МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РФ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
Томский государственный университет систем управления и радиоэлектроники

УТВЕРЖДАЮ

Проректор по учебной работе

_____ П.Е.Троян

«___» _____ 2017 г.

РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ
ФУНКЦИОНАЛЬНОЕ И ЛОГИЧЕСКОЕ ПРОГРАММИРОВАНИЕ

Уровень основной образовательной программы бакалавриат

Направление подготовки 01.03.02 Прикладная математика и информатика

Форма обучения очная

Факультет систем управления

Кафедра автоматизированных систем управления

Курс 3

Семестр 6

Учебный план набора 2013 года и последующих лет

Распределение рабочего времени:

Виды учебной работы	Семестр 6	Всего	Единицы
Лекции	18	18	Часов
Лабораторные работы	36	36	часов
Практические занятия			Часов
Курсовой проект/работа (КРС) (аудиторная)			Часов
Всего аудиторных занятий	54	54	Часов
Из них в интерактивной форме	8	8	Часов
Самостоятельная работа студентов (СРС)	54	54	Часов
Всего (без экзамена)	108	108	Часов
Самост. работа на подготовку и сдачу экзамена			Часов
Общая трудоемкость	108	108	часов
(в зачетных единицах)	3	3	ЗЕТ

Зачет 6 **семестр**

Томск 2017

Согласована на портале № 16227

Рабочая программа составлена с учетом требований Федерального Государственного образовательного стандарта высшего образования (ФГОС ВО) по направлению 01.02.03 Прикладная математика и информатика (квалификация (степень) бакалавр), утвержденного Приказом Министерства образования и науки Российской Федерации 12.03.2016 №228, рассмотрена и утверждена на заседании кафедры «12» января 2017 г., протокол № 5.

Разработчик старший преподаватель каф. АСУ _____ А.В. Ковшов

Зав. обеспечивающей кафедрой АСУ
д.т.н., профессор _____ А.М. Кориков

Рабочая программа согласована с факультетом, профилирующей и выпускающей кафедрами специальности.

Кандидат технических наук, доцент
Декан, к.т.н., доцент _____ П.В. Сенченко

Заведующий профилирующей и выпускающей
кафедрой АСУ, д.т.н., профессор _____ А.М. Кориков

Эксперт:

Кафедра АСУ, доцент _____ А.И. Исакова
(место работы) (занимаемая должность)

Согласована на портале № 16227

1. ЦЕЛИ И ЗАДАЧИ ДИСЦИПЛИНЫ

Дисциплина «Функциональное и логическое программирование» (ФЛП) читается в 6 семестре и предусматривает чтение лекций, проведение лабораторных работ, получение различного рода консультаций.

Предметом изучения в рассматриваемой дисциплине являются функциональные языки программирования, логические языки программирования, математически основы функциональных и логических языков.

Цель дисциплины – формирование у студентов профессиональных знаний и практических навыков по разработке и созданию моделей и систем с помощью языков функционального и логического программирования. Функциональный и логический стили программирования являются альтернативными способами решения задач по отношению к традиционным стилям программирования - процедурному и объектно-ориентированному. Его отличительными чертами являются “математичность” и “человечность”. В функциональном программировании единственным действием является вызов функции, в нем не существует прямого обращения к ячейкам памяти, операторов присваивания, циклов, блок-схем и передачи управления. Логическое программирование - это переход на еще более высокий уровень программирования. Применение его позволяет автоматически строить доказательства теорем и решать задачи искусственного интеллекта.

Задачей дисциплины является изучение общих концепций и методов современного декларативного программирования и, в частности, таких его разновидностей, как функциональное и логическое программирование, позволяющих эффективно решать задачи, связанные с обработкой символьной информации, нетипизированных данных, построения систем поддержки принятия решения, искусственного интеллекта, а также экспертных систем

2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ООП

Дисциплина «Функциональное и логическое программирование» относится к числу дисциплин по выбору. Теоретический материал закрепляется на лабораторных занятиях путем написания программ на языках Lisp и Prolog.

Успешное овладение данной дисциплиной ФЛП предполагает предварительное изучение дисциплин: "Математический анализ", "Дискретная математика", "Математическая логика и теория алгоритмов", "Основы программирования".

Знания и навыки, полученные при изучении дисциплины ФЛП, студенты смогут использовать при дальнейшем изучении дисциплины «Научно-исследовательская работа».

3. ТРЕБОВАНИЯ К РЕЗУЛЬТАТАМ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Процесс изучения дисциплины направлен на формирование следующих компетенций общепрофессиональных (ОПК) и профессиональных компетенций (ПК):

1. Способность к разработке алгоритмических и программных решений в области системного и прикладного программирования, математических, информационных и имитационных моделей, созданию информационных ресурсов глобальных сетей, образовательного контента, прикладных баз данных, тестов и средств тестирования систем и средств на соответствие стандартам и исходным требованиям (ОПК-3);

2. Способность к разработке и применению алгоритмических и программных решений в области системного и прикладного программного обеспечения (ПК-7).

Студент, изучивший дисциплину «Функциональное и логическое программирование», **должен**:

знать: языки функционального и логического программирования; основные методы и средства эффективной разработки программного продукта; типовые роли в процессе разработки программного обеспечения; методологии разработки программного обеспечения; математические основы лямбда-исчисления, предикатов первого порядка.

уметь: использовать методы и технологии разработки для генерации исполняемого кода; анализировать поставленные задачи, разрабатывать алгоритмы, представлять знания для решения поставленных задач; разрабатывать модели различных классов систем с применением языков функционального и логического программирования; программировать на языках Лисп и Пролог.

владеть: основными методологиями процессов разработки программного обеспечения; математическим аппаратом, применяемым в функциональном и логическом программировании; языками Лисп и Пролог для построения моделей искусственного интеллекта.

Согласована на портале № 16227

4. ОБЪЕМ ДИСЦИПЛИНЫ И ВИДЫ УЧЕБНОЙ РАБОТЫ

Общая трудоемкость дисциплины составляет **3** зачетные единицы.

Вид учебной работы	Всего	
	часов	6 семестр
Аудиторные занятия (всего)	**54**	**54**
В том числе:		
Лекции	18	18
Лабораторные работы (ЛР)	36	36
Самостоятельная работа (всего)	**54**	**54**
В том числе:		
Проработка лекционного материала	9	9
Подготовка к лабораторным занятиям	36	36
Самостоятельное изучение тем теоретической части	9	9
Подготовка к экзамену		
Вид промежуточной аттестации (зачет, экзамен)		зачет
Общая трудоемкость час	**108**	**108**
зач. ед.	**3**	**3**

5. СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

5.1. Разделы дисциплин и виды занятий

№ п/п	Наименование раздела дисциплины	Лек.	Лаб.	Сем.	СРС	Всего час.	Формируемые компетенции
1.	Декларативное программирование. Введение в функциональное программирование	2			2	4	ОПК-3, ПК-7
2.	Основы языка Лисп. Внутреннее представление списков	2	4		6	12	ОПК-3, ПК-7
3.	Математические основы языка лисп. Элементы лямбда-исчисления	2	4		6	12	ОПК-3, ПК-7
4.	Рекурсия в Лиспе	2	8		10	20	ОПК-3, ПК-7
5.	Функции более высокого порядка	2	4		6	12	ОПК-3, ПК-7
6.	Введение в логическое программирование	2			2	4	ОПК-3, ПК-7
7.	Основы языка Пролог. Рекурсия в Прологе	2	8		10	20	ОПК-3, ПК-7
8.	Порядок предложений и целей. Ограничение перебора	2	4		6	12	ОПК-3, ПК-7
9.	Метапрограммирование. Программирование второго порядка	2	4		6	12	ОПК-3, ПК-7

5.2. Содержание разделов дисциплины (по лекциям)

№ п/п	Наименование разделов	Содержание разделов	Трудо-емкость (час.)	Формируемые компетенции
1	Декларативное программирование. Введение в функциональное программирование	Современные концептуальные подходы в программировании. Компьютерная модель Фон-Неймана и ее недостатки. Декларативное программирование как концепция. Языки сверхвысокого уровня. Программирование, управляемое данными. Функциональный взгляд на вычисления. Функция как «черный ящик». Чистые функции. Функциональность.	2	ОПК-3, ПК-7
2	Основы языка лисп. Внутреннее представление списков	Символьная обработка данных. Одинаковая форма данных и программы. Автоматическое и динамическое управление памятью. Безтиповой язык программирования. Основные структуры данных: символы и списки. Список как средство представления знаний. Различные интерпретации списка. Единообразная префиксная нотация. Диалог с интерпретатором Лиспа. Иерархия вызовов. Блокировка вычисления выражений - quote. Принудительное вычисление выражения – eval. Функции обработки списков: first, rest, cons. Основные встроенные операторы. Накапливающие параметры. Локальные переменные и функции. Память и ссылочные ячейки. Указатели car и cdr. Логическое и физическое равенство списков.	2	ОПК-3, ПК-7

Согласована на портале № 16227

3	Математические основы языка лисп. Элементы лямбда-исчисления	Ламбда-исчисление как формальная система. Синтаксис и семантика λ-исчисления. Вычисление λ-выражений. Подстановка, конверсия, равенство. Порядок редукций и нормальные формы. β-редукция и проблема конфликта имен. Рекурсивные выражения. Комбинаторы. Чистое λ-исчисление. Ламбда-выражения в Лиспе. λ-вызов.	2	ОПК-3, ПК-7
4	Рекурсия в лиспе	Рекурсия как способ программирования повторяющихся вычислений и функций, определяемых через самих себя. Отличие рекурсии и простого цикла. Простая рекурсия. Примеры рекурсивных функций. Встроенные рекурсивные функции для обработки списков. Накапливающие параметры. Хвостовая рекурсия. Другие формы рекурсии: параллельная, взаимная, удаленная, рекурсия более высокого порядка.	2	ОПК-3, ПК-7
5	Функции более высокого порядка	Функционалы. Функциональный аргумент, функциональное значение функции. Способы композиции функций. Функции более высокого порядка. Применяющие функционалы. Отображающие функционалы. Композиция функционалов. Функциональное. Абстрактный подход - обобщение функций, имеющих одинаковый вид. Параметризованное определение функций. Автофункции. Автоаппликация и авторепликация.	2	ОПК-3, ПК-7
6	Введение в логическое программирование	Логический вывод. Метод резолюций. Унификация. Применение метода резолюций для ответа на вопросы. Введение в Пролог. Особенности языка Пролог. Пример программы: родственные отношения. Фразы Хорна как способ представления знаний. Алгоритм работы интерпретатора Пролога.	2	ОПК-3, ПК-7
7	Основы языка пролог. Рекурсия в прологе	Символы и списки. Константы и переменные. Синтаксис языка Пролог. Арифметические выражения, арифметические функции, арифметические предикаты. Составные термы (структуры), пример программы “Упрощение цепей”. Основные предикаты обработки списков: member, append, select. Примеры использования.	2	ОПК-3, ПК-7
8	Порядок предложений и целей. Ограничение перебора	Декларативная процедурная и семантика Пролога. Отсечение. Отсечения, меняющие процедурный и декларативный смыслы программы. Формальный алгоритм работы отсечения. Примеры, использующие отсечение. Отрицание как неудача. Трудности с отсечением и отрицанием. Программирование повторяющихся операций.	2	ОПК-3, ПК-7
9	Метапрограммирование. Программи-рование второго порядка	Эквивалентность данных и программ. Предположение об открытости мира. Внелогические предикаты: доступ к программам и обработка программ. Ввод и вывод. Программы, которые учатся у пользователя. Запоминающие функции. Модификация синтаксиса (операторная запись).	2	ОПК-3, ПК-7

5.3. Разделы дисциплины и междисциплинарные связи с обеспечивающими (предыдущими) и последующими дисциплинами

№ п/п	Наименование обеспечивающих (предыдущих) дисциплин	№ № разделов данной дисциплины, для которых необходимо изучение обеспечивающих (предыдущих) дисциплин								
		1	2	3	4	5	6	7	8	9
1.	Математический анализ	+	+	+	+	+				
2.	Дискретная математика						+	+	+	+
3.	Математическая логика и теория алгоритмов						+	+	+	+
4.	Основы программирования	+	+	+	+	+	+	+	+	+

№ п/п	Наименование последующих дисциплин	№ № разделов данной дисциплины, которые необходимы при изучении последующих дисциплин								
		1	2	3	4	5	6	7	8	9
1.	Научно-исследовательская работа	+	+	+	+	+	+	+	+	+

Согласована на портале № 16227

5.4. Соответствие компетенций, формируемых при изучении дисциплины, и видов занятий

Перечень компетенций	Лекц.	Лаб.	СРС	Формы контроля *(примеры)*
ОПК-3	+	+	+	Тест, конспект, письменный отчет по лабораторной работе
ПК-7	+	+	+	Тест, конспект, письменный отчет по лабораторной работе

Л – лекция, Лаб – лабораторные работы, СРС – самостоятельная работа студента

6. МЕТОДЫ И ФОРМЫ ОРГАНИЗАЦИИ ОБУЧЕНИЯ

Для успешного освоения дисциплины применяются различные образовательные технологии, которые обеспечивают достижение планируемых результатов обучения согласно основной образовательной программе, с учетом требований к объему занятий в интерактивной форме.

Технологии интерактивного обучения при разных формах занятий

Формы / Методы	Лекции (час)	Лабораторные занятия (час)	Всего (час)
IT-методы	2		**2**
Case-study (метод конкретных ситуаций)		2	**2**
Игра	2		**2**
Поисковый метод		2	**2**
Итого интерактивных занятий	**4**	**4**	**8**

Примечание.

1. IT-методы: на экран выводится алгоритм на тему лекции, от студентов требуется найти ошибку.
2. Работа в команде происходит при коллективном выполнении заданий практических работ (поиск оптимальных решений и case-study).
3. Различные игровые или конкретные ситуации из практики предлагаются студентам во время лекций в качестве демонстрации.
4. Поисковый метод используется для поиска логических ошибок алгоритма.

7. ЛАБОРАТОРНЫЙ ПРАКТИКУМ

Лабораторные занятия предусматривают закрепление основных теоретических вопросов данного курса на конкретных индивидуальных задачах, решение которых требует практическое применение полученных теоретических знаний.

№ п/п	№ раздела дисциплины	Наименование лабораторных работ	Трудо-емкость (час.)	Компетенции
1.	**2**	Лисп. Хранение и обработка данных в списочных структурах	4	ОПК-3, ПК-7
2.	**3**	Лисп. Обработка ламбда-выражений	4	ОПК-3, ПК-7
3.	**4**	Лисп. Программирование с использованием различных форм рекурсии	8	ОПК-3, ПК-7
4.	**5**	Лисп. Программирование функционалов	4	ОПК-3, ПК-7
5.	**7**	Пролог. Программирование с использованием различных форм рекурсии	8	ОПК-3, ПК-7
6.	**8**	Пролог. Программирование рекурсивных предикатов с ограничением перебора	4	ОПК-3, ПК-7
7.	**9**	Пролог. Программирование функционалов	4	ОПК-3, ПК-7

8. ПРАКТИЧЕСКИЕ ЗАНЯТИЯ (СЕМИНАРЫ) – не предусмотрены.

9. САМОСТОЯТЕЛЬНАЯ РАБОТА

№ п/п	№ раздела дисциплины из табл. 5.1	Тематика самостоятельной работы *(детализация)*	Трудо-емкость (час.)	Компетенции	Контроль выполнения работы (Опрос, тест, дом.задание, и т.д)
1.	1 ÷ 9	Проработка лекционного материала	9	ОПК-3, ПК-7	Опрос на занятиях (устно)
2.	2,5,6,8,9	Самостоятельное изучение тем теоретической части	9	ОПК-3, ПК-7	Домашнее задание
3.	2,3,4,5,7,8,9	Подготовка к лабораторным занятиям	36	ОПК-3, ПК-7	Отчет, защита лабораторной работы

Согласована на портале № 16227

Темы для самостоятельного изучения

1) Логическое и физическое равенство списков (тема 2, 1 час);
2) Автофункции. Автоаппликация и авторепликация. (тема 5, 2 часа);
3) Фразы Хорна как способ представления знаний (тема 6, 1 час);
4) Программирование повторяющихся операций (тема 8, 2 часа);
5) Ввод и вывод. Программы, которые учатся у пользователя (тема 9, 3 часа).

10. ПРИМЕРНАЯ ТЕМАТИКА КУРСОВЫХ ПРОЕКТОВ – не предусмотрены.

11. БАЛЛЬНО-РЕЙТИНГОВАЯ СИСТЕМА

Курс 3, семестр 6 Контроль обучения – Зачет

Максимальный семестровый рейтинг – **100 баллов.**

По дисциплине «Функциональное и логическое программирование» итоговой формой отчетности в 6 семестре является зачет, все 100 баллов входят в семестровую составляющую.

Для стимулирования планомерности работы студента в семестре в раскладку баллов по элементам контроля введен компонент своевременности, который применяется только для студентов, без опозданий, отчитывающихся по предусмотренным элементам контроля.

На протяжении всего семестра текущая успеваемость оценивается только в баллах нарастающим итогом, в том числе и результаты контрольных точек.

Текущий контроль изучения дисциплины состоит из следующих видов:

– контроль усвоения теоретического материала – проведение тестов во время проведения двух контрольных точек и по окончании семестра;

– контрольные работы на лабораторных занятиях.

В таблице 11.1 содержится распределение баллов в течение семестра для дисциплины «Функциональное и логическое программирование» завершающейся **зачетом** и содержащей 9 лекций (18 часов) и лабораторные работы (36 часов). В таблице 11.2 представлен пересчет суммы баллов по 1 и 2 контрольной точке в традиционную оценку.

Таблица 11.1 – Распределение баллов по дисциплине «Функциональное и логическое программирование»

Элементы учебной деятельности	Максимальный балл на 1-ую КТ с начала семестра	Максимальный балл за период между 1КТ и 2КТ	Максимальный балл за период между 2КТ и на конец семестра	Всего за семестр
Посещение занятий	4	4	4	**12**
Тестовый контроль	9	9	9	**27**
Выполнение лабораторных работ	14	21	14	**49**
Компонент своевременности	4	4	4	**12**
Итого максимум за период	**31**	**38**	**31**	**100**
Нарастающим итогом	**31**	**69**	**100**	

После окончания семестра студент, набравший менее 60 баллов, считается неуспевающим, не получившим зачет. **Студент, выполнивший все запланированные практические работы** и набравший сумму 60 и более баллов, получает зачет «автоматом».

Таблица 11.2 – Пересчет баллов в оценки за контрольные точки

Баллы на дату контрольной точки	Оценка
≥ 90 % от максимальной суммы баллов на дату КТ	5
От 70% до 89% от максимальной суммы баллов на дату КТ	4
От 60% до 69% от максимальной суммы баллов на дату КТ	3
< 60 % от максимальной суммы баллов на дату КТ	2

Согласована на портале № 16227

12. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

12.1. Основная литература

1. Ленивое функциональное программирование: учебное пособие / В. М. Зюзьков; Томский государственный университет систем управления и радиоэлектроники. - Томск: Издательство Томского университета, 2007. - 293[1] с. (51 экз.)

2. Логическое программирование: учебное пособие / В. М. Зюзьков; Томский государственный университет систем управления и радиоэлектроники. - 2-е изд., перераб. и доп. - Томск: Издательство Томского университета, 2007. – 142 с. (33 экз.)

12.2. Дополнительная литература

1. Хювёнен Э. Мир Лиспа: в 2 т.: пер. с фин. / Э. Хювёнен, Й. Сеппянен; пер. А. А. Рейтсакас, ред. В. Л. Стефанюк. - М.: Мир, 1990 - . Т. 1: Введение в язык Лисп и функциональное программирование: научное издание. - М.: Мир, 1990. – 446 с. (22 экз.)

2. Зюзьков В. М. Функциональное программирование: Учебное пособие / В. М. Зюзьков; Министерство образования Российской Федерации, Томский государственный университет систем управления и радиоэлектроники, Кафедра автоматизированных систем управления. - Томск: ТМЦДО, 2000. - 52 с. (10 экз.)

3. Программирование на языке Пролог: учебное пособие / И. А. Абрамов; Пензенский государственный педагогический университет им. В. Г. Белинского (Пенза). - Пенза: ПГПУ, 2011. - 116 с. (10 экз.).

4. Братко, И. Программирование на языке Пролог для искусственного интеллекта: Пер. с англ. / И. Братко. - М.: Мир, 1990. - 560 с. (78 экз.)

5. Шрайнер П. А. Основы программирования на языке Пролог: Курс лекций. Учебное пособие / П. А. Шрайнер; Интернет-Университет Информационных Технологий. - М.: Интернет-Университет Информационных Технологий, 2005. - 172 с. (20 экз.)

12.3. Перечень пособий, методических указаний и материалов, используемых в учебном процессе

Перечень методических указаний по лабораторным работам:

1. Монастырный, Е. А. Статистические методы в управлении инновациями: Методические рекомендации к лабораторным работам, практическим занятиям и самостоятельной работе [Электронный ресурс] / Монастырный Е. А. — Томск: ТУСУР, 2012. — 15 с. — Режим доступа:https://edu.tusur.ru/publications/1953

Перечень методических указаний по самостоятельной работе студентов:

3. Ковшов А.В. Функциональное и логическое программирование: методические указания по самостоятельной и индивидуальной работе студентов всех форм обучения для специальности 230105 – ПО ВТ и АС/ А. В. Ковшов; Министерство образования и науки Российской Федерации, Томский государственный университет систем управления и радиоэлектроники, Кафедра автоматизированных систем управления. - Томск, ТУСУР, 2012. - 9 с. − [Электронный ресурс]. – Режим доступа: http://el.asu.tusur.ru/course/view.php?id=17 (для зарегистрированных пользователей)

12.4. Учебно-методические пособия для лиц с ограниченными возможностями здоровья

Учебно-методические материалы для самостоятельной и аудиторной работы обучающихся из числа 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.

Согласована на портале № 16227

12.5. Базы данных, информационно-справочные, поисковые системы и требуемое программное обеспечение

1. Информационно-справочные и поисковые системы сети Интернет.

13. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

13.1. Общие требования к материально-техническому обеспечению дисциплины

13.1.1. Материально-техническое обеспечение для лекционных занятий

Для проведения занятий лекционного типа, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации используется учебная аудитория, с количеством посадочных мест не менее 22-24, оборудованная доской и стандартной учебной мебелью. Имеются наглядные пособия в виде презентаций по лекционным разделам дисциплины.

13.1.2. Материально-техническое обеспечение для лабораторных работ

Для проведения лабораторных занятий используется учебно-исследовательская вычислительная лаборатория, расположенная по адресу 634034, Томская область, г. Томск, Вершинина улица, д. 74, 4 этаж, ауд. 437, 438, 439. Состав оборудования: Учебная мебель; Экран с электроприводом DRAPER BARONET – 1 шт.; Мультимедийный проектор TOSHIBA – 1 шт.; Компьютеры класса не ниже Intel Pentium G3220 (3.0GHz/4Mb)/4GB RAM/ 500GB с широкополосным доступом в Internet, с мониторами типа Samsung 18.5" S19C200N– 10 шт.; Используется лицензионное программное обеспечение, пакеты версией не ниже: Microsoft Windows XP Professional with SP3; Visual Studio 2008 EE with SP1; Microsoft Office Visio 2010; Microsoft SQL-Server 2005; Matlab v6.5.

13.1.3. Материально-техническое обеспечение для самостоятельной работы

Для самостоятельной работы используется учебная аудитория (компьютерный класс), расположенная по адресу 634034, г. Томск, ул. Вершинина, 74, 1 этаж, ауд. 100. Состав оборудования: учебная мебель; компьютеры класса не ниже ПЭВМ INTEL Celeron D336 2.8ГГц. - 4 шт.; компьютеры подключены к сети ИНТЕРНЕТ и обеспечивают доступ в электронную информационно-образовательную среду университета.

13.2. Материально-техническое обеспечение дисциплины для лиц с ограниченными возможностями здоровья

Освоение дисциплины лицами с ОВЗ осуществляется с использованием средств обучения общего и специального назначения.

При обучении студентов **с нарушениями слуха** предусмотрено использование звукоусиливающей аппаратуры, мультимедийных средств и других технических средств приема/передачи учебной информации в доступных формах для студентов с нарушениями слуха, мобильной системы обучения для студентов с инвалидностью, портативной индукционной системы. Учебная аудитория, в которой обучаются студенты с нарушением слуха, оборудована компьютерной техникой, аудиотехникой, видеотехникой, электронной доской, мультимедийной системой.

При обучении студентов **с нарушениями зрениями** предусмотрено использование в лекционных и учебных аудиториях возможности просмотра удаленных объектов (например, текста на доске или слайда на экране) при помощи видеоувеличителей для удаленного просмотра.

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

14. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

14.1. Основные требования к фонду оценочных средств и методические рекомендации

Фонд оценочных средств и типовые контрольные задания, используемые для оценки сформированности и освоения закрепленных за дисциплиной компетенций при проведении текущей, промежуточной аттестации по дисциплине приведен в приложении к рабочей программе.

14.2. Требования к фонду оценочных средств для лиц с ограниченными возможностями здоровья

Для студентов с инвалидностью предусмотрены дополнительные оценочные средства, перечень которых указан в таблице 14.1.

Согласована на портале № 16227

Таблица 14.1 – Дополнительные средства оценивания для студентов с инвалидностью

Категории студентов	Виды дополнительных оценочных средств	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, исходя из состояния обучающегося на момент проверки

14.3. Методические рекомендации по оценочным средствам для лиц с ограниченными возможностями здоровья

Для студентов с ОВЗ 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.

Студентам с инвалидностью увеличивается время на подготовку ответов на контрольные вопросы. Для таких студентов 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 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 обучающихся с инвалидностью процедура оценивания результатов обучения может проводиться в несколько этапов.

Согласована на портале № 16227

Приложение к рабочей программе

Федеральное государственное бюджетное образовательное учреждение высшего образования

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

УТВЕРЖДАЮ

Проректор по учебной работе

_____ П. Е. Троян

«___» _____ 2017 г.

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ПО УЧЕБНОЙ ДИСЦИПЛИНЕ «ФУНКЦИОНАЛЬНОЕ И ЛОГИЧЕСКОЕ ПРОГРАММИРОВАНИЕ»

Уровень основной образовательной программы бакалавриат

Направление подготовки 01.03.02 Прикладная математика и информатика

Форма обучения очная

Факультет систем управления

Кафедра автоматизированных систем управления

Курс 3

Семестр 6

Учебный план набора 2013 года

Зачет 6 семестр

Томск 2017

Согласована на портале № 16227

1. ВВЕДЕНИЕ

Фонд оценочных средств (ФОС) является приложением к рабочей программе дисциплины «Функциональное и логическое программирование» и представляет собой совокупность контрольно-измерительных материалов (типовые задачи (задания), контрольные работы, тесты и др.) и методов их использования, предназначенных для измерения уровня достижения студентом установленных результатов обучения.

ФОС по дисциплине используется при проведении текущего контроля успеваемости и промежуточной аттестации студентов.

Перечень закрепленных за дисциплиной «Функциональное и логическое программирование» компетенций приведен в таблице 1.

Таблица 1 – Перечень закрепленных за дисциплиной компетенций

Код	Формулировка компетенции	Этапы формирования компетенции
ОПК-3	– способность к разработке алгоритмических и программных решений в области системного и прикладного программирования, математических, информационных и имитационных моделей, созданию информационных ресурсов глобальных сетей, образовательного контента, прикладных баз данных, тестов и средств тестирования систем и средств на соответствие стандартам и исходным требованиям	**Знать:** – Принципы построения функциональных и логических программ, типы данных, базовые конструкции, рекурсивные вычисления в языках программирования Lisp и Prolog. **Уметь:** – Работать в специализированных и типовых средах разработки, составлять рекуррентные последовательности для рекурсивных вычислений, писать программы на языке функционального программирования Lisp и языке логического программирования Prolog. **Владеть:** – Основными приемами декларативного программирования на примере функционального (язык Lisp) и логического (язык Prolog) подходов к программированию.
ПК-7	– способность к разработке и применению алгоритмических и программных решений в области системного и прикладного программного обеспечения	**Знать:** – основную цель и задачи декларативного программирования, реализуемого с применением функциональных и логических языков; – принципы построения декларативных программ, не имеющих четко определенного алгоритма решения, характерного для структурного/императивного программирования; – методы и подходы в функциональном и логическом программировании; – порядок разработки и выполнения программ, написанных на функциональных и логических языках; – применимость декларативного программирования для дальнейшей деятельности; **Уметь:** – применять имеющиеся знания для решения практических задач и тестовых заданий; – связывать программирование с математическими моделями; – пользоваться различными режимами при работе с декларативными языками программирования. **Владеть:** – сведениями о других существующих парадигмах и языках программирования; – навыками программирования в конкретных ситуациях и в зависимости от поставленной цели и предметной области.

Согласована на портале № 16227

2. РЕАЛИЗАЦИЯ КОМПЕТЕНЦИЙ

2.1 Компетенция ОПК-3

ОПК-3 - способность к разработке алгоритмических и программных решений в области системного и прикладного программирования, математических, информационных и имитационных моделей, созданию информационных ресурсов глобальных сетей, образовательного контента, прикладных баз данных, тестов и средств тестирования систем и средств на соответствие стандартам и исходным требованиям.

Для формирования компетенции необходимо осуществить ряд этапов. Этапы формирования компетенции, применяемые для этого виды занятий и используемые средства оценивания, представлены в таблице 2.

Таблица 2 – Этапы формирования компетенции и используемые средства оценивания

Состав	Знать	Уметь	Владеть
Содержание этапов	– Принципы построения функциональных и логических программ, типы данных, базовые конструкции, рекурсивные вычисления в языках программирования Lisp и Prolog.	– Работать в специализированных и типовых средах разработки, составлять рекуррентные последовательности для рекурсивных вычислений, писать программы на языке функционального программирования Lisp и языке логического программирования Prolog.	– Основными приемами декларативного программирования на примере функционального (язык Lisp) и логического (язык Prolog) подходов к программированию.
Виды занятий	– Лекции; – Лабораторные работы; – Групповые консультации.	– Лабораторные работы; – Самостоятельная работа студентов; – Зачет.	– Лабораторные работы; – Самостоятельная работа студентов; – Зачет.
Используемые средства оценивания	– Тестирование; – Реферат; – Защита лабораторных работ; – Устный опрос (зачет).	– Защита лабораторных работ; – Устный опрос (зачет).	– Защита лабораторных работ; – Устный опрос (зачет).

Общие характеристики показателей и критериев оценивания компетенции на всех этапах приведены в таблице 3.

Таблица 3 – Общие характеристики показателей и критериев оценивания компетенции по этапам

Показатели и критерии	Знать	Уметь	Владеть
ОТЛИЧНО (высокий уровень)	Обладает фактическими и теоретическими знаниями в пределах изучаемой области с пониманием границ применимости	Обладает диапазоном практических умений, требуемых для развития творческих решений, абстрагирования проблем	Самостоятельно планирует и выполняет работу, используя современный информационный инструментарий
ХОРОШО (базовый уровень)	Знает факты, принципы, процессы, общие понятия в пределах изучаемой области	Обладает диапазоном практических умений, требуемых для решения определенных проблем в области исследования	Берет ответственность за завершение задач в исследовании, приспосабливает свое поведение к обстоятельствам в решении проблем
УДОВЛЕТВО-РИТЕЛЬНО (низкий уровень)	Обладает низким уровнем общих знаний	Обладает умениями на низком уровне, которые не достаточны для выполнения даже простых задач	Работает только при прямом наблюдении

Согласована на портале № 16227

Формулировка показателей и критериев оценивания данной компетенции приведена в таблице 4.

Таблица 4 – Показатели и критерии оценивания компетенции на этапах

Показатели и критерии	Знать	Уметь	Владеть
ОТЛИЧНО (высокий уровень)	– Глубоко понимает принципы построения функциональных и логических программ, их типы данных, базовые конструкции, рекурсивных вычислений в языках программирования Lisp и Prolog.	– Обладает диапазоном практических умений, требуемых для развития творческих решений, абстрагирования проблем; – Уверенно работает со средами разработки и широко использует их возможности; – Разрабатывает оптимальные с точки зрения производительности программы на языках Lisp и Prolog с обработкой некорректных данных.	– Владеет навыками декларативного программирования на языках Lisp и Prolog, оценки сложности алгоритмов и программ; – Уверенно контролирует разработку программ, совершенствует программный код.
ХОРОШО (базовый уровень)	– Понимает принципы построения функциональных и логических программ, способен программировать рекурсивные вычисления на языках программирования Lisp и Prolog.	– Обладает диапазоном практических умений, требуемых для решения определенных проблем в области исследования; – Работает со средами разработки; – Пишет работоспособные программы на языках Lisp и Prolog, не всегда наиболее оптимальные по производительности.	– Берет ответственность за завершение задач, приспосабливает свое поведение к обстоятельствам в решении проблем; – Не склонен оптимизировать программный код.
УДОВЛЕТВО-РИТЕЛЬНО (низкий уровень)	– Имеет общее представление о принципа построения функциональных и логических программ, испытывает затруднения при программировании рекурсивных вычислений на языках программирования Lisp и Prolog, программный код не оптимальный.	– Обладает умениями на низком уровне, которые не достаточны для выполнения даже простых задач; – В общих чертах и с затруднениями пишет неоптимальный программный код на языках Lisp и Prolog.	– Слабо ориентируется во владении методами и приемами декларативного программирования; – Работает только при прямом наблюдении.

2.2 Компетенция ПК-7

ПК-7 – способность к разработке и применению алгоритмических и программных решений в области системного и прикладного программного обеспечения.

Для формирования компетенции необходимо осуществить ряд этапов. Этапы формирования компетенции, применяемые для этого виды занятий и используемые средства оценивания, представлены в таблице 5.

Таблица 5 – Этапы формирования компетенции и используемые средства оценивания

Состав	Знать	Уметь	Владеть
Содержание этапов	– основную цель и задачи декларативного программирования, реализуемого с применением функциональных и логических языков; – принципы построения декларативных программ, не имеющих четко определенного алгоритма решения, характерного для структурного/императивного	– применять имеющиеся знания для решения практических задач и тестовых заданий; – связывать программирование с математическими моделями; – пользоваться	– сведениями о других существующих парадигмах и языках программирования; – навыками программирования в конкретных ситуациях и в зависимости от поставленной цели и

Согласована на портале № 16227

	программирования; – методы и подходы в функциональном и логическом программировании; – порядок разработки и выполнения программ, написанных на функциональных и логических языках; – применимость декларативного программирования для дальнейшей деятельности.	различными режимами при работе с декларативными языками программирования.	предметной области.
Виды занятий	– Лекции; – Лабораторные работы; – Групповые консультации.	– Лабораторные работы; – Самостоятельная работа студентов; – Зачет.	– Лабораторные работы; – Самостоятельная работа студентов; – Зачет.
Используемые средства оценивания	– Тестирование; – Реферат; – Защита лабораторных работ; – Устный опрос (зачет).	– Защита лабораторных работ; – Устный опрос (зачет).	– Защита лабораторных работ; – Устный опрос (зачет).

Общие характеристики показателей и критериев оценивания компетенции на всех этапах приведены в таблице 6.

Таблица 6 – Общие характеристики показателей и критериев оценивания компетенции по этапам

Показатели и критерии	**Знать**	**Уметь**	**Владеть**
ОТЛИЧНО (высокий уровень)	Обладает фактическими и теоретическими знаниями в пределах изучаемой области с пониманием границ применимости	Обладает диапазоном практических умений, требуемых для развития творческих решений, абстрагирования проблем	Самостоятельно планирует и выполняет работу, используя современный информационный инструментарий
ХОРОШО (базовый уровень)	Знает факты, принципы, процессы, общие понятия в пределах изучаемой области	Обладает диапазоном практических умений, требуемых для решения определенных проблем в области исследования	Берет ответственность за завершение задач в исследовании, приспосабливает свое поведение к обстоятельствам в решении проблем
УДОВЛЕТВО-РИТЕЛЬНО (низкий уровень)	Обладает низким уровнем общих знаний	Обладает умениями на низком уровне, которые не достаточны для выполнения даже простых задач	Работает только при прямом наблюдении

Формулировка показателей и критериев оценивания данной компетенции приведена в таблице 7.

Таблица 7 – Показатели и критерии оценивания компетенции на этапах

Показатели и критерии	**Знать**	**Уметь**	**Владеть**
ОТЛИЧНО (высокий уровень)	– Глубоко понимает основную цель и задачи декларативного программирования, реализуемого с применением функциональных и логических языков, применимость декларативного	– Обладает диапазоном практических умений, требуемых для развития творческих решений, абстрагирования проблем – Уверенно применяет алгоритмические и	– Уверенно рассуждает о других существующих парадигмах и языках программирования; – Уверенно владеет навыками программирования в

Согласована на портале № 16227

	программирования для дальнейшей деятельности; – Знает принципы построения декларативных программ, не имеющих четко определенного алгоритма решения, характерного для структурного/императивного программирования;	программные решения в области прикладного программного обеспечения.	зависимости от поставленной задачи и предметной области – Уверенно контролирует разработку программ, совершенствует программный код
ХОРОШО (базовый уровень)	– Понимает основную цель и задачи декларативного программирования, реализуемого с применением функциональных и логических языков, применимость декларативного программирования для дальнейшей деятельности; – Понимает принципы построения функциональных и логических программ, способен программировать рекурсивные вычисления на языках программирования Lisp и Prolog.	– Обладает диапазоном практических умений, требуемых для решения определенных проблем в области исследования – Применяет алгоритмические и программные решения в области прикладного программного обеспечения.	– Владеет на хорошем уровне методами разработки алгоритмических и программных решений с применением функционального и логического подхода в области прикладного программного обеспечения.
УДОВЛЕТВО-РИТЕЛЬНО (низкий уровень)	– Имеет общие представления о целях и задачах декларативного программирования, реализуемого с применением функциональных и логических языков, применимость декларативного программирования для дальнейшей деятельности; – Поверхностно знает принципы построения декларативных программ.	– Неуверенно применяет алгоритмические и программные решения в области прикладного программного обеспечения. – В общих чертах и с затруднениями пишет неоптимальный программный код на языках Lisp и Prolog	– Слабо владеет методами и приемами декларативного программирования – Слабо владеет методами разработки алгоритмических и программных решений в области системного и прикладного программного обеспечения.

3. ТИПОВЫЕ КОНТРОЛЬНЫЕ ЗАДАНИЯ

Для реализации вышеперечисленных задач обучения используются следующие материалы: 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, в составе, приведенном ниже.

3.1 Темы лабораторных занятий

1. Лисп. Хранение и обработка данных в списочных структурах
2. Лисп. Обработка ламбда-выражений
3. Лисп. Программирование с использованием различных форм рекурсии
4. Лисп. Программирование функционалов
5. Пролог. Программирование с использованием различных форм рекурсии
6. Пролог. Программирование рекурсивных предикатов с ограничением перебора
7. Пролог. Программирование функционалов

3.2 Темы для самостоятельной работы (темы рефератов)

1. Современные концептуальные подходы в программировании.
2. Компьютерная модель Фон-Неймана и ее недостатки.
3. Декларативное программирование как концепция.

Согласована на портале № 16227

4. Языки сверхвысокого уровня.
5. Программирование, управляемое данными.
6. Функциональный взгляд на вычисления.
7. Функция как «черный ящик». Чистые функции. Функциональность.
8. Символьная обработка данных.
9. Автоматическое и динамическое управление памятью.
10. Список как средство представления знаний.
11. Различные интерпретации списка в Лиспе.
12. Локальные переменные и функции.
13. Представление списка в памяти Лисп-системы. Логическое и физическое равенство списков.
14. Ламбда-исчисление как формальная система.
15. Синтаксис и семантика □-исчисления. Вычисление □-выражений. Подстановка, конверсия, равенство. Порядок редукций и нормальные формы. □-редукция и проблема конфликта имен.
16. Рекурсия как способ программирования повторяющихся вычислений и функций, определяемых через самих себя.
17. Виды и формы рекурсии: простая, параллельная, взаимная, удаленная, рекурсия более высокого порядка.
18. Функционалы.
19. Абстрактный подход - обобщение функций, имеющих одинаковый вид. Параметризованное определение функций.
20. Автофункции. Автоаппликация и авторепликация.
21. Логический вывод.
22. Метод резолюций.
23. Особенности языка Пролог.
24. Алгоритм работы интерпретатора Пролога.
25. Декларативная процедурная и семантика Пролога.
26. Методы ограничения перебора в прологе.
27. Эквивалентность данных и программ в Прологе.
28. Метаинтерпретатор Пролога.

3.3 Контрольные вопросы для подготовки к зачету

1. Декларативное программирование. Особенности.
2. Отличия декларативного и процедурного программирования.
3. Функциональный взгляд на вычисления. Функция как черный ящик.
4. Свойство функциональности (прозрачности по ссылкам).
5. Чистое функциональное программирование.
6. Особенности языка Лисп.
7. Основные структуры данных Лиспа. Символьные выражения.
8. Список как средство представления данных и знаний. Списки в Лиспе.
9. Список как средство представления данных и знаний. Интерпретация списков.
10. Понятие функции. Особенности вызова функций и передачи аргументов в Лиспе.
11. Иерархия вызовов функций в Лиспе. Примеры.
12. Блокирование вычислений выражений.
13. Базовые функции обработки списков.
14. Внутреннее представление списков. Указатели.
15. Внутреннее представление списков. Логическое и физическое равенство списков.
16. Внутреннее представление объектов Лиспа в памяти. «Сборка мусора».
17. Функции сравнения: =, eq, eql, equal.
18. Определение функций в Лиспе.
19. Связывание. Формы set и setq. Побочный эффект от их использования. Примеры.
20. Разветвление вычислений в Лиспе. Функции cond и if. Примеры использования.
21. Локальные связи. Форма let. Примеры использования.
22. Локальные функции. Форма flet. Примеры использования.
23. Локальные функции. Форма labels. Примеры использования.

Согласована на портале № 16227

24. Рекурсия. Рекурсивные структуры.
25. Рекурсия. Прямая, косвенная, линейная рекурсия. Примеры.
26. Рекурсия. Рекурсивное определение функций length и count.
27. Рекурсия. Рекурсивное определение функций member и reverse.
28. Рекурсия. Рекурсивное определение функций append и remove.
29. Оптимизация рекурсивных вычислений: накапливающие параметры.
30. Оптимизация рекурсивных вычислений: хвостовая рекурсия.
31. Удаленная рекурсия.
32. Ламбда-исчисление. Синтаксис и семантика ламбда-исчисления. Ламбда-термы.
33. Ламбда-исчисление. Свободные и связанные переменные. Примеры.
34. Ламбда-исчисление. Вычисление ламбда-выражений. Правила подстановки.
35. Ламбда-исчисление. Коллизии переменных. Примеры.
36. Ламбда-исчисление. Вычисление ламбда-выражений. Правила конверсии.
37. Ламбда-исчисление. Вычисление ламбда-выражений. Равенства. Принцип экстенсиональности.
38. Ламбда-исчисление. Нормальные формы. Отношение редукции.
39. Ламбда-исчисление. Аппликативный порядок редукций. Пример.
40. Ламбда-исчисление. Нормальный порядок редукций. Пример.
41. Ламбда-исчисление. Функция, отбрасывающая свой аргумент.
42. Ламбда-абстракция в Лиспе. Форма lambda. Примеры.
43. Функциональные аргументы и значения. Функционалы.
44. Применяющие функционалы. Функции apply, funcall. Примеры использования.
45. Применяющие функционалы. Отображение списка. Функция mapcar. Примеры использования.
46. Применяющие функционалы. Фильтрация списка. Функция mapcar. Примеры использования.
47. Замыкание (функциональная блокировка).
48. Абстракция вычислений. Примеры использования.
49. Абстракция вычислений с помощью встроенной функции reduce.
50. Автофункции.

4. МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ

Для обеспечения процесса обучения и решения задач обучения используются следующие материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций (приведены в разделах 12.1 – 12.3 рабочей программы):

1. Ленивое функциональное программирование : учебное пособие / В. М. Зюзьков ; Томский государственный университет систем управления и радиоэлектроники. - Томск : Издательство Томского университета, 2007. - 293[1] с. (51 экз.)

2. Логическое программирование : учебное пособие / В. М. Зюзьков ; Томский государственный университет систем управления и радиоэлектроники. - 2-е изд., перераб. и доп. - Томск : Издательство Томского университета, 2007. – 142 с. (33 экз.)

3. Хювёнен Э. Мир Лиспа : в 2 т.: пер. с фин. / Э. Хювёнен, Й. Сеппянен; пер. А. А. Рейтсакас, ред. В. Л. Стефанюк. - М. : Мир, 1990 - . Т. 1 : Введение в язык Лисп и функциональное программирование : научное издание. - М. : Мир, 1990. – 446 с. (22 экз.)

4. Зюзьков В. М. Функциональное программирование : Учебное пособие / В. М. Зюзьков; Министерство образования Российской Федерации, Томский государственный университет систем управления и радиоэлектроники, Кафедра автоматизированных систем управления. - Томск : ТМЦДО, 2000. - 52 с. (11 экз.)

5. Программирование на языке Пролог: учебное пособие / И. А. Абрамов ; Пензенский государственный педагогический университет им. В. Г. Белинского (Пенза). - Пенза : ПГПУ, 2011. - 116 с. (10 экз.).

Согласована на портале № 16227

6. Братко, И. Программирование на языке Пролог для искусственного интеллекта : Пер. с англ. / И. Братко. - М. : Мир, 1990. - 560 с. (81 экз.)

7. Шрайнер П. А. Основы программирования на языке Пролог : Курс лекций. Учебное пособие / П. А. Шрайнер; Интернет-Университет Информационных Технологий. - М.: Интер-нет-Университет Информационных Технологий, 2005. - 172 с. (20 экз.)

8. Методические указания по выполнению лабораторных работ в [1] п. 3.3 – стр. 6-39.

Согласована на портале № 16227