

Документ подписан простой электронной подписью

Информация о владельце:

ФИО: Агафонов Александр Викторович

Должность: директор филиала

Дата подписания: 19.06.2025 15:33:26

Уникальный программный ключ: 2534

2534

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ

«МОСКОВСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»
ЧЕБОКСАРСКИЙ ИНСТИТУТ (ФИЛИАЛ) МОСКОВСКОГО ПОЛИТЕХНИЧЕСКОГО УНИВЕРСИТЕТА

**Кафедра Информационных технологий
и систем управления**



РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

«Функциональное и логическое программирование»

(наименование дисциплины)

| | |
|-------------------------------------|--|
| Направление подготовки | 09.03.01 «Информатика и вычислительная техника» (код и наименование направления подготовки) |
| Направленность (профиль) подготовки | «Программное обеспечение вычислительной техники и автоматизированных систем» (наименование профиля подготовки) |
| Квалификация выпускника | бакалавр |
| Форма обучения | очная, заочная |
| Год начала обучения | 2025 |

Рабочая программа дисциплины разработана в соответствии с:

- Федеральный государственный образовательный стандарт высшего образования - бакалавриат по направлению подготовки 09.03.01 – Информатика и вычислительная техника, утвержденный приказом Министерства науки и высшего образования Российской Федерации № 929 от 19 сентября 2017 г. зарегистрированный в Минюсте 10 октября 2017 года, рег. номер 48489 (далее – ФГОС ВО).

- учебным планом (очной, заочной форм обучения) по специальности 09.03.01 – Информатика и вычислительная техника.

Рабочая программа дисциплины включает в себя оценочные материалы для проведения текущего контроля успеваемости и промежуточной аттестации по дисциплине (п.8 Фонд оценочных средств для проведения текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины)

Автор Ковалёв Сергей Васильевич, кандидат технических наук, доцент
кафедры информационных технологий и систем управления

(указать ФИО, ученую степень, ученое звание или должность)

Программа одобрена на заседании кафедры Информационных технологий и систем управления (протокол № 8 от 12.04.2025г).

1. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы (Цели освоения дисциплины)

1.1. Целями освоения дисциплины «Функциональное и логическое программирование» являются: формирование у студентов профессиональных знаний и практических навыков по разработке и созданию моделей и систем с помощью языков функционального и логического программирования.

Задачами освоения дисциплины «Функциональное и логическое программирование» являются: получение знаний об основных понятиях связанных с изучением общих концепций и методов современного декларативного программирования и, в частности, таких его разновидностей, как функциональное и логическое программирование, позволяющих эффективно решать задачи, связанные с обработкой символьной информации, не типизированных данных, построения систем поддержки принятия решения, искусственного интеллекта, а также экспертных систем.

1.2. Области профессиональной деятельности и(или) сферы профессиональной деятельности, в которых выпускники, освоившие программу, могут осуществлять профессиональную деятельность:

Об Связь, информационные и коммуникационные технологии (в сфере проектирования, разработки, внедрения и эксплуатации средств вычислительной техники и информационных систем, управления их жизненным циклом).

1.3. К основным задачам изучения дисциплины относится подготовка обучающихся к выполнению трудовых функций в соответствии с профессиональными стандартами:

| Код и наименование профессионального стандарта | Обобщенные трудовые функции | | | Трудовые функции | | |
|---|-----------------------------|---|----------------------|---|--------|-----------------------------------|
| | код | наименование | уровень квалификации | наименование | код | уровень (подуровень) квалификации |
| 06.001 Программист Профессиональный стандарт «Программист», утвержденный приказом Министерством труда и социальной защиты Российской Федерации от 18.11.2013 № 679н (зарегистрирован в Министерстве труда и социальной защиты Российской Федерации от 18 декабря 2013 г. №30635) | D | Разработка требований и проектирование программного обеспечения | 6 | Анализ требований к программному обеспечению | D/01.6 | 6 |
| | | | 6 | Разработка технических спецификаций на программные компоненты и их взаимодействие | D/02.6 | |
| | | | 6 | Проектирование программного обеспечения | D/03.6 | |
| 06.028 Системный | A | Разработка | 6 | Разработка | A/01.6 | 6 |

| Код и наименование профессионального стандарта | Обобщенные трудовые функции | | | Трудовые функции | | |
|---|-----------------------------|--|----------------------|--|--------|-----------------------------------|
| | код | наименование | уровень квалификации | наименование | код | уровень (подуровень) квалификации |
| программист Профессиональный стандарт «Системный программист», утв. Приказом Министерством труда и социальной защиты РФ от 29 сентября 2020 года N 678н | | компонентов системных программных продуктов | 6 | драйверов устройств | | |
| | | | | Разработка компиляторов, загрузчиков, сборщиков | A/02.6 | 6 |
| | | | | Разработка системных утилит | A/03.6 | 6 |
| | | | | Создание инструментальных средств программирования | A/04.6 | 6 |
| 06.015 Специалист по информационным системам Профессиональный стандарт «Специалист по информационным системам», утв. приказом Министерством труда и социальной защиты РФ 18 ноября 2014 г. №896н | С | Выполнение работ и управление работами по созданию (модификации) и сопровождению ИС, автоматизирующих задачи организационного управления и бизнес-процессы | 6 | Разработка модели бизнес-процессов заказчика | C/08.6 | 6 |
| | | | | Выявление требований к ИС | C/11.6 | 6 |
| | | | | Анализ требований | C/12.6 | 6 |
| | | | | Разработка архитектуры ИС | C/14.6 | 6 |
| | | | | Проектирование и дизайн ИС | C/16.6 | 6 |
| | | | | Разработка баз данных ИС | C/17.6 | 6 |
| | | | | Организационное и технологическое обеспечение кодирования на языках программирования | C/18.6 | 6 |
| | | | | Организационное и технологическое обеспечение модульного тестирования ИС (верификации) | C/19.6 | 6 |
| | | | | Создание пользовательской документации к ИС | C/22.6 | 6 |
| | | | | Организация репозитория хранения данных о создании (модификации) и вводе ИС в эксплуатацию | C/40.6 | 6 |

1.4. Компетенции обучающегося, формируемые в результате освоения дисциплины

| Наименование категории (группы) компетенций | Код и наименование компетенций | Код и наименование индикатора достижения | Перечень планируемых результатов обучения |
|---|--------------------------------|--|---|
|---|--------------------------------|--|---|

| | | компетенции | |
|---|--|---|---|
| <p>Разработка, отладка, модификация и поддержка системного программного обеспечения</p> | <p>ПК-3. Способен разрабатывать компоненты системных программных продуктов</p> | <p>ПК-3.1 Разрабатывает драйверы устройств</p> | <p><i>на уровне знаний:</i> знать языки функционального и логического программирования; основные методы и средства эффективной разработки программного продукта. <i>на уровне умений:</i> уметь использовать методы и технологии разработки для генерации исполняемого кода. <i>на уровне навыков:</i> владеть основными методологиями процессов разработки программного обеспечения.</p> |
| | | <p>ПК-3.2 Разрабатывает компиляторы, загрузчики, сборщики</p> | <p><i>на уровне знаний:</i> знать типовые роли в процессе разработки программного обеспечения. <i>на уровне умений:</i> уметь анализировать поставленные задачи, уметь разрабатывать алгоритмы, представлять знания для решения поставленных задач. <i>на уровне навыков:</i> владеть математическим аппаратом, применяемым в функциональном и логическом программировании.</p> |
| | | <p>ПК-3.3 Разрабатывает системные утилиты</p> | <p><i>на уровне знаний:</i> знать методологии разработки программного обеспечения. знать математические основы лямбда- исчисления, предикатов первого порядка. <i>на уровне умений:</i> уметь разрабатывать модели различных классов систем с применением языков функционального и логического программирования. уметь программировать на языках Лисп и Пролог. <i>на уровне навыков:</i> владеть логическим программированием. языками Лисп и Пролог для построения моделей искусственного интеллекта.</p> |
| | | <p>ПК-3.4. Создает инструментальные средства программирования</p> | <p><i>на уровне знаний:</i> знать языки функционального и логического программирования, их ключевые концепции и подходы. основные методы и инструменты для эффективной разработки программного обеспечения. <i>на уровне умений:</i> уметь анализировать поставленные задачи и разрабатывать модели различных типов систем с</p> |

| | | |
|--|--|---|
| | | использованием языков функционального и логического программирования. <i>на уровне навыков:</i> владеть основными методологиями процесса разработки программного обеспечения, языками программирования Лисп и Пролог. |
|--|--|---|

2. Место дисциплины в структуре ОПОП

Дисциплина Б1.Д(М).В.1 «Функциональное и логическое программирование» реализуется в рамках части, формируемой в рамках части, формируемой участниками образовательных отношений Блока 1 «Элективные дисциплины (модули)» программы бакалавриата.

Дисциплина «Функциональное и логическое программирование» преподается обучающимся по очной форме обучения – в 6-м семестре, по заочной форме – в 6-м семестре.

Дисциплина «Функциональное и логическое программирование» является промежуточным этапом формирования компетенций ПК-3 процессе освоения ОПОП.

Дисциплина «Функциональное и логическое программирование» основывается на знаниях, умениях и навыках, приобретенных при изучении дисциплин Перспективы развития информатики и вычислительной техники, Введение в информатику, Учебная практика: ознакомительная практика, Объектно-ориентированное программирование, Программирование для мобильных устройств, Структуры и алгоритмы обработки данных, Криптографические методы защиты информации, Защита информации, Учебная практика: научно-исследовательская работа (получение первичных навыков научно-исследовательской работы), Учебная практика: технологическая (проектно-технологическая) практика, Микропроцессорные устройства систем управления, Интернет-технологии, Интернет-программирование и является предшествующей для изучения дисциплин Системное программирование, Государственная итоговая аттестация: подготовка к сдаче и сдача государственного экзамена, Государственная итоговая аттестация: выполнение и защита выпускной квалификационной работы

Формой промежуточной аттестации знаний обучающихся по очной форме обучения является экзамен во 6-м семестре, по заочной форме экзамен на 6-м семестре.

3. Объем дисциплины

очная форма обучения:

| Вид учебной работы по дисциплине | Всего в з.е. и часах | Семестр 6 в часах |
|---|---------------------------|----------------------|
| Общая трудоёмкость дисциплины | 5 з.е. -180 ак.час | 180 ак.час |
| <i>Контактная работа - Аудиторные занятия</i> | 73 | 73 |
| <i>Лекции</i> | 36 | 36 |

| | | |
|--|------------------|------------------|
| <i>Лабораторные занятия</i> | 36 | 36 |
| <i>Семинары, практические занятия</i> | - | - |
| <i>Консультация</i> | 1 | 1 |
| <i>Самостоятельная работа</i> | 71 | 71 |
| Курсовая работа (курсовой проект) | - | - |
| Вид промежуточной аттестации | Экзамен-36 часов | Экзамен-36 часов |

заочная форма обучения:

| Вид учебной работы по дисциплине | Всего в з.е. и часах | Семестр 6 в часах |
|--|-----------------------------|--------------------------|
| Общая трудоёмкость дисциплины | 5 з.е. -180 ак.час | 180 ак.час |
| <i>Контактная работа - Аудиторные занятия</i> | 17 | 17 |
| <i>Лекции</i> | 6 | 6 |
| <i>Лабораторные занятия</i> | 10 | 10 |
| <i>Семинары, практические занятия</i> | - | - |
| <i>Консультация</i> | 1 | 1 |
| <i>Самостоятельная работа</i> | 154 | 154 |
| Курсовая работа (курсовой проект) | - | - |
| Вид промежуточной аттестации | Экзамен-9 часов | Экзамен-9 часов |

4. Содержание дисциплины, структурированное по темам (разделам) дисциплины с указанием их объемов (в академических часах) и видов учебных занятий

4.1. Учебно-тематический план

Очная форма обучения

| Тема (раздел) | Распределение часов | | | Самостоятельная работа | Формируемые компетенции (код) |
|--|---------------------|----------------------|----------------------|------------------------|----------------------------------|
| | Лекции | Лабораторные занятия | Практические занятия | | |
| Тема 1. Основные элементы языка ПРОЛОГ. Согласование целевых утверждений в ПРОЛОГе. Рекурсивное представление данных и программ в ПРОЛОГе. | 8 | 8 | - | 17 | ПК-3.1, ПК-3.2 ПК-3.3, ПК-3.4 |
| Тема 2. Декларативная и операционная семантика. Constraint-технология в логическом программировании | 8 | 8 | - | 18 | ПК-3.1, ПК-3.2 ПК-3.3, ПК-3.4 |
| Тема 3. Основы функциональной парадигмы. Строго функциональный язык. | 10 | 10 | - | 18 | ПК-3.1, ПК-3.2 ПК-3.3, ПК-3.4 |
| Тема 4. Представление и интерпретация функциональных программ | 10 | 10 | | 18 | ПК-3.1, ПК-3.2 ПК-3.3, ПК-3.4 |
| Консультации | 1 | | | - | ПК-3.1, ПК-3.2 ПК-3.3, ПК-3.4 |
| Контроль (экзамен) | 36 | | | | ПК-3.1, ПК-3.2 ПК-3.3, ПК-3.4 |

| | | | |
|--------------|-----------|-----------|--|
| ИТОГО | 73 | 71 | |
|--------------|-----------|-----------|--|

Заочная форма обучения

| Тема (раздел) | Распределение часов | | | Самостоятельная работа | Формируемые компетенции (код) |
|--|---------------------|----------------------|----------------------|------------------------|----------------------------------|
| | Лекции | Лабораторные занятия | Практические занятия | | |
| Тема 1. Основные элементы языка ПРОЛОГ. Согласование целевых утверждений в ПРОЛОГе. Рекурсивное представление данных и программ в ПРОЛОГе. | 2 | 2 | - | 38 | ПК-3.1, ПК-3.2 ПК-3.3, ПК-3.4 |
| Тема 2. Декларативная и операционная семантика. Constraint-технология в логическом программировании | 2 | 2 | - | 38 | ПК-3.1, ПК-3.2 ПК-3.3, ПК-3.4 |
| Тема 3. Основы функциональной парадигмы. Строго функциональный язык. | - | 4 | - | 38 | ПК-3.1, ПК-3.2 ПК-3.3, ПК-3.4 |
| Тема 4. Представление и интерпретация функциональных программ | 2 | 2 | - | 40 | ПК-3.1, ПК-3.2 ПК-3.3, ПК-3.4 |
| Консультации | 1 | | | - | ПК-3.1, ПК-3.2 ПК-3.3, ПК-3.4 |
| Контроль (экзамен) | 9 | | | | ПК-3.1, ПК-3.2 ПК-3.3, ПК-3.4 |
| ИТОГО | 17 | | | 154 | |

4.2 Содержание дисциплины

Тема 1. Основные элементы языка ПРОЛОГ. Согласование целевых утверждений в ПРОЛОГе. Рекурсивное представление данных и программ в ПРОЛОГе

Понятие языка ПРОЛОГ и его основные элементы. Структура программ на языке ПРОЛОГ.

Согласование целевых утверждений и фактов в ПРОЛОГе. Процесс обратного вывода и его значение.

Рекурсивное представление данных в ПРОЛОГе: списки, деревья и другие структуры данных.

Применение рекурсии в программировании на ПРОЛОГе. Примеры рекурсивных предикатов.

Тема 2. Декларативная и операционная семантика. Constraint-технология в логическом программировании

Понятие декларативной и операционной семантики в программировании. Сравнение декларативного и императивного подходов.

Constraint-технология в логическом программировании: основные

принципы и применение.

Методы ограничения в ПРОЛОГе и их использование для решения задач.

Примеры применения constraint-технологий в логическом программировании.

Тема 3. Основы функциональной парадигмы. Строго функциональный язык.

Понятие функциональной парадигмы и её основные характеристики. Отличия функционального программирования от других парадигм.

Строго функциональный язык: примеры и особенности. Применение чистых функций и неизменяемых данных.

Лямбда-выражения и функциональные комбинации. Роль высших функций в функциональном программировании.

Тема 4. Представление и интерпретация функциональных программ

Структура функциональных программ и их интерпретация. Основные компоненты функциональных программ.

Методы компиляции и интерпретации функциональных языков. Примеры реализации интерпретаторов функциональных языков.

Применение функциональных программ для решения задач и их эффективность. Сравнение с императивными подходами.

5. Учебно-методическое обеспечение самостоятельной работы студентов

Самостоятельная работа проводится с целью: систематизации и закрепления полученных теоретических знаний и практических умений обучающихся; углубления и расширения теоретических знаний студентов; формирования умений использовать нормативную, правовую, справочную документацию, учебную и специальную литературу; развития познавательных способностей и активности обучающихся: творческой инициативы, самостоятельности, ответственности, организованности; формирование самостоятельности мышления, способностей к саморазвитию, совершенствованию и самоорганизации; формирования профессиональных компетенций; развитию исследовательских умений студентов.

Формы и виды самостоятельной работы студентов: чтение основной и дополнительной литературы – самостоятельное изучение материала по рекомендуемым литературным источникам; работа с библиотечным каталогом, самостоятельный подбор необходимой литературы; работа со словарем, справочником; поиск необходимой информации в сети Интернет; конспектирование источников; реферирование источников; составление аннотаций к прочитанным литературным источникам; составление рецензий и отзывов на прочитанный материал; составление обзора публикаций по теме; составление и разработка терминологического словаря; составление хронологической таблицы; составление библиографии (библиографической картотеки); подготовка к различным формам текущей и промежуточной аттестации (к тестированию, контрольной работе, экзамену); выполнение домашних контрольных работ; самостоятельное выполнение практических

заданий репродуктивного типа (ответы на вопросы, задачи, тесты; выполнение творческих заданий).

Технология организации самостоятельной работы обучающихся включает использование информационных и материально-технических ресурсов образовательного учреждения: библиотеку с читальным залом, компьютерные классы с возможностью работы в Интернет; аудитории (классы) для консультационной деятельности.

Перед выполнением обучающимися внеаудиторной самостоятельной работы преподаватель проводит консультирование по выполнению задания, который включает цель задания, его содержания, сроки выполнения, ориентировочный объем работы, основные требования к результатам работы, критерии оценки. Во время выполнения обучающимися внеаудиторной самостоятельной работы и при необходимости преподаватель может проводить индивидуальные и групповые консультации.

Контроль самостоятельной работы студентов предусматривает: соотнесение содержания контроля с целями обучения; объективность контроля; валидность контроля (соответствие предъявляемых заданий тому, что предполагается проверить); дифференциацию контрольно-измерительных материалов.

Формы контроля самостоятельной работы: просмотр и проверка выполнения самостоятельной работы преподавателем; организация самопроверки, взаимопроверки выполненного задания в группе; обсуждение результатов выполненной работы на занятии; проведение письменного опроса; проведение устного опроса; организация и проведение индивидуального собеседования; организация и проведение собеседования с группой.

Перечень вопросов, отводимых на самостоятельное освоение дисциплины, формы внеаудиторной самостоятельной работы

| Наименование тем (разделов) дисциплины | Перечень вопросов, отводимых на самостоятельное освоение | Формы внеаудиторной самостоятельной работы |
|---|--|---|
| <p>Тема 1. Основные элементы языка ПРОЛОГ. Согласование целевых утверждений в ПРОЛОГе. Рекурсивное представление данных и программ в ПРОЛОГе.</p> | <ol style="list-style-type: none"> 1. Определение основных элементов языка ПРОЛОГ. 2. Принципы согласования целевых утверждений в ПРОЛОГе. 3. Как работает механизм backtracking в ПРОЛОГе. 4. Рекурсивное представление данных: примеры и их применение. 5. Как реализовать рекурсивные предикаты в ПРОЛОГе. 6. Сравнение рекурсивного и итеративного подходов в ПРОЛОГе. 7. Примеры использования списков в ПРОЛОГе. 8. Как обрабатывать ошибки и исключения в ПРОЛОГе. 9. Влияние порядка правил на согласование в ПРОЛОГе. 10. Примеры сложных структур данных и их рекурсивное представление. | <p>Разработка небольшой программы на языке ПРОЛОГ, демонстрирующей использование рекурсивных предикатов и обработку списков. Написание краткого отчета о реализации и анализе кода.</p> |
| <p>Тема 2. Декларативная и операционная</p> | <ol style="list-style-type: none"> 1. Определение декларативной и операционной семантики. 2. Сравнение декларативного и процедурного подходов в программировании. | <p>Исследование и написание реферата о принципах работы декларативной</p> |

| | | |
|--|--|--|
| семантика. Constraint-технология в логическом программировании | <ol style="list-style-type: none"> 3. Принципы работы с constraint-технологией. 4. Как реализовать ограничения в логическом программировании. 5. Примеры использования constraint-технологий в практике. 6. Влияние семантики на разработку программ. 7. Как декларативный подход упрощает решение задач. 8. Примеры задач, решаемых с помощью constraint-технологий. 9. Как проверять корректность моделей в логическом программировании. 10. Применение декларативных языков в современных системах. | семантики и ее применении в современных языках программирования. Разработка небольшого проекта, использующего constraint-технологии. |
| Тема 3. Основы функциональной парадигмы. Строго функциональный язык. | <ol style="list-style-type: none"> 1. Определение функциональной парадигмы и ее основные характеристики. 2. Принципы работы строго функциональных языков. 3. Как осуществляется передача аргументов в функциональных языках. 4. Роль функций высшего порядка в функциональном программировании. 5. Примеры использования лямбда-выражений. 6. Как реализовать каррирование функций. 7. Сравнение функционального и императивного подходов. 8. Как обрабатывать состояния и побочные эффекты в функциональных языках. 9. Примеры применения функциональных языков в реальных задачах. 10. Как функциональная парадигма влияет на производительность программ. | Разработка проекта на строго функциональном языке, включающего функции высшего порядка и каррирование. Анализ кода на предмет ошибок и оптимизации. |
| Тема 4. Представление и интерпретация функциональных программ | <ol style="list-style-type: none"> 1. Как представляются функциональные программы в памяти. 2. Принципы интерпретации функциональных программ. 3. Сравнение различных подходов к интерпретации. 4. Роль компиляции и интерпретации в функциональных языках. 5. Как реализовать механизмы ленивой оценки. 6. Примеры представления функциональных выражений. 7. Как работает механизм рекурсии в интерпретаторах. 8. Влияние использования оптимизаций на производительность. 9. Как отлаживать функциональные программы. 10. Примеры интерпретаторов функциональных языков. | Написание реферата о методах интерпретации функциональных программ и их сравнении. Разработка простого интерпретатора для функционального языка с анализом его работы. |

Шкала оценивания

| Шкала оценивания | Критерии оценивания |
|---------------------|---|
| «Отлично» | Обучающийся глубоко и содержательно раскрывает тему самостоятельной работы, не допустив ошибок. Ответ носит развернутый и исчерпывающий характер |
| «Хорошо» | Обучающийся в целом раскрывает тему самостоятельной работы, однако ответ хотя бы на один из них не носит развернутого и исчерпывающего характера |
| «Удовлетворительно» | Обучающийся в целом раскрывает тему самостоятельной работы и допускает ряд неточностей, фрагментарно раскрывает содержание теоретических вопросов или их раскрывает |

| | |
|-----------------------|---|
| | содержательно, но допуская значительные неточности. |
| «Неудовлетворительно» | Обучающийся не владеет выбранной темой самостоятельной работы |

6. Оценочные материалы (фонд оценочных средств) для проведения текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины

6.1. Паспорт фонда оценочных средств

| № | Контролируемые разделы (темы) дисциплины | Код и наименование компетенции | Индикатор достижения компетенции | Наименование оценочного средства |
|----|--|---|---|---------------------------------------|
| 1. | Тема 1. Основные элементы языка ПРОЛОГ. Согласование целевых утверждений в ПРОЛОГе. Рекурсивное представление данных и программ в ПРОЛОГе. | ПК-3. Способен разрабатывать компоненты системных программных продуктов | ПК-3.1 Разрабатывает драйверы устройств ПК-3.2 Разрабатывает компиляторы, загрузчики, сборщики ПК-3.3 Разрабатывает системные утилиты ПК-3.4. Создает инструментальные средства программирования | Опрос, тест, доклад, реферат, экзамен |
| 2. | Тема 2. Декларативная и операционная семантика. Constraint-технология в логическом программировании | ПК-3. Способен разрабатывать компоненты системных программных продуктов | ПК-3.1 Разрабатывает драйверы устройств ПК-3.2 Разрабатывает компиляторы, загрузчики, сборщики ПК-3.3 Разрабатывает системные утилиты ПК-3.4. Создает инструментальные средства программирования | Опрос, тест, доклад, реферат, экзамен |
| 3. | Тема 3. Основы функциональной парадигмы. Строго функциональный язык. | ПК-3. Способен разрабатывать компоненты системных программных продуктов | ПК-3.1 Разрабатывает драйверы устройств ПК-3.2 Разрабатывает компиляторы, загрузчики, сборщики ПК-3.3 Разрабатывает системные утилиты ПК-3.4. Создает инструментальные средства программирования | Опрос, тест, доклад, реферат, экзамен |
| 4. | Тема 4. Представление и интерпретация функциональных программ | ПК-3. Способен разрабатывать компоненты системных программных продуктов | ПК-3.1 Разрабатывает драйверы устройств ПК-3.2 Разрабатывает компиляторы, загрузчики, сборщики | Опрос, тест, доклад, реферат, экзамен |

| | | | | |
|--|--|--|--|--|
| | | | ПК-3.3 Разрабатывает системные утилиты ПК-3.4. Создает инструментальные средства программирования | |
|--|--|--|--|--|

Этапы формирования компетенций в процессе освоения ОПОП прямо связаны с местом дисциплин в образовательной программе. Каждый этап формирования компетенции, характеризуется определенными знаниями, умениями и навыками и (или) опытом профессиональной деятельности, которые оцениваются в процессе текущего контроля успеваемости, промежуточной аттестации по дисциплине (практике) и в процессе итоговой аттестации.

Дисциплина «Функциональное и логическое программирование» является промежуточным этапом комплекса дисциплин, в ходе изучения которых у студентов формируются компетенции ПК-3.

Формирования компетенции ПК-3 начинается с изучения дисциплин «Перспективы развития информатики и вычислительной техники», «Введение в информатику», Учебная практика: ознакомительная практика, «Объектно-ориентированное программирование», «Программирование для мобильных устройств», «Структуры и алгоритмы обработки данных», «Криптографические методы защиты информации», «Защита информации», Учебная практика: научно-исследовательская работа (получение первичных навыков научно-исследовательской работы), Учебная практика: технологическая (проектно-технологическая) практика, «Микропроцессорные устройства систем управления», «Интернет-технологии», «Интернет-программирование».

Завершается работа по формированию у студентов указанных компетенций в ходе дисциплины «Системное программирование».

Итоговая оценка сформированности компетенций ПК-3 определяется в период Государственная итоговая аттестация: подготовка к сдаче и сдача государственного экзамена, Государственная итоговая аттестация: выполнение и защита выпускной квалификационной работы

В процессе изучения дисциплины, компетенции также формируются поэтапно.

Основными этапами формирования ПК-3 при изучении дисциплины Б1.Д(М).В.1 «Функциональное и логическое программирование» является последовательное изучение содержательно связанных между собой тем учебных занятий. Изучение каждой темы предполагает овладение студентами необходимыми дескрипторами (составляющими) компетенций. Для оценки уровня сформированности компетенций в процессе изучения дисциплины предусмотрено проведение текущего контроля успеваемости по темам (разделам) дисциплины и промежуточной аттестации по дисциплине – экзамен.

6.2. Контрольные задания и материалы, необходимые для оценки знаний, умений и навыков и (или) опыта деятельности, характеризующих

этапы формирования компетенций в процессе освоения образовательной программы

6.2.1. Контрольные вопросы по темам (разделам) для опроса на занятиях

| Тема (раздел) | Вопросы |
|--|---|
| Тема 1. Основные элементы языка ПРОЛОГ. Согласование целевых утверждений в ПРОЛОГе. Рекурсивное представление данных и программ в ПРОЛОГе. | ПК-3 1. Основные синтаксические конструкции языка ПРОЛОГ. 2. Предикаты в ПРОЛОГе и их использование. 3. Объясните, как осуществляется согласование (unification) в ПРОЛОГе. 4. Работа переменных в ПРОЛОГе и их особенности. 5. Факты и правила, и их взаимодействие в ПРОЛОГе. 6. Пример использования рекурсии для обработки списков в ПРОЛОГе. 7. Реализация условных конструкций в ПРОЛОГе. 8. Основные ошибки, которые могут возникнуть при использовании рекурсивных предикатов. 9. Динамические базы данных в ПРОЛОГе и их применение. 10. Использование встроенных предикатов для работы с данными в ПРОЛОГе. |
| Тема 2. Декларативная и операционная семантика. Constraint-технология в логическом программировании | ПК-3 1. Основные характеристики декларативной семантики в логическом программировании. 2. Принцип работы операционной семантики в ПРОЛОГе. 3. Логические выводы для реализации декларативного подхода. 4. Ограничения в контексте ПРОЛОГа. 5. Типы ограничений и их применение в логическом программировании. 6. Разница между декларативным и операционным подходами к программированию. 7. Преимущества использования декларативного подхода в программировании. 8. Роль семантики в тестировании и отладке логических программ. |
| Тема 3. Основы функциональной парадигмы. Строго функциональный язык. | ПК-3 1. Ключевые концепции функционального программирования. 2. Каррирование и его применение в функциональных языках. 3. Принципы работы с функциями высшего порядка. 4. Ленивые вычисления и их реализация в функциональных языках. 5. Основные особенности строгих функциональных языков программирования. 6. Примеры использования анонимных функций в |

| | |
|---|---|
| | <p>функциональном программировании.</p> <p>7. Реализация структуры данных в строгих функциональных языках.</p> <p>8. Преимущества и недостатки строгих функциональных языков по сравнению с другими парадигмами.</p> <p>9. Механизмы обработки ошибок и исключений в функциональных языках.</p> |
| Тема 4. Представление и интерпретация функциональных программ | <p>ПК-3</p> <p>1. Структура типичной функциональной программы.</p> <p>2. Основные этапы компиляции функциональных программ.</p> <p>3. Семантический анализ и его реализация в интерпретаторах.</p> <p>4. Основные методы оптимизации функциональных программ.</p> <p>5. Роль компиляции в повышении производительности функциональных программ.</p> <p>6. Интерпретация функциональных программ отличается от компиляции.</p> <p>7. Механизм замыкания в функциональном программировании.</p> |

Шкала оценивания ответов на вопросы

| Шкала оценивания | Критерии оценивания |
|-----------------------|--|
| «Отлично» | Обучающийся глубоко и содержательно раскрывает ответ на каждый теоретический вопрос, не допустив ошибок. Ответ носит развернутый и исчерпывающий характер, может обосновать свои суждения, применить знания на практике, привести необходимые примеры не только по учебнику, но и самостоятельно составленные; |
| «Хорошо» | Обучающийся в целом раскрывает теоретические вопросы, однако ответ хотя бы на один из них не носит развернутого и исчерпывающего характера. |
| «Удовлетворительно» | Обучающийся в целом раскрывает теоретические вопросы и допускает ряд неточностей, фрагментарно раскрывает содержание теоретических вопросов или их раскрывает содержательно, но допуская значительные неточности. |
| «Неудовлетворительно» | Обучающийся не знает ответов на поставленные теоретические вопросы. |

6.2.2. Темы для докладов

| Тема (раздел) | Вопросы |
|--|--|
| Тема 1. Основные элементы языка ПРОЛОГ. Согласование целевых утверждений в ПРОЛОГе. Рекурсивное представление данных и программ в ПРОЛОГе. | <p>ПК-3</p> <p>1. Основные конструкции языка ПРОЛОГ и их применение в логическом программировании.</p> <p>2. Механизм согласования (unification) в ПРОЛОГе: принципы и примеры.</p> <p>3. Рекурсия в ПРОЛОГе: как представлять и обрабатывать рекурсивные структуры данных.</p> <p>4. Списки в ПРОЛОГе: создание, обработка и применение в</p> |

| | |
|--|---|
| | <p>программировании.</p> <ol style="list-style-type: none"> 5. Использование предикатов для реализации логических задач в ПРОЛОГе. 6. Применение арифметических выражений в ПРОЛОГе: синтаксис и семантика. 7. Динамические базы данных в ПРОЛОГе: работа с изменяемыми данными. 8. Как использовать предикаты для создания сложных логических структур в ПРОЛОГе. 9. Примеры создания и использования модулей в ПРОЛОГе для организации кода. 10. Разработка простых игр и головоломок с использованием ПРОЛОГа. |
| <p>Тема 2. Декларативная и операционная семантика. Constraint-технология в логическом программировании</p> | <p>ПК-3</p> <ol style="list-style-type: none"> 1. Сравнение декларативной и операционной семантики: влияние на разработку программ. 2. Принципы работы с ограничениями (constraint) в логическом программировании. 3. Примеры применения constraint-технологий в решении задач оптимизации. 4. Декларативные языки программирования и их применение в современных системах. 5. Применение декларативного подхода в разработке программного обеспечения. 6. Основы логического программирования на примерах: как декларативный подход упрощает решения. 7. Методики реализации constraint-решателей: алгоритмы и примеры. 8. Применение логического программирования в области искусственного интеллекта и обработки знаний. |
| <p>Тема 3. Основы функциональной парадигмы. Строго функциональный язык.</p> | <p>ПК-3</p> <ol style="list-style-type: none"> 1. Определение функциональной парадигмы и ее основные характеристики. 2. Принципы работы строго функциональных языков: особенности и преимущества. 3. Использование функций высшего порядка и лямбда-выражений в функциональном программировании. 4. Сравнение функциональной и императивной парадигм: ключевые отличия и преимущества каждого подхода. 5. Принципы чистоты функций в строгих функциональных языках: без побочных эффектов. 6. Как реализуются алгебраические типы данных в функциональных языках. 7. Примеры использования рекурсии и хвостовой рекурсии в функциональном программировании. 8. Сравнительный анализ функциональных языков программирования: Haskell, Scala, F#. |
| <p>Тема 4. Представление и интерпретация функциональных программ</p> | <p>ПК-3</p> <ol style="list-style-type: none"> 1. Как представляются функциональные программы в памяти: внутренние структуры и механизмы. 2. Принципы интерпретации функциональных программ: алгоритмы и методы. 3. Ленивые вычисления в функциональном программировании: |

| | |
|--|--|
| | <p>преимущества и примеры использования.</p> <p>4. Оптимизация функциональных программ: методы и подходы.</p> <p>5. Механизмы компиляции и интерпретации функциональных языков: ключевые отличия.</p> <p>6. Применение оптимизаций в интерпретаторах функциональных языков: примеры и методы.</p> <p>7. Как реализовать механизмы обработки исключений в функциональных программах.</p> <p>8. Примеры реальных приложений на функциональных языках: от веб-разработки до анализа данных.</p> |
|--|--|

Шкала оценивания

| Шкала оценивания | Критерии оценивания |
|-----------------------|--|
| «Отлично» | Обучающийся глубоко и содержательно раскрывает тему доклада, не допустив ошибок. Ответ носит развернутый и исчерпывающий характер. |
| «Хорошо» | Обучающийся в целом раскрывает тему доклада, однако ответ хотя бы на один из них не носит развернутого и исчерпывающего характера. |
| «Удовлетворительно» | Обучающийся в целом раскрывает тему доклада и допускает ряд неточностей, фрагментарно раскрывает содержание теоретических вопросов или их раскрывает содержательно, но допуская значительные неточности. |
| «Неудовлетворительно» | Обучающийся не владеет выбранной темой |

6.2.3. Оценочные средства остаточных знаний (тест)

ПК-3

1. Данные, которые принимаются за истину и не требуют доказательства, в ПРОЛОГе называются:

- 1) предикатами
- 2) правилами
- 3) доказательствами
- 4) фактами

2. Части правила вывода отделяются знаком ...

- 1) :-
- 2) -
- 3) *
- 4) =

3. В ПРОЛОГе знак «;» означает ...

- 1) Логическое И
- 2) Логическое ИЛИ
- 3) Логическое НЕТ
- 4) Эквиваленция

4. Внутренние запросы ПРОЛОГа ...

- 1) Вводятся в окне диалога
- 2) Записываются в текст программы

- 3) Вызываются через меню
 - 4) Записываются в массив
-
5. Арность предиката - ...
 - 1) Правило его использования
 - 2) Его условное обозначение
 - 3) Количество его аргументов
 - 4) Конструкция
-
6. Какая конструкция не имеет смысла в ПРОЛОГе?
 - 1) $X=10$
 - 2) $10=X$
 - 3) $X=X+10$
 - 4) $\text{mod}(X,3)$
-
7. Секция описания типов в ПРОЛОГе называется
 - 1) DOMAINS
 - 2) TYPES
 - 3) PREDICATES
 - 4) GOAL
-
8. Поиск с возвратом в ПРОЛОГе включает ...
 - 1) Предикат fail
 - 2) Предикат !
 - 3) Предикат repeat
 - 4) Предикат keysort
-
9. Процедура, вызывающая сама себя, называется
 - 1) возвратной
 - 2) циклической
 - 3) рекурсивной
 - 4) повторяющейся
-
10. Список в ПРОЛОГе заключается в ...
 - 1) круглые скобки
 - 2) квадратные скобки
 - 3) фигурные скобки
-
11. Языком функционального программирования является ...
 - 1) ЛИСП
 - 2) ПРОЛОГ
 - 3) DELPHI
-
12. В ЛИСПе основной тип данных - ...
 - 1) дерево

- 2) список
- 3) символ
- 4) стек

13. Константа nil в ЛИСПе означает:

- 1) истина
- 2) нуль
- 3) ложь

14. Символы, числа и константы в ЛИСПе называются ...

- 1) атомами
- 2) элементами
- 3) объектами

15. В ЛИСПе имя функции и ее аргументы образуют ...

- 1) массив
- 2) дерево
- 3) список
- 4) хэш-таблицу

16. Множество всех связей между элементами совокупности образует ...

- 1) правило
- 2) квантор
- 3) предикат
- 4) отношение

17. Имя свойства или отношения между объектами с последовательностью аргументов называется

- 1) правилом
- 2) квантором
- 3) предикатом
- 4) идентификатором

18. Логическая операция, ограничивающая область истинности какого-либо предиката, называется ...

- 1) правилом
- 2) квантором
- 3) отношением
- 4) функцией

19. Формальный язык, позволяющий высказывать общие точные утверждения о свойствах классов (множеств) объектов - это ...

- 1) язык логики предикатов
- 2) язык логики классов объектов
- 3) метаязык

4) логический язык программирования

20. Утверждение (факт), принимаемое истинным без доказательства - это ...

- 1) правило
- 2) предикат
- 3) теорема
- 4) аксиома

21. Утверждение, требующее доказательства, называется ...

- 1) предикатом
- 2) правилом вывода
- 3) фактом
- 4) доменом

22. Свойствами исчисления предикатов являются ...

- 1) оправданность аксиоматизации
- 2) надежность
- 3) непротиворечивость
- 4) полнота

23. Приведенная форма формулы логики предикатов, в которой все кванторы стоят в ее начале, а область действия каждого из них распространяется до конца формулы, называется ...

- 1) скулемовской формой
- 2) клаузуальной формой
- 3) предваренной нормальной формой
- 4) приведенной формой

24. Лямбда-исчисление реализовано в языке ...

- 1) ПРОЛОГ
- 2) ЛИСП
- 3) C++
- 4) АДА

25. В языке ЛИСП блокирует вызов функции функция ...

- 1) quote
- 2) nil
- 3) fail
- 4) defun

26. Для определения собственной функции в ЛИСП используется функция ...

- 1) quote
- 2) setf

- 3) fail
- 4) defun

27. Для отделения головы списка в ЛИСП используется функция ...

- 1) cond
- 2) cons
- 3) car
- 4) cdr

28. Для соединения элемента и списка в новый список в ЛИСП используется функция ...

- 1) cond
- 2) cons
- 3) car
- 4) cdr

29. Для организации ветвления в ЛИСП используется предложение ...

- 1) cond
- 2) do
- 3) let
- 4) cons

30. Для организации циклов в ЛИСП используется предложение ...

- 1) cond
- 2) do
- 3) for
- 4) while

31. В языке ПРОЛОГ предикат, который всегда возвращает истину, называется ...

- 1) обязательным
- 2) истинным
- 3) тривиальным
- 4) ложным

32. Утверждение, описывающее связь между объектами, называется ...

- 1) фактом
- 2) правилом
- 3) предикатом
- 4) аксиомой

33. В ПРОЛОГе переменная обозначается ...

- 1) строчной буквой

- 2) заглавной буквой
- 3) символом
- 4) числом

34. В ПРОЛОГе для определения факта используется оператор ...

- 1) :-
- 2) :->
- 3) =
- 4) :=

35. В языке ЛИСП функция, которая возвращает последний элемент списка, называется ...

- 1) last
- 2) lastcar
- 3) cdr
- 4) car

36. В функциональном программировании функции могут быть ...

- 1) изменяемыми
- 2) чистыми
- 3) локальными
- 4) глобальными

37. В языке ПРОЛОГ для создания правила используется символ ...

- 1) :=
- 2) =
- 3) :-
- 4) ->

38. В каком языке программирования используется концепция "замыкания"?

- 1) ЛИСП
- 2) Java
- 3) C
- 4) Python

39. В языке ЛИСП, для определения локальной переменной используется ...

- 1) let
- 2) define
- 3) lambda
- 4) quote

40. В каком из языков программирования поддерживается концепция

"функций высшего порядка"?

- 1) C++
- 2) Java
- 3) ЛИСП
- 4) Python

41. В функциональном программировании, что означает "чистая функция"?

- 1) Функция, которая изменяет состояние программы
- 2) Функция, которая всегда возвращает один и тот же результат для одних и тех же аргументов
- 3) Функция, которая не принимает аргументы
- 4) Функция, которая вызывает другие функции

42. Какой из следующих операторов используется для определения условного выражения в ПРОЛОГе?

- 1) if-then
- 2) cond
- 3) case
- 4) select

43. В языке ЛИСП, для обработки ошибок используется функция ...

- 1) error
- 2) catch
- 3) try
- 4) assert

44. В языке ЛИСП, для создания анонимной функции используется ...

- 1) lambda
- 2) defun
- 3) quote
- 4) let

45. Какой из приведенных языков является строго функциональным?

- 1) Java
- 2) C#
- 3) Haskell
- 4) Python

Ключ к тесту:

| | | | | | | | | |
|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|
| 1.4 | 2.1 | 3.2 | 4.2 | 5.3 | 6.3 | 7.2 | 8.1 | 9.3 |
| 10.2 | 11.1 | 12.2 | 13.3 | 14.1 | 15.3 | 16.4 | 17.3 | 18.2 |
| 19.1 | 20.4 | 21.4 | 22.3 | 23.4 | 24.3 | 25.1 | 26.4 | 27.3 |
| 28.2 | 29.1 | 30.2 | 31.3 | 32.3 | 33.2 | 34.1 | 35.1 | 36.2 |

| | | | | | | | | |
|------|------|------|------|------|------|------|------|------|
| 37.3 | 38.1 | 39.1 | 40.3 | 41.2 | 42.2 | 43.2 | 44.1 | 45.3 |
|------|------|------|------|------|------|------|------|------|

Шкала оценивания результатов тестирования

| % верных решений (ответов) | Шкала оценивания |
|----------------------------|---------------------|
| 85 - 100 | отлично |
| 70 - 84 | хорошо |
| 50- 69 | удовлетворительно |
| 0 - 49 | неудовлетворительно |

6.2.4. Примеры задач при разборе конкретных ситуаций

Тема 1. Основные элементы языка ПРОЛОГ. Согласование целевых утверждений в ПРОЛОГе. Рекурсивное представление данных и программ в ПРОЛОГе.

ПК-3.

1. Написать программу на языке ПРОЛОГ для описания семейных отношений. Определить факты, описывающие семью (например, кто чей родитель). Затем создать предикат, который будет определять, является ли один человек родителем другого. Написать пример запроса для проверки, является ли определенный человек родителем другого.

2. Создать рекурсивный предикат для вычисления длины списка в ПРОЛОГе. Привести пример использования предиката, чтобы определить длину списка [a, b, c, d].

3. Написать программу на ПРОЛОГе, которая определяет, является ли число четным. Использовать рекурсию для проверки четности и привести примеры запросов с числами.

Тема 2. Декларативная и операционная семантика. Constraint-технология в логическом программировании

ПК-3.

1. Описать, как работает декларативная семантика в ПРОЛОГе. Привести пример, где используется декларативный подход для решения задачи о поиске всех возможных комбинаций элементов из списка.

2. Используя constraint-технологии в логическом программировании, создать программу, которая решает задачу о размещении четырех ферзей на шахматной доске так, чтобы они не угрожали друг другу. Описать, какие ограничения применялись.

3. Объяснить, как операционная семантика влияет на выполнение программы на ПРОЛОГе. Привести пример, где разница между декларативным и операционным подходом может привести к различным результатам в решении задачи.

Тема 3. Основы функциональной парадигмы. Строго функциональный язык.

ПК-3.

1. Определить основные принципы функционального программирования. Написать короткую программу на ЛИСПе, которая

демонстрирует использование чистых функций и высших функций.

2. Создать программу на строго функциональном языке, которая вычисляет факториал числа с использованием рекурсии. Объяснить, как работает код и какие преимущества он имеет по сравнению с императивным подходом.

3. Обсудить, как ленивые вычисления могут быть реализованы в функциональных языках. Написать пример кода, который демонстрирует ленивые вычисления при работе с бесконечными списками.

Тема 4: Представление и интерпретация функциональных программ

ПК-3.

1. Описать структуру функциональной программы. Написать пример, который включает в себя функции, аргументы и возвращаемые значения, чтобы продемонстрировать основные компоненты.

2. Сравнить компиляцию и интерпретацию функциональных программ. Написать краткий обзор того, как каждая из этих стратегий работает, и привести примеры, когда может быть полезнее использовать одну стратегию над другой.

3. Рассмотреть механизм замыкания в функциональных языках. Написать пример кода на ЛИСПе, который использует замыкания, и объяснить, как это помогает в организации кода.

Шкала оценивания

| Шкала оценивания | Критерии оценивания |
|-----------------------|--|
| «Отлично» | обучающийся ясно изложил условие задачи, решение обосновал |
| «Хорошо» | обучающийся ясно изложил условие задачи, но в обосновании решения имеются сомнения; |
| «Удовлетворительно» | обучающийся изложил решение задачи, но обосновал его формулировками обыденного мышления; |
| «Неудовлетворительно» | обучающийся не уяснил условие задачи, решение не обосновал либо не сдал работу на проверку (в случае проведения решения задач в письменной форме). |

6.2.5. Темы для рефератов

| Тема (раздел) | Вопросы |
|---------------|---------|
|---------------|---------|

| | |
|---|--|
| <p>Тема 1. Основные элементы языка ПРОЛОГ. Согласование целевых утверждений в ПРОЛОГе. Рекурсивное представление данных и программ в ПРОЛОГе.</p> | <p>ПК-3</p> <ol style="list-style-type: none"> 1. История и развитие языка ПРОЛОГ: эволюция языков логического программирования. 2. Сравнение синтаксиса ПРОЛОГа с другими языками программирования: отличия и сходства. 3. Применение рекурсии в ПРОЛОГе: примеры и лучшие практики. 4. Обратный вывод в ПРОЛОГе: механизмы работы и примеры использования. 5. Ошибка и отладка в ПРОЛОГе: распространенные ошибки и их устранение. |
| <p>Тема 2. Декларативная и операционная семантика. Constraint-технология в логическом программировании</p> | <p>ПК-3</p> <ol style="list-style-type: none"> 1. Основные принципы декларативного программирования: особенности и преимущества. 2. Сравнение декларативной и операционной семантики: плюсы и минусы каждого подхода. 3. Constraint-технология в логическом программировании: применение и примеры. 4. Решение задач с использованием ограничений: практические примеры. 5. Будущее декларативного программирования и его развитие: тенденции и прогнозы. |
| <p>Тема 3. Основы функциональной парадигмы. Строго функциональный язык.</p> | <p>ПК-3</p> <ol style="list-style-type: none"> 1. Функциональная парадигма: основные концепции и принципы: введение в функциональное программирование. 2. Каррирование и его применение: как каррирование упрощает код. 3. Ленивые вычисления в функциональных языках: преимущества и примеры использования. 4. Строго функциональные языки программирования: характеристика и примеры. 5. Функциональное программирование в контексте современной разработки: его влияние на технологии. |
| <p>Тема 4. Представление и интерпретация функциональных программ</p> | <p>ПК-3</p> <ol style="list-style-type: none"> 1. Структура функциональной программы: основные компоненты и их взаимосвязи. 2. Методы компиляции функциональных языков: как работает компилятор. 3. Интерпретаторы функциональных языков: принципы работы и примеры. 4. Оптимизация функциональных программ: методы и стратегии. 5. Роль функционального программирования в решении реальных задач: примеры успешных проектов. |

Шкала оценивания

| Шкала оценивания | Критерии оценивания |
|------------------|---|
| «Отлично» | Обучающийся глубоко и содержательно раскрывает тему самостоятельной работы, не допустив ошибок. Ответ носит развернутый и исчерпывающий характер. |

| | |
|-----------------------|---|
| «Хорошо» | Обучающийся в целом раскрывает тему самостоятельной работы, однако ответ хотя бы на один из них не носит развернутого и исчерпывающего характера. |
| «Удовлетворительно» | Обучающийся в целом раскрывает тему самостоятельной работы и допускает ряд неточностей, фрагментарно раскрывает содержание теоретических вопросов или их раскрывает содержательно, но допуская значительные неточности. |
| «Неудовлетворительно» | Обучающийся не владеет выбранной темой самостоятельной работы |

6.3. ОЦЕНОЧНЫЕ СРЕДСТВА ПРОМЕЖУТОЧНОГО КОНТРОЛЯ

Вопросы для подготовки к промежуточной аттестации по итогам освоения дисциплины Функциональное и логическое программирование:

ПК-3.

1. Сущность декларативной парадигмы и ее отличие от императивной.
2. Предложения: факты и правила
3. Структура программы на Прологе.
4. Общая схема доказательства целевого утверждения. Понятие свободной и связанной переменной, анонимная переменная.
5. Ввод-вывод в Прологе. Встроенные предикаты.
6. Отсечение, назначение и основные приемы использования.
7. Список. Описание, приемы работы.
8. Деревья. Описание, приемы работы.
9. ЛИСП как язык функционального программирования. Базовый ЛИСП: форма представления программы, основные функции.
10. Не строго функциональные элементы ЛИСПа. ЛИСП как язык разработки систем искусственного интеллекта и систем аналитических преобразований на ЭВМ.
11. Декларативная и операционная семантика.
12. Использование Пролога для программирования баз данных.
13. Использование Пролога для решения задач искусственного интеллекта.
14. Использование Пролога для создания прототипов экспертных систем.
15. Способы реализации нечеткой логики в Прологе.
16. Соответствие между функциональными и императивными программами. Императивный язык. Формальное описание семантики через интерпретатор императивного языка.
17. Основы функциональной парадигмы. Программирование функций и процедур.
18. Рекурсивные функции и лямбда-исчисление А.Черча.
19. Строго функциональный язык. Элементарные понятия. Символьные данные: лямбда-выражения и представление данных.
20. Constraint-технология в логическом программировании.
21. Понятие число функционального языка.
22. Элементарные селекторы и конструкторы лямбда-выражений.

Элементарные предикаты и арифметика.

23. Рекурсивные функции: разбор случаев, рекурсивные определения, выборподфункций.

24. Списки. Примеры обработки списков.

25. Приемы программирования. Накапливающие параметры.

26. Локальные определения. Лямбда - выражения.

27. Функционалы и другие функции высших порядков. Точечная записьлямбда-выражений.

28. Примеры простых функциональных программ: аналитическое дифференцирование; поиск по дереву - по ширине и в глубину.

29. Представление и интерпретация программ. Абстрактная и контрольная формы функциональной программы.

30. Функциональные эквиваленты императивных программ. Преобразованиеимперативных программ в функциональные.

31. Понятие "целевое утверждение" в Прологе. Как оно формируется и используется?

32. Как реализуются ограничения в Прологе? Примеры применения.

33. Различия между явными и неявными переменными в Прологе. Как это влияет на выполнение программы?

34. Основные методы работы с ассоциативными массивами в ЛИСПе. Примеры использования.

35. Как реализуется механизм обработки исключений в функциональных языках программирования?

36. В чем заключается различие между "чистыми" и "нечистыми" функциями в функциональном программировании?

37. Как происходит компиляция и интерпретация программ в функциональных языках?

38. «Первоклассные функции» в ЛИСПе и как они могут быть использованы для создания более абстрактного и модульного кода.

39. Как использовать " мемоизацию" для оптимизации рекурсивных функций?

40. Основные концепции системы типов в функциональных языках программирования. Как они помогают избежать ошибок?

41. Примеры использования операторов высшего порядка в ЛИСПе. Как они повышают выразительность кода?

42. Как реализуются механизмы ленивых вычислений в функциональных языках? Примеры.

43. Основные принципы работы с потоками данных в функциональных языках программирования.

44. Как реализовать параллелизм в функциональных языках? Примеры подходов.

45. Понятие "интерфейсы" в функциональном программировании. Как они влияют на проектирование программ?

46. Как реализовать "методы слияния" для списков в функциональном программировании?

47. Что такое "рекурсивные типы данных" и как они используются в функциональном стиле?
48. Как реализовать "интерпретатор" для простого функционального языка?
49. Как реализовать "функции высшего порядка" в Прологе? Примеры их применения.
50. Что такое "модульность" программ в функциональном программировании и как она достигается? Примеры.
51. Как обрабатываются ошибки в Прологе? Примеры обработки исключений.
52. Как в ЛИСПе реализуется концепция "первоклассных объектов"? Примеры.
53. Бесконечные списки. Способы реализации в функциональном программировании. Примеры.
54. Множества в ПРОЛОГе. Принципы работы с множествами. Примеры.
55. Операционная и декларативная семантика с точки зрения выполнения программ на языке Пролог.
56. «Чистые» и «нечистые» функции в функциональном программировании. Влияние на тестируемость и отладку кода.
57. Основные подходы к реализации «параллелизма». Реализация и использование «поток» данных в функциональном программировании.
58. Метод «разделяй и властвуй». Преимущества. Сравнение с итеративными подходами.
59. «Ленивые вычисления» и их влияние на производительность программ.
60. «Рекурсивные типы данных». Принципы работы. Примеры использования.
61. Экспертные системы. Основные принципы разработки экспертных систем с использованием ПРОЛОГа.
62. Каковы основные концепции "систем типов" в функциональных языках программирования и как они помогают избежать ошибок во время компиляции?
63. Как использовать "интерфейсы" и "протоколы" в функциональном программировании для создания гибких и расширяемых приложений?
64. Реализация «автоматического тестирования» функциональных программ и подходы для тестирования рекурсивных функций.
65. Каковы основные подходы к реализации "многопоточности" в функциональных языках, и как они отличаются от подходов в императивных языках?
66. Реализация и использование "параллельных вычислений" в ЛИСПе, и какие преимущества это может предоставить.
67. Параметрический полиморфизм и адекватный полиморфизм.
68. Реализация генераторов в функциональных языках и задачи, которые они решают.

69. Использование «функции обратного вызова» в ЛИСПе для реализации асинхронного программирования.

70. Реализация и использование «модели» в функциональных языках.

71. Реализация «параметрических функций» в функциональном программировании и их преимущества.

72. Каковы основные принципы работы с логическими переменными в Прологе и как они отличаются от обычных переменных?

73. Использование предикатов для построения базы данных в ПРОЛОГе и примеры создания и запроса данных.

74. Доказательство утверждений. Механизм резолюции и его реализация.

75. Механизм backtracking в ПРОЛОГе. Пример работы и использования механизма.

6.4. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций

Основной целью проведения промежуточной аттестации является определение степени достижения целей по учебной дисциплине или ее разделам. Осуществляется это проверкой и оценкой уровня теоретической знаний, полученных обучающимися, умения применять их в решении практических задач, степени овладения обучающимися практическими навыками и умениями в объеме требований рабочей программы по дисциплине, а также их умение самостоятельно работать с учебной литературой.

Организация проведения промежуточной аттестации регламентирована «Положением об организации образовательного процесса в федеральном государственном автономном образовательном учреждении «Московский политехнический университет»

6.4.1. Показатели оценивания компетенций на различных этапах их формирования, достижение обучающимися планируемых результатов обучения по дисциплине

| Код и наименование компетенции ПК-3. Способен разрабатывать компоненты системных программных продуктов | | | | |
|---|---|--|---|--|
| Этап (уровень) | Критерии оценивания | | | |
| | неудовлетворительно | удовлетворительно | хорошо | отлично |
| знать | Обучающийся демонстрирует полное отсутствие или недостаточное соответствие следующих знаний: назначение, организацию, принципы | Обучающийся демонстрирует неполное соответствие следующих знаний: назначение, организацию, принципы функционирования, | Обучающийся демонстрирует частичное соответствие следующих знаний: назначение, организацию, принципы | Обучающийся демонстрирует полное соответствие следующих знаний: назначение, организацию, принципы функционирования, |

| | | | | |
|----------------|---|---|--|---|
| | функционирования, последовательность и этапы разработки системных, инструментальных и прикладных программ, программных комплексов и систем; | последовательность и этапы разработки системных, инструментальных и прикладных программ, программных комплексов и систем; | функционирования, последовательность и этапы разработки системных, инструментальных и прикладных программ, программных комплексов и систем; | последовательность и этапы разработки системных, инструментальных и прикладных программ, программных комплексов и систем; |
| уметь | Обучающийся не умеет или в недостаточной степени умеет: применять современные инструментальные средства и технологии программирования при разработке компонентов аппаратно-программных комплексов и баз данных; | Обучающийся демонстрирует неполное соответствие следующих умений: применять современные инструментальные средства и технологии программирования при разработке компонентов аппаратно-программных комплексов и баз данных; | Обучающийся демонстрирует частичное соответствие следующих умений: применять современные инструментальные средства и технологии программирования при разработке компонентов аппаратно-программных комплексов и баз данных; | Обучающийся демонстрирует полное соответствие следующих умений: применять современные инструментальные средства и технологии программирования при разработке компонентов аппаратно-программных комплексов и баз данных; |
| владеть | Обучающийся не владеет или в недостаточной степени владеет: методами и средствами разработки компонентов аппаратно-программных комплексов и баз данных; | Обучающийся владеет в неполном и проявляет недостаточность владения: методами и средствами разработки компонентов аппаратно-программных комплексов и баз данных; | Обучающимся допускаются незначительные ошибки, неточности, затруднения, частично владеет: методами и средствами разработки компонентов аппаратно-программных комплексов и баз данных; | Обучающийся свободно применяет полученные навыки, в полном объеме владеет: методами и средствами разработки компонентов аппаратно-программных комплексов и баз данных; |

6.4.2. Методика оценивания результатов промежуточной аттестации

Показателями оценивания компетенций на этапе промежуточной аттестации по дисциплине «Функциональное и логическое программирование» являются результаты обучения по дисциплине.

Оценочный лист результатов обучения по дисциплине

| Код компетенции | Знания | Умения | Навыки | Уровень сформированности компетенции на данном этапе / оценка |
|-----------------|--------|--------|--------|---|
|-----------------|--------|--------|--------|---|

| | | | | |
|--|---|---|---|--|
| <p>ПК-3. Способен разрабатывать компоненты системных программных продуктов</p> | <p>на уровне знаний: знать методы и средства разработки компонентов аппаратно-программных комплексов и баз данных; знать назначение, организацию, принципы функционирования, последовательность и этапы разработки системных, инструментальных и прикладных программ, программных комплексов и систем; знать стандарты, методические и нормативные материалы, определяющие проектирование и разработку компонентов аппаратно-программных комплексов и баз данных; знать модели, методы и формы организации процесса разработки компонентов аппаратно-программных комплексов и баз данных; знать методы и средства обеспечения информационной безопасности разрабатываемых компонентов АПК</p> | <p>на уровне умений: уметь применять современные инструментальные средства и технологии программирования при разработке компонентов аппаратно-программных комплексов и баз данных; уметь применять современные программно-методические комплексы автоматизированного проектирования объектов профессиональной деятельности.</p> | <p>на уровне навыков: методами и средствами разработки компонентов аппаратно-программных комплексов и баз данных; методами организации процесса разработки компонентов аппаратно-программных комплексов и баз данных.</p> | |
| <p>Оценка по дисциплине (среднее арифметическое)</p> | | | | |

Оценка по дисциплине зависит от уровня сформированности компетенций, закрепленных за дисциплиной и представляет собой среднее арифметическое от выставленных оценок по отдельным результатам обучения (знания, умения, навыки).

Оценка «отлично» выставляется, если среднее арифметическое находится в интервале от 4,5 до 5,0.

Оценка «хорошо» выставляется, если среднее арифметическое находится в интервале от 3,5 до 4,4.

Оценка «удовлетворительно» выставляется, если среднее арифметическое находится в интервале от 2,5 до 3,4.

Оценка «неудовлетворительно» выставляется, если среднее арифметическое находится в интервале от 0 до 2,4.

Промежуточная аттестация обучающихся в форме экзамена проводится по результатам выполнения всех видов учебной работы, предусмотренных учебным планом по дисциплине «Функциональное и логическое программирование», 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 «отлично», «хорошо», «удовлетворительно» или «неудовлетворительно».

| Шкала оценивания | Описание |
|-------------------------|---|
| Отлично | Выполнены все виды учебной работы, предусмотренные учебным планом. Студент демонстрирует соответствие знаний, умений, навыков приведенным в таблицах показателей, оперирует приобретенными знаниями, умениями, навыками, применяет их в ситуациях повышенной сложности. При этом могут быть допущены незначительные ошибки, неточности, затруднения при аналитических операциях, переносе знаний и умений на новые, нестандартные ситуации. |
| Хорошо | Выполнены все виды учебной работы, предусмотренные учебным планом. Студент демонстрирует неполное, правильное соответствие знаний, умений, навыков приведенным в таблицах показателей, либо если при этом были допущены 2-3 несущественные ошибки. |
| Удовлетворительно | Выполнены все виды учебной работы, предусмотренные учебным планом. Студент демонстрирует соответствие знаний, в котором освещена основная, наиболее важная часть материала, но при этом допущена одна значительная ошибка или неточность. |
| Неудовлетворительно | 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. |
|--|---|

7. Электронная информационно-образовательная среда

Каждый обучающийся в течение всего периода обучения обеспечивается индивидуальным неограниченным доступом к электронной информационно-образовательной среде Чебоксарского института (филиала) Московского политехнического университета из любой точки, в которой имеется доступ к информационно-телекоммуникационной сети «Интернет» (далее – сеть «Интернет»), как на территории филиала, так и вне ее.

Электронная информационно-образовательная среда – совокупность информационных и телекоммуникационных технологий, соответствующих технологических средств, обеспечивающих освоение обучающимися образовательных программ в полном объеме независимо от места нахождения обучающихся.

Электронная информационно-образовательная среда обеспечивает:

а) доступ к учебным планам, рабочим программам дисциплин (модулей), практик, электронным учебным изданиям и электронным образовательным ресурсам, указанным в рабочих программах дисциплин (модулей), практик;

б) формирование электронного портфолио обучающегося, в том числе сохранение его работ и оценок за эти работы;

в) фиксацию хода образовательного процесса, результатов промежуточной аттестации и результатов освоения программы бакалавриата;

г) проведение учебных занятий, процедур оценки результатов обучения, реализация которых предусмотрена с применением электронного обучения, дистанционных образовательных технологий;

д) взаимодействие между участниками образовательного процесса, в том числе синхронное и (или) асинхронное взаимодействия посредством сети "Интернет".

Функционирование электронной информационно-образовательной среды обеспечивается соответствующими средствами информационно-коммуникационных технологий и квалификацией работников, ее использующих и поддерживающих.

Функционирование электронной информационно-образовательной среды соответствует законодательству Российской Федерации.

Основными составляющими ЭИОС филиала являются:

а) сайт института в сети Интернет, расположенный по адресу www.polytech21.ru, <https://chebpolytech.ru/> который обеспечивает:

- доступ обучающихся к учебным планам, рабочим программам дисциплин, практик, к изданиям электронных библиотечных систем, электронным информационным и образовательным ресурсам, указанных в рабочих программах (разделы сайта «Сведения об образовательной организации»);

- информирование обучающихся обо всех изменениях учебного процесса (новостная лента сайта, лента анонсов);

- взаимодействие между участниками образовательного процесса (подразделы сайта «Задать вопрос директору»);

б) официальные электронные адреса подразделений и сотрудников института с Яндекс-доменом @polytech21.ru (список контактных данных подразделений Филиала размещен на официальном сайте Филиала в разделе «Контакты», списки контактных официальных электронных данных преподавателей размещены в подразделах «Кафедры») обеспечивают взаимодействие между участниками образовательного процесса;

в) личный кабинет обучающегося (портфолио) (вход в личный кабинет размещен на официальном сайте Филиала в разделе «Студенту» подразделе «Электронная информационно-образовательная среда») включает в себя портфолио студента, электронные ведомости, рейтинг студентов и обеспечивает:

- фиксацию хода образовательного процесса, результатов промежуточной аттестации и результатов освоения образовательных программ обучающимися,

- формирование электронного портфолио обучающегося, в том числе с сохранение работ обучающегося, рецензий и оценок на эти работы,

г) электронные библиотеки, включающие электронные каталоги, полнотекстовые документы и обеспечивающие доступ к учебно-методическим материалам, выпускным квалификационным работам и т.д.:

Чебоксарского института (филиала) - «ИРБИС»

д) электронно-библиотечные системы (ЭБС), включающие электронный каталог и полнотекстовые документы:

- ЭБС «ЛАНЬ» -<https://e.lanbook.com/>

- Образовательная платформа Юрайт - <https://urait.ru>

- IPR SMART -<https://www.iprbookshop.ru/>

е) платформа цифрового образования Политеха -<https://lms.mospolytech.ru/>

ж) система «Антиплагиат» -<https://www.antiplagiat.ru/>

з) система электронного документооборота DIRECTUM Standard — обеспечивает документооборот между Филиалом и Университетом;

и) система «IC Управление ВУЗом Электронный деканат» (Московский политехнический университет) обеспечивает фиксацию хода образовательного процесса, результатов промежуточной аттестации и результатов освоения образовательных программ обучающимися;

к) система «POLYTECH systems» обеспечивает информационное, документальное автоматизированное сопровождение образовательного процесса;

л) система «Абитуриент» обеспечивает документальное автоматизированное сопровождение работы приемной комиссии.

8. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины

Основная литература:

1. Трофимов, В. В. Алгоритмизация и программирование : учебник для

вузов / В. В. Трофимов, Т. А. Павловская. — 4-е изд. — Москва : Издательство Юрайт, 2025. — 108 с. — (Высшее образование). — ISBN 978-5-534-20430-8. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/562040>.

2. Свердлов, С. З. Языки программирования и методы трансляции : учебное пособие для вузов / С. З. Свердлов. — 5-е изд., стер. — Санкт-Петербург : Лань, 2025. — 564 с. — ISBN 978-5-507-50570-8. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/447398>. — Режим доступа: для авториз. пользователей.

3. Алгоритмизация и программирование : учебник для вузов / Д. А. Бархатова, Н. И. Пак, А. А. Петрова, И. А. Яшина. — Санкт-Петербург : Лань, 2025. — 348 с. — ISBN 978-5-507-52244-6. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/482915>. — Режим доступа: для авториз. пользователей.

4. Фридман, Д. The Little Schemer. Чудесное функциональное программирование : руководство / Д. Фридман, М. Феллейзен ; рисунки Д. Библи. — Москва : ДМК Пресс, 2024. — 230 с. — ISBN 978-5-93700-234-1. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/456698>. — Режим доступа: для авториз. пользователей.

Дополнительная литература:

1. Черпаков, И. В. Основы программирования : учебник и практикум для вузов / И. В. Черпаков. — 2-е изд. — Москва : Издательство Юрайт, 2025. — 196 с. — (Высшее образование). — ISBN 978-5-534-18759-5. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/560807>.

2. Пантелеев, Е. Р. Основы разработки приложений семантического Web на платформе SWI-Prolog : учебное пособие для вузов / Е. Р. Пантелеев. — Санкт-Петербург : Лань, 2025. — 120 с. — ISBN 978-5-507-52206-4. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/482999>. — Режим доступа: для авториз. пользователей.

Периодика:

Научный периодический журнал «Вестник Южно-Уральского государственного университета. Серия «Компьютерные технологии, управление, радиоэлектроника»: Научный рецензируемый журнал. <https://vestnik.susu.ru/ctcr> - Текст: электронный.

9.Профессиональные базы данных и информационно-справочные системы

| | |
|---|---|
| Профессиональная база данных и информационно-справочные системы | Информация о праве собственности (реквизиты договора) |
|---|---|

| | |
|---|--|
| Университетская информационная система РОССИЯ https://uisrussia.msu.ru/ | Тематическая электронная библиотека и база для прикладных исследований в области экономики, управления, социологии, лингвистики, философии, филологии, международных отношений, права. свободный доступ |
| научная электронная библиотека Elibrary http://elibrary.ru/ | Научная электронная библиотека eLIBRARY.RU - это крупнейший российский информационно-аналитический портал в области науки, технологии, медицины и образования, содержащий рефераты и полные тексты более 26 млн научных статей и публикаций, в том числе электронные версии более 5600 российских научно-технических журналов, из которых более 4800 журналов в открытом доступе свободный доступ |
| сайт Института научной информации по общественным наукам РАН. http://www.inion.ru | Библиографические базы данных ИНИОН РАН по социальным и гуманитарным наукам ведутся с начала 1980-х годов. Общий объём массивов составляет более 3 млн. 500 тыс. записей (данные на 1 января 2012 г.). Ежегодный прирост — около 100 тыс. записей. В базы данных включаются аннотированные описания книг и статей из журналов и сборников на 140 языках, поступивших в Фундаментальную библиотеку ИНИОН РАН. Описания статей и книг в базах данных снабжены шифром хранения и ссылками на полные тексты источников из Научной электронной библиотеки. |
| Федеральный портал «Российское образование» [Электронный ресурс] – http://www.edu.ru | Федеральный портал «Российское образование» – уникальный интернет-ресурс в сфере образования и науки. Ежедневно публикует самые актуальные новости, анонсы событий, информационные материалы для широкого круга читателей. Еженедельно на портале размещаются эксклюзивные материалы, интервью с ведущими специалистами – педагогами, психологами, учеными, репортажи и аналитические статьи. Читатели получают доступ к нормативно-правовой базе сферы образования, они могут пользоваться самыми различными полезными сервисами – такими, как онлайн-тестирование, опросы по актуальным темам и т.д. |
| Информационные технологии – периодическое научно-техническое издание в области информационных технологий, автоматизированных систем и использования информатики в различных приложениях novtex.ru | Издательство выпускает теоретические и прикладные научно-технические журналы, обеспечивающие научной, производственной, обзорно-аналитической и образовательной информацией руководящих работников и специалистов промышленных предприятий, научных академических и отраслевых организаций, а также учебных заведений в области приоритетных направлений развития науки и технологий. |
| Ассоциация инженерного образования России http://www.ac-raee.ru/ | Совершенствование образования и инженерной деятельности во всех их проявлениях, относящихся к учебному, научному и технологическому направлениям, включая процессы преподавания, консультирования, исследования, разработки инженерных решений, оказания широкого спектра образовательных услуг, обеспечения связей с общественностью, производством, наукой и интеграции в международное научно-образовательное пространство. свободный доступ |

10. Программное обеспечение (лицензионное и свободно распространяемое), используемое при осуществлении образовательного процесса

| Аудитория | Программное обеспечение | Информация о праве собственности (реквизиты) |
|-----------|-------------------------|--|
|-----------|-------------------------|--|

| | | договора, номер лицензии и т.д.) |
|--|--|--|
| <p>№ 2196 Учебная аудитория для проведения учебных занятий всех видов, предусмотренных программой бакалавриата/специалитета/ магистратуры, оснащенная оборудованием и техническими средствами обучения, состав которых определяется в рабочих программах дисциплин (модулей) Кабинет систем управления ООО «НПО «Каскад-ГРУП»</p> | Windows 7 OLPNLAcadmс | договор №Д03 от 30.05.2012) с допсоглашениями от 29.04.14 и 01.09.16 (бессрочная лицензия) |
| | Kaspersky Endpoint Security для бизнеса – Расширенный Russian Edition. 150-249 Node 2 year Educational Renewal License | Сублицензионный договор № ППИ-126/2023 от 14.12.2023 |
| | Google Chrome | Свободное распространяемое программное обеспечение (бессрочная лицензия) |
| | Zoom | свободно распространяемое программное обеспечение (бессрочная лицензия) |
| | Microsoft Office Standard 2007(Microsoft DreamSpark Premium Electronic Software Delivery Academic(Microsoft Open License | номер лицензии-42661846 от 30.08.2007) с допсоглашениями от 29.04.14 и 01.09.16 (бессрочная лицензия) |
| <p>№ 2066 Учебная аудитория для проведения учебных занятий всех видов, предусмотренных программой среднего профессионального образования/бакалавриата/специалитета/ магистратуры, оснащенная оборудованием и техническими средствами обучения, состав которых определяется в рабочих программах дисциплин (модулей) Лаборатория информационных технологий</p> | Kaspersky Endpoint Security для бизнеса – Расширенный Russian Edition. 150-249 Node 2 year Educational Renewal License | Сублицензионный договор № ППИ-126/2023 от 14.12.2023 |
| | Windows 7 OLPNLAcadmс Windows Server 2012 | договор №Д03 от 30.05.2012) с допсоглашениями от 29.04.14 и 01.09.16 (бессрочная лицензия) |
| | MS Windows 10 Pro | договор № 392_469.223.3К/19 от 17.12.19 (бессрочная лицензия) |
| | Microsoft Office Standard 2019(Microsoft DreamSpark Premium Electronic Software Delivery Academic(Microsoft Open License | номер лицензии-42661846 от 30.08.2007) с допсоглашениями от 29.04.14 и 01.09.16 (бессрочная лицензия) |
| | Eclipse IDE for Java EE Developers, NET Framework, JDK 8, Microsoft SQL Server Express Edition, Microsoft Visio Professional, Microsoft Visual Studio, MySQL Installer for Windows, NetBeans, SQL Server Management Studio, Microsoft SQL Server Java Connector, Android Studio, IntelliJ IDEA. | свободно распространяемое программное обеспечение (бессрочная лицензия) |
| | КОМПАС-3D v20 и v21 | Сублицензионный договор № Нп-22-00044 от 21.03.2022 (бессрочная лицензия) |
| | MathCADv.15 | Сублиц.договор |

| | | |
|--|--|---|
| | | №39331/МОС2286 от 6.05.2013) номер лицензии-42661846 от 30.08.2007) (бессрочная лицензия) |
| | SimInTech | Отечественное программное обеспечение |
| | AdobeReader | свободно распространяемое программное обеспечение (бессрочная лицензия) |
| | AdobeFlashPlayer | свободно распространяемое программное обеспечение (бессрочная лицензия) |
| | Microsoft Visual Studio 2019 | свободно распространяемое программное обеспечение (бессрочная лицензия) |
| | Python 3.7 | свободно распространяемое программное обеспечение (бессрочная лицензия) |
| | AIMP | отечественное свободно распространяемое программное обеспечение (бессрочная лицензия) |
| № 1126 Помещение для самостоятельной работы обучающихся | Kaspersky Endpoint Security для бизнеса – Расширенный Russian Edition. 150-249 Node 2 year Educational Renewal License | Сублицензионный договор № ППИ-126/2023 от 14.12.2023 |
| | Windows 7 OLPNLAcadmс | договор №Д03 от 30.05.2012) с допсоглашениями от 29.04.14 и 01.09.16 (бессрочная лицензия) |
| | AdobeReader | свободно распространяемое программное обеспечение (бессрочная лицензия) |
| | Гарант- справочно-правовая система | Договор №С-002-2025 от 09.01.2025 |
| | Yandex браузер | свободно распространяемое программное обеспечение (бессрочная лицензия) |
| | Microsoft Office Standard 2007(Microsoft DreamSpark Premium Electronic Software Delivery Academic(Microsoft Open License | номер лицензии-42661846 от 30.08.2007) с допсоглашениями от 29.04.14 и 01.09.16 (бессрочная лицензия) |
| | Zoom | свободно распространяемое программное обеспечение (бессрочная лицензия) |
| | AIMP | отечественное свободно распространяемое программное обеспечение (бессрочная лицензия) |

11. Материально-техническое обеспечение дисциплины

| Тип и номер помещения | Перечень основного оборудования и технических средств обучения |
|-----------------------|--|
|-----------------------|--|

| | |
|---|---|
| <p>Учебная аудитория для проведения учебных занятий всех видов, предусмотренных программой бакалавриата/специалитета/ магистратуры, оснащенная оборудованием и техническими средствами обучения, состав которых определяется в рабочих программах дисциплин (модулей) Кабинет систем управления ООО «НПО «Каскад-ГРУП» 2196 (Чебоксары, ул. К.Маркса, д.60)</p> | <p><u>Оборудование:</u> комплект мебели для учебного процесса; доска учебная; стенды <u>Технические средства обучения:</u> мультимедийное оборудование (проектор, экран)</p> |
| <p>Учебная аудитория для проведения учебных занятий всех видов, предусмотренных программой среднего профессионального образования/бакалавриата/специалитета/ магистратуры, оснащенная оборудованием и техническими средствами обучения, состав которых определяется в рабочих программах дисциплин (модулей) Лаборатория информационных технологий №2066 (Чебоксары, ул. К.Маркса, д.60)</p> | <p><u>Оборудование:</u> комплект мебели для учебного процесса; доска учебная; автоматизированные рабочие места, автоматизированное рабочее место преподавателя, проектор и экран; маркерная доска; <u>Технические средства обучения:</u> компьютерная техника (процессор Core i3, оперативная память объемом не менее 8 Гб); сервер в лаборатории (8-ядерный процессор с частотой не менее 3 ГГц, оперативная память объемом не менее 16 Гб, жесткие диски общим объемом не менее 1 Тб)</p> |
| <p>Помещение для самостоятельной работы обучающихся № 1126 (г. Чебоксары, ул. К.Маркса. 60)</p> | <p><u>Оборудование:</u> комплект мебели для учебного процесса; <u>Технические средства обучения:</u> компьютерная техника с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду Филиала</p> |

12. Методические указания для обучающегося по освоению дисциплины

Методические указания для занятий лекционного типа

В ходе лекционных занятий обучающемуся необходимо вести конспектирование учебного материала, обращать внимание на категории, формулировки, раскрывающие содержание тех или иных явлений и процессов, научные выводы и практические рекомендации.

Необходимо задавать преподавателю уточняющие вопросы с целью уяснения теоретических положений, разрешения спорных ситуаций. Целесообразно дорабатывать свой конспект лекции, делая в нем соответствующие записи из основной и дополнительной литературы, рекомендованной преподавателем и предусмотренной учебной программой дисциплины.

Методические указания для занятий лабораторного типа.

Выполнению лабораторных работ предшествует проверка знаний студентов – их теоретической готовности к выполнению задания. Проверка

знаний проводится в форме, которую определяет преподаватель дисциплины (тестирование, опрос).

При проведении лабораторных занятий выделяют следующие разделы:

- общие положения (перечень лабораторных или практических занятий);
- ообщие требования к выполнению работ;
- инструкция по каждой работе;
- справочные материалы и т. д.

Лабораторные занятия позволяют развивать у обучающегося творческое теоретическое мышление, умение самостоятельно изучать литературу, анализировать практику; учат четко формулировать мысль, вести дискуссию, то есть имеют исключительно важное значение в развитии самостоятельного мышления.

Готовясь к докладу или выступлению в рамках интерактивной формы, при необходимости, следует обратиться за помощью к преподавателю.

Методические указания к самостоятельной работе.

Самостоятельная работа обучающегося является основным средством овладения учебным материалом во время, свободное от обязательных учебных занятий. Самостоятельная работа обучающегося над усвоением учебного материала по учебной дисциплине может выполняться в библиотеке университета, учебных кабинетах, компьютерных классах, а также в домашних условиях. Содержание и количество самостоятельной работы обучающегося определяется учебной программой дисциплины, методическими материалами, практическими заданиями и указаниями преподавателя.

Самостоятельная работа в аудиторное время может включать:

- 1) конспектирование (составление тезисов) лекций;
- 2) выполнение контрольных работ;
- 3) решение задач;
- 4) работу со справочной и методической литературой;
- 5) работу с нормативными правовыми актами;
- 6) выступления с докладами, сообщениями на семинарских занятиях;
- 7) защиту выполненных работ;
- 8) участие в оперативном (текущем) опросе по отдельным темам изучаемой дисциплины;
- 9) участие в беседах, деловых (ролевых) играх, дискуссиях, круглых столах, конференциях;
- 10) участие в тестировании и др.

Самостоятельная работа во внеаудиторное время может состоять из:

- 1) повторения лекционного материала;
- 2) подготовки к практическим занятиям;
- 3) изучения учебной и научной литературы;
- 4) изучения нормативных правовых актов (в т.ч. в электронных базах данных);
- 5) решения задач, и иных практических заданий;

- 6) подготовки к контрольным работам, тестированию и т.д.;
- 7) подготовки к практическим занятиям устных докладов (сообщений);
- 8) выполнения курсовых работ, предусмотренных учебным планом;
- 9) выполнения выпускных квалификационных работ и др.;
- 10) выделения наиболее сложных и проблемных вопросов по изучаемой теме, получение разъяснений и рекомендаций по данным вопросам с преподавателями на консультациях;
- 11) проведения самоконтроля путем ответов на вопросы текущего контроля знаний, решения представленных в учебно-методических материалах кафедры задач, тестов, докладов;
- 12) текущий контроль осуществляется в форме устных, тестовых опросов, докладов;

В случае пропусков занятий, наличия индивидуального графика обучения и для закрепления практических навыков студентам могут быть выданы типовые индивидуальные задания, которые должны быть сданы в установленный преподавателем срок.

13. Особенности реализации дисциплины для инвалидов и лиц с ограниченными возможностями здоровья

Обучение по дисциплине «Функциональное и логическое программирование» инвалидов и лиц с ограниченными возможностями здоровья (далее ОВЗ) осуществляется преподавателем с учетом особенностей психофизического развития, индивидуальных возможностей и состояния здоровья таких обучающихся.

Для студентов с нарушениями опорно-двигательной функции и с ОВЗ по слуху предусматривается сопровождение лекций и практических занятий мультимедийными средствами, раздаточным материалом.

Для студентов с ОВЗ по зрению предусматривается применение технических средств усиления остаточного зрения, а также предусмотрена возможность разработки аудиоматериалов.

По дисциплине «Функциональное и логическое программирование» обучение инвалидов и лиц с ограниченными возможностями здоровья может осуществляться как в аудитории, так и с использованием электронной информационно-образовательной среды, образовательного портала и электронной почты.

ЛИСТ ДОПОЛНЕНИЙ И ИЗМЕНЕНИЙ
рабочей программы дисциплины

Рабочая программа дисциплины рассмотрена, обсуждена и одобрена для исполнения в 202__-202__ учебном году на заседании кафедры, протокол № ____ от « » 202 г.

Внесены дополнения и изменения _____

Рабочая программа дисциплины рассмотрена, обсуждена и одобрена для исполнения в 202__-202__ учебном году на заседании кафедры, протокол № ____ от « » 202 г.

Внесены дополнения и изменения _____

Рабочая программа дисциплины рассмотрена, обсуждена и одобрена для исполнения в 202__-202__ учебном году на заседании кафедры, протокол № ____ от « » 202 г.

Внесены дополнения и изменения _____

Рабочая программа дисциплины рассмотрена, обсуждена и одобрена для исполнения в 202__-202__ учебном году на заседании кафедры, протокол № ____ от « » 202 г.

Внесены дополнения и изменения _____
