

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
ПЕНЗЕНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ФАКУЛЬТЕТ ВЫЧИСЛИТЕЛЬНОЙ ТЕХНИКИ

УТВЕРЖДАЮ
Декан факультета
вычислительной техники



(Подпись)

Фионова Л.Р.

(Фамилия, Инициалы)

« _____ » _____ 2015 г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Б1.2.23.2 Программирование на декларативных языках

Направление подготовки: 01.03.02 Прикладная математика и информатика

Профиль подготовки: Системное программирование и компьютерные технологии

Квалификация (степень) выпускника – Бакалавр

Форма обучения: Очная

Пенза – 2015

1. Цели освоения дисциплины

Целью освоения дисциплины «Программирование на декларативных языках» является формирование и развитие у студентов общекультурных и профессиональных компетенций, формирование систематизированных знаний, умений и навыков в области прикладной математики и информатики, позволяющих подготовить конкурентоспособного выпускника для соответствующей сферы, готового к инновационной творческой реализации в области системного программирования и компьютерных технологий.

Задачи изучаемой дисциплины:

Исходя из общих целей подготовки бакалавра прикладной математики и информатики по профилю «Системное программирование и компьютерные технологии»:

- содействовать средствами дисциплины «Программирование на декларативных языках» развитию у студентов мотивации к профессиональной деятельности, творческого мышления, коммуникативной готовности, общей культуры;
- научить студентов ясно, точно, грамотно излагать свои мысли в устной и письменной речи.

Исходя из конкретного содержания дисциплины:

- формирование у будущих системных программистов систематизированных знаний в области современных программных средств логического и функционального программирования;
- изучение основных понятий и конструкций современных языков логического и функционального программирования, множества задач, решаемых с применением логического и функционального подходов к программированию;
- изучение синтаксиса и семантики современных языков логического и функционального программирования (Lisp, Turbo Prolog и Visual Prolog).

2. Место дисциплины в структуре ОПОП бакалавриата

Дисциплина «Программирование на декларативных языках» относится к дисциплинам по выбору вариативной части учебного плана.

Для освоения дисциплины студенты используют знания, умения, навыки, сформированные в процессе изучения предметов «Основы информатики», «Алгоритмы и алгоритмические языки», «Математическая логика», «Языки и методы программирования».

Освоение данной дисциплины является основой для последующего освоения дисциплины профессионального цикла «Искусственный интеллект», последующего прохождения практики и подготовки к итоговой государственной аттестации.

3. Компетенции обучающегося, формируемые в результате освоения дисциплины «Программирование на декларативных языках»

Процесс изучения дисциплины направлен на формирование элементов следующих компетенций в соответствии с ФГОС ВО по данному направлению:

Коды компетенции	Наименование компетенции	Структурные элементы компетенции (в результате освоения дисциплины обучающийся должен знать, уметь, владеть)
ОПК-1	способность использовать базовые знания естественных наук, математики и информатики, основные факты, концепции, принципы теорий, связанные с прикладной математикой и информатикой	Знать: основные концепции и факты математики, которые используются при программном решении задач
		Уметь: применять математические знания при программировании
		Владеть: основными методами объектно-ориентированного программирования

ОПК-3	способностью 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: сущность и значение понятия алгоритма, его применения в области анализа и проектирования программного обеспечения с помощью объектно-ориентированного подхода.
		Уметь: использовать объектно-ориентированный подход для создания программного обеспечения
		Владеть: основными методами создания программного обеспечения с использованием объектно-ориентированного подхода.
ОПК-4	способностью решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности.	Знать: основные стандартные задачи создания программного обеспечения на основе объектно-ориентированного подхода.
		Уметь: использовать современные библиотеки алгоритмов, современные среды программирования.
		Владеть: основными алгоритмическими и системными решениями в области системного и прикладного ПО
ПК-7	способность к разработке и применению алгоритмических и программных решений в области системного и прикладного ПО	Знать: основные направления современных научных исследований в области объектно-ориентированного проектирования программного обеспечения.
		Уметь: использовать данные современных научных исследований, необходимые для формирования выводов в области методологии объектно-ориентированного проектирования
		Владеть: основными алгоритмическими и системными решениями в области системного и прикладного ПО

ПСК-1	Способностью к формализации и алгоритмизации поставленных задач	<p>Знать: основные этапы решения задач с помощью ЭВМ.</p> <p>Владеть: современными способами решения задач с помощью ЭВМ.</p> <p>Уметь: применять полученные знания для решения конкретных задач</p>
ПСК-2	Способностью к написанию программного кода с использованием языков программирования, определения и манипулирования данными	<p>Знать: основные конструкции языка C++.</p> <p>Владеть: современными средствами создания программного кода.</p> <p>Уметь: создавать приложения в среде MS Visual Studio</p>

4. Структура и содержание дисциплины «Логическое и функциональное программирование»

4.1. Структура дисциплины

Общая трудоемкость дисциплины составляет 6 зачетных единиц 216 часов.

№ п/п	Наименование разделов и тем дисциплины	Семестр	Недели семестра	Виды учебной работы, включая самостоятельную работу студентов и трудоемкость (в часах)							Формы текущего контроля успеваемости (по неделям семестра)		
				Аудиторная работа			Самостоятельная работа				Опрос на лабораторном занятии	Проверка отчёта выполнения лабораторной работы	
				Всего	Лекция	Лабораторные занятия	Всего	Подготовка к аудиторным занятиям	Подготовка отчёта лабораторной работы	Подготовка к экзамену			
1	2	3	4	5	6	7	8	9	11	12	13	14	
1.	Раздел 1. Функциональная и логическая парадигмы программирования	6	1-5	30	12	18	19	11	8				
1.1.	Декларативный подход в программировании	6	1	5	2	3	10	3	2			1	1
1.2.	Основные понятия функциональных языков. Базовые функции	6	2-3	10	4	6	10	3	2			1	1
1.3.	Функции в функциональном программировании	6	4-5	5	2	3	10	3	2			1	1
1.4.	Рекурсивные функции	6	5-6	10	4	6	9	2	2			1	1
2.	Раздел 2. Логические методы программирования	6	7-13	35	14	21	28	14	14				
2.1.	Введение в логическое программирование	6	7	5	2	3	7	2	2			1	1
2.2.	Основные понятия языка Пролог	6	8	5	2	3	6	2	2			1	1

2.3.	Правила поиска в языке Пролог	6	9	5	2	3	6	2	2		1	1
2.4.	Простые и составные объекты данных	6	10	5	2	3	5	2	2		1	1
2.5.	Организация повторов в языке Пролог	6	11	5	2	3	5	2	2		1	1
2.6.	Рекурсивные структуры данных	6	12	5	2	3	5	2	2		1	1
2.7.	Внутренняя база фактов	6	13	5	2	3	5	2	2		1	1
3.	Раздел 3. Программирование на языках LISP и F#	6	14-20	35	14	21	33	14	19			
3.1.	Методы разработки функциональных программ	6	14	5	2	3	8	2	3		1	1
3.2.	Управляющие структуры	6	15-16	10	4	6	8	4	4		1	1
3.3.	Работа со списками	6	17-18	10	4	6	8	4	4		1	1
3.4.	Свойства символов	6	19	5	2	3	7	2	4		1	1
3.5.	Производные структуры данных	6	20	5	2	3	7	2	4		1	1
	Общая трудоемкость, в часах										Промежуточная аттестация	
		216	100	40	60	116	39	41	36	Форма	Сем естр	
										Зачет	6	
										Экзамен	6	

4.2. Содержание дисциплины

4.2.1. Содержание лекционных занятий

Раздел 1. Функциональная и логическая парадигмы программирования

Тема 1.1. Декларативный подход в программировании как альтернатива процедурной парадигме

Функциональное и логическое программирование как научная дисциплина. Декларативные и процедурные способы представления знаний. Понятие декларативного программирования. Две семантики декларативных языков. Область применения языков функционального и логического программирования.

Тема 1.2. Основные понятия функциональных языков. Базовые функции

О языках функционального программирования. Особенности функционального программирования. Понятие строго функционального языка. Примеры функциональных языков программирования. Атом, выражение, список. Селекторы, конструкторы и предикатные функции. Функции конструирования списков.

Тема 1.3. Функции в функциональном программировании

Описание функций Неименованные функции, лямбда-определение и лямбда-вызов. Именованные функции, виды описаний именованных функций.

Тема 1.4. Рекурсивные функции

Способы организации рекурсии. Требования к рекурсивным функциям. Виды условий завершения рекурсивных вызовов.

Раздел 2. Логические методы программирования

Тема 2.1. Введение в логическое программирование

Определения базовых понятий: логическая программа, факт, правила и целевое утверждение. Правило резолюции и логический вывод. Вычисление цели логической программой и абстрактный интерпретатор логических программ. Означивание цели и значение логической программы. Общие сведения о языках логического программирования. Области применения языка логического программирования Пролог.

Тема 2.2. Основные понятия языка Пролог

Основные элементы языка. Предложения: факты и правила. Запросы. Предикаты. Переменные и константы. Цели. Проблемы, связанные с использованием отрицания и предиката отсечения.

Тема 2.3. Правила поиска в языке Пролог

Сопоставление. Возвраты. Использование встроенных предикатов fail и not. Отсечение. Декларативный смысл Пролог-программы. Процедурная семантика.

Тема 2.4. Простые и составные объекты данных

Простые объекты информации: переменные, константы, атомы. Составные объекты данных и функторы. Сопоставление составных объектов. Объявление типов составных объектов. Сопоставление составных объектов. Объявление типов составных объектов. Составные объекты смешанных типов.

Тема 2.5. Организация повторов в языке Пролог

Поиск с возвратом. Использование отката с петлями. Рекурсивные правила. Хвостовая рекурсия. Способы задания хвостовой рекурсии.

Тема 2.6. Рекурсивные структуры данных

Представление списков. Операции над списками. Сортировка. Рекурсия в списках. Структурные списки. Поиск всех решений для цели. Дерево двоичного поиска. Рекурсия в информационных структурах. Деревья в информационных структурах.

Тема 2.7. Внутренняя база фактов

Объявление внутренней базы фактов. Использование внутренних баз фактов. Обновление внутренней базы фактов. Занесение фактов во время выполнения программы. Считывание фактов из файла. Удаление фактов во время выполнения программы. Удаление нескольких

фактов сразу. Сохранение базы фактов во время работы программы. Работа с фактами как с терминами.

Раздел 3. Программирование на языках LISP и F#

Тема 3.1. Методы разработки функциональных программ

Методы восходящего и нисходящего проектирования функциональных программ. Основные и вспомогательные функции. Локальные определения. Рекурсивные локальные определения.

Тема 3.2. Управляющие структуры

Организация последовательных вычислений. Условные предложения. Организация циклических вычислений.

Тема 3.3. Работа со списками

Внутреннее представление списков. Логическая и физическая эквивалентность структуры списков. Точечные пары. Организация деревьев.

Тема 3.4. Свойства символов

Системные свойства символов. Функции для определения и модификации списка свойств символа.

Тема 3.5. Производные структуры данных

Записи и структуры, основные действия над ними. Массивы, работа с массивами.

5. Образовательные технологии

В ходе освоения дисциплины «Программирование на декларативных языках», при проведении аудиторных занятий, используются технологии традиционных и нетрадиционных учебных занятий.

Технология традиционного обучения предусматривает такие методы и формы изучения материала как лекция, лабораторные занятия.

- Проведение интерактивной лекции, демонстрирующей сравнение структуры императивной и декларативной программы (Тема 2.1. Введение в логическое программирование).
- Проведение проблемной лекции (Тема 2.6. Рекурсивные структуры данных).

При изучении дисциплины «Программирование на декларативных языках» используются активные и интерактивные технологии обучения, такие как:

- кейс-технология (Тема 1.4. Рекурсивные функции).

Занятия, проводимые в интерактивной форме, в том числе с использованием интерактивных технологий составляют 25% от общего количества аудиторных занятий.

Самостоятельная работа студентов подразумевает работу под руководством преподавателя (консультации, коллоквиумы) и индивидуальную работу студента, выполняемую, в том числе, в компьютерном классе с выходом в сеть «Интернет» на факультете ВТ университета.

При реализации образовательных технологий используются следующие виды самостоятельной работы:

- работа с конспектом лекции;
- подготовка к лабораторной работе;
- подготовка к контрольной и проверочной работе;
- поиск информации в сети «Интернет» и дополнительной и справочной литературе.

В целях реализации индивидуального подхода к обучению студентов, осуществляющих учебный процесс по собственной траектории в рамках индивидуального рабочего плана, изучение данной дисциплины базируется на следующих возможностях: обеспечение внеаудиторной работы со студентами в том числе в электронной образовательной среде с использованием соответствующего программного оборудования, дистанционных форм обучения, возможностей интернет-ресурсов, индивидуальных консультаций и т.д.

<p>1.3.</p>	<ul style="list-style-type: none"> • <i>Подготовка к аудиторному занятию: работа с конспектом лекции;</i> <p>Рассмотреть примеры описания функций в языках программирования.</p> <ul style="list-style-type: none"> • <i>Подготовка к лабораторной работе</i> <p>Рассмотреть вопросы:</p> <ol style="list-style-type: none"> 1. λ-исчисление. Редукция. 2. Соответствие между вычислениями функциональных программ и редукцией. <p>Выполнение индивидуального занятия Подготовка отчета</p>	<p>3,4</p>	<p>2</p> <p>4</p> <p>2</p>
<p>1.4.</p>	<ul style="list-style-type: none"> • <i>Подготовка к аудиторному занятию: работа с конспектом лекции;</i> <p>Рассмотреть примеры описания рекурсивных функций в языках программирования.</p> <ul style="list-style-type: none"> • <i>Подготовка к лабораторной работе</i> <p>Рассмотреть вопросы:</p> <ol style="list-style-type: none"> 1. Способы организации рекурсии в языках программирования. 2. Базис и шаг рекурсии. 3. Условия выхода из рекурсии. <ul style="list-style-type: none"> • <i>Подготовка к экзамену</i> <p>Рассмотреть вопросы:</p> <ol style="list-style-type: none"> 1. Функциональное и логическое программирование как научная дисциплина. Декларативные и процедурные способы представления знаний. 2. Понятие декларативного программирования. Две семантики декларативных языков. Область применения языков функционального и логического программирования. 3. Особенности функционального программирования. Понятие строго функционального языка. Примеры функциональных языков программирования. 4. Атом, выражение, список. Селекторы, конструкторы и предикатные функции. Функции конструирования списков. 5. Функции в функциональном программировании. 6. Рекурсивные функции. 7. Методы разработки функциональных программ. <p>Выполнение индивидуального занятия Подготовка отчета</p>	<p>3,4</p>	<p>2</p> <p>4</p> <p>2</p>

6.3. Материалы для проведения текущего и промежуточного контроля знаний студентов

Контроль освоения компетенций

№ п\п	Вид контроля	Контролируемые темы (разделы)	Компетенции, компоненты которых контролируются
1	Текущий: проверка выполнения индивидуального задания на лабораторной работе	Разделы 1 – 3	ОПК-1,3,4; ПК-7 ПСК-1,2
2	Текущий: собеседование при защите лабораторных работ	Разделы 1 – 3	ОПК-1,3,4; ПК-7 ПСК-1,2

Средства текущего контроля

Текущий контроль выполняется на лабораторных занятиях при защите выполненных студентом лабораторных работ. Защита предполагает представление отчета с обоснованием примененных решений выполнения лабораторного задания, ответы на контрольные вопросы преподавателя.

Темы лабораторных работ

1. Представление рекурсивных функций.
2. Знакомство с основами функционального программирования. Описание и вызов функций в языке Лисп.
3. Методы разработки функциональных программ. Локальные определения.
4. Внутреннее представление списков.
5. Основы рекурсии. Другие формы рекурсии.
6. Структура программы на языке Пролог. Составление правил.
7. Использование встроенных предикатов в языке Пролог.
8. Рекурсивные правила.
9. Обработка строк, строковые предикаты.
10. Обработка списков.
11. Внутренние динамические базы данных.
12. Работа с деревьями.
13. Обработка файлов в языке Пролог.
14. Решение логических задач средствами языка Пролог.
15. Особенности работы среды Visual Prolog.

Примерные задания для зачета

1. Найти полу сумму наибольшего и наименьшего из трех чисел.
2. Подсчитать количество цифр в строке.
3. Перевести число из десятичной системы счисления в восьмеричную.
4. Обратить список.
5. Найти максимальную цифру в строке.
6. Заполнить список гласными буквами строки.
7. Написать СУБД «Магазин», содержащую данные о товарах в виде:

<отдел> <товар> <цена>. СУБД может осуществлять следующий сервис:

- a) Добавление записи в БД на диске.
- b) Удалять запись об определенном товаре в БД.
- c) Удалять записи о товарах определенного отдела.
- d) Удалять самые дешевые товары.
- e) Выводить содержимое БД на экран.
- f) Выводить товары, принадлежащие определенному отделу, с их расценками.
- g) Выводить наименование самого дорогого товара.
- h) Осуществлять выход из СУБД.

Вопросы для экзамена

1. Функциональное и логическое программирование как научная дисциплина. Декларативные и процедурные способы представления знаний.
2. Понятие декларативного программирования. Две семантики декларативных языков. Область применения языков функционального и логического программирования.
3. Особенности функционального программирования. Понятие строго функционального языка. Примеры функциональных языков программирования.
4. Атом, выражение, список. Селекторы, конструкторы и предикатные функции. Функции конструирования списков.
5. Функции в функциональном программировании.
6. Рекурсивные функции.
7. Методы разработки функциональных программ.
8. Управляющие структуры в языке Лисп.
9. Работа со списками в языке Лисп.
10. Производные структуры данных в языке Лисп.
11. Введение в логическое программирование на языке Пролог
12. Логические основы языка Пролог: n-местные предикаты. Операции над предикатами.
13. Структура программы на языке Пролог.
14. Встроенные предикаты Пролога: предикаты ввода-вывода, управления экраном, арифметические, алгебраические предикаты.
15. Правила в Прологе.
16. Встроенный предикат fail. Внутренний механизм унификации Пролога.
17. Рекурсивные правила в Прологе.
18. Встроенные строковые предикаты в Прологе.
19. Списки в Прологе, определение, описание. Основные действия со списками в Прологе.
20. Сортировка списков.
21. Внутренние динамические базы данных. Встроенные предикаты по обработке динамических баз данных.
22. Применение динамических предикатов в Прологе.
23. Описание файлового домена в Прологе.
24. Встроенные предикаты по работе с файлами в Прологе. Основные действия по работе с файлами в Прологе.
25. Примеры обработки файловых величин в Прологе.
26. Деревья в Прологе, определение, описание.
27. Основные действия с деревьями в Прологе.
28. Двоичные справочники в Прологе.
29. Реализация поиска на графах: пространство состояний задач.
30. Стратегия поиска в глубину.
31. Стратегия поиска в ширину.
32. Реализация ООП в Prolog.

7. Учебно-методическое и информационное обеспечение дисциплины «Программирование на декларативных языках»

а) основная литература:

1. Братко, Иван. Программирование на языке пролог для искусственного интеллекта - М. : Мир, 1990. - 559 с (16)

http://kleopatra.pnzgu.ru/cgi-bin/irbis64r_91/cgiirbis_64.exe?P21DBN=KATL&I21DBN=KATL_PRINT&S21FMT=fullw_print&C21COM=F&Z21MFN=4025

2. Острейковский В. А. Информатика. М. : Высш. шк., 2005. - 511 с. (10)

http://kleopatra.pnzgu.ru/cgi-bin/irbis64r_91/cgiirbis_64.exe?P21DBN=KATL&I21DBN=KATL_PRINT&S21FMT=fullw_print&C21COM=F&Z21MFN=7681

3. Гринченков Д.В. Математическая логика и теория алгоритмов для программистов.- М. : КНОРУС, 2017. - 206 с. (10)

http://kleopatra.pnzgu.ru/cgi-bin/irbis64r_91/cgiirbis_64.exe?P21DBN=KATL&I21DBN=KATL_PRINT&S21FMT=fullw_print&C21COM=F&Z21MFN=18119

б) дополнительная литература:

4. Функциональное и логическое программирование : учебное пособие / А. Л. Ездаков. - 2-е изд. - М. : БИНОМ. Лаборатория знаний, 2011. - 119 с. (5)

http://kleopatra.pnzgu.ru/cgi-bin/irbis64r_91/cgiirbis_64.exe?P21DBN=KATL&I21DBN=KATL_PRINT&S21FMT=fullw_print&C21COM=F&Z21MFN=14824

в) программное обеспечение и Интернет-ресурсы

1. Зыков С. Введение в теорию программирования. Функциональный подход: курс лекций НОУ Интуит <http://www.intuit.ru/studies/courses/39/39/info>

2. Сошников Д. Логическое программирование: курс лекций НОУ Интуит <http://www.intuit.ru/studies/courses/558/414/info>

3. Сошников Д. Функциональное программирование: курс лекций НОУ Интуит <http://www.intuit.ru/studies/courses/471/327/info>

2. Шрайнер П. Основы программирования на языке Пролог: курс лекций НОУ Интуит <http://www.intuit.ru/studies/courses/44/44/info>

8. Материально-техническое обеспечение дисциплины

В целях оптимизации учебного процесса студенты используют рабочие места в компьютерном классе, оборудованном локальной сетью и выходом в Internet, имеющиеся в библиотеке учебники. Все работы выполняются на персональных компьютерах, на соответствующих практических занятиях - с использованием среды программирования Visual Prolog.

Рабочая программа дисциплины «Программирование на декларативных языках» составлена в соответствии с требованиями ФГОС ВО по направлению подготовки 01.03.02 "Прикладная математика и информатика".

Программу составил:
Горюнов Ю.Ю., к.ф.-м.н., доцент



(подпись)

Настоящая программа не может быть воспроизведена ни в какой форме без предварительного письменного разрешения кафедры-разработчика программы.

Программа одобрена на заседании кафедры "Компьютерные технологии"

Протокол № 8^а

от "16" апреля 2015 года

Зав. кафедрой КТ



В.И. Горбаченко

Программа одобрена методической комиссией ФВТ

Протокол № 5^а

от "17" апреля 2015 года

Председатель методической комиссии
ФВТ



Н. Н. Казанов

Сведения о переутверждении программы на очередной учебный год и регистрации изменений

Учебный год	Решение кафедры (№ протокола, дата, подпись зав. кафедрой)	Внесенные изменения	Номера листов (страниц)		
			замененных	новых	аннулированных
2015/2016	Переутвердить б/д изменения №1 от 31.08.2015 <i>[подпись]</i>				
2016/2017	Переутвердить б/д изменения №1 от 30.08.2016 <i>[подпись]</i>				