

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 УР
Сенченко П.В.
«22» 02 2023 г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

ФУНКЦИОНАЛЬНОЕ 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**

Семестр: **7**

Учебный план набора 2023 года

Объем дисциплины и виды учебной деятельности

Виды учебной деятельности	7 семестр	Всего	Единицы
Лабораторные занятия	4	4	часов
Самостоятельная работа	151	151	часов
Самостоятельная работа под руководством преподавателя	14	14	часов
Контрольные работы	2	2	часов
Подготовка и сдача экзамена	9	9	часов
Общая трудоемкость (включая промежуточную аттестацию)	180	180	часов
		5	з.е.

Формы промежуточной аттестация	Семестр	Количество
Экзамен	7	
Контрольные работы	7	1

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Сенченко П.В.
Должность: Проректор по УР
Дата подписания: 22.02.2023
Уникальный программный ключ:
a1119608-cdff-4455-b54e-5235117c185c

Томск

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 программирования.

1.2. Задачи дисциплины

1. Получить знания и овладеть понятийным аппаратом: рекурсия; функциональное программирование; λ -исчисление; функционалы; предикаты первого порядка; 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.01.10.

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		

ПК-1. Способен осуществлять концептуальное, функциональное и логическое проектирование систем среднего и крупного масштаба и сложности	ПК-1.1. Знает методы концептуального, функционального и логического проектирования программного обеспечения	Знает типовые роли в процессе разработки программного обеспечения; математические основы предикатов первого порядка ; математические основы лямбда-исчисления
	ПК-1.2. Умеет разрабатывать концептуальные, функциональные и логические модели программного обеспечения	Умеет разрабатывать модели различных классов систем с применением языков функционального программирования; осуществлять разработку программного обеспечения на языках Лисп
	ПК-1.3. Владеет навыками использования современных инструментальных средств концептуального, функционального и логического проектирования программного обеспечения	Владеет математическим аппаратом, применяемым в функциональном 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 5 зачетных единиц, 180 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
Контактная аудиторная работа обучающихся с преподавателем, всего	20	20
Лабораторные занятия	4	4
Самостоятельная работа под руководством преподавателя	14	14
Контрольные работы	2	2
Самостоятельная работа обучающихся, в т.ч. контактная внеаудиторная работа обучающихся с преподавателем, всего	151	151
Самостоятельное изучение тем (вопросов) теоретической части дисциплины	119	119
Подготовка к контрольной работе	16	16
Подготовка к лабораторной работе	8	8
Написание отчета по лабораторной работе	8	8
Подготовка и сдача экзамена	9	9
Общая трудоемкость (в часах)	180	180
Общая трудоемкость (в з.е.)	5	5

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.

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

Названия разделов (тем) дисциплины	Лаб. раб.	Контр. раб.	СРП, ч.	Сам. раб., ч	Всего часов (без промежуточной аттестации)	Формируемые компетенции
7 семестр						
1 Что такое декларативное программирование?	-	2	2	20	24	ПК-1
2 Функциональный взгляд на вычисления	4		2	30	36	ПК-1
3 Основы языка ЛИСП	-		4	25	29	ПК-1
4 Рекурсия	-		1	26	27	ПК-1
5 Лямбды-исчисления	-		2	26	28	ПК-1
6 Функции высокого порядков	-		2	4	6	ПК-1
7 Несколько более сложных задач	-		1	20	21	ПК-1
Итого за семестр	4	2	14	151	171	
Итого	4	2	14	151	171	

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	2	ПК-1
	Итого	2	
2 Функциональный взгляд на вычисления	Определение функции. Функциональная композиция.	2	ПК-1
	Итого	2	
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	4	ПК-1
	Итого	4	
4 Рекурсия	Знакомство с рекурсией . Встроенные рекурсивные функции для списков. Различные виды рекурсии	1	ПК-1
	Итого	1	
5 Лямбды-исчисления	Лямбда-исчисление как формальная система. Лямбда-исчисление как язык программирования.	2	ПК-1
	Итого	2	
6 Функции высокого порядков	Функции высших порядков. Различие между данными и функциями. Функционалы. Работа с графами и деревьями: представление, обработка, поиск пути на графе.	2	ПК-1
	Итого	2	

7 Несколько более сложных задач	Работа со списками. Задания с функциями и рекурсиями	1	ПК-1
	Итого	1	
Итого за семестр		14	
Итого		14	

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	ПК-1
Итого за семестр		2	
Итого		2	

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	ПК-1
Итого		4	
Итого за семестр		4	
Итого		4	

5.5. Контроль самостоятельной работы (курсовой проект / курсовая работа)

Не предусмотрено учебным планом

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	18	ПК-1	Тестирование, Экзамен
	Подготовка к контрольной работе	2	ПК-1	Контрольная работа
	Итого	20		

2 Функциональный взгляд на вычисления	Подготовка к контрольной работе	4	ПК-1	Контрольная работа
	Подготовка к лабораторной работе	4	ПК-1	Лабораторная работа
	Написание отчета по лабораторной работе	4	ПК-1	Отчет по лабораторной работе
	Самостоятельное изучение тем (вопросов) теоретической части дисциплины	18	ПК-1	Тестирование, Экзамен
	Итого	30		
3 Основы языка ЛИСП	Подготовка к лабораторной работе	4	ПК-1	Лабораторная работа
	Написание отчета по лабораторной работе	4	ПК-1	Отчет по лабораторной работе
	Самостоятельное изучение тем (вопросов) теоретической части дисциплины	15	ПК-1	Тестирование, Экзамен
	Подготовка к контрольной работе	2	ПК-1	Контрольная работа
	Итого	25		
4 Рекурсия	Самостоятельное изучение тем (вопросов) теоретической части дисциплины	24	ПК-1	Тестирование, Экзамен
	Подготовка к контрольной работе	2	ПК-1	Контрольная работа
	Итого	26		
5 Лямбды-исчисления	Самостоятельное изучение тем (вопросов) теоретической части дисциплины	24	ПК-1	Тестирование, Экзамен
	Подготовка к контрольной работе	2	ПК-1	Контрольная работа
	Итого	26		

6 Функции высокого порядков	Самостоятельное изучение тем (вопросов) теоретической части дисциплины	2	ПК-1	Тестирование, Экзамен
	Подготовка к контрольной работе	2	ПК-1	Контрольная работа
	Итого	4		
7 Несколько более сложных задач	Самостоятельное изучение тем (вопросов) теоретической части дисциплины	18	ПК-1	Тестирование, Экзамен
	Подготовка к контрольной работе	2	ПК-1	Контрольная работа
	Итого	20		
Итого за семестр		151		
	Подготовка и сдача экзамена	9		Экзамен
Итого		160		

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.	Конт.Раб.	СРП	Сам. раб.	
ПК-1	+	+	+	+	Контрольная работа, Лабораторная работа, 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

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/>.

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, 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

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;
- 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.

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?	ПК-1	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	ПК-1	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	ПК-1	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	ПК-1	Контрольная работа	Примерный перечень тем и тестовых заданий на контрольные работы
		Тестирование	Примерный перечень тестовых заданий
		Экзамен	Перечень экзаменационных вопросов
5 Лямбды-исчисления	ПК-1	Контрольная работа	Примерный перечень тем и тестовых заданий на контрольные работы
		Тестирование	Примерный перечень тестовых заданий
		Экзамен	Перечень экзаменационных вопросов
6 Функции высокого порядков	ПК-1	Контрольная работа	Примерный перечень тем и тестовых заданий на контрольные работы
		Тестирование	Примерный перечень тестовых заданий
		Экзамен	Перечень экзаменационных вопросов

7 Несколько более сложных задач	ПК-1	Контрольная работа	Примерный перечень тем и тестовых заданий на контрольные работы
		Тестирование	Примерный перечень тестовых заданий
		Экзамен	Перечень экзаменационных вопросов

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.

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

- 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) ?
 - ((d) f g)
 - (a s)
 - s
 - (d)
- Что будет получено в результате вызова следующей суперпозиции базовых функций языка Лисп: (cons (car '(1 2 3)) '(+ 2 6)) ?
 - (1 2 3 8)
 - (1 + 2 6)
 - (1 8)
 - (1 . 8)
- 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)) ?
 - 0
 - 1
 - 2
 - 3
- 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))
 - 24
 - 4
 - Nil
 - 144
- Программа на функциональном языке Лисп представляет собой последовательность вычисляемых выражений. Что будет выдано программой в результате следующей последовательности вызова вычисляемых выражений?


```
> (setq x 10)
10
> (defun f(x y) (+ (* x x) y))
F
> (f 2 3)
?
```

 - 103
 - 10
 - 7
 - error

- Механизм рекурсивного вызова является одним из основных принципов

- 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 (caddr 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 вычисляемых выражений с использованием разрушающей функции?
- ```
(setq x `(2 3))
(setq y (cons 1 x))
(рlaca x 7)
у - ?
```
- 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 вычисляемого 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)

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 вычисляемых 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

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) findsl(X,животное(_,_),X).

3) findall (X,животное(_,_),L).

4) findall(X,животное(_,_),L),[X|L]).

9.1.2. Перечень экзаменационных вопросов

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 og;
- 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 подписков в подписках (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)))))?

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 тем и тестовых заданий на контрольные работы

Функциональное программирование

1. Как называется часть списка без первого элемента?
 1. голова;
 2. хвост;
 3. ячейка;
 4. функционал;
 5. атом.
2. Какое обозначение имеет пустой список?
 1. TRUE;
 2. T;
 3. NIL;
 4. FALSE;
 5. (NIL).
3. В рекурсивном определении важен порядок следования условий. Какая ветвь должна быть первой в COND?
 1. проверка на NULL;
 2. рекурсивная;
 3. терминальная;
 4. функциональная;
 5. проверка на истину.
4. Какие задачи мы должны решить при планировании рекурсивной ветви?
 1. когда функция может вернуть значение без рекурсивного вызова;
 2. в каком порядке поставить терминальную и рекурсивную ветви;
 3. определить соотношение между терминальной и рекурсивной ветвями;
 4. как упрощать аргумент, приближая его к конечному значению;

5. как построить форму, называемую рекурсивным отношением.
5. Что выполняет следующая рекурсивная функция, аргументом которой является список?
`(defun q (z) (cond ((null (cdr z)) 1) (t (* (cadr z) (q (cddr z)))))`
- 1) перемножает элементы списка;
 - 2) перемножает элементы списка, стоящие на нечетных позициях;
 - 3) перемножает элементы списка, стоящие на четных позициях;
 - 4) перемножает элементы списка, не равные нулю.
6. Что выполняет следующая рекурсивная функция, аргументами которой являются два списка?
`(defun q (a b) (cond ((null a) nil) (t (cons (+ (car a)(car b)) (q (cdr a)(cdr b)))))`
- 1) формирует список из поэлементных сумм исходных списков;
 - 2) создает список из двух сумм элементов – 1-го и 2-го списка;
 - 3) суммирует все элементы в обоих списках;
 - 4) подсчитывает количество элементов в кратчайшем списке.
7. Что является результатом работы функции (MAKE-ARRAY N)?
- 1) создание массива ячеек размерности N
 - 2) создание списка из атомов NIL размерностью N
 - 3) создание массива с именем N
 - 4) объявление массива размерностью N
8. Определите порядок приведенных ниже выражений, в результате которого должен сформироваться следующий список свойств: (a 1 s (2 3) d qwe f 5)
- 1) `(setf (get `sp `s) `(2 3))`
 - 2) `(setf (get `sp `f) 5)`
 - 3) `(setf (get `sp `a) 1)`
 - 4) `(setf (get `sp `d) qwe)`
9. Определите порядок приведенных ниже выражений, в результате которого должен сформироваться следующий список свойств: (k1 45 k2 asd k3 (z x c) k4 1)
- 1) `(setf (get `sp `k4) 1)`
 - 2) `(setf (get `sp `k3) `(z x c))`
 - 3) `(setf (get `sp `k1) 45)`
 - 4) `(setf (get `sp `k2) asd)`
10. Создан массив ячеек A. Его значение равно: #(a s d f)
 Чему будет равно значение переменной X после выполнения следующего вычислимого выражения?
`(setq X (cons (aref A 3) `(1 2)))`

9.1.4. Темы лабораторных работ

1. Разработка программы с использованием языка Лисп

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 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.

ЛИСТ СОГЛАСОВАНИЯ

Рассмотрена и одобрена на заседании кафедры АОИ
протокол № 13 от «15» 12 2022 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