Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Макаренко Елена Николаевна
Должность: Ректор
Дата подписания: 09.09.2026 16:07:04
Уникальный программный ключ:
c098bc0c1041cb2a4cf926cf171d6715d99a6ae00adc8e27b55cbe1e2dbd7c78

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение высшего образования
«Ростовский государственный экономический университет (РИНХ)»

УТВЕРЖДАЮ
Начальник
учебно-методического управления
Т.К. Платонова
«25» мая 2026 г.

Рабочая программа дисциплины
Функциональное и логическое программирование

Направление подготовки
09.03.04 Программная инженерия

Направленность (профиль) программы бакалавриата
09.03.04.01 Системное и прикладное программное обеспечение

Для набора 2026 года

Квалификация
Бакалавр

КАФЕДРА **Информационные технологии и программирование**

Распределение часов дисциплины по семестрам / курсам

Курс	5		Итого	
Вид занятий	УП	РП		
Лекции	6	6	6	6
Лабораторные	8	8	8	8
Итого ауд.	14	14	14	14
Контактная работа	14	14	14	14
Сам. работа	193	193	193	193
Часы на контроль	9	9	9	9
Итого	216	216	216	216

ОСНОВАНИЕ

Учебный план утвержден учёным советом Университета (протокол № 9 от 03.03.2026 г.).

Программу составил(и): к.э.н., доцент, Жилина Е.В.

Зав. кафедрой: к.э.н., доцент Е.В. Ефимова

Методический совет: д.э.н., профессор Е.Н. Тищенко

1. ЦЕЛИ ОСВОЕНИЯ ДИСЦИПЛИНЫ

1.1	Изучение основ функционального и логического программирования, формирование профессиональных знаний и практических навыков по разработке и созданию моделей и систем с помощью языков функционального и логического программирования

2. ТРЕБОВАНИЯ К РЕЗУЛЬТАТАМ ОСВОЕНИЯ ДИСЦИПЛИНЫ

ПК-3. способен разрабатывать компоненты программных комплексов (в том числе интерфейсы, драйвера, компиляторы, загрузчики, сборщики, системные утилиты) и баз данных с использованием современных инструментальных средств и технологий программирования

ПК-1. способен использовать методы и инструментальные средства исследования объектов профессиональной деятельности, формализации предметной области с учетом ограничений

В результате освоения дисциплины обучающийся должен:

Знать:

основные понятия из области разработки программных систем, применяемые метрики, методы и инструментальные средства (соотнесено с индикатором ПК-1.1);
основы информатики и программирования (соотнесено с индикатором ПК-3.1).

Уметь:

применять основные методы разработки программного обеспечения; применять основные инструменты разработки программного обеспечения (соотнесено с индикатором ПК-1.2);
использовать современные технологии разработки программных продуктов (соотнесено с индикатором ПК-3.2).

Владеть:

основными методами разработки программного обеспечения; основными инструментами разработки программного обеспечения (соотнесено с индикатором ПК-1.3);
навыками разработки алгоритмов в виде блок-схемы и составления плана ручного тестирования разрабатываемого программного продукта (соотнесено с индикатором ПК-3.3).

3. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

Раздел 1. Теоретические аспекты функционального программирования

№	Наименование темы, краткое содержание	Вид занятия / работы / форма ПА	Семестр / Курс	Количество часов	Компетенции
1.1	Концепция функционального программирования и определение функций. Концепция и особенности функционального программирования. Свойства функциональных языков. Основные особенности Лиспа, достоинства языка.Элементарные понятия языка Лисп: атомы и списки. Программа на языке Лисп. Вычислимые выражения. Понятие функции, префиксная нотация. Вычисление лямбда-выражений. Определение функций в Лиспе.Базовые функции языка, предикаты.	Лекционные занятия	5	2	ПК-3 ПК-1
1.2	Рекурсия. Понятие рекурсии. Правила записи рекурсивной функции. Терминальная ветвь, рекурсивная ветвь. Прямая и косвенная рекурсия. Рекурсия с несколькими терминальными ветвями, рекурсивными ветвями.	Лекционные занятия	5	2	ПК-3 ПК-1
1.3	Основы логической парадигмы. Исчисление предикатов как язык представления знаний. Виды импликации. Логика предикатов первого порядка, хорновские дизъюнкты. Вычислительная модель. Анализ структуры термов. Принцип Робинсона.	Лекционные занятия	5	2	ПК-3 ПК-1
1.4	Основы функционального программирования на Haskell. Строго функциональный язык. Символьные данные: лямбдавыражения и представление данных. Элементарные селекторы и конструкторы лямбда-выражений. Элементарные предикаты и арифметика. Рекурсивные функции: разбор случаев, рекурсивные определения, выбор подфункций. Примеры обработки списков. Дополнительные возможности. Приемы программирования.	Самостоятельная работа	5	8	ПК-3 ПК-1
1.5	Аккумуляторы. Локальные определения. Функционалы и другие функции высших порядков. Примеры простых функциональных программ. Значения и типы. Полиморфные типы. Типы, определяемые пользователем. Бинарные конструкторы данных. Рекурсивные типы. Синонимы типов. Встроенные типы.	Самостоятельная работа	5	8	ПК-3 ПК-1
1.6	Генераторы списков и арифметические последовательности. Строки. Функции. Инфиксные операции. Секции. Декларация пользовательских операций. Бесконечные структуры данных и ленивые вычисления. Сопоставление с образцом. As-образцы.	Самостоятельная работа	5	8	ПК-3 ПК-1

	Универсальные образцы. Семантика сопоставления с образцом. Выражение case.				
1.7	Ленивые образцы. Лексическая видимость и вложенные формы. Let-выражения. Предложение where. Двумерный синтаксис. Классы типов и перегрузка. Наследование Множественное наследование. Типы высшего порядка. Контроль корректности типов.	Самостоятельная работа	5	8	ПК-3 ПК-1
1.8	Описание newtype. Метки полей. Строгие конструкторы данных. Понятие монады. Система ввода-вывода. Базисные операции ввода-вывода. Сетевое программирование.	Самостоятельная работа	5	8	ПК-3 ПК-1

Раздел 2. Разработка программных систем

№	Наименование темы, краткое содержание	Вид занятия / работы / форма ПА	Семестр / Курс	Количество часов	Компетенции
2.1	Навыки работы с интерпретатором языка Haskell. Основные типы языка Haskell. Простейшие логические функции. Простейшие списочные функции.	Лабораторные занятия	5	2	ПК-3 ПК-1
2.2	Интерпретатор языка Haskell. Символьные функции. Простейшие кортежные функции. Теоретико-множественные операции. Сортировка. Вспомогательные функции. Отладка	Лабораторные занятия	5	2	ПК-3 ПК-1
2.3	Лабораторное задание Списочные функции высших порядков. Арифметические последовательности. Генераторы списков.	Самостоятельная работа	5	8	ПК-3 ПК-1
2.4	Лабораторное задание Бесконечные списки. Ввод-вывод. Нетривиальные функции.	Самостоятельная работа	5	8	ПК-3 ПК-1
2.5	Лабораторное задание Простые числа и факторизация. Деревья. Деревья вычислений.	Самостоятельная работа	5	8	ПК-3 ПК-1
2.6	Лабораторное задание Функции высшего порядка Let-связывание. Охраняющие условия. Пользовательские типы. Определение операторов. Рекурсивные типы. Модули. Абстрактные типы данных. Операции ввода-вывода	Самостоятельная работа	5	8	ПК-3 ПК-1

Раздел 3. Методы проектирования при использовании алгоритмов функционального программирования

№	Наименование темы, краткое содержание	Вид занятия / работы / форма ПА	Семестр / Курс	Количество часов	Компетенции
3.1	Проработка лекционного материала. Решение задач на Haskell	Самостоятельная работа	5	8	ПК-3 ПК-1
3.2	Подготовка к лабораторным занятиям. Модули. Абстрактные типы данных. Операции ввода-вывода	Самостоятельная работа	5	8	ПК-3 ПК-1

Раздел 4. Теоретические основы логического программирования

№	Наименование темы, краткое содержание	Вид занятия / работы / форма ПА	Семестр / Курс	Количество часов	Компетенции
4.1	Введение в логическое программирование. Основы языка Visual Prolog. Унификация и поиск с возвратом	Самостоятельная работа	5	8	ПК-3 ПК-1
4.2	Основные элементы языка Prolog. Алфавит языка. Термы. Виды термов: константы, переменные, структуры. Литеры и их типы. Интерпретация литер. Операторы. Свойства операторов (позиция, приоритет, ассоциативность). Инфиксные, префиксные, постфиксные операторы. Факты. Правила. Запись фактов и правил. Предикаты. Цели, конъюнкция целей. Общая схема доказательства целевого утверждения.	Самостоятельная работа	5	8	ПК-3 ПК-1
4.3	Арифметика в языке Prolog. Встроенные предикаты для сравнения чисел. Вычисление арифметических выражений. Предикат is и его варианты в различных версиях. Примеры программ с выполнением арифметических операций. Согласование целевых утверждений. Доказательство целевых утверждений при использовании механизма возврата. Правила установления соответствия. Недетерминизм первого и второго рода. Понятие свободной и связанной переменной. Примеры программ с использованием механизма возврата. Операционная модель вычисления Prolog-программ.	Самостоятельная работа	5	8	ПК-3 ПК-1
4.4	Рекурсивное представление данных и программ. Рекурсивные функции. Структуры и деревья. Список как частный вид структуры. Формы записи списков. Работа со списками Граничные условия и способы использования рекурсии. Примеры программ с рекурсивными определениями. Сортировка списков. Отсечение и способы его использования в языке. Причины использования отсечения. Предикат cut. Диаграмма работы программы с использованием отсечения. Общие случаи использования отсечения. Проблемы, связанные с использованием отсечения. Развитие	Самостоятельная работа	5	8	ПК-3 ПК-1

	отсечения в отсечение-отрезок в некоторых вариантах.				
4.5	Разработка графического интерфейса пользователя в среде Visual Prolog. Возможности визуальной среды разработки Visual Prolog	Самостоятельная работа	5	8	ПК-3 ПК-1

Раздел 5. Разработка логических информационных систем

№	Наименование темы, краткое содержание	Вид занятия / работы / форма ПА	Семестр / Курс	Количество часов	Компетенции
5.1	Знакомство с основами языка логического программирования Prolog. Возможности программной оболочки. Способы представления данных в виде фактов и правил вывода для записи программы на языке Prolog. Используя возможности, предоставляемые программной оболочкой, проверить правильность работы созданной программы. Оценка полученных результатов.	Лабораторные занятия	5	2	ПК-3 ПК-1
5.2	Глобальные разделы программы Visual Prolog. Операции над списками. Арифметические действия. Стандартный предикат findall. Задача о классификации объектов	Лабораторные занятия	5	2	ПК-3 ПК-1
5.3	Лабораторное задание Отсечение. Внутренняя база фактов Visual Prolog. Рекурсивные структуры данных. Составные списки	Самостоятельная работа	5	8	ПК-3 ПК-1
5.4	Лабораторное задание Разработка программ, использующих для нахождения решений рекурсию и поиск с возвратом. Исследование различия в двух методах нахождения решений: рекурсии и поиске с возвратом. Исследование различия в двух методах нахождения решений: хвостовой и нехвостовой рекурсии. Оценка полученных результатов.	Самостоятельная работа	5	8	ПК-3 ПК-1
5.5	Лабораторное задание Разработка программ для работы с рекурсивными структурами данных (списками и деревьями). Способы представления данных в виде фактов и правил вывода для записи программы на языке Prolog. Исследование методов работы с рекурсивными структурами данных (списками и деревьями. Оценка полученных результатов.	Самостоятельная работа	5	8	ПК-3 ПК-1

Раздел 6. Методы проектирования при использовании алгоритмов логического программирования

№	Наименование темы, краткое содержание	Вид занятия / работы / форма ПА	Семестр / Курс	Количество часов	Компетенции
6.1	Проработка лекционного материала. пользователя в среде Visual Prolog	Самостоятельная работа	5	8	ПК-3 ПК-1
6.2	Подготовка к лабораторным занятиям.	Самостоятельная работа	5	33	ПК-3 ПК-1
6.3	Подготовка к промежуточной аттестации	Экзамен	5	9	ПК-3 ПК-1

4. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

Структура и содержание фонда оценочных средств для проведения текущего контроля и промежуточной аттестации представлены в Приложении 1 к рабочей программе дисциплины.

5. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

5.1. Учебные, научные и методические издания

	Авторы, составители	Заглавие	Издательство, год	Библиотека / Количество
1	Прыкина Е. Н.	Основы логического программирования в среде Турбо Пролог: учебное пособие	Кемерово: Кемеровский государственный университет культуры и искусств (КемГУКИ), 2006	ЭБС «Университетская библиотека онлайн»
2	Городняя Л. В.	Основы функционального программирования: курс: учебное пособие	Москва: Интернет-Университет Информационных Технологий (ИНТУИТ), 2004	ЭБС «Университетская библиотека онлайн»
3		Информационные системы и технологии: журнал	Орел: Госуниверситет – УНПК, 2013	ЭБС «Университетская библиотека онлайн»

	Авторы, составители	Заглавие	Издательство, год	Библиотека / Количество
4	Ефимова Е. А.	Основы программирования на языке Visual Prolog: учебное пособие	Москва: Национальный Открытый Университет «ИНТУИТ», 2016	ЭБС «Университетская библиотека онлайн»
5		Информационные системы и технологии: журнал	Орел: Госуниверситет – УНПК, 2015	ЭБС «Университетская библиотека онлайн»
6	Салмина Н. Ю.	Функциональное программирование и интеллектуальные системы: учебное пособие	Томск: ТУСУ, 2016	ЭБС «Университетская библиотека онлайн»
7	Галкина, М. Ю.	Функциональное и логическое программирование: практикум	Новосибирск: Сибирский государственный университет телекоммуникаций и информатики, 2008	ЭБС «IPR SMART»
8	Козырева, Г. Ф.	Функциональное и логическое программирование: учебно-методическое пособие	Саратов: Ай Пи Эр Медиа, 2018	ЭБС «IPR SMART»
9	Галкина, М. Ю.	Функциональное и логическое программирование: учебное пособие	Новосибирск: Сибирский государственный университет телекоммуникаций и информатики, 2018	ЭБС «IPR SMART»

5.2. Профессиональные базы данных и информационные справочные системы

ИСС "КонсультантПлюс"
ИСС "Гарант"http://www.internet.garant.ru/
Общероссийский математический портал (информационная система)http://www.mathnet.ru/

5.3. Перечень программного обеспечения

Операционная система РЕД ОС
Visual Studio Code

5.4. Учебно-методические материалы для обучающихся с ограниченными возможностями здоровья

При необходимости по заявлению обучающегося с ограниченными возможностями здоровья учебно-методические материалы предоставляются в формах, адаптированных к ограничениям здоровья и восприятия информации. Для лиц с нарушениями зрения: в форме аудиофайла; в печатной форме увеличенным шрифтом. Для лиц с нарушениями слуха: в форме электронного документа; в печатной форме. Для лиц с нарушениями опорно-двигательного аппарата: в форме электронного документа; в печатной форме.

6. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

Помещения для всех видов работ, предусмотренных учебным планом, укомплектованы необходимой специализированной учебной мебелью и техническими средствами обучения:
- столы, стулья;
- персональный компьютер / ноутбук (переносной);
- проектор;
- экран / интерактивная доска.

Лабораторные занятия проводятся в компьютерных классах, рабочие места в которых оборудованы необходимыми лицензионными и/или свободно распространяемыми программными средствами и выходом в Интернет, и/или в специализированных лабораториях, предусмотренных образовательной программой.
Помещения для самостоятельной работы обучающихся оснащены компьютерной техникой с возможностью подключения к сети "Интернет" и обеспечением доступа к электронной информационно-образовательной среде.

7. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ

Методические указания по освоению дисциплины представлены в Приложении 2 к рабочей программе дисциплины.

Приложение 1

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

1 Описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкал оценивания

1.1 Показатели и критерии оценивания компетенций:

ЗУН, составляющие компетенцию	Показатели оценивания	Критерии оценивания	Средства оценивания
ПК-1: способен использовать методы и инструментальные средства исследования объектов профессиональной деятельности, формализации предметной области с учетом ограничений			
З. основные понятия из области разработки программных систем, применяемые метрики, методы и инструментальные средства	знает методологии разработки программного обеспечения; правила написания понятной тестовой документации (тест-кейсы, тестпланы) при подготовке к опросу и экзамену	сформировавшееся систематическое знание методологии разработки программного обеспечения; правил написания понятной тестовой документации (тест-кейсы, тестпланы) при ответе на вопросы опроса и экзамена	Вопросы к экзамену (1-109), практико-ориентированные задания к экзамену (1-10), лабораторные задания (раздел 2, 5), вопросы для опроса (1-109)
У. применять основные методы разработки программного обеспечения; применять основные инструменты разработки программного обеспечения	применяет методы управления процессами разработки требований, оценки рисков, приобретения, проектирования, конструирования, тестирования, эволюции и сопровождения при выполнении лабораторных и практико-ориентированных заданий	корректность применения методов управления процессами разработки требований, оценки рисков, приобретения, проектирования, конструирования, тестирования, эволюции и сопровождения при выполнении лабораторных и практико-ориентированных заданий	Вопросы к экзамену (1-109), практико-ориентированные задания к экзамену (1-10), лабораторные задания (раздел 2, 5), вопросы для опроса (1-109)
В. основными методами разработки программного обеспечения; основными инструментами разработки программного обеспечения	разрабатывает программное обеспечение при выполнении лабораторных и практико-ориентированных заданий	корректность разработки программного обеспечения при выполнении лабораторных и практико-ориентированных заданий	Вопросы к экзамену (1-109), практико-ориентированные задания к экзамену (1-10), лабораторные задания (раздел 2, 5), вопросы для опроса (1-109)
ПК-3: способен разрабатывать компоненты программных комплексов (в том числе интерфейсы, драйвера, компиляторы, загрузчики, сборщики, системные утилиты) и баз данных с использованием современных инструментальных средств и технологий программирования			
З: основы информатики и программирования	знает программные средства системного, прикладного и специального назначения, инструментальные средства, языки и системы программирования при подготовке к опросу и экзамену	сформировавшееся систематическое знание программных средств системного, прикладного и специального назначения, инструментальных средств, языков и систем программирования при ответе на вопросы опроса и экзамена	Вопросы к экзамену (1-109), практико-ориентированные задания к экзамену (1-10), лабораторные задания (раздел 2, 5), вопросы для опроса (1-109)
У: использовать современные технологии разработки программных продуктов	использует программные средства системного, прикладного и специального назначения, инструментальные средства, языки и системы программирования при выполнении лабораторных и практико-ориентированных заданий	сформировавшееся систематическое умение использования программных средств системного, прикладного и специального назначения, инструментальных средств, языков и систем программирования при выполнении лабораторных и практико-ориентированных заданий	Вопросы к экзамену (1-109), практико-ориентированные задания к экзамену (1-10), лабораторные задания (раздел 2, 5), вопросы для опроса (1-109)
В: навыками разработки алгоритмов в виде блок-схемы и составления плана ручного тестирования разрабатываемого программного продукта	владеет навыками применения программных средств системного, прикладного и специального назначения, инструментальных средств, языков и систем программирования при выполнении лабораторных и практико-ориентированных заданий	сформировавшееся систематическое владение навыками применения программных средств системного, прикладного и специального назначения, инструментальных средств, языков и систем программирования при выполнении лабораторных и практико-ориентированных заданий	Вопросы к экзамену (1-109), практико-ориентированные задания к экзамену (1-10), лабораторные задания (раздел 2, 5), вопросы для опроса (1-109)

1.2 Шкалы оценивания:

Текущий контроль успеваемости и промежуточная аттестация осуществляется в рамках накопительной балльно-рейтинговой системы в 100-балльной шкале.
- 84-100 баллов (оценка «отлично»);
- 67-83 баллов (оценка «хорошо»);
- 50-66 баллов (оценка удовлетворительно);
- 0-49 баллов (оценка неудовлетворительно).

2 Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций в процессе освоения образовательной программы

Вопросы к экзамену

1. Особенности и отличия от других методов.
2. Сравнение императивного и функционального программирования.
3. Выявление структур и закономерностей.
4. Основы функционального программирования на Haskell.
5. Строго функциональный язык.
6. Символьные данные: лямбда-выражения и представление данных.
7. Элементарные селекторы и конструкторы лямбда-выражений.
8. Элементарные предикаты и арифметика.
9. Рекурсивные функции: разбор случаев, рекурсивные определения, выбор подфункций.
10. Примеры обработки списков.
11. Дополнительные возможности.
12. Приемы программирования.
13. Рекурсивные функции и функционалы.
14. Решение различных типов задач на Haskell.
15. Основы лямбда-исчисления.
16. Лямбда-исчисление как формальная и структурированная система.
17. Представление данных в лямбда-исчислении.
18. Типы в лямбда-исчислении с учетом типов и структурных особенностей задач.
19. Исчисление предикатов как язык представления знаний.
20. Виды импликации.
21. Логика предикатов первого порядка, хорновские дизъюнкты.
22. Вычислительная модель.
23. Анализ структуры термов.
24. Принцип Робинсона.
25. Аккумуляторы.
26. Локальные определения.
27. Функционалы и другие функции высших порядков.
28. Примеры простых функциональных программ.
29. Значения и типы.
30. Полиморфные типы.
31. Типы, определяемые пользователем.
32. Бинарные конструкторы данных.
33. Рекурсивные типы.
34. Синонимы типов.
35. Встроенные типы.
36. Генераторы списков и арифметические последовательности.
37. Строки.
38. Функции.
39. Инфиксные операции.
40. Секции.
41. Декларация пользовательских операций.
42. Бесконечные структуры данных и ленивые вычисления.
43. Сопоставление с образцом.
44. As-образцы.
45. Универсальные образцы.
46. Семантика сопоставления с образцом.
47. Выражение case.
48. Ленивые образцы.
49. Лексическая видимость и вложенные формы.
50. Let-выражения.
51. Предложение where.
52. Двумерный синтаксис.
53. Классы типов и перегрузка.
54. Наследование

55. Множественное наследование.
56. Типы высшего порядка.
57. Контроль корректности типов.
58. Описание newtype.
59. Метки полей.
60. Строгие конструкторы данных.
61. Понятие монады.
62. Система ввода-вывода.
63. Базисные операции ввода- вывода.
64. Сетевое программирование.
65. Основы языка Visual Prolog.
66. Унификация и поиск с возвратом.
67. Списки и рекурсия.
68. Арифметические вычисления и сравнения.
69. Внутренняя база фактов Visual Prolog.
70. Возможности визуальной среды разработки Visual Prolog.
71. Основные элементы языка Prolog.
72. Алфавит языка Prolog.
73. Термы языка Prolog.
74. Виды термов языка Prolog: константы, переменные, структуры.
75. Литеры языка Prolog и их типы.
76. Интерпретация литер.
77. Операторы.
78. Свойства операторов (позиция, приоритет, ассоциативность).
79. Инфиксные, префиксные, постфиксные операторы.
80. Факты. Правила. Запись фактов и правил.
81. Предикаты. Цели, конъюнкция целей.
82. Общая схема доказательства целевого утверждения.
83. Встроенные предикаты для сравнения чисел.
84. Вычисление арифметических выражений.
85. Предикат is и его варианты в различных версиях.
86. Примеры программ с выполнением арифметических операций.
87. Согласование целевых утверждений.
88. Доказательство целевых утверждений при использовании механизма возврата.
89. Правила установления соответствия.
90. Недетерминизм первого и второго рода.
91. Понятие свободной и связанной переменной.
92. Примеры программ с использованием механизма возврата.
93. Операционная модель вычисления Prolog-программ.
94. Рекурсивное представление данных и программ.
95. Рекурсивные функции.
96. Структуры и деревья.
97. Список как частный вид структуры.
98. Формы записи списков.
99. Работа со списками.
100. Граничные условия и способы использования рекурсии.
101. Примеры программ с рекурсивными определениями.
102. Сортировка списков.
103. Отсечение и способы его использования в языке.
104. Причины использования отсечения.
105. Предикат cut.
106. Диаграмма работы программы с использованием отсечения.
107. Общие случаи использования отсечения.
108. Проблемы, связанные с использованием отсечения.
109. Развитие отсечения в отсечение-отрезок в некоторых вариантах.

Практико-ориентированные задания к экзамену

Задание 1. Результат вычисления выражения (quote (+ 5 7)) =? Определить функцию EXPP(n,k) = n*n* n * (k раз) - возведение в степень через умножение.

Задание 2. Результат вычисления выражения (EVAL(quote (+ 5 7))) =? Определить функцию maxeven(t) вычисления максимума среди четных элементов в линейном списке t из положительных чисел.

Задание 3. Результат вычисления выражения (car '(1 2 3)) Определить функцию ISSORT(v), где v - линейный список из вещественных чисел. Значением функции является true (T), если линейный список отсортирован отношением <= , и false (NIL) - в противном случае.

Задание 4. Результат вычисления выражения (car '((1 2) 3)) Определить функцию ISALLEV(v), где v - линейный список из целых чисел. Значением функции является true (T), если все элементы линейного списка четные числа, и false (NIL) - в противном случае.
Задание 5. Результат вычисления выражения (car (cdr '((1 2) 3))) Определить функцию ISALLOD(v), где v - линейный список из целых чисел. Значением функции является true (T), если все элементы линейного списка нечетные, и false (NIL) - в противном случае
Задание 6 Результат вычисления выражения '(+ 5 7) =? Определить функцию REPL (n) = (n n-1 ? 2 1).
Задание 7 Результат вычисления выражения (car (cdr '((1 2) 3))). Определить функцию delzero(v), где v - линейный список из вещественных чисел. Значением функции является линейный список, который получается в результате удаления всех нулевых элементов из списка v.
Задание 8 Результат вычисления выражения (car '(+1 2)). Определить функцию iseq(v), где v - линейный список из целых чисел. Значением функции является true (T), если все элементы линейного списка одинаковые, и false (NIL) - в противном случае.
Задание 9 Результат вычисления выражения (cons '1 '(2 3)). Определить функцию EXPALL(v,k), где v - линейный список из целых чисел, k -целое число>=0. Значением функции является линейный список (v1 k , v2 k, ?.), то есть все элементы списка v возвести в k - ую степень.
Задание 10 Результат вычисления выражения (cons '(1) '(2 3)). Определить функцию FEXP(v,k), где v - линейный список из целых чисел, k -целое число>=0. Значением функции является линейный список (v1 1 , v2 2 , ?.), где vi I - возведение элемента vi в i -ую степень.

Критерии оценивания:

- 84-100 баллов (оценка «отлично») - изложенный материал фактически верен, наличие глубоких исчерпывающих знаний в объеме пройденной программы дисциплины в соответствии с поставленными программой курса целями и задачами обучения; правильные, уверенные действия по применению полученных знаний на практике, грамотное и логически стройное изложение материала при ответе, усвоение основной и знакомство с дополнительной литературой;
- 67-83 баллов (оценка «хорошо») - наличие твердых и достаточно полных знаний в объеме пройденной программы дисциплины в соответствии с целями обучения, правильные действия по применению знаний на практике, четкое изложение материала, допускаются отдельные логические и стилистические погрешности, обучающийся усвоил основную литературу, рекомендованную в рабочей программе дисциплины;
- 50-66 баллов (оценка «удовлетворительно») - наличие твердых знаний в объеме пройденного курса в соответствии с целями обучения, изложение ответов с отдельными ошибками, уверенно исправленными после дополнительных вопросов; правильные в целом действия по применению знаний на практике;
- 0-49 баллов (оценка «неудовлетворительно») - ответы не связаны с вопросами, наличие грубых ошибок в ответе, непонимание сущности излагаемого вопроса, неумение применять знания на практике, неуверенность и неточность ответов на дополнительные и наводящие вопросы.

Лабораторные задания

1.Тематика лабораторных работ по разделам и темам

Раздел 2 «Разработка программных систем».

Лабораторное задание 1. Навыки работы с интерпретатором языка Haskell. Основные типы языка Haskell. Простейшие логические функции. Простейшие списочные функции.
Лабораторное задание 2 Интерпретатор языка Haskell. Символьные функции. Простейшие кортежные функции. Теоретико-множественные операции. Сортировка. Вспомогательные функции. Отладка.
Лабораторное задание 3 Списочные функции высших порядков. Арифметические последовательности. Генераторы списков.
Лабораторное задание 4 Бесконечные списки. Ввод-вывод. Нетривиальные функции.
Лабораторное задание 5 Простые числа и факторизация. Деревья. Деревья вычислений.
Лабораторное задание 6 Функции высшего порядка Let-связывание. Охраняющие условия. Пользовательские типы. Определение операторов. Рекурсивные типы. Модули. Абстрактные типы данных. Операции ввода-вывода

Раздел 5 «Разработка логических информационных систем».

Лабораторное задание **7** Знакомство с основами языка логического программирования Prolog. Возможности программной оболочки. Способы представления данных в виде фактов и правил вывода для записи программы на языке Prolog. Используя возможности, предоставляемые программной оболочкой, проверить правильность работы созданной программы. Оценка полученных результатов.
Лабораторное задание 8 Глобальные разделы программы Visual Prolog. Операции над списками. Арифметические действия. Стандартный предикат findall. Задача о классификации объектов
Лабораторное задание 9 Отсечение. Внутренняя база фактов Visual Prolog. Рекурсивные структуры данных. Составные списки
Лабораторное задание 10 Разработка программ, использующих для нахождения решений рекурсию и поиск с возвратом. Исследование различия в двух методах нахождения решений: рекурсии и поиске с возвратом. Исследование различия в двух методах нахождения решений: хвостовой и нехвостовой рекурсии. Оценка полученных результатов.
Лабораторное задание 11 Разработка программ для работы с рекурсивными структурами данных (списками и деревьями). Способы представления данных в виде фактов и правил вывода для записи программы на языке Prolog. Исследование методов работы с рекурсивными структурами данных (списками и деревьями. Оценка полученных результатов.

Критерии оценивания:

Максимальное количество баллов: 88 баллов.
Каждое задание оценивается максимум 8 баллов.

8 б. – задание выполнено верно;
7-5 б. – при выполнении задания были допущены неточности, не влияющие на результат;
4-3 б. – при выполнении задания были допущены ошибки;
2-1 б. – при выполнении задания были допущены существенные ошибки.
0 б. – задание не выполнено.

Вопросы для опроса

1. Особенности и отличия от других методов.
2. Сравнение императивного и функционального программирования.
3. Выявление структур и закономерностей.
4. Основы функционального программирования на Haskell.
5. Строго функциональный язык.
6. Символьные данные: лямбда-выражения и представление данных.
7. Элементарные селекторы и конструкторы лямбда-выражений.
8. Элементарные предикаты и арифметика.
9. Рекурсивные функции: разбор случаев, рекурсивные определения, выбор подфункций.
10. Примеры обработки списков.
11. Дополнительные возможности.
12. Приемы программирования.
13. Рекурсивные функции и функционалы.
14. Решение различных типов задач на Haskell.
15. Основы лямбда-исчисления.
16. Лямбда-исчисление как формальная и структурированная система.
17. Представление данных в лямбда-исчислении.
18. Типы в лямбда-исчислении с учетом типов и структурных особенностей задач.
19. Исчисление предикатов как язык представления знаний.
20. Виды импликации.
21. Логика предикатов первого порядка, хорновские дизъюнкты.
22. Вычислительная модель.
23. Анализ структуры термов.
24. Принцип Робинсона.
25. Аккумуляторы.
26. Локальные определения.
27. Функционалы и другие функции высших порядков.
28. Примеры простых функциональных программ.
29. Значения и типы.
30. Полиморфные типы.
31. Типы, определяемые пользователем.
32. Бинарные конструкторы данных.
33. Рекурсивные типы.
34. Синонимы типов.
35. Встроенные типы.
36. Генераторы списков и арифметические последовательности.
37. Строки.
38. Функции.
39. Инфиксные операции.
40. Секции.
41. Декларация пользовательских операций.
42. Бесконечные структуры данных и ленивые вычисления.
43. Сопоставление с образцом.
44. As-образцы.
45. Универсальные образцы.
46. Семантика сопоставления с образцом.
47. Выражение case.
48. Ленивые образцы.
49. Лексическая видимость и вложенные формы.
50. Let-выражения.
51. Предложение where.
52. Двумерный синтаксис.
53. Классы типов и перегрузка.
54. Наследование
55. Множественное наследование.
56. Типы высшего порядка.
57. Контроль корректности типов.
58. Описание newtype.
59. Метки полей.

60. Строгие конструкторы данных.
61. Понятие монады.
62. Система ввода-вывода.
63. Базисные операции ввода- вывода.
64. Сетевое программирование.
65. Основы языка Visual Prolog.
66. Унификация и поиск с возвратом.
67. Списки и рекурсия.
68. Арифметические вычисления и сравнения.
69. Внутренняя база фактов Visual Prolog.
70. Возможности визуальной среды разработки Visual Prolog.
71. Основные элементы языка Prolog.
72. Алфавит языка Prolog.
73. Термы языка Prolog.
74. Виды термов языка Prolog: константы, переменные, структуры.
75. Литеры языка Prolog и их типы.
76. Интерпретация литер.
77. Операторы.
78. Свойства операторов (позиция, приоритет, ассоциативность).
79. Инфиксные, префиксные, постфиксные операторы.
80. Факты. Правила. Запись фактов и правил.
81. Предикаты. Цели, конъюнкция целей.
82. Общая схема доказательства целевого утверждения.
83. Встроенные предикаты для сравнения чисел.
84. Вычисление арифметических выражений.
85. Предикат is и его варианты в различных версиях.
86. Примеры программ с выполнением арифметических операций.
87. Согласование целевых утверждений.
88. Доказательство целевых утверждений при использовании механизма возврата.
89. Правила установления соответствия.
90. Недетерминизм первого и второго рода.
91. Понятие свободной и связанной переменной.
92. Примеры программ с использованием механизма возврата.
93. Операционная модель вычисления Prolog-программ.
94. Рекурсивное представление данных и программ.
95. Рекурсивные функции.
96. Структуры и деревья.
97. Список как частный вид структуры.
98. Формы записи списков.
99. Работа со списками.
100. Граничные условия и способы использования рекурсии.
101. Примеры программ с рекурсивными определениями.
102. Сортировка списков.
103. Отсечение и способы его использования в языке.
104. Причины использования отсечения.
105. Предикат cut.
106. Диаграмма работы программы с использованием отсечения.
107. Общие случаи использования отсечения.
108. Проблемы, связанные с использованием отсечения.
109. Развитие отсечения в отсечение-отрезок в некоторых вариантах.

Критерии оценивания:
Максимальное количество баллов: 12 баллов.
Во время опроса обучаемому задаются 4 вопросов.
За один ответ обучаемый получает:
3 б. – за правильный ответ;
2 б. – при ответе были допущены ошибки;
1 б. – при ответе были допущены существенные ошибки.
0 б. – не ответил на вопрос.

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

Процедуры оценивания включают в себя текущий контроль и промежуточную аттестацию.

Текущий контроль успеваемости проводится с использованием оценочных средств, представленных в п. 2 данного приложения. Результаты текущего контроля доводятся до сведения студентов до промежуточной аттестации.

Промежуточная аттестация проводится в форме экзамена.

Экзамен проводится по расписанию промежуточной аттестации в письменном виде. Количество вопросов в экзаменационном задании – 3: два теоретических вопроса и одно практико-ориентированное задание. Проверка ответов и объявление результатов производится в день экзамена. Результаты аттестации заносятся в ведомость и зачетную книжку студента. Студенты, не прошедшие промежуточную аттестацию по графику сессии, должны ликвидировать задолженность в установленном порядке.

Приложение 2

МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ

Учебным планом предусмотрены следующие виды занятий:
- лекции;
- лабораторные занятия.

В ходе лекционных занятий рассматриваются основные теоретические вопросы, даются рекомендации для самостоятельной работы и подготовке к лабораторным занятиям.

В ходе лабораторных занятий углубляются и закрепляются знания студентов по ряду рассмотренных на лекциях вопросов, развиваются навыки практической работы.

При подготовке к лабораторным занятиям каждый студент должен:
– изучить рекомендованную учебную литературу;
– изучить конспекты лекций;
– подготовить ответы на все вопросы по изучаемой теме.

В процессе подготовки к лабораторным занятиям студенты могут воспользоваться консультациями преподавателя.

Вопросы, не рассмотренные на лабораторных занятиях, должны быть изучены студентами в ходе самостоятельной работы. Контроль самостоятельной работы студентов над учебной программой курса осуществляется в ходе занятий методом опроса. В ходе самостоятельной работы каждый студент обязан прочитать основную и по возможности дополнительную литературу по изучаемой теме, дополнить конспекты недостающим материалом, выписками из рекомендованных первоисточников. Выделить непонятные термины, найти их значение в энциклопедических словарях. Студент должен готовиться к предстоящему лабораторному занятию по всем, обозначенным в рабочей программе дисциплины вопросам.

Для подготовки к занятиям, текущему контролю и промежуточной аттестации студенты могут воспользоваться электронно-библиотечными системами. Также обучающиеся могут взять на дом необходимую литературу на абонементе университетской библиотеки или воспользоваться читальными залами.