачи, умеет реализовывать пройденные алгоритмы для прикладных задач	Обучающийся имеет навыки создания программного обеспечения, способен написать завершенную программу исходя из постановки задачи, умеет реализовывать пройденные алгоритмы для прикладных задач, умеет находить и пользоваться готовыми библиотеками, встраивать вызовы библиотечных функций в собственный код
владеть: различными методами обработки информации; методами теории алгоритмов; методами оценки качества программ; навыками работы с командной строкой ОС Linux	Обучающийся не владеет методами обработки информации и теории алгоритмов	Обучающийся владеет методами обработки информации и теории алгоритмов, но может оценить качество разработанного программного кода	Обучающийся владеет методами обработки информации, теории алгоритмов и оценки качества программ	Обучающийся владеет методами обработки информации, теории алгоритмов и оценки качества программ, демонстрирует уверенную работу в терминале, использует командные вызовы утилит

Шкалы оценивания результатов промежуточной аттестации и их описание:

Форма промежуточной аттестации в первом семестре: зачет

Промежуточная аттестация обучающихся в форме зачета проводится по результатам выполнения всех видов учебной работы, предусмотренных учебным планом по данной дисциплине, при этом учитываются результаты текущего контроля успеваемости в течение семестра. Оценка степени достижения обучающимися планируемых результатов обучения по дисциплине проводится преподавателем, ведущим занятия по дисциплине методом экспертной оценки. По итогам промежуточной аттестации по дисциплине выставляется оценка «зачтено» или «незачтено».

К промежуточной аттестации допускаются только студенты, выполнившие все виды учебной работы, предусмотренные рабочей программой по дисциплине «Информатика» (которые прошли промежуточный контроль, выполнили практические работы, выступили с докладом по презентации).

Шкала оценивания	Описание
Зачтено	Выполнены все виды учебной работы, предусмотренные учебным планом. Студент демонстрирует удовлетворительное и более чем удовлетворительное соответствие знаний, умений, навыков приведенным в таблицах показателей, допускаются умеренные ошибки, проявляется достаточное наличие знаний, умений, навыков по ряду показателей, при этом студент может испытывать затруднения при оперировании знаниями и умениями при их переносе на новые ситуации.
Незачтено	Не выполнен один или более видов учебной работы, предусмотренных учебным планом. Студент демонстрирует неполное соответствие знаний, умений, навыков приведенным в таблицах показателей, допускаются значительные ошибки, проявляется отсутствие знаний, умений, навыков по ряду показателей, студент испытывает значительные затруднения при оперировании знаниями и умениями при их переносе на новые ситуации.

Фонды оценочных средств представлены в приложении 1 к рабочей программе.

7. Учебно-методическое и информационное обеспечение дисциплины

а) основная литература:

1. Гончарук С. В. Администрирование ОС Linux. Москва: Национальный Открытый Университет «ИНТУИТ», 2016. — 165 с. — Режим доступа: http://www.knigafund.ru/books/176443.

б) дополнительная литература:

1. Куль Т. П. Операционные системы: учеб. пособие — Москва: РИПО, 2015. — 312 с. — Режим доступа: http://www.knigafund.ru/books/207985.

в) программное обеспечение и интернет-ресурсы:

1. Операционная система Linux (свободное ПО)
2. Браузер MozillaFirefox (свободное ПО)
3. Офисные приложения LibreOffice для Linux (свободное ПО)
4. Офисные приложения Microsoft Office 2013(или ниже) - Microsoft Open License. Лицензия № 61984042
5. Интерпретатор языка Ruby (свободное ПО)
6. Компилляторы gcc, gcc+ (свободное ПО)
7. Система контроля версий Git (свободное ПО)
8. Справочная информация по офисному пакету LibreOffice. (https://help.libreoffice.org/Main_Page/ru)
9. Интерактивная система просмотра системных руководств проекта OpenNET. (http://www.opennet.ru/man_1.shtml)

8. Материально-техническое обеспечение дисциплины

- Компьютерные классы с ОС Linux в аудиториях АВ1201, АВ1202.
- Лекционная аудитория с проектором АВ1310.

9. Методические рекомендации для самостоятельной работы студентов

Изучение дисциплины осуществляется в строгом соответствии с целевой установкой в тесной взаимосвязи с учебным планом. Основой теоретической и

практической подготовки студентов являются лабораторные и практические занятия. При рассмотрении учебного материалы рекомендуется делать акцент на практические примеры, демонстрировать их реальную работу с помощью проектора. Полезно также сосредоточить внимание студентов на умении работать в коллективе (группе). Следует уделить внимание таким аспектам как свободно-распространяемое программное обеспечение, а так же предложить установить на домашних компьютерах альтернативную операционную систему (в связке Windows-Linux), а также офисный пакет LibreOffice.

В процессе самостоятельной работы студенты закрепляют и углубляют знания, полученные во время аудиторных занятий, а также самостоятельно изучают отдельные темы учебной программы.

Важным обстоятельством является привлечение внимания студентов к обсуждаемой проблеме, стимулирование интереса к ней и организация активного обсуждения, как структуры проблемы, так и составляющих ее наиболее актуальных тем. Для повышения эффективности проведения занятия требуется предварительная подготовка всех его участников. В этой связи рекомендуется заблаговременно (не менее, чем за неделю) оповестить студентов о теме занятия, дать перечень литературы по теме.

При проведении практического занятия преподаватель выполняет, в основном, функции ведущего – направляет студентов в правильное русло решения задач, рассматривает оптимальность предложенных решений, корректирует возможные ошибки.

Активная работа студента на практическом занятии учитывается при определении итоговой оценки его знаний по дисциплине на экзамене.

При проведении практических занятий и организации самостоятельной работы важным аспектом является организация работы студентов в группах. Это способствует развитию навыков работы в коллективе и активной коммуникации.

Проведение лабораторных работ осуществляется в виде выполнения заданий небольшими студенческими группами, где каждый студент получает возможность обрести навык выполнения той или иной задачи, а также проверить результаты своей работы в связке с партнером по команде.

При выполнении контрольных работ и индивидуальных заданий используется балльная система оценки. В ходе их выполнения студент набирает баллы за выполненные задания. Максимально студент может набрать 20 баллов. Освоение компетенций зависит от суммарного балла: 18-20 баллов - компетенции считаются освоенными на продвинутом уровне; 12-17 баллов - компетенции считаются освоенными на хорошем уровне; 8-11 баллов - компетенции считаются освоенными на базовом уровне; 0-7 баллов - компетенции считаются не освоенными. Каждая неудачная попытка сдачи уменьшает наибольшее возможное количество баллов за выполнение проекта на 1 балл.

Самостоятельная работа по дисциплине «Информатика» предполагает: выполнение студентами домашних заданий, подготовке докладов и рефератов. Домашние задания являются, как правило, продолжением практических занятий и содействуют овладению практическими навыками по основным разделам дисциплины. Самостоятельная работа студентов предполагает изучение теоретического и практического материала по актуальным вопросам дисциплины. Рекомендуется самостоятельное изучение учебной и научной литературы, использование справочной литературы и др.

При выдаче заданий на самостоятельную работу используется дифференцированный подход к студентам. Перед выполнением студентами самостоятельной внеаудиторной работы преподаватель проводит инструктаж по

выполнению задания, который включает: цель здания, его содержание, сроки выполнения, ориентировочный объем работы, основные требования к результатам работы, критерии оценки. В процессе инструктажа преподаватель предупреждает студентов о возможных типичных ошибках, встречающихся при выполнении задания. Инструктаж проводится преподавателем за счет объема времени, отведенного на изучение дисциплины.

Самостоятельная работа осуществляется индивидуально. Контроль самостоятельной работы организуется в двух формах:

- самоконтроль и самооценка студента;
- контроль со стороны преподавателей (текущий и промежуточный).

Текущий контроль осуществляется на практических занятиях, промежуточный контроль осуществляется на зачете в письменной (устной) форме.

Критериями оценки результатов самостоятельной работы студента являются:

- уровень освоения студентом учебного материала;
- умения студента использовать теоретические знания при выполнении практических задач;
- сформированность умений;
- оформление материала в соответствии с требованиями.

10. Методические указания для преподавателя

Проведение занятий по дисциплине «Информатика» осуществляется в строгом соответствии с целевой установкой и в тесной взаимосвязи с учебным планом. При рассмотрении учебных материалов рекомендуется делать акцент на практические примеры, демонстрировать их реальную работу с помощью проектора.

В процессе самостоятельной работы студенты закрепляют и углубляют знания, полученные во время аудиторных занятий, дорабатывают конспекты лекций, готовятся к зачету и экзамену, а также самостоятельно изучают отдельные темы учебной программы.

Важным обстоятельством является привлечение внимания студентов к обсуждаемой проблеме, стимулирование интереса к ней и организация активного обсуждения, как структуры проблемы, так и составляющих ее наиболее актуальных тем. Для повышения эффективности проведения занятия требуется предварительная подготовка всех его участников. В этой связи рекомендуется заблаговременно (не менее, чем за неделю) оповестить студентов о теме занятия, дать перечень литературы по теме.

При проведении практического занятия преподаватель выполняет, в основном, функции ведущего – направляет студентов в правильное русло решения задач, рассматривает оптимальность предложенных решений, корректирует возможные ошибки.

Активная работа студента на практическом занятии учитывается при определении итоговой оценки его знаний по дисциплине на экзамене.

Самостоятельная работа по дисциплине «Информатика» предполагает: выполнение студентами домашних заданий. Домашние задания являются, как правило, продолжением практических занятий и содействуют овладению практическими навыками по основным разделам дисциплины. Самостоятельная работа студентов предполагает изучение теоретического и практического материала по актуальным вопросам дисциплины. Рекомендуется самостоятельное изучение учебной и научной литературы, использование справочной литературы и др.

При выдаче заданий на самостоятельную работу используется дифференцированный подход к студентам. Перед выполнением студентами самостоятельной внеаудиторной работы преподаватель проводит инструктаж по выполнению задания, который включает: цель здания, его содержание, сроки выполнения, ориентировочный объем работы, основные требования к результатам работы, критерии оценки. В процессе инструктажа преподаватель предупреждает студентов о возможных типичных ошибках, встречающихся при выполнении задания. Инструктаж проводится преподавателем за счет объема времени, отведенного на изучение дисциплины.

Текущий контроль осуществляется на практических занятиях, промежуточный контроль осуществляется на экзамене в письменной или устной форме.

Самостоятельная работа осуществляется индивидуально. Контроль самостоятельной работы организуется в двух формах:

- самоконтроль и самооценка студента;
- контроль со стороны преподавателей (текущий и промежуточный).
- Критериями оценки результатов самостоятельной работы студента являются:
- уровень освоения студентом учебного материала;
- умения студента использовать теоретические знания при выполнении практических задач;
- сформированность умений;
- оформление материала в соответствии с требованиями.

Приложение 1 к
рабочей программе

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное образовательное учреждение высшего образования
«МОСКОВСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

Направление подготовки: 09.03.01 «Информатика и вычислительная техника»

Профиль подготовки Информатика и вычислительная техника

Форма обучения: очная

Кафедра: «Прикладная информатика»

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

ПО ДИСЦИПЛИНЕ «Информатика»

Состав: 1. Паспорт фонда оценочных средств

2. Описание оценочных средств:
тематика рефератов,

темы творческих заданий,

перечень кейс-задач,

варианты заданий контрольной работы,

вопросы к коллоквиуму,

перечень вопросов к зачету

Москва, 2022 год

Таблица 1

ПОКАЗАТЕЛЬ УРОВНЯ СФОРМИРОВАННОСТИ КОМПЕТЕНЦИЙ

Информатика					
ФГОС ВО **09.03.01 «Информатика и вычислительная техника»**					
В процессе освоения данной дисциплины студент формирует и демонстрирует следующие **общекультурные и общеобразовательные компетенции**:					
КОМПЕТЕНЦИИ		**Перечень компонентов**	**Технология формирования компетенций**	**Форма оценочного средства****	**Степени уровней освоения компетенций**
ИН-ДЕКС	**ФОРМУЛИРОВКА**				
ОПК-6	способность работать в коллективе, толерантно воспринимая социальные, этнические, конфессиональные и культурные различия	**знать:** способы обмена информацией между компонентами программного обеспечения при работе в команде **уметь:** создавать и редактировать документы в различных офисных пакетах; использовать в профессиональной деятельности средства поиска и обмена информацией **владеть:** офисным пакетом LibreOffice, навыками работы в облачных хранилищах, публиковать документы и предоставлять доступ к ним другим пользователям	лекция, самостоятельная работа, семинарские занятия	К-З, К, К/Р, Р, ТЗ, З	**Базовый уровень** работать в коллективе при решении практических задач **Повышенный уровень** способен быть лидером команды при решении практических задач

ОПК-1	способность инсталлировать программное и аппаратное обеспечение для информационных и автоматизированных систем	**знать:** виды основных операционных систем, различать операционные системы по областям их применения, способы установки драйверов и другого специализированного программного обеспечения, собирать программное обеспечение из исходных кодов **уметь:** устанавливать операционные системы, пакеты программного обеспечения, пользоваться менеджерами приложений, настраивать работу операционных систем и отдельных приложений **владеть:** навыками использования утилит (поиск информации по названию и контексту, поиск документации, файловые утилиты)	лекция, самостоятельная работа	К-З, К, К/Р, Р, ТЗ, З	**Базовый уровень:** способность инсталлировать программное и аппаратное обеспечение **Повышенный уровень:** способен решать проблемы, возникающие при инсталляции программного и аппаратного обеспечения

ОПК-2	способность осваивать методики использования программных средств для решения практических задач	**знать:** понятие информации и методы её измерения, хранения и переработки; теоретические основы информатики; современные тенденции развития информатики и вычислительной техники, компьютерных технологий; основы логики высказываний и предикатов; введение в теорию алгоритмов и алгоритмических языков; элементы теории сложности **уметь:** применять методики разработки и использования программного обеспечения, создавать программное обеспечение для решения конкретных практических задач, реализовывать в программном коде известные алгоритмы, а также пользоваться готовыми библиотеками **владеть:** различными методами обработки информации; методами теории алгоритмов; методами оценки качества программ; навыками работы с командной строкой ОС Linux	лекция, самостоятельная работа, семинарские занятия	К-З, К, К/Р, Р, ТЗ З	**Базовый уровень** умеет применять методики, изученные в ходе изучения дисциплины **Повышенный уровень** способен самостоятельно изучить методики, необходимые для решения новых современных практических задач

Перечень оценочных средств по дисциплине «Информатика»

№ ОС	Наименование оценочного средства	Краткая характеристика оценочного средства	Представление оценочного средства в ФОС
1	Зачет (З)	Средство контроля, позволяющее оценить знания, умения и владения обучающегося по учебной дисциплине по двухбальной системе. Рекомендуется для оценки знаний, умений и владений студентов.	Комплект вопросов к зачету
2	Кейс-задача (К-З)	Проблемное задание, в котором обучающемуся предлагают осмыслить реальную профессионально-ориентированную ситуацию, необходимую для решения данной проблемы.	Задания для решения кейс-задачи
3	Коллоквиум (К)	Средство контроля усвоения учебного материала темы, раздела или разделов дисциплины, организованное как учебное занятие в виде собеседования педагогического работника с обучающимися.	Вопросы по темам/разделам дисциплины
4	Контрольная работа (К/Р)	Средство проверки умений применять полученные знания для решения задач определенного типа по теме или разделу	Комплект контрольных заданий по вариантам
5	Реферат (Р)	Продукт самостоятельной работы студента, представляющий собой краткое изложение в письменном виде полученных результатов теоретического анализа определенной научной (учебно- исследовательской) темы, где автор раскрывает суть исследуемой проблемы, приводит различные точки зрения, а также собственные взгляды на нее.	Темы рефератов
6	Творческое задание (ТЗ)	Частично регламентированное задание, имеющее нестандартное решение и позволяющее диагностировать умения, интегрировать знания различных областей, аргументировать собственную точку зрения. Может выполняться в индивидуальном порядке или группой обучающихся.	Темы групповых и/или индивидуальных творческих заданий

Примерный перечень вопросов к зачету по курсу «Информатика»:

1. Предмет науки информатика. Алгоритмы и их основные свойства.
2. Основные понятия информатики.
3. Принцип построения ЭВМ.
4. Основные поколения и этапы развития ЭВМ.
5. Общие определения грамматики и языка, ей порождённого. Задание грамматики с помощью нормальной формы Бэкуса-Наура (НФБН). Примеры.
6. Задание синтаксиса языка предикатов с помощью нормальной формы Бэкуса-Наура (НФБН). Цепочки и деревья вывода. Примеры.
7. Задание семантики предикатов с помощью таблиц истинности. Тавтологии, законы эквивалентности и примеры доказательств.
8. Расширение понятия предиката. Ограничения на использование кванторов в предикатах. Построение отрицаний к предикатам. Примеры.
9. Спецификация программы {Q} S {R}. Слабейшее предусловие wp(S,R) и связь между Q и wp(S,R). Взаимосвязь между предикатами и множествами, ими задаваемыми. Понятие правильной программы. Примеры.
10. Определение простейших операторов языка Ruby в терминах wp: пустой оператор, оператор выхода, последовательное выполнение операторов. Примеры.
11. Определение простейших операторов языка Ruby в терминах wp: оператор простого и множественного присваивания. Примеры.
12. Массивы с функциональной точки зрения, понятие вырезки. Определение оператора присваивания элементу массива. Примеры.
13. Определение условного оператора в терминах wp. Примеры.
14. Базисные схемы обработки информации. Рекурсия и итерация, их взаимосвязь. Методы построения и доказательства правильности рекурсивных программ. Принцип математической индукции. Примеры.
15. Математическая модель итерации. Спецификация задачи на итерацию. Три способа конкретизации общей модели итерации. Идея и основные определения схемы проектирования цикла при помощи инварианта. Примеры.
16. Схема вычисления инвариантной функции и её связь со схемой проектирования цикла при помощи инварианта. Примеры.
17. Индуктивные функции на пространстве последовательностей. Критерий индуктивности. Схема вычисления индуктивной функции. Примеры.
18. Схема проектирования цикла при помощи инварианта. Условия правильности цикла. Примеры написания программ и доказательств их правильности.
19. Основные методы построения инвариантов цикла. Теория воздушного шарика. Метод устранения конъюнктивного члена. Примеры.
20. Основные методы построения инвариантов цикла. Замена константы переменной и расширение области значений переменной. Примеры.
21. Функции на пространстве последовательностей. Их стационарные значения и доопределения. Способы построения индуктивного расширения и обобщённая схема вычисления индуктивной функции. Примеры.
22. Минимальное индуктивное расширение. Формулировка теоремы о единственности минимального индуктивного расширения. Основной практический результат теории индуктивных функций.
23. Каноническое индуктивное расширение и доказательство теоремы существования и единственности минимального индуктивного расширения.
24. Критерий минимальности (с доказательством). Примеры минимальных индуктивных расширений.

25. Основные понятия объектно-ориентированного программирования. Классы и объекты. Объекты и переменные, как ссылки на них. Методы и их вызов. Операторный синтаксис. Методы экземпляра и класса.
26. Языки и грамматики. Выводимые цепочки и деревья вывода. Язык правильных арифметических формул и его задание различными грамматиками.
27. Операционные системы: понятие, назначение, разновидности.
28. Основные достоинства и недостатки операционной системы Linux.
29. Какие основные черты свободного программного обеспечения отличают его от проприетарного?
30. Какие есть достоинства у программного обеспечения с открытым исходным кодом?
31. Особенности лицензии GPL.
32. Файловые системы и файлы. Типы файлов.
33. Система прав доступа к файлам и директориям в ОС Linux.
34. Работа с файловой системой в командной строке Linux.
35. Перечислите основные технические средства, используемые в настоящее время для долговременного хранения информации.
36. Что такое компьютерные вирусы и какие средства используются для борьбы с ними?
37. Принцип кодирования обычных текстовых файлов. Смена кодировки.
38. Просмотр и редактирование простых текстовых файлов. Редакторы с подсветкой синтаксиса.
39. Работа с архивами в командной строке Linux.
40. Поиск по файловой системе. Внутритекстовый поиск.
41. Стандартные потоки ввода-вывода командной строки Linux, перенаправление ввода-вывода, конвейер.
42. Всемирная паутина (World Wide Web) и Интернет.
43. Понятие об едином указателе ресурсов (URL).
44. Понятие о гипертексте и языке HTML.
45. Каскадные таблицы стилей и язык CSS.
46. Использование сценариев для управления динамическими страницами.
47. Браузеры и их назначение. Поиск информации в Интернет. Электронная почта (e-mail).
48. Сетевые операционные системы. Понятие о локальных и удаленных процессах.
49. Работа на удаленном компьютере (ssh) и передача файлов по сети (ftp, scp, sftp).
50. Понятие о WYSIWYG-редакторах.
51. Различные офисные пакеты. Их достоинства и недостатки.
52. Создание документов сложной структуры. Структура текстового документа.
53. Структура и принципы работы локальных и глобальных сетей.
54. Создание документов сложной структуры. Форматирование символов и абзацев, страниц.
55. Создание документов сложной структуры. Колонтитулы.
56. Создание документов сложной структуры. Включение графики, таблиц и формул.
57. Создание документов сложной структуры. Оформление математических формул.
58. Создание документов сложной структуры. Стили.
59. Электронные таблицы. Общие принципы работы.
60. Электронные таблицы. Относительная и абсолютная адресация. Копирование формул.
61. Электронные таблицы. Сводные таблицы и работа с несколькими листами.
62. Электронные таблицы. Диаграммы и их виды.
63. Электронные таблицы. Оформление электронных таблиц. Границы и заливка.
64. Электронные таблицы. Объединение ячеек. Типы ячеек. Условное форматирование.
65. Форматы графических файлов. Особенности растровой и векторной графики.
66. Базовые примитивы векторной графики.
67. Редакторы презентаций. Условия комфортного восприятия слайдов.
68. Редакторы презентаций. Макеты слайдов.
69. Редакторы презентаций. Стили.

70. Редакторы презентаций. Время показа слайдов и различные переходы между ними.
71. Редакторы презентаций. Добавление аудио и видео.
72. Форматы файлов, используемые для хранения документов, созданных в офисных пакетах.

Примерная тематика рефератов по курсу «Информатика»

1. Основные понятия информатики.
2. Информатизация и компьютеризация общества.
3. Информационные системы и технологии.
4. Аппаратное и программное обеспечение компьютера.
5. Информационно-логические основы построения ЭВМ.
6. Принцип построения ЭВМ.
7. Функциональная и структурная организация ЭВМ.
8. Персональные компьютеры и микропроцессоры.
9. Запоминающие устройства персональных компьютеров.
10. Периферийные устройства персональных компьютеров.
11. Компьютерные сети.

Критерии оценки:

К оформлению работы выдвигается ряд требований. Документ должен использовать единый стиль оформления, иметь автоматически генерируемое оглавление, правильно оформленный список литературы, колонтитулы (название и ФИО, нумерация страниц, титульный лист не содержит номера), таблицы и списки, вставленные изображения. В соответствии с проделанной работой студент получает оценку от 0 до 9 баллов. Освоение компетенций зависит от полученной оценки: 8-9 баллов — компетенции считаются освоенными на продвинутом уровне; 4-7 — компетенции считаются освоенными на базовом уровне; 0-3 — компетенции считаются не освоенными.

Темы творческих заданий

1. Исследование движения мяча, брошенного под углом к горизонту.
2. Исследование колебаний пружинного маятника.
3. Исследование характеристик тока в электрической цепи.
4. Моделирование процесса радиоактивного распада атомов образца.
5. Моделирование дисперсии света при прохождении через границу двух сред.
6. Расчет стоимости оптовых закупок со скидкой.
7. Расчет прибыли фирмы с учетом налогообложения.
8. Расчет потребления и стоимости электроэнергии.
9. Расчет калорийности пищи и процента от нормы ежедневного потребления.
10. Расчет выплат по кредиту.

Критерии оценки:

Исследование необходимо провести с помощью электронных таблиц. Таблицы должны быть аккуратно оформлены, а начальные условия эксперимента должны легко поддаваться изменениям, так что все остальные величины автоматически и корректно пересчитываются. Результаты работы должны быть проиллюстрированы графиками и/или диаграммами. В соответствии с проделанной работой студент получает оценку от 0 до 9 баллов. Освоение компетенций зависит от полученной оценки: 8-9 баллов — компетенции считаются освоенными на продвинутом уровне; 4-7 — компетенции считаются освоенными на базовом уровне; 0-3 — компетенции считаются не освоенными.

Кейс-задача

«Я пользователь ПК»

Не так давно Вы устроились в компьютерную фирму в должности программиста, и сегодня Ваш первый рабочий день. Вы осваиваетесь на рабочем месте и включаете свой персональный компьютер. Спустя некоторое время перед Вами появляется заставка операционной системы Scientific Linux, а чуть позже — приглашение войти в систему. Вы вспоминаете и вводите собственные логин и пароль, после чего попадаете на экран рабочего стола. Осмотревшись, Вы замечаете, что в операционной системе используется окружение рабочего стола GNOME, а среди программ присутствуют браузер Firefox, офисный пакет LibreOffice, графические редакторы GIMP и Inkscape, средство мгновенного обмена сообщениями Pidgin, почтовое приложение Evolution и текстовый редактор gedit.

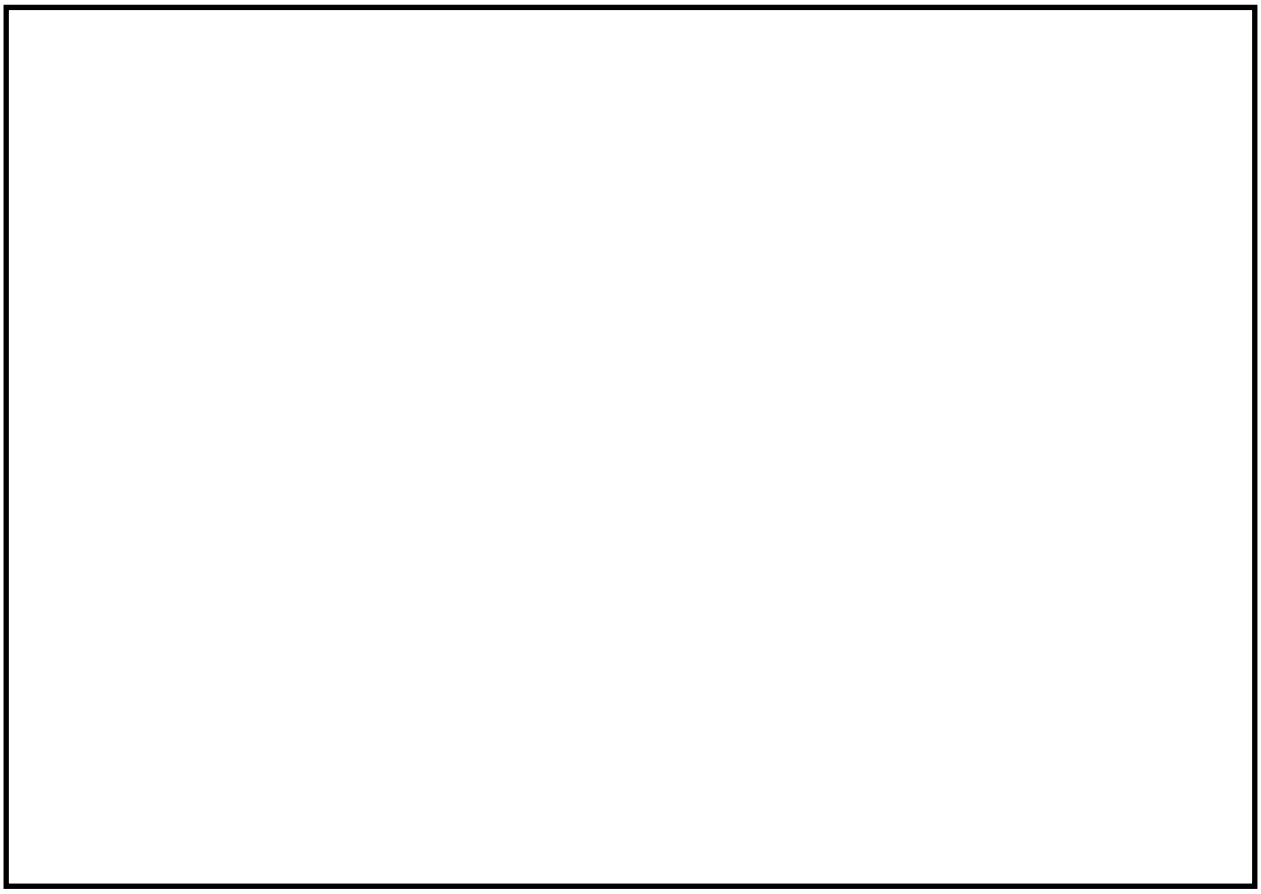
Задание №1. Вы желаете создать на рабочем столе ярлыки для программ, которые, по Вашему мнению, могли бы Вам пригодиться в дальнейшей профессиональной деятельности. Может быть Вы желаете создать ярлыки для некоторых программ, которые не удалось сейчас обнаружить. (Их готов для Вас установить Ваш системный администратор.) Ярлыки каких программ Вы желаете увидеть на Вашем рабочем столе?

Ответ:

- _____
- _____
- _____
- _____
- _____
- _____
- _____
- _____

После того, как на рабочем столе наведен порядок, Вы запускаете файловый менеджер и обнаруживаете, что Ваша домашняя директория пуста.

Задание №2. Вам необходимо подумать и решить, какую структуру каталогов Вы организуете внутри собственной домашней директории. Постарайтесь учесть специфику Вашей профессии и подумайте, с какими документами и другими файлами Вам придется иметь дело. Решите, как можно было бы удобно распределить данные файлы по различным каталогам. Не забудьте, что для некоторых директорий может быть логичным организовать групповой доступ для совместной работы с коллегами. Изобразите, какую структуру каталогов Вы создадите в Вашей домашней директории. Для каждой директории подпишите ее назначение и права доступа к ней.



Ответ:

Сделав перерыв на обед, Вы возвращаетесь на свое рабочее место и читаете электронную почту. Вам пришло письмо с заданием: необходимо подключиться к разработке проекта под кодовым названием «Полиэдр».

Задание №3. Разработку ведет программист Василий, и проект хранится в его директории ~pve13/polyhedron. Вы запускаете командную строку и вводите команду, которая позволяет Вам скопировать к себе проект «Полиэдр». Что это за команда?

Ответ: _____

Задание №4. Вместе с проектом скопировались и временные файлы, оставленные текстовым редактором, который использует Василий. Все такие файлы заканчиваются на специальный символ «~». С помощью какой команды Вы удалите все временные файлы в Вашей копии проекта?

Ответ: _____

Задание №5. Вы желаете оценить приблизительный объем проекта. Конкретно Вас интересует количество строк в файлах исходного кода, имеющих расширение «.rb». Все такие файлы находятся в поддиректории src директории проекта. Вложенных в src директорий нет. С помощью какой команды Вы определите суммарное число строк во всех файлах исходного кода?

Ответ: _____

Перед тем, как продолжить работу над проектом, Вы решаете немного отдохнуть и собраться с мыслями. Сегодняшний день — Ваш первый день на новом рабочем месте. Вашей первоочередной задачей было освоиться и подготовить свое рабочее пространство для дальнейшей профессиональной деятельности. И Вы с этой задачей успешно справились. Впереди предстоит еще немалая работа, но к ней Вы вернетесь позже…

Критерии оценки:

Данная кейс-задача содержит в себе несколько заданий различной сложности, не имеющий однозначного решения и требующих в большей или меньшей степени творческого подхода. Для каждой задачи следует оценивать не только правильность ответа, но и его полноту и оригинальность. В соответствии с проделанной работой студент получает суммарную оценку от 0 до 9 баллов. Освоение компетенций зависит от полученной оценки: 8-9 баллов — компетенции считаются освоенными на продвинутом уровне; 4-7 — компетенции считаются освоенными на базовом уровне; 0-3 — компетенции считаются не освоенными.

Варианты заданий контрольной работы

Данная контрольная содержит 5 вариантов с различными заданиями. В зависимости от размеров группы можно сформировать достаточное число вариантов, используя различные комбинации заданий из первых 5-ти вариантов.

Вариант 1

1. С помощью какой команды командной строки Linux можно найти в текущей директории (и только в ней, без учета поддиректорий) все файлы архивов RAR и ZIP (определять по расширению файла), созданные позже чем двое суток назад?
2. С помощью какой команды командной строки Linux можно изменить кодировку всех текстовых файлов (расширение «.txt») в текущей директории на UTF-8?
3. С помощью какой команды командной строки Linux можно определить имена всех файлов в текущей директории, содержащих в своем тексте произвольные числа?
4. С помощью какой команды командной строки Linux можно создать сжатый по алгоритму GZIP TAR-архив, который должен содержать файлы a, b и c?
5. С помощью какой команды командной строки Linux можно объединить файлы 1, 2 и 3 в один файл с именем 4?

Вариант 2

1. С помощью какой команды командной строки Linux можно найти в текущей директории и всех ее поддиректориях все файлы архивов TAR, сжатых по алгоритму GZIP (определять по расширению файла), созданные не позже чем двое суток назад?
2. С помощью какой команды командной строки Linux можно изменить кодировку всех текстовых файлов (расширение «.html») в текущей директории на CP1251?
3. С помощью какой команды командной строки Linux можно определить имена всех файлов в текущей директории, содержащих в своем тексте произвольные даты в формате ДД.ММ.ГГГГ? (Допускается предположить, что документы не содержат «неправильных» дат, например, 32.02.0205.)
4. С помощью какой команды командной строки Linux можно создать сжатый по алгоритму BZIP2 TAR-архив, который должен содержать директорию d?
5. С помощью какой команды командной строки Linux можно просмотреть постранично содержимое сразу трех файлов 1, 2 и 3?

Вариант 3

1. С помощью какой команды командной строки Linux можно найти в текущей директории (и только в ней, без учета поддиректорий) все директории, название которых содержит подстроку «dir»?
2. С помощью какой команды командной строки Linux можно изменить кодировку всех текстовых файлов (расширение «.tex») в текущей директории на KOI8-R?
3. С помощью какой команды командной строки Linux можно определить имена всех файлов в текущей директории, содержащих в своем тексте названия дней недели в именительном падеже?
4. С помощью какой команды командной строки Linux можно создать сжатый по алгоритму ZIP архив, который должен содержать директории d1 и d2?

5. С помощью какой команды командной строки Linux можно вывести на стандартный поток вывода содержимое сразу трех файлов 1, 2 и 3?

Вариант 4

1. С помощью какой команды командной строки Linux можно найти в текущей директории (и ее поддиректориях не более второго уровня вложения) все файлы (не директории), размер которых превышает 5Мб?
2. С помощью какой команды командной строки Linux можно изменить кодировку всех текстовых файлов (расширение «.rb») в текущей директории на UTF-16?
3. С помощью какой команды командной строки Linux можно определить имена всех файлов в текущей директории, содержащих в своем тексте слова длиннее 10 букв?
4. С помощью какой команды командной строки Linux можно создать сжатый по алгоритму RAR архив, который должен содержать файлы a, b и c директории d?
5. С помощью какой команды командной строки Linux можно дописать в конец файла f строку end?

Вариант 5

1. С помощью какой команды командной строки Linux можно найти в текущей директориях текущей директории (но не в ней самой) все обычные файлы и директории (исключая файлы специальных типов), редактирование которых разрешено только владельцу?
2. С помощью какой команды командной строки Linux можно изменить кодировку всех текстовых файлов (расширение «.c») в текущей директории на KOI8-U?
3. С помощью какой команды командной строки Linux можно определить имена всех файлов в текущей директории, которые не содержат знаков русской пунктуации? (Можно считать, что в файлах используются только символы, которые можно найти на обычной клавиатуре.)
4. С помощью какой команды командной строки Linux можно распаковать архив ar.rar в директорию ar?
5. С помощью какой команды командной строки Linux можно просмотреть содержимое сразу трех файлов 1, 2 и 3 в режиме прокрутки?

Критерии оценки:

В соответствии с проделанной работой студент получает оценку от 0 до 9 баллов. Освоение компетенций зависит от полученной оценки: 8-9 баллов — компетенции считаются освоенными на продвинутом уровне; 4-7 — компетенции считаются освоенными на базовом уровне; 0-3 — компетенции считаются не освоенными.

Вопросы к коллоквиуму

На коллоквиуме студенту задается два теоритических вопроса.

1. Предмет науки информатика. Алгоритмы и их основные свойства.
2. Основные понятия информатики.
3. Принцип построения ЭВМ.
4. Общие определения грамматики и языка, ей порождённого. Задание грамматики с помощью нормальной формы Бэкуса-Наура (НФБН). Примеры.
5. Задание синтаксиса языка предикатов с помощью нормальной формы Бэкуса-Наура (НФБН). Цепочки и деревья вывода. Примеры.
6. Задание семантики предикатов с помощью таблиц истинности. Тавтологии, законы эквивалентности и примеры доказательств.
7. Расширение понятия предиката. Ограничения на использование кванторов в предикатах. Построение отрицаний к предикатам. Примеры.
8. Спецификация программы {Q} S {R}. Слабейшее предусловие wp(S,R) и связь между Q и wp(S,R). Взаимосвязь между предикатами и множествами, ими задаваемыми. Понятие правильной программы. Примеры.
9. Определение простейших операторов языка Ruby в терминах wp: пустой оператор, оператор выхода, последовательное выполнение операторов. Примеры.
10. Определение простейших операторов языка Ruby в терминах wp: оператор простого и множественного присваивания. Примеры.
11. Массивы с функциональной точки зрения, понятие вырезки. Определение оператора присваивания элементу массива. Примеры.
12. Определение условного оператора в терминах wp. Примеры.
13. Базисные схемы обработки информации. Рекурсия и итерация, их взаимосвязь. Методы построения и доказательства правильности рекурсивных программ. Принцип математической индукции. Примеры.
14. Математическая модель итерации. Спецификация задачи на итерацию. Три способа конкретизации общей модели итерации. Идея и основные определения схемы проектирования цикла при помощи инварианта. Примеры.
15. Схема вычисления инвариантной функции и её связь со схемой проектирования цикла при помощи инварианта. Примеры.
16. Индуктивные функции на пространстве последовательностей. Критерий индуктивности. Схема вычисления индуктивной функции. Примеры.
17. Схема проектирования цикла при помощи инварианта. Условия правильности цикла. Примеры написания программ и доказательств их правильности.
18. Основные методы построения инвариантов цикла. Теория воздушного шарика. Метод устранения конъюнктивного члена. Примеры.
19. Основные методы построения инвариантов цикла. Замена константы переменной и расширение области значений переменной. Примеры.
20. Функции на пространстве последовательностей. Их стационарные значения и доопределения. Способы построения индуктивного расширения и обобщённая схема вычисления индуктивной функции. Примеры.
21. Минимальное индуктивное расширение. Формулировка теоремы о единственности минимального индуктивного расширения. Основной практический результат теории индуктивных функций.

22. Каноническое индуктивное расширение и доказательство теоремы существования и единственности минимального индуктивного расширения.
23. Критерий минимальности (с доказательством). Примеры минимальных индуктивных расширений.
24. Основные понятия объектно-ориентированного программирования. Классы и объекты. Объекты и переменные, как ссылки на них. Методы и их вызов. Операторный синтаксис. Методы экземпляра и класса.
25. Языки и грамматики. Выводимые цепочки и деревья вывода. Язык правильных арифметических формул и его задание различными грамматиками.
26. Основные поколения и этапы развития ЭВМ.
27. Операционные системы: понятие, назначение, разновидности.
28. Основные достоинства и недостатки операционной системы Linux.
29. Какие основные черты свободно распространяемого программного обеспечения отличают его от коммерческого?
30. Какие есть достоинства и программного обеспечения с открытым исходным кодом?
31. Особенности лицензии GPL.
32. Файловые системы и файлы. Типы файлов.
33. Система прав доступа к файлам и директориям в ОС Linux.
34. Работа с файловой системой в командной строке Linux.
35. Перечислите основные технические средства, используемые в настоящее время для долговременного хранения информации.
36. Что такое компьютерные вирусы и какие средства используются для борьбы с ними?
37. Принцип кодирования обычных текстовых файлов. Смена кодировки.
38. Просмотр и редактирование простых текстовых файлов. Редакторы с подсветкой синтаксиса.
39. Работа с архивами в командной строке Linux.
40. Поиск по файловой системе. Внутритекстовый поиск.
41. Стандартные потоки ввода-вывода командной строки Linux, перенаправление ввода-вывода, конвейер.
42. Структура и принципы работы локальных и глобальных сетей.
43. Всемирная паутина (World Wide Web) и Интернет.
44. Понятие об едином указателе ресурсов (URL).
45. Понятие о гипертексте и языке HTML.
46. Каскадные таблицы стилей и язык CSS.
47. Использование сценариев для управления динамическими страницами.
48. Браузеры и их назначение. Поиск информации в Интернет. Электронная почта (e-mail).
49. Сетевые операционные системы. Понятие о локальных и удаленных процессах.
50. Работа на удаленном компьютере (ssh) и передача файлов по сети (ftp, scp, sftp).

Критерии оценки:

В соответствии с проделанной работой студент получает оценку от 0 до 2 баллов. Освоение компетенций зависит от полученной оценки: 2 балла — компетенции считаются освоенными на продвинутом уровне; 1 балл — компетенции считаются освоенными на базовом уровне; 0 — компетенции считаются не освоенными.

Приложение2
к рабочей программе

Структура и содержание дисциплины «Информатика»
по направлению подготовки 09.03.01 «Информатика и вычислительная техника» (Бакалавр)

п/п	Раздел	Семестр	Неделя семестра	Виды учебной работы, включая самостоятельную работу студентов, и трудоемкость в часах					Виды самостоятельной работы студентов					Формы аттестации	
				Л	П/С	Лаб	СРС	КСР	К.Р.	К.П.	РГР	Реферат	К/р	Э	З
	Первый семестр														
	Теоретические основы информатики	1	1	1	2		20								
	Основы работы в ОС Linux. Свободно распространяемое программное обеспечение	1	2-3	1	2		20	+							
	Работа с файлами: редактирование, поиск, архивирование	1	4-6	1	2		20						+		
	Локальная сеть и Интернет	1	7-9	1	1		20						+		
	Работа в текстовых процессорах	1	10-11	0	1		20					+			
	Работа в табличных процессорах	1	12-14	0	1		20	+							
	Работа в редакторах презентаций	1	15-17	0	1		10	+							
	Форма аттестации		***18-21***												+
	Всего часов по дисциплине в первом семестре			**4**	**10**		**130**								