МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования

«ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ СИСТЕМ
УПРАВЛЕНИЯ И РАДИОЭЛЕКТРОНИКИ»
(ТУСУР)

УТВЕРЖДАЮ
Проректор по учебной работе
_____П.В. Сенченко
«23» _____12_____ 2020 г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

ФУНКЦИОНАЛЬНОЕ И ЛОГИЧЕСКОЕ ПРОГРАММИРОВАНИЕ

Уровень образования: **высшее образование - бакалавриат**
Направление подготовки / специальность: **09.03.04 Программная инженерия**
Направленность (профиль) / специализация: **Индустриальная разработка программных продуктов**
Форма обучения: **заочная (в том числе с применением дистанционных образовательных технологий)**
Факультет: **Факультет дистанционного обучения (ФДО)**
Кафедра: **Кафедра автоматизации обработки информации (АОИ)**
Курс: **4, 5**
Семестр: **7, 8, 9**
Учебный план набора 2021 года

Объем дисциплины и виды учебной деятельности

Виды учебной деятельности	7 семестр	8 семестр	9 семестр	Всего	Единицы
Лабораторные занятия	4	4		8	часов
Курсовой проект			4	4	часов
Самостоятельная работа	121	151	62	334	часов
Самостоятельная работа под руководством преподавателя	8	14	6	28	часов
Контрольные работы	2	2		4	часов
Подготовка и сдача экзамена	9	9		18	часов
Общая трудоемкость (включая промежуточную аттестацию)	144	180	72	396	часов
				11	з.е.

Формы промежуточной аттестация	Семестр	Количество
Экзамен	7	
Контрольные работы	7	1
Экзамен	8	
Контрольные работы	8	1
Курсовой проект	9	

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Сенченко П.В.
Должность: Проректор по УР
Дата подписания: 23.12.2020
Уникальный программный ключ:
a1119608-cdff-4455-b54e-5235117c185c

Томск

Согласована на портале № 66608

1. Общие положения

1.1. Цели дисциплины

1. формирование у студентов профессиональных знаний и практических навыков по разработке и созданию моделей интеллектуальных систем с помощью языков функционального и логического программирования.

1.2. Задачи дисциплины

1. Получить знания и овладеть понятийным аппаратом: рекурсия; функциональное программирование; λ-исчисление; функционалы; предикаты первого порядка; логическое программирование; интеллектуальные системы.
2. Получить практические навыки написания программ с использованием технологий функционального программирования.

2. Место дисциплины в структуре ОПОП

Блок дисциплин: Б1. Дисциплины (модули).
Часть блока дисциплин: Часть, формируемая участниками образовательных отношений.
Модуль дисциплин: Модуль направленности (профиля) (major).
Индекс дисциплины: Б1.В.01.01.
Реализуется с применением электронного обучения, дистанционных образовательных технологий.

3. Перечень планируемых результатов обучения по дисциплине, соотнесенных с индикаторами достижения компетенций

Процесс изучения дисциплины направлен на формирование следующих компетенций в соответствии с ФГОС ВО и основной образовательной программой (таблица 3.1):
Таблица 3.1 – Компетенции и индикаторы их достижения

Компетенция	Индикаторы достижения компетенции	Планируемые результаты обучения по дисциплине
Универсальные компетенции		
-	-	-
Общепрофессиональные компетенции		
-	-	-
Профессиональные компетенции		
ПКР-4. Готовность к использованию методов и инструментальных средств исследования объектов профессиональной деятельности	ПКР-4.1. Знает современные инструментальные средства программного обеспечения.	Знает языки функционального и логического программирования; основные методы и средства эффективной разработки программного продукта.
	ПКР-4.2. Умеет анализировать и выбирать инструментальные средства программного обеспечения.	Умеет использовать методы и технологии разработки для генерации исполняемого кода; анализировать поставленные задачи, разрабатывать алгоритмы, представлять данные для решения поставленных задач.
	ПКР-4.3. Владеет навыками использования методов и инструментальных средств исследования программного обеспечения.	Владеет основными методологиями процессов разработки программного обеспечения; языками Лисп и Пролог для построения моделей искусственного интеллекта.

Согласована на портале № 66608

ПКР-6. Владение навыками моделирования, анализа и использования формальных методов конструирования программного обеспечения	ПКР-6.1. Знает основы моделирования и формальные методы конструирования программного обеспечения.	Знает типовые роли в процессе разработки программного обеспечения; математические основы предикатов первого порядка ; математические основы лямбда-исчисления.
	ПКР-6.2. Умеет использовать формальные методы конструирования программного обеспечения.	Умеет разрабатывать модели различных классов систем с применением языков функционального и логического программирования; осуществлять разработку программного обеспечения на языках Лисп и Пролог
	ПКР-6.3. Владеет методами формализации и моделирования программного обеспечения.	Владеет математическим аппаратом, применяемым в функциональном и логическом программировании.

4. Объем дисциплины в зачетных единицах с указанием количества академических часов, выделенных на контактную работу обучающихся с преподавателем и на самостоятельную работу обучающихся

Общая трудоемкость дисциплины составляет 11 зачетных единиц, 396 академических часов.

Распределение трудоемкости дисциплины по видам учебной деятельности представлено в таблице 4.1.

Таблица 4.1 – Трудоемкость дисциплины по видам учебной деятельности

Виды учебной деятельности	Всего часов	Семестры		
		7 семестр	8 семестр	9 семестр
Контактная аудиторная работа обучающихся с преподавателем, всего	44	14	20	10
Лабораторные занятия	8	4	4	
Курсовой проект	4			4
Самостоятельная работа под руководством преподавателя	28	8	14	6
Контрольные работы	4	2	2	
Самостоятельная работа обучающихся, в т.ч. контактная внеаудиторная работа обучающихся с преподавателем, всего	334	121	151	62
Самостоятельное изучение тем (вопросов) теоретической части дисциплины	234	91	115	28
Подготовка к контрольной работе	38	14	24	
Подготовка к лабораторной работе	16	8	8	
Написание отчета по лабораторной работе	12	8	4	
Выполнение курсового проекта	18			18
Написание отчета по курсовому проекту	16			16
Подготовка и сдача экзамена	18	9	9	
Общая трудоемкость (в часах)	396	144	180	72
Общая трудоемкость (в з.е.)	11	4	5	2

5. Структура и содержание дисциплины

5.1. Разделы (темы) дисциплины и виды учебной деятельности

Структура дисциплины по разделам (темам) и видам учебной деятельности приведена в таблице 5.1.

Согласована на портале № 66608

Таблица 5.1 – Разделы (темы) дисциплины и виды учебной деятельности

Названия разделов (тем) дисциплины	Лаб. раб.	Контр. раб.	Курс. пр.	СРП, ч.	Сам. раб., ч	Всего часов (без промежуточной аттестации)	Формируемые компетенции
7 семестр							
1 Что такое декларативное программирование?	-	2	-	1	10	13	ПКР-4, ПКР-6
2 Функциональный взгляд на вычисления	4		-	1	36	41	ПКР-4, ПКР-6
3 Основы языка ЛИСП	-		-	2	25	27	ПКР-4, ПКР-6
4 Рекурсия	-		-	1	14	15	ПКР-4, ПКР-6
5 Лямбды-исчисления. Функции высокого порядков	-		-	2	26	28	ПКР-4, ПКР-6
6 Несколько более сложных задач	-		-	1	10	11	ПКР-4, ПКР-6
Итого за семестр	4	2	0	8	121	135	
8 семестр							
7 Математические основы логического программирования	-	2	-	7	66	75	ПКР-4, ПКР-6
8 Введение в Пролог	4		-	7	85	96	ПКР-4, ПКР-6
Итого за семестр	4	2	0	14	151	171	
9 семестр							
9 Структуры данных	-	-	4	2	22	28	ПКР-4, ПКР-6
10 Управление повторением в Прологе	-	-		2	18	20	ПКР-4, ПКР-6
11 Внелогические предикаты Пролога	-	-		2	22	24	ПКР-4, ПКР-6
Итого за семестр	0	0	4	6	62	72	
Итого	8	4	4	28	334	378	

5.2. Содержание разделов (тем) дисциплины

Содержание разделов (тем) дисциплины приведено в таблице 5.2.

Таблица 5.2 – Содержание разделов (тем) дисциплины

Названия разделов (тем) дисциплины	Содержание разделов (тем) дисциплины	СРП, ч	Формируемые компетенции
7 семестр			
1 Что такое декларативное программирование?	Машина фон Неймана и процедурное программирование . От программирования процедурного к декларативному	1	ПКР-4, ПКР-6
	Итого	1	
2 Функциональный взгляд на вычисления	Определение функции. Функциональная композиция.	1	ПКР-4, ПКР-6
	Итого	1	

Согласована на портале № 66608

3 Основы языка ЛИСП	О языке Лисп (история и особенности). Основные структуры данных (символы, числа, списки). Понятие функции. Базовые функции. Определение функций. Некоторые специальные формы	2	ПКР-4, ПКР-6
	Итого	2	
4 Рекурсия	Знакомство с рекурсией . Встроенные рекурсивные функции для списков. Различные виды рекурсии	1	ПКР-4, ПКР-6
	Итого	1	
5 Лямбды-исчисления. Функции высокого порядков	Ламбда-исчисление как формальная система. Ламбда-исчисление как язык программирования. Функции высших порядков. Различие между данными и функциями. Функционалы. Работа с графами и деревьями: представление, обработка, поиск пути на графе.	2	ПКР-4, ПКР-6
	Итого	2	
6 Несколько более сложных задач	Работа со списками. Задания с функциями и рекурсиями	1	ПКР-4, ПКР-6
	Итого	1	
	Итого за семестр	8	
8 семестр			
7 Математические основы логического программирования	История. Логический язык первого порядка. Формальные теории первого порядка. Доказательство от противного. Задача об автоматическом доказательстве теорем. Предваренная нормальная форма . Сколемизация.. Конъюнктивная нормальная форма.. Сведение к дизъюнктам. Правило резолюции для исчисления высказываний. Унификация. Правило резолюции для исчисления предикатов. Алгоритм резолюций. Опровержение методом резолюций	7	ПКР-4, ПКР-6
	Итого	7	
8 Введение в Пролог	Хорновская логическая программа. Сеанс работы с интерпретатором Пролога. Пример Пролог-программы: родственные отношения. Общие принципы поиска ответов на вопросы системой Пролог. Декларативная и процедурная семантика программ. 6 Синтаксис языка SWI-Prolog. Порядок предложений и целей	7	ПКР-4, ПКР-6
	Итого	7	
	Итого за семестр	14	
9 семестр			

Согласована на портале № 66608

9 Структуры данных	Предикат унификации. Арифметические выражения. Примеры программ с числами. Дифференцирование. Списки. Синтаксис и семантика списков. Некоторые предикаты для списков. Структуры. Модификация синтаксиса (операторная запись)	2	ПКР-4, ПКР-6
	Итого	2	
10 Управление повторением в Прологе	Отсечение. Определение отсечения. Примеры программ с отсечением. Отрицание как неудача. Трудности с отсечением и отрицанием. Рекурсия	2	ПКР-4, ПКР-6
	Итого	2	
11 Внелогические предикаты Пролога	Анализ и синтез термов. Проверка типа термов. Создание и декомпозиция термов. Ввод и вывод. Метапрограммирование. Операции с базой данных	2	ПКР-4, ПКР-6
	Итого	2	
Итого за семестр		6	
Итого		28	

5.3. Контрольные работы

Виды контрольных работ и часы на контрольные работы приведены в таблице 5.3.

Таблица 5.3 – Контрольные работы

№ п.п.	Виды контрольных работ	Трудоемкость, ч	Формируемые компетенции
7 семестр			
1	Контрольная работа с автоматизированной проверкой	2	ПКР-4, ПКР-6
Итого за семестр		2	
8 семестр			
2	Контрольная работа	2	ПКР-4, ПКР-6
Итого за семестр		2	
Итого		4	

5.4. Лабораторные занятия

Наименование лабораторных работ приведено в таблице 5.4.

Таблица 5.4 – Наименование лабораторных работ

Названия разделов (тем) дисциплины	Наименование лабораторных работ	Трудоемкость, ч	Формируемые компетенции
7 семестр			
2 Функциональный взгляд на вычисления	Разработка программы с использованием языка Лисп	4	ПКР-4, ПКР-6
	Итого	4	
Итого за семестр		4	
8 семестр			
8 Введение в Пролог	Разработка программ с использованием языка Пролог	4	ПКР-4, ПКР-6
	Итого	4	
Итого за семестр		4	

Согласована на портале № 66608

Итого	8	

5.5.Контроль самостоятельной работы (курсовой проект)

Содержание самостоятельной работы и ее трудоемкость, а также формируемые компетенции в рамках выполнения курсового проекта представлены в таблице 5.5.

Таблица 5.5 – Содержание самостоятельной работы и ее трудоемкость в рамках выполнения курсового проекта

Содержание самостоятельной работы в рамках выполнения курсового проекта	Трудоемкость, ч	Формируемые компетенции
9 семестр		
Выполнение курсовой работы по логическому программированию ("логическое программирование" понимается в "широком" смысле этого слова) требует решения одной из вариантов задач и, как результат, создания программы на Прологе или Лиспе и написания пояснительной записки к работе.	4	ПКР-4, ПКР-6
Итого за семестр	4	
Итого	4	

Примерная тематика курсовых проектов:

1. Упрощение электрических цепей
2. Программа для алгебраических вычислений
3. Упрощение арифметических выражений
4. Определение связности графа на Прологе
5. Определение связности графа на Лиспе

5.6. Самостоятельная работа

Виды самостоятельной работы, трудоемкость и формируемые компетенции представлены в таблице 5.6.

Таблица 5.6. – Виды самостоятельной работы, трудоемкость и формируемые компетенции

Названия разделов (тем) дисциплины	Виды самостоятельной работы	Трудоемкость, ч	Формируемые компетенции	Формы контроля
7 семестр				
1 Что такое декларативное программирование?	Самостоятельное изучение тем (вопросов) теоретической части дисциплины	8	ПКР-4, ПКР-6	Тестирование, Экзамен
	Подготовка к контрольной работе	2	ПКР-4, ПКР-6	Контрольная работа
	Итого	10		

Согласована на портале № 66608

2 Функциональный взгляд на вычисления	Самостоятельное изучение тем (вопросов) теоретической части дисциплины	24	ПКР-4, ПКР-6	Тестирование, Экзамен
	Подготовка к контрольной работе	4	ПКР-4, ПКР-6	Контрольная работа
	Подготовка к лабораторной работе	4	ПКР-4, ПКР-6	Лабораторная работа
	Написание отчета по лабораторной работе	4	ПКР-4, ПКР-6	Отчет по лабораторной работе
	Итого	36		
3 Основы языка ЛИСП	Подготовка к лабораторной работе	4	ПКР-4, ПКР-6	Лабораторная работа
	Написание отчета по лабораторной работе	4	ПКР-4, ПКР-6	Отчет по лабораторной работе
	Самостоятельное изучение тем (вопросов) теоретической части дисциплины	15	ПКР-4, ПКР-6	Тестирование, Экзамен
	Подготовка к контрольной работе	2	ПКР-4, ПКР-6	Контрольная работа
	Итого	25		
4 Рекурсия	Самостоятельное изучение тем (вопросов) теоретической части дисциплины	12	ПКР-4, ПКР-6	Тестирование, Экзамен
	Подготовка к контрольной работе	2	ПКР-4, ПКР-6	Контрольная работа
	Итого	14		
5 Лямбды-исчисления. Функции высокого порядков	Самостоятельное изучение тем (вопросов) теоретической части дисциплины	24	ПКР-4, ПКР-6	Тестирование, Экзамен
	Подготовка к контрольной работе	2	ПКР-4, ПКР-6	Контрольная работа
	Итого	26		

Согласована на портале № 66608

6 Несколько более сложных задач	Самостоятельное изучение тем (вопросов) теоретической части дисциплины	8	ПКР-4, ПКР-6	Тестирование, Экзамен
	Подготовка к контрольной работе	2	ПКР-4, ПКР-6	Контрольная работа
	Итого	10		
Итого за семестр		121		
	Подготовка и сдача экзамена	9		Экзамен
8 семестр				
7 Математические основы логического программирования	Самостоятельное изучение тем (вопросов) теоретической части дисциплины	54	ПКР-4, ПКР-6	Тестирование, Экзамен
	Подготовка к контрольной работе	12	ПКР-4, ПКР-6	Контрольная работа
	Итого	66		
8 Введение в Пролог	Самостоятельное изучение тем (вопросов) теоретической части дисциплины	61	ПКР-4, ПКР-6	Тестирование, Экзамен
	Подготовка к контрольной работе	12	ПКР-4, ПКР-6	Контрольная работа
	Подготовка к лабораторной работе	8	ПКР-4, ПКР-6	Лабораторная работа
	Написание отчета по лабораторной работе	4	ПКР-4, ПКР-6	Отчет по лабораторной работе
	Итого	85		
Итого за семестр		151		
	Подготовка и сдача экзамена	9		Экзамен
9 семестр				
9 Структуры данных	Самостоятельное изучение тем (вопросов) теоретической части дисциплины	12	ПКР-4, ПКР-6	Тестирование
	Выполнение курсового проекта	6	ПКР-4, ПКР-6	Курсовой проект
	Написание отчета по курсовому проекту	4	ПКР-4, ПКР-6	Отчет по курсовому проекту
	Итого	22		

Согласована на портале № 66608

10 Управление повторением в Прологе	Самостоятельное изучение тем (вопросов) теоретической части дисциплины	8	ПКР-4, ПКР-6	Тестирование
	Выполнение курсового проекта	6	ПКР-4, ПКР-6	Курсовой проект
	Написание отчета по курсовому проекту	4	ПКР-4, ПКР-6	Отчет по курсовому проекту
	Итого	18		
11 Внелогические предикаты Пролога	Самостоятельное изучение тем (вопросов) теоретической части дисциплины	8	ПКР-4, ПКР-6	Тестирование
	Выполнение курсового проекта	6	ПКР-4, ПКР-6	Курсовой проект
	Написание отчета по курсовому проекту	8	ПКР-4, ПКР-6	Отчет по курсовому проекту
	Итого	22		
Итого за семестр		62		
Итого		352		

5.7. Соответствие компетенций, формируемых при изучении дисциплины, и видов учебной деятельности

Соответствие компетенций, формируемых при изучении дисциплины, и видов учебной деятельности представлено в таблице 5.7.

Таблица 5.7 – Соответствие компетенций, формируемых при изучении дисциплины, и видов занятий

Формируемые компетенции	Виды учебной деятельности					Формы контроля
	Лаб. раб.	Курс. пр.	Конт.Раб.	СРП	Сам. раб.	
ПКР-4	+	+	+	+	+	Контрольная работа, Курсовой проект, Лабораторная работа, Отчет по курсовому проекту, Отчет по лабораторной работе, Тестирование, Экзамен
ПКР-6	+	+	+	+	+	Контрольная работа, Курсовой проект, Лабораторная работа, Отчет по курсовому проекту, Отчет по лабораторной работе, Тестирование, Экзамен

6. Рейтинговая система для оценки успеваемости обучающихся

Рейтинговая система не используется

7. Учебно-методическое и информационное обеспечение дисциплины

7.1. Основная литература

Согласована на портале № 66608

1. Зюзьков В.М. Функциональное программирование: Учебное пособие. — Томск : Томский межвузовский центр дистанционного образования, 2005. Доступ из личного кабинета студента. [Электронный ресурс]: — Режим доступа: https://study.tusur.ru/study/library/.

2. Зюзьков В.М. Логическое программирование: Учебное пособие. — Томск : Томский межвузовский центр дистанционного образования, 2005. Доступ из личного кабинета студента. [Электронный ресурс]: — Режим доступа: https://study.tusur.ru/study/library/.

7.2. Дополнительная литература

1. Функциональное программирование и интеллектуальные системы : Учебное пособие / Салмина Н. Ю. - 2016. Доступ из личного кабинета студента. [Электронный ресурс]: — Режим доступа: https://study.tusur.ru/study/library/.

7.3. Учебно-методические пособия

7.3.1. Обязательные учебно-методические пособия

1. Функциональное и логическое программирование : учебное методическое пособие / В. М. Зюзьков, Н. Ю. Салмина. – Томск : ФДО, ТУСУР, 2019. – 66 с Доступ из личного кабинета студента. [Электронный ресурс]: — Режим доступа: https://study.tusur.ru/study/library/.

2. Зюзьков В.М. Функциональное и логическое программирование : методические указания по организации самостоятельной работы для студентов заочной формы обучения технических направлений, обучающихся с применением дистанционных образовательных технологий / В.М. Зюзьков. – Томск : ФДО, ТУСУР, 2018. Доступ из личного кабинета студента. [Электронный ресурс]: — Режим доступа: https://study.tusur.ru/study/library/.

7.3.2. Учебно-методические пособия для лиц с ограниченными возможностями здоровья и инвалидов

Учебно-методические материалы для самостоятельной и аудиторной работы обучающихся из числа лиц с ограниченными возможностями здоровья и инвалидов предоставляются в формах, адаптированных к ограничениям их здоровья и восприятия информации.

Для лиц с нарушениями зрения:

– в форме электронного документа;

– 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.

7.4. Иное учебно-методическое обеспечение

1. Зюзьков В.М. Функциональное и логическое программирование [Электронный ресурс]: электронный курс / В. М. Зюзьков. – Томск ТУСУР, ФДО, 2018. (доступ из личного кабинета студента) .

2. Зюзьков В.М. Функциональное и логическое программирование [Электронный ресурс]: электронный курс / В. М. Зюзьков, Н.Ю. Салмина . – Томск ТУСУР, ФДО, 2019. (доступ из личного кабинета студента) .

7.5. Современные профессиональные базы данных и информационные справочные системы

1. При изучении дисциплины рекомендуется обращаться к современным базам данных, информационно-справочным и поисковым системам, к которым у ТУСУРа открыт доступ: https://lib.tusur.ru/ru/resursy/bazy-dannyh.

8. Материально-техническое и программное обеспечение дисциплины

8.1. Общие требования к материально-техническому и программному обеспечению дисциплины

Согласована на портале № 66608

Учебные аудитории для проведения занятий лабораторного типа, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации, для самостоятельной работы студентов

634034, Томская область, г. Томск, Вершинина улица, д. 74, 207 ауд.

Описание имеющегося оборудования:

- Веб-камера - 6 шт.;
- Наушники с микрофоном - 6 шт.;
- Комплект специализированной учебной мебели;
- Рабочее место преподавателя.

Программное обеспечение:

- 7-Zip;
- Google Chrome;
- Kaspersky Endpoint Security для Windows;
- LibreOffice 7.0.6.2;
- Microsoft Windows;
- SWI-Prolog (с возможностью удаленного доступа);
- XLisp (с возможностью удаленного доступа);

8.2. Материально-техническое и программное обеспечение для самостоятельной работы

Для самостоятельной работы используются учебные аудитории (компьютерные классы), расположенные по адресам:

- 634050, Томская область, г. Томск, Ленина проспект, д. 40, 233 ауд.;
- 634045, Томская область, г. Томск, ул. Красноармейская, д. 146, 209 ауд.;
- 634034, Томская область, г. Томск, Вершинина улица, д. 47, 126 ауд.;
- 634034, Томская область, г. Томск, Вершинина улица, д. 74, 207 ауд.

Описание имеющегося оборудования:

- учебная мебель;
- компьютеры;
- компьютеры подключены к сети «Интернет» и обеспечивают доступ в электронную информационно-образовательную среду ТУСУРа.

Перечень программного обеспечения:

- Microsoft Windows;
- OpenOffice;
- Kaspersky Endpoint Security 10 для Windows;
- 7-Zip;
- Google Chrome.

8.3. Материально-техническое обеспечение дисциплины для лиц с ограниченными возможностями здоровья и инвалидов

Освоение дисциплины лицами с ограниченными возможностями здоровья и инвалидами осуществляется с использованием средств обучения общего и специального назначения.

При занятиях с обучающимися с **нарушениями слуха** предусмотрено использование звукоусиливающей аппаратуры, мультимедийных средств и других технических средств приема/передачи учебной информации в доступных формах, мобильной системы преподавания для обучающихся с инвалидностью, портативной индукционной системы. Учебная аудитория, в которой занимаются обучающиеся с нарушением слуха, оборудована компьютерной техникой, аудиотехникой, видеотехникой, электронной доской, мультимедийной системой.

При занятиях с обучающимися с **нарушениями зрения** предусмотрено использование в лекционных и учебных аудиториях возможности просмотра удаленных объектов (например, текста на доске или слайда на экране) при помощи видеоувеличителей для комфортного просмотра.

При занятиях с обучающимися с **нарушениями опорно-двигательного аппарата** используются альтернативные устройства ввода информации и другие технические средства приема/передачи учебной информации в доступных формах, мобильной системы обучения для людей с инвалидностью.

Согласована на портале № 66608

9. Оценочные материалы и методические рекомендации по организации изучения дисциплины

9.1. Содержание оценочных материалов для текущего контроля и промежуточной аттестации

Для оценки степени сформированности и уровня освоения закрепленных за дисциплиной компетенций используются оценочные материалы, представленные в таблице 9.1.

Таблица 9.1 – Формы контроля и оценочные материалы

Названия разделов (тем) дисциплины	Формируемые компетенции	Формы контроля	Оценочные материалы (ОМ)
1 Что такое декларативное программирование?	ПКР-4, ПКР-6	Контрольная работа	Примерный перечень тем и тестовых заданий на контрольные работы
		Тестирование	Примерный перечень тестовых заданий
		Экзамен	Перечень экзаменационных вопросов
2 Функциональный взгляд на вычисления	ПКР-4, ПКР-6	Контрольная работа	Примерный перечень тем и тестовых заданий на контрольные работы
		Лабораторная работа	Темы лабораторных работ
		Тестирование	Примерный перечень тестовых заданий
		Экзамен	Перечень экзаменационных вопросов
		Отчет по лабораторной работе	Темы лабораторных работ
3 Основы языка ЛИСП	ПКР-4, ПКР-6	Контрольная работа	Примерный перечень тем и тестовых заданий на контрольные работы
		Лабораторная работа	Темы лабораторных работ
		Тестирование	Примерный перечень тестовых заданий
		Экзамен	Перечень экзаменационных вопросов
		Отчет по лабораторной работе	Темы лабораторных работ
4 Рекурсия	ПКР-4, ПКР-6	Контрольная работа	Примерный перечень тем и тестовых заданий на контрольные работы
		Тестирование	Примерный перечень тестовых заданий
		Экзамен	Перечень экзаменационных вопросов

Согласована на портале № 66608

5 Лямбды-исчисления. Функции высокого порядков	ПКР-4, ПКР-6	Контрольная работа	Примерный перечень тем и тестовых заданий на контрольные работы
		Тестирование	Примерный перечень тестовых заданий
		Экзамен	Перечень экзаменационных вопросов
6 Несколько более сложных задач	ПКР-4, ПКР-6	Контрольная работа	Примерный перечень тем и тестовых заданий на контрольные работы
		Тестирование	Примерный перечень тестовых заданий
		Экзамен	Перечень экзаменационных вопросов
7 Математические основы логического программирования	ПКР-4, ПКР-6	Контрольная работа	Примерный перечень тем и тестовых заданий на контрольные работы
		Тестирование	Примерный перечень тестовых заданий
		Экзамен	Перечень экзаменационных вопросов
8 Введение в Пролог	ПКР-4, ПКР-6	Контрольная работа	Примерный перечень тем и тестовых заданий на контрольные работы
		Лабораторная работа	Темы лабораторных работ
		Тестирование	Примерный перечень тестовых заданий
		Экзамен	Перечень экзаменационных вопросов
		Отчет по лабораторной работе	Темы лабораторных работ
9 Структуры данных	ПКР-4, ПКР-6	Отчет по курсовому проекту	Примерный перечень тематик курсовых проектов
		Тестирование	Примерный перечень тестовых заданий
		Курсовой проект	Примерный перечень тематик курсовых проектов
10 Управление повторением в Прологе	ПКР-4, ПКР-6	Отчет по курсовому проекту	Примерный перечень тематик курсовых проектов
		Тестирование	Примерный перечень тестовых заданий
		Курсовой проект	Примерный перечень тематик курсовых проектов

Согласована на портале № 66608

11 Внелогические предикаты Пролога	ПКР-4, ПКР-6	Отчет по курсовому проекту	Примерный перечень тематик курсовых проектов
		Тестирование	Примерный перечень тестовых заданий
		Курсовой проект	Примерный перечень тематик курсовых проектов

Шкала оценки сформированности отдельных планируемых результатов обучения по дисциплине приведена в таблице 9.2.

Таблица 9.2 – Шкала оценки сформированности планируемых результатов обучения по дисциплине

Оценка	Баллы за ОМ	Формулировка требований к степени сформированности планируемых результатов обучения		
		знать	уметь	владеть
2 (неудовлетворительно)	< 60% от максимальной суммы баллов	отсутствие знаний или фрагментарные знания	отсутствие умений или частично освоенное умение	отсутствие навыков или фрагментарные применение навыков
3 (удовлетворительно)	от 60% до 69% от максимальной суммы баллов	общие, но не структурированные знания	в целом успешно, но не систематически осуществляемое умение	в целом успешное, но не систематическое применение навыков
4 (хорошо)	от 70% до 89% от максимальной суммы баллов	сформированные, но содержащие отдельные проблемы знания	в целом успешное, но содержащие отдельные пробелы умение	в целом успешное, но содержащие отдельные пробелы применение навыков
5 (отлично)	≥ 90% от максимальной суммы баллов	сформированные систематические знания	сформированное умение	успешное и систематическое применение навыков

Шкала комплексной оценки сформированности компетенций приведена в таблице 9.3.

Таблица 9.3 – Шкала комплексной оценки сформированности компетенций

Оценка	Формулировка требований к степени компетенции
2 (неудовлетворительно)	Не имеет необходимых представлений о проверяемом материале или Знать на уровне **ориентирования**, представлений. Обучающийся знает основные признаки или термины изучаемого элемента содержания, их отнесенность к определенной науке, отрасли или объектам, узнает в текстах, изображениях или схемах и знает, к каким источникам нужно обращаться для более детального его усвоения.
3 (удовлетворительно)	Знать и уметь на **репродуктивном** уровне. Обучающихся знает изученный элемент содержания репродуктивно: произвольно воспроизводит свои знания устно, письменно или в демонстрируемых действиях.

Согласована на портале № 66608

4 (хорошо)	Знать, уметь, владеть на **аналитическом** уровне. Зная на репродуктивном уровне, указывать на особенности и взаимосвязи изученных объектов, на их достоинства, ограничения, историю и перспективы развития и особенности для разных объектов усвоения.
5 (отлично)	Знать, уметь, владеть на **системном** уровне. Обучающийся знает изученный элемент содержания системно, произвольно и доказательно воспроизводит свои знания устно, письменно или в демонстрируемых действиях, учитывая и указывая связи и зависимости между этим элементом и другими элементами содержания дисциплины, его значимость в содержании дисциплины.

9.1.1. Примерный перечень тестовых заданий

1. Одним из основных методов в функциональном программировании является суперпозиция функций. Рассматриваются суперпозиции функций CAR и CDR. Дан список (setq x `(a s (d) f g)). Что вернет функция (caddr x) ?
   1) ((d) f g)
   2) (a s)
   3) s
   4) (d)
2. Что будет получено в результате вызова следующей суперпозиции базовых функций языка Лисп: (cons (car `(1 2 3)) `(+ 2 6)) ?
   1) (1 2 3 8)
   2) (1 + 2 6)
   3) (1 8)
   4) (1 . 8)
3. Для разветвления вычислений в функциональном языке Лисп используется условное предложение COND. Задан список (setq x `((1) (2) 3 (4))). Что будет получено в результате работы следующего выражения
   (cond ((null x) 0) ((atom (car x)) 1) ((eq (cadr x) `(2)) 2) (t 3)) ?
   1) 0
   2) 1
   3) 2
   4) 3
4. В основе всех функциональных языков лежит лямбда-исчисление в том смысле, что все функциональные программы можно преобразовать в лямбда-выражение. Что выдаст следующее лямбда-выражение, описанное на языке Лисп?
   ((lambda (x y) (cond ((zerop x) (* y y)) ((< x 0) (+ y y)) (t (+ x y)))) (+ -10 2) (+ 2 10))
   1) 24
   2) 4
   3) Nil
   4) 144
5. Программа на функциональном языке Лисп представляет собой последовательность вычислимых выражений. Что будет выдано программой в результате следующей последовательности вызова вычислимых выражений?
   > (setq x 10)
   10
   > (defun f (x y) (+ (* x x) y))
   F
   > (f 2 3)
   ?
   1) 103
   2) 10
   3) 7
   4) error
6. Механизм рекурсивного вызова является одним из основных принципов

Согласована на портале № 66608

функционального программирования. Что выполняет следующая рекурсивная функция, аргументом которой является список?
(defun q (z) (cond ((null z) nil)
(t (append [q (cdr z)] [list (car z)]))))
1) переставляет последний элемент списка в начало;
2) меняет первый и последний элемент списка местами;
3) переставляет первый элемент списка в конец списка;
4) переставляет элементы списка в обратном порядке.

7. Любая рекурсивная функция должна иметь терминальные ветви (определяющие правило останова) и рекурсивные ветви. Какое количество терминальных ветвей содержит следующая рекурсивная функция?
(defun q (z) (cond ((null z) nil)
((null (cdr z)) 0)
((not (numberp (car z))) nil)
(t (+ [*(car z)(cadr z)] [q (cddr z)]))))
1) 1
2) 2
3) 3
4) 4

8. Функции, которые не формируют новые списки, а изменяют структуру существующих списков, называются разрушающими. Чему будет равен Y в результате следующей последовательности вызова вычислимых выражений с использованием разрушающей функции?
(setq x `(2 3))
(setq y (cons 1 x))
(rplaca x 7)
y - ?
1) (1 2 3)
2) (1 7)
3) 7
4) (1 7 3)

9. Любой функциональный язык содержит функционалы: функции, имеющие аргументы, значением которых являются функции. Что будет получено в результате работы следующего функционала?
(mapcar `length `((1 2 3)(a s d f)(4 5)))
1) (3 4 2)
2) (3 2 1)
3) (2 4 3)
4) Nil

10. Использование механизма циклов вместо рекурсии позволяет экономить память и строить, зачастую, более эффективные программы. Что будет получено в результате работы следующего вычислимого выражения с использованием цикла?
(let ((x 0)(y nil))(loop (setq x (+ 1 x))(setq y (cons x y))(cond((= x 5)(return y)))))
1) 5
2) nil
3) (5 4 3 2 1)
4) (1 2 3 4 5)

11. Применение функционалов в программе основано на том, что программы и данные в языках функционального программирования представляются одинаково. Определена функция SUM, аргументом которой является список, а результат работы – сумма элементов списка. Что будет получено в результате работы следующего функционала, где функция SUM рассматривается как аргумент другой функции?
(maplist `sum `(1 2 3 4 5))
1) 15
2) (15 14 12 9 5)
3) (5 9 12 14 15)
4) (1 2 3 4 5)

Согласована на портале № 66608

12. Основная структура данных в языках функционального программирования – списки. По сути, любая функция на языке Лисп является функцией обработки списков. Какой список свойств получится в результате выполнения следующей последовательности вычислимых выражений?
    (setf (get `as `v4) `(4))
    (setf (get `as `v2) `(2))
    (setf (get `as `v3) `(3))
    (setf (get `as `v1) `(1))
    1) (v4 4 v3 3 v2 2 v1 1)
    2) (v1 1 v3 3 v2 2 v4 4)
    3) (v4 (4) v2 (2) v3 (3) v1 (1))
    4) (v1 (1) v3 (3) v2 (2) v4 (4))
13. Особый вид списков, используемых в языке Лисп – ассоциативные списки, элементами которых являются точечные пары. Какой ассоциативный список получится в результате работы следующей функции:
    (pairlis `((a) (b) (c)) `(1 2 3) ())?
    1) ((c . 3)(b . 2)(a . 1))
    2) (c 3 b 2 a 1)
    3) ((a . 1)(b . 2)(c . 3))
    4) ((a 1)(b 2)(c 3))
14. Фундаментальной операцией над объектами в логическом программировании является механизм унификации – сопоставление термов и переменных. Проанализируйте, унифицируемы ли следующие предикаты?
    Pred1 (X, Y, Y) ? Pred1 (5, 10, 12)
    1) Да
    2) Нет: переменные и константы не сравнимы
    3) Нет: в предикате не может быть две одинаковые переменные
    4) Нет, одна переменная не может иметь два разных значения
15. Задана следующая последовательность предикатов.
    F(0,1).
    F(1,2).
    F(2,3).
    F(3,4).
    Pr(X,Y) :- f(X,Z), f(Z,Y).
    Чему будут равны значения переменных A и Z в результате следующего вопроса:
    ? – pr(0,A), pr(A,Z).
    1) A=1, Z=2
    2) A=1, Z=3
    3) A=3, Z=4
    4) Нет решения
16. Приведенная ниже процедура описывает следующее знание: «Мэри любит всех животных, кроме змей».
    Love (mary, X) :- snake(X), fail.
    Love (mary, X) :- animal(X), not(snake(X)).
    Перепишите процедуру, убрав отрицание, используя при необходимости механизм отсечения.
    1) Love (mary, X) :- snake(X),fail,!.
    Love (mary, X) :- animal(X).
    2) Love (mary, X) :- !,snake(X),fail.
    Love (mary, X) :- animal(X).
    3) Love (mary, X) :- snake(X),!,fail.
    Love (mary, X) :- animal(X).
    4) Love (mary, X) :- snake(X),!.
    Love (mary, X) :- animal(X).
17. Рекурсия – это способ задания функции путем определения каждого его значения в терминах ранее определенных значений. Рекурсивный механизм является мощнейшим инструментом построения программ в логическом программировании. Что выполняет

Согласована на портале № 66608

следующая рекурсивная процедура, аргументом которой является список?
a ([ ], 1).
a ([X | XT], P) :- a (XT, P1), P=P1*X.
1) перемножает элементы списка, стоящие на нечетных позициях;
2) перемножает элементы списка, стоящие на четных позициях;
3) перемножает элементы списка, не равные нулю;
4) перемножает элементы списка.

18. Задана следующая рекурсивная процедура:
Q ([ ], 0).
Q ([X], X).
Q ([X, _| Y], N) :- Q (Y, N1), N=N1+X.
Чему будет равно X в результате следующего вопроса?
?- q ([3,1,5,2,6], X).
1) 14
2) 5
3) 8
4) 3
19. Язык логического программирования Пролог создан для задач анализа и понимания естественного языка. Как можно определить предикат внук(X,Y) через предикат родитель(X,Y), основываясь на понятиях родственных отношений?
1) Внук(X,Y):-родитель(X,Y),родитель(X,Y).
2) Внук(X,Y):-родитель(X,Z),родитель(Z,Y).
3) Внук(X,Y):-родитель(X,X),родитель(Y,Y).
4) Внук(X,Y):-родитель(Z,X),родитель(Z,Y).
20. Работа программы в логическом программировании основана, прежде всего, на обработке существующих фактов. Дана база фактов: животное(<наименование>,<ареал>,<количество_особей>). Как будет выглядеть предикат, формирующий список, элементами которого являются <количество_особей> по всем животным?
1) findall(животное(_,_,X)).
2) findsll(X,животное(_,_,X),X).
3) findall (X,животное(_,_,X),L).
4) findall(X,животное(_,_,L),[X|L]).

9.1.2. Перечень экзаменационных вопросов

Приведены примеры типовых заданий из банка экзаменационных тестов, составленных по пройденным разделам дисциплины.

1. Лисп. Какая ошибка в определении функции, проверяющей, является ли данный список одноуровневым?
(defun f (s)
(if (not (atom (car s))) nil (f (cdr s))))
1) без ошибок;
2) перепутаны случаи "то" и "иначе" в условной функции;
3) надо пользоваться предикатом or;
4) нет окончания рекурсии.
2. Лисп. Сколько элементов самого верхнего уровня в следующих списках:
1) ((1 2 3));
2) ((a b) c (d (e)));
3) (a ((())) nil nil);
4) ((((a (b (c d) e) f) g) h ((i (j) k) l) m) n)
3. Лисп. Каково общее число подсписков в подсписках (т. е. списки уровня 3; исходный список имеет уровень 1) в следующих списках:
1) (6 (3 6 (7 (4 5) 8)) (9 3));
2) (((a 9 ((b 7) c)) (d 5)));
3) (quote (s (3 5) (7 (9 8))));
4) (d (q 2 7 4) (+ 1 (* 7 (+ 3 (- 2)))))?

Согласована на портале № 66608

4. Лисп. Какие из следующих утверждений верны?
   1. Язык XLisp - функциональный язык только с ленивыми вычислениями.
   2. Язык XLisp - функциональный язык только с энергичными вычислениями.
   3. Язык XLisp - функциональный язык с энергичными и ленивыми вычислениями.
   4. Язык C++ - функциональный язык.
5. Лисп. Даны определения функций
   (defun twice (f)
   (function (lambda (x) (funcall f (funcall f x)))))
   (defun do (x) (funcall (twice 'list) x))
   Чему равно значение (do '0) ? Введите символьное выражение без пробелов.
6. Лисп. Дано определения функций
   (defun many (f x)
   (mapcar (function (lambda (g) (funcall g x))) f))
   (defun f1 (x) (+ x x))
   (defun f2 (x) (* x x))
   Чему равно значение (length (many '(f1 f2) 1)) ?
7. Лисп. Дано определение функции
   (defun create (x y)
   (eval (cons 'defun (cons x (cdr y)))))
   Вызов этой функции приводит к определению некоторой новой функции f.
   (create 'f '(lambda (x) (* x x)))
   Чему равно значение (f 2)?
8. Лисп. Дано определение функции
   (defun factor (n)
   (if (< n 2) '(1) (append (factor (- n 1)) (list '* n))))
   Чему равно значение (length (factor 3)) ?
9. Лисп. Каково значение следующего выражения
   (eval (cons (quote >) (cons 5 (list ((lambda (x y) (- x y)) 3 7))))) ?
   Введите значение маленькими буквами.
10. Лисп. Дано определение функции
    (defun f (x s)
    (if (= x (car s)) 1 (+ 1 (f x (cdr s)))))
    Чему равно значение (f 2 '(1 3 2)) ?

9.1.3. Примерный перечень тематик курсовых проектов

1. Упрощение электрических цепей
2. Программа для алгебраических вычислений
3. Упрощение арифметических выражений
4. Определение связности графа на Прологе
5. Определение связности графа на Лиспе

9.1.4. Примерный перечень тем и тестовых заданий на контрольные работы

Функциональное и логическое программирование
Контрольная работа 1

1. Дана цель для интерпретатора Пролога
   ?- not parent(X,pat).
   на которую был получен отрицательный ответ. Какое из следующих трех предложений правильно передает логический смысл этого ответа:
   1) "Не существует родитель у Pat."
   2) "Не все X являются родителями Pat."
   3) "Есть родители, но не у Pat."
   1) первое предложение;
   2) второе предложение;
   3) третье предложение;
   4) все три предложения не передают логический смысл ответа.

Согласована на портале № 66608

2. Пусть дано отношение 'родитель' на Прологе. Определим отношение "X - родственник Y" следующим образом :
   'родственник'(X,Y):-'родитель'(X,Y).
   'родственник'(X,Y): - 'родитель'(Y,X).
   'родственник'(X,Y):-'родственник'(X,Z),'родственник'(Z,Y).
   Какие из следующих утверждений верны?
   1. Одно из первых двух правил лишнее.
   2. Третье правило потенциально опасное - оно может привести в некоторых запросах к бесконечной рекурсии.
   3. Любой запрос к данной программе приводит к конечной работе интерпретатора Пролога.
   4. Правила неправильно задают отношение 'родственник'.
3. Пролог. Определим предикат length для вычисления длины списка:
   length([], 0).
   length([_|T], N) :- length(T, N1), N is 1+N1.
   Если во втором правиле в его теле поменять две цели местами, то при вызове
   ?- length([1, 2, 3], N).
   произойдет следующее:
   1) интерпретатор не сможет вычислить цель, а сообщит о ошибке;
   2) N получит значение равное 3;
   3) цель успешно вычислится, но N в качестве значения получит не число;
   4) интерпретатор ответит: No.
4. Пролог. Последовательность чисел Фибоначчи имеет вид
   1, 1, 2, 3, 5, 8, 13,…
   Каждый член последовательности, за исключением первых двух, представляет собой сумму предыдущих двух членов. Какой метод программирования позволяет написать предикат fib(N, F), эффективно (т. е. линейно по времени) вычисляющий N-ое число Фибоначчи F?
   1) использование запоминающих функций;
   2) такого метода нет, рекурсия всегда не эффективна;
   3) использование отсечений;
   4) изменение порядка целей и предложений;
   5) использование накапливающих параметров.
5. Пролог. Определите, будет ли каждая из следующих пар термов унифицируемой:
   1) 'книга'('название'('Ферма животных'), 'автор'('Джордж Оруэлл')) и 'книга'(название'(T)', Avtor);
   2) 'дата'('день недели'('среда'), 'число'(12), 'месяц'(M), 'год'(1986)) и 'дата'(W,D,X,Y);
   3) 'праздник'('рождество', 'дата'('день'(25), 'месяц'('декабрь'), 'год'(Y))) и 'праздник'(H, 'дата'(D,M, 'год'(1986));
   4) 'праздник'('Первомай'(1, 'май')) и 'праздник'('Первомай', 1, 'май').
6. Пролог. Какие следующие утверждения истинны?
   1. Правило имеет заголовок (голову) и тело.
   2. Унификация цели может вызвать необходимость унификации подцелей.
   3. Пролог всегда совершает возврат для повторной унификации любой цели.
   4. Пытаясь согласовать цель повторно, Пролог начинает поиск снова с начала программы.
   5. Пользователь может заставить Пролог совершить возврат, отвергнув полученный ответ.
7. Пролог. К чему приведет следующий вызов предиката
   ?- max(4+7, 8*9, N).
   если предикат max/3 определен следующим образом:
   max(X,Y,X):-X>=Y.
   max(X,Y,Y):-Y>X.
   1) Пролог ответит Yes и выдаст N=72;
   2) Пролог ответит Yes и выдаст N=8*9;
   3) Пролог ответит No;
   4) Пролог сообщит об ошибке в операциях сравнения.
8. В каком порядке Пролог ищет утверждения программы для унификации с целью?

Согласована на портале № 66608

1) в порядке размещения клауз (предложений) в тексте программы - сверху вниз;
2) сначала просматриваются факты в программе, потом - правила сверху вниз;
3) Пролог сам устанавливает порядок, исходя из эффективности программы.

9. Пролог. Какие следующие утверждения истинны?
   1. В Прологе единственная структура данных - термы.
   2. Структура характеризуется своим функтором. Не допускается использовать структуры с одинаковым функтором и разной местностью (арностью).
   3. Мы задаем вопросы, используя термы в качестве целей.
   4. Структура относится к рекурсивному типу данных.
   5. Переменная - это "забронированное" место. Любой терм может заменить переменную, но для разных вхождений переменной в структуру не обязательна должна иметь место одна и та же замена.
   6. Когда два терма унифицируются, переменные в них заменяются на некоторые значения.
10. Пролог. Определите, будет ли каждая из следующих пар термов унифицируемой:
    1) 'книга'('название'('Ферма животных'), 'автор'('Джордж Оруэлл')) и 'книга'(название'(T)', Avtor);
    2) 'дата'('день недели'('среда'), 'число'(12), 'месяц'(M), 'год'(1986)) и 'дата'(W,D,X,Y);
    3) 'праздник'('рождество', 'дата'('день'(25), 'месяц'('декабрь'), 'год'(Y))) и 'праздник'(H, 'дата'(D,M, 'год'(1986));
    4) 'праздник'('Первомай'(1, 'май')) и 'праздник'('Первомай', 1, 'май').
    Введите через пробел номера пар (в порядке возрастания), для которых, вы считаете, проходит унификация.

Контрольная работа 2

Разработка программы для написания простых предикатов с использованием языка Пролог.

Задача 1

Условие

Постройте предикат position_max(+L, -M, -N), который в списке L находит максимальное значение M и порядковый номер N этого значения. Решение Будем использовать метод накапливающего параметра. Для этого введем вспомогательный предикат position_max(+List, +I, +M0, +N0, -M, -N), где I равен номеру рассматриваемого элемента в исходном списке, M0 – текущее значение максимума, N0 – позиция текущего максимума.

```
position_max([X|T],M,N): - position_max(T,1,X,1,M,N).
position_max([],_,M,N,M,N).
position_max([A|T],I,X,_,M,N):- A>X,
K is I+1, position_max(T,K,A,K,M,N).

position_max([A|T],I,X,Y,M,N): - A=0, Y1 is Y-1,
 p(X,Y1,R1),
R is R1+X. p(X,Y,R): - Y<0, Y1 is -Y,
```

```
p(X,Y1,R).
```

Задача 2

Условие

Определите умножение целых чисел через сложение и вычитание. Решение Определим предикат p(+X,+Y,?R), где X и Y – сомножители, R – произведение. Делаем рекурсию по второму аргументу предиката, используя рекурсивное определение X*Y=X*(Y-1)+X.

```
p(X,0,0).
p(X,Y,R):- Y>0,
Y1 is Y-1,
p(X,Y1,R1),
R is R1+X.
p(X,Y,R):- Y<0, Y1 is -Y, p(X,Y1,R).
```

Вариант 1

1. Определите возведение в целую степень через умножение и деление.

2. Напишите предикат p(+L, -N), истинный тогда и только тогда, когда N – предпоследний элемент списка L, имеющего не менее двух элементов.

Вариант 2

Согласована на портале № 66608

1. Напишите предикат p(+X, +N, -L), истинный тогда и только тогда, когда L – список из N раз повторенных элементов X.

2. Напишите предикат p(+L, -S), истинный тогда и только тогда, когда S – список списков элементов списка L. Например, p([a, b, c],[[a], [b], [c]]) – истина.

Вариант 3

1. Напишите предикат p(+L, -S), истинный тогда и только тогда, когда L – список списков, а S – список, объединяющий все эти списки в один.

2. Напишите предикат p(+L, -S), истинный тогда и только тогда, когда список S есть циклическая перестановка элементов списка L, например, p([f, g, h, j], [g, h, j, f]) – истина.

Вариант 4

1. Напишите предикат p(+X, +N, +V, -L), истинный тогда и только тогда, когда список L получается после добавления X на N-е место в список V.

2. Напишите предикат p(+N, +V, -L), истинный тогда и только тогда, когда список L получается после удаления N-го элемента из списка V.

Вариант 5

1. Напишите предикат p(+V, -L), истинный тогда и только тогда, когда список L получается после удаления всех повторных вхождений элементов в список V, например, p([a, b, c, d, d, a], [a, b, c, d]) – истина.

2. Напишите предикат p(+V, -L), истинный тогда и только тогда, когда список L получается после удаления из списка V всех элементов, стоящих на четных местах, например, p([1, 2, 3, 4, 5, 6], [1, 3, 5]) – истина.

9.1.5. Темы лабораторных работ

1. Разработка программы с использованием языка Лисп
2. Разработка программ с использованием языка Пролог

9.2. Методические рекомендации

Учебный материал излагается в форме, предполагающей самостоятельное мышление студентов, самообразование. При этом самостоятельная работа студентов играет решающую роль в ходе всего учебного процесса.

Начать изучение дисциплины необходимо со знакомства с рабочей программой, списком учебно-методического и программного обеспечения. Самостоятельная работа студента включает работу с учебными материалами, выполнение контрольных мероприятий, предусмотренных учебным планом.

В процессе изучения дисциплины для лучшего освоения материала необходимо регулярно обращаться к рекомендуемой литературе и источникам, указанным в учебных материалах; пользоваться через кабинет студента на сайте Университета образовательными ресурсами электронно-библиотечной системы, а также общедоступными интернет-порталами, содержащими научно-популярные и специализированные материалы, посвященные различным аспектам учебной дисциплины.

При самостоятельном изучении тем следуйте рекомендациям:

– чтение или просмотр материала осуществляйте со скоростью, достаточной для индивидуального понимания и освоения материала, выделяя основные идеи; на основании изученного составить тезисы. Освоив материал, попытаться соотнести теорию с примерами из практики;

– если в тексте встречаются незнакомые или малознакомые термины, следует выяснить их значение для понимания дальнейшего материала;

– осмысливайте прочитанное и изученное, отвечайте на предложенные вопросы.

Студенты могут получать индивидуальные консультации, в т.ч. с использованием средств телекоммуникации.

По дисциплине могут проводиться дополнительные занятия, в т.ч. в форме вебинаров. Расписание вебинаров и записи вебинаров публикуются в электронном курсе по дисциплине.

9.3. Требования к оценочным материалам для лиц с ограниченными возможностями здоровья и инвалидов

Для лиц с ограниченными возможностями здоровья и инвалидов предусмотрены

Согласована на портале № 66608

дополнительные оценочные материалы, перечень которых указан в таблице 9.4.

Таблица 9.4 – Дополнительные материалы оценивания для лиц с ограниченными возможностями здоровья и инвалидов

Категории обучающихся	Виды дополнительных оценочных материалов	Формы контроля и оценки результатов обучения
С нарушениями слуха	Тесты, письменные самостоятельные работы, вопросы к зачету, контрольные работы	Преимущественно письменная проверка
С нарушениями зрения	Собеседование по вопросам к зачету, опрос по терминам	Преимущественно устная проверка (индивидуально)
С нарушениями опорно-двигательного аппарата	Решение дистанционных тестов, контрольные работы, письменные самостоятельные работы, вопросы к зачету	Преимущественно дистанционными методами
С ограничениями по общемедицинским показаниям	Тесты, письменные самостоятельные работы, вопросы к зачету, контрольные работы, устные ответы	Преимущественно проверка методами, определяющимися исходя из состояния обучающегося на момент проверки

9.4. Методические рекомендации по оценочным материалам для лиц с ограниченными возможностями здоровья и инвалидов

Для лиц с ограниченными возможностями здоровья и инвалидов предусматривается доступная форма предоставления заданий оценочных средств, а именно:

– в печатной форме;
– в печатной форме с увеличенным шрифтом;
– в форме электронного документа;
– методом чтения ассистентом задания вслух;
– предоставление задания с использованием сурдоперевода.

Лицам с ограниченными возможностями здоровья и инвалидам увеличивается время на подготовку ответов на контрольные вопросы. Для таких обучающихся предусматривается доступная форма предоставления ответов на задания, а именно:

– письменно на бумаге;
– набор ответов на компьютере;
– набор ответов с использованием услуг ассистента;
– представление ответов устно.

Процедура оценивания результатов обучения лиц с ограниченными возможностями здоровья и инвалидов по дисциплине предусматривает предоставление информации в формах, адаптированных к ограничениям их здоровья и восприятия информации:

Для лиц с нарушениями зрения:
– в форме электронного документа;
– 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.

При необходимости для лиц с ограниченными возможностями здоровья и инвалидов процедура оценивания результатов обучения может проводиться в несколько этапов.

Согласована на портале № 66608

ЛИСТ СОГЛАСОВАНИЯ

Рассмотрена и одобрена на заседании кафедры АОИ
протокол № 6 от «10» 12 2020 г.

СОГЛАСОВАНО:

Должность	Инициалы, фамилия	Подпись
Заведующий выпускающей каф. АОИ	А.А. Сидоров	Согласовано, d4895b45-5d88-49f8-b7c7-e8bf0196776a
Заведующий обеспечивающей каф. АОИ	А.А. Сидоров	Согласовано, d4895b45-5d88-49f8-b7c7-e8bf0196776a
Декан ФДО	И.П. Черкашина	Согласовано, 4580bdea-d7a1-4d22-bda1-21376d739cfc

ЭКСПЕРТЫ:

Доцент, каф. АОИ	Ю.В. Морозова	Согласовано, 8461038d-613f-4932-8e22-2b7293a14b92
Доцент, каф. АОИ	Н.Ю. Салмина	Согласовано, ed28a52c-a209-461c-b4ed-4e958affbfc7

РАЗРАБОТАНО:

Доцент, каф. АОИ	Ю.В. Морозова	Разработано, 8461038d-613f-4932-8e22-2b7293a14b92
Доцент, каф. АОИ	Н.Ю. Салмина	Разработано, ed28a52c-a209-461c-b4ed-4e958affbfc7

Согласована на портале № 66608