

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ
**«БЕЛГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ
ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ им. В.Г.ШУХОВА»**
(БГТУ им. В.Г. Шухова)



РАБОЧАЯ ПРОГРАММА
дисциплины

Функциональное и логическое программирование

Направление подготовки:
09.03.04 Программная инженерия

профиль подготовки:
Разработка программно-информационных систем

Квалификация (степень)
бакалавр

Форма обучения
очная


Институт информационных технологий и управляющих систем

**Кафедра программного обеспечения вычислительной техники и
автоматизированных систем**

Белгород – 2015

Рабочая программа составлена на основании требований:

- Федерального государственного образовательного стандарта высшего образования по направлению подготовки 09.03.04 «Программная инженерия» (уровень бакалавриата), утверждённого приказом Министерства образования и науки Российской Федерации № 229 от 12 марта 2015 г.
- плана учебного процесса БГТУ им. В.Г. Шухова по направлению подготовки 09.03.04 «Программная инженерия», профиль «Разработка программно-информационных систем».

Составитель: старший преподаватель  (В.К. Поляков)
(ученая степень и звание, подпись) (инициалы, фамилия)

Рабочая программа согласована с выпускающей кафедрой
Программного обеспечения вычислительной техники и автоматизированных систем

Заведующий кафедрой: к.т.н., доцент  (В.М. Поляков)
(ученая степень и звание, подпись) (инициалы, фамилия)

« 16 » 04 2015 г.

Рабочая программа обсуждена на заседании кафедры
Программного обеспечения вычислительной техники и автоматизированных систем

« 16 » 04 2015 г., протокол № 11

Заведующий кафедрой: к.т.н., доцент  (В.М. Поляков)
(ученая степень и звание, подпись) (инициалы, фамилия)

Рабочая программа одобрена методической комиссией института
Информационных технологий и управляющих систем

« 23 » 04 2015 г., протокол № 3/12

Председатель: доцент  (Ю.И. Солопов)
(ученая степень и звание, подпись) (инициалы, фамилия)

1. ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ

Формируемые компетенции			Требования к результатам обучения
№	Код компетенции	Компетенция	
Профессиональные			
3	ПК-3	Владение навыками использования различных технологий разработки программного обеспечения	Знать: особенности программирования на языках функционального и логического программирования. Уметь: разрабатывать ПО на языках функционального и логического программирования с использованием внешних библиотек. Владеть: навыками использования программных средств написания, компиляции (интерпретации), профилирования, отладки программ на функциональных и логических языках программирования.

2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ

Содержание дисциплины основывается и является логическим продолжением следующих дисциплин:

№	Наименование дисциплины (модуля)
1	Основы программирования
2	Информатика
3	Алгоритмы и структуры данных
4	Математическая логика и теория алгоритмов

Содержание дисциплины служит основой для изучения следующих дисциплин:

№	Наименование дисциплины (модуля)
1	Спецификация, архитектура и проектирования программных систем
2	Конструирование программных систем

3. ОБЪЁМ ДИСЦИПЛИНЫ

Общая трудоемкость дисциплины составляет 3 зач. единицы, 108 часов.

Вид учебной работы	Всего часов	Семестр №4
Общая трудоемкость дисциплины, час	108	108
Контактная работа (аудиторные занятия), в т.ч.:	51	51
лекции	17	17
лабораторные	34	34
практические	–	–
Самостоятельная работа студентов, в том числе:	57	57
Курсовой проект	–	–
Курсовая работа	–	–
Расчетно-графическое задание	–	–
Индивидуальное домашнее задание	–	–
<i>Другие виды самостоятельной работы</i>	57	57
Форма промежуточной аттестации (зачёт)	<i>дифф. зачёт</i>	<i>дифф. зачёт</i>

4. СОДЕРЖАНИЕ ДИСЦИПЛИНЫ
4.1 Наименование тем, их содержание и объем
Курс 2 Семестр 4

№ п/п	Наименование раздела	Объём на тематический раздел, час			
		Кол-во лекционных часов	Практические и др. занятия	Лабораторные занятия	Самостоятельная работа
1	2	3	4	5	6
Функциональное программирование					
(наименование тематического раздела)					
1	Введение в функциональное программирование. Понятие императивного и декларативного программирования. Недостатки современных языков программирования. Сильные стороны функционального программирования. Развитие многоядерных процессоров и языка функционального программирования. Области и примеры применения функциональных языков. Сильные стороны языка Haskell.	1		4	6
2	Рекурсивные функции. Абстракции списков. Определение функций с помощью шаблонов. Определение функций с помощью сопоставления с образцом. Образцы на списках. Основные принципы использования рекурсии. Хвостовая рекурсия. Использование накапливающих параметров. Рекурсивные функции на списках. Рекурсия по нескольким аргументам. Взаимная рекурсия.	1		2	4
3	Определение типов. Понятие типа данных. Множества значений и допустимых операций. Полиморфизм и перегрузка типов функций. Основные классы типов: Eq, Ord, Show, Read, Num, Integral, Fractional, Enum. Определение синонимов типов с помощью декларации type. Определение новых типов данных с помощью декларации data. Наследование методов класса типов с помощью декларации deriving. Определение новых методов класса типов с помощью декларации instance. Определение параметризованных (полиморфных) типов. Определение рекурсивных типов.	1		2	3

4	<p>Функции высшего порядка. Понятие функции высшего порядка. Примеры типичных шаблонов программирования. Функция <code>map</code>, <code>filter</code>. Примеры использования. Карринг, частичная параметризация и операторные секции. Функции <code>curry</code> и <code>uncurry</code>. Пример использования. Функция <code>\$</code>. Примеры использования. Функции <code>takeWhile</code> и <code>dropWhile</code>. Функция <code>zipWith</code>. Пример: вычисление длины ломаной и периметра многоугольника. Композиция функций. Примеры использования. Группа функций свертки <code>fold*</code>. Универсальность функции <code>foldr</code>. Общий подход построения программ на функциональных языках «снизу вверх».</p>	2		2	4
5	<p>Средства ввода-вывода. Принципиальные трудности реализации в функциональных программах ввода-вывода, параллельных процессов, хранения состояния, взаимодействия с программами на других языках. Отличие функций и действий. Тип <code>IO</code>. Внутренняя организация команды <code>do</code>. Операции <code>>>=</code> и <code>>></code>. Организация императивной последовательности выполнения команд с помощью <code>do</code>. Действия (<code>Actions</code>) для консольного ввода-вывода и генерации случайных чисел. Пример программы для угадывания чисел. Организация ветвлений и циклов. Принципиальное отличие императивных циклов и циклов в языке Haskell. Действия для файлового ввода-вывода. Действия для сетевого ввода-вывода. Особенности ленивых вычислений при работе с файлами. Обработка ошибок ввода-вывода.</p>	2		4	6
6	<p>Монады. Подход с передачей состояния как переменной <code>World</code>. Понятие монады. Использование управляющих конструкций для структурирования вычислений. Использование монад для реализации хранения изменяемого состояния (<code>IORef</code>). Использование <code>unsafePerformIO</code> для ввода-вывода из функционального ядра. Синхронизация параллельных процессов (<code>forkIO</code>, <code>MVar</code>, организация каналов). Обработка исключительных ситуаций. Синхронные и асинхронные</p>	2		4	6

	исключения. Взаимодействие с программами на других языках. Импорт и экспорт функций. Маршаллинг. Управление памятью. Указатели и внешние объекты. Отличие императивных языков программирования от императивных средств в функциональных языках.				
7	Примеры использования языка Haskell. Распознавание и трансляция выражений. Понятие и определение типа функции-парсера. Три основных парсера: <code>return</code> , <code>failure</code> , <code>item</code> . Организация последовательного и выборочного применения парсеров. Организация повторного применения парсеров. Парсеры для идентификаторов, натуральных чисел и пробелов. Парсеры для лексем. Построение упрощенной грамматики арифметических выражений. Реализация парсера упрощенной грамматики арифметических выражений.	1		4	5
8	Примеры использования языка Haskell. Работа с арифметическими выражениями. Средства тестирования. Сравнение алгоритмической и низкоуровневой оптимизации. Понятие автоматизированного тестирования. Обзор возможностей средства автоматизированного тестирования QuickCheck. Создание свойств для автоматизированной проверки.	1		4	6
	ВСЕГО	11		26	40
Логическое программирование					
(наименование тематического раздела)					
9	Введение в логическое программирование. Основы языка Пролог. Понятие и основные особенности логического программирования. История возникновения и развития Пролога. Проект ЭВМ пятого поколения. Области использования Пролога. Перспектива применения в качестве языка запросов к базам данных. Основные понятия Пролога. Предложения, цели, факты и правила. Факты. Хранение данных в программах. Механизм поиска и достижения цели в программе на языке Пролог. Запросы. Понятие унификации и	2		2	5

	<p>работа механизма вывода пролог-машины. Внутренняя структура подцели в механизме логического вывода. Понятие и назначение CALL, EXIT, REDO, FAIL. Работа механизма логического вывода. Обработка правил в подцелях. Предикаты специального вида: I/O, fail. Правила. Передача управления внутри подцелей. Особенности использования арифметических операторов.</p>				
10	<p>Обзор основных возможностей языка Пролог. Управление исполнением программ. Списки. Предикаты работы с внутренней базой данных. Использование динамических баз данных для повышения эффективности выполнения программ. Предикаты специального вида: gereat, !. Предикат отсечения и особенности логического вывода. Особенности использования арифметических операторов. Представление предикатов в виде операторов. Рекурсия. Практические аспекты применения рекурсии и оптимизация выполнения. Рекурсия. Влияние порядка правил и предикатов на эффективность работы программы. Использование динамических баз данных для повышения эффективности выполнения программ. Списки. Особенности работы со списками. Неоднозначное использование предикатов и особенности механизма вывода на примере проверки вхождения элемента в список и конкатенации.</p>	1		2	3
11	<p>Представление и обработка деревьев в языке Пролог. Обработка грамматики, разностные списки. Представление бинарных деревьев и двоичных справочников в языке Пролог. Примеры базовых процедур работы с бинарными деревьями. Разбор структуры естественного языка средствами языка Пролог. Запись правил грамматики. Разностные списки.</p>	1		2	4
12	<p>Применение функционального и логического программирования. Искусственный интеллект. Понятие и основные области исследования искусственного интеллекта. Основные вехи исторического развития искусственного интеллекта. Понятие</p>	2		2	5

	теста Тьюринга. Логические игры. Игры для двух лиц с полной информацией. Методы обхода пространства состояний. Минимаксный принцип в логических играх. Алгоритм альфа-бета отсечения для эффективной реализации минимаксного принципа. Методы улучшения эффективности альфа-бета алгоритма: эвристическое отсечение, последовательное углубление просмотра дерева игры, просмотр до спокойных позиций, экспертная система особых ситуаций (эндшпилей). Понятие, виды, структура и функциональность экспертных систем. Принципы реализации экспертных систем средствами языка Пролог. Реализация ответов на вопросы «как» и «почему».				
	ВСЕГО	6		8	17
	Итого	17		34	57

4.3. Содержание лабораторных занятий

№ п/п	Наименование раздела	Наименование лабораторной работы	Кол-во часов	Кол-во часов СРС
1	Рекурсивные функции.	Основы языка Haskell. Рекурсивные функции в языке Haskell	4	6
2	Определение типов. Понятие типа данных.	Пользовательские типы данных в языке Haskell	4	6
3	Определение типов. Понятие типа данных.	Рекурсивные типы данных в языке Haskell	2	4
4	Функции высшего порядка.	Функции высшего порядка в языке Haskell	8	13
5	Средства ввода-вывода.	Операции ввода-вывода в языке Haskell	8	12
6	Введение в логическое программирование. Основы языка Пролог.	Основы языка Prolog. Рекурсивные правила в языке Prolog.	3	6
7	Обзор основных возможностей языка Пролог.	Работа с линейными списками в языке Prolog.	3	6
8	Представление и обработка деревьев в языке Пролог	Работа с нелинейными структурами данных в языке Prolog.	2	4
	ИТОГО		34	57

5. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ДЛЯ ПРОВЕДЕНИЯ ТЕКУЩЕГО КОНТРОЛЯ УСПЕВАЕМОСТИ, ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ПО ИТОГАМ ОСВОЕНИЯ ДИСЦИПЛИНЫ

5.1. Перечень контрольных вопросов (типовых заданий)

№ п/п	Название раздела	Наименование вопросов
1	Введение в функциональное программирование	Сильные стороны функционального и логического программирования.
2	Введение в функциональное программирование	Основные области применения и задачи функционального и логического программирования.
3	Рекурсивные функции.	Функции. Способы задания функций в языке Haskell.
4	Рекурсивные функции.	Характеристика и использование стандартных типов языка Haskell.
5	Рекурсивные функции.	Понятие полиморфизма и перегрузки функций.
6	Рекурсивные функции.	Понятие карринга, частичной параметризации и операторных секций.
7	Функции высшего порядка.	Функции высшего порядка. Композиция функций.
8	Функции высшего порядка.	Функции высшего порядка. Функционал filter.
9	Функции высшего порядка.	Функции высшего порядка. Функционал map.
10	Функции высшего порядка.	Функции высшего порядка. Функционал fold*.
11	Функции высшего порядка.	Синтаксис и применение лямбда-функций.
12	Функции высшего порядка.	Работа с кортежами и списками в языке Haskell. Абстракции списков с примерами применения.
14	Определение типов. Понятие типа данных.	Определение пользовательских типов. Определение типа, параметризованного другим типом.
15	Определение типов. Понятие типа данных.	Определение пользовательских типов. Определение рекурсивных типов.
16	Определение типов. Понятие типа данных.	Определение пользовательских типов. Классы типов, инкапсуляция и наследование.
17	Определение типов. Понятие типа данных.	Модульная реализация программы на языке Haskell.
18	Средства ввода-вывода.	Особенности ввода-вывода в языке Haskell. Функции экранного и файлового ввода-вывода.
19	Средства ввода-вывода.	Двумерный синтаксис и локальные определения языка Haskell. Примеры использования.
20	Введение в логическое программирование.	Списки. Рекурсивное определение списка. Операции над списками.

	Основы языка Пролог.	
21	Основы языка Пролог.	Бинарные деревья, двоичные справочники и операции над ними.
22	Обзор основных возможностей языка Пролог.	Искусственный интеллект. Множественный перебор. Альфа-бета отсечение.
23	Обзор основных возможностей языка Пролог.	Переменные свободные и связанные. Анонимная переменная. Отсечение. "Зеленые" и "красные" отсечения.
24	Обзор основных возможностей языка Пролог.	Предложения: факты и правила. Цели внутренние и внешние. Отношения (предикаты).
25	Обзор основных возможностей языка Пролог.	Рекурсия. Достоинства и недостатки рекурсии. Хвостовая рекурсия.
26	Обзор основных возможностей языка Пролог.	Организация циклов на основе рекурсии. Вычисление факториала.
27	Обзор основных возможностей языка Пролог.	Метод поиска в глубину. Откат после неудачи. Отсечение и откат. Метод поиска, определяемый пользователем.
28	Обзор основных возможностей языка Пролог.	Рекурсия. Применение накапливающих параметров.
29	Обзор основных возможностей языка Пролог.	Работа с динамическими базами данных: добавление фактов в базу, удаление фактов из базы.
30	Обзор основных возможностей языка Пролог.	Искусственный интеллект. Создание логики игровых программ.
31	Обзор основных возможностей языка Пролог.	Семантические модели Пролога: декларативная и процедурная.
32	Представление и обработка деревьев в языке Пролог	Организация циклов на основе рекурсии. Вычисление факториала.
33	Представление и обработка деревьев в языке Пролог	Работа с динамическими базами данных: добавление фактов в базу, удаление фактов из базы.
34	Представление и обработка деревьев в языке Пролог	Структура программы на PDC-Прологе. Стандартные, списковые, составные, альтернативные домены.
35	Представление и обработка деревьев в языке Пролог	Списки. Рекурсивное определение списка. Операции над списками.
36	Обзор основных возможностей языка Пролог.	Искусственный интеллект. Тест Тьюринга. Экспертные системы.

**5.2. Перечень тем курсовых проектов, курсовых работ,
их краткое содержание и объем.**

Планом не предусмотрено

**5.3. Перечень индивидуальных домашних заданий,
расчетно-графических заданий.**

Планом не предусмотрено

5.4. Перечень контрольных работ.

Планом не предусмотрено

6. ОСНОВНАЯ И ДОПОЛНИТЕЛЬНАЯ ЛИТЕРАТУРА

6.1. Перечень основной литературы

1. Городняя Л.В. Основы функционального программирования. — М.: ИНТУИТ, 2004. — 280 с.
2. Братко И. Программирование на языке Пролог для искусственного интеллекта.: Пер. с англ. — М.: Мир, 1990. — 560 с.: ил.

6.2. Перечень дополнительной литературы

1. Стерлинг Л., Шапиро Э. Искусство программирования на языке Пролог. — М.: Мир, 1990. — 235 с.
2. Непейвода Н.Н. Стили и методы программирования. — М.: ИНТУИТ, 2005. — 320 с.
3. Хювёнен Э., Сеппянен Й. Мир Лиспа. — М.: Мир, 1990. — 447 с.
4. Адаменко А.Н., Кучуков А.М. Логическое программирование и Visual Prolog. — СПб.: БХВ-Петербург, 2003. — 992 с.
5. Хендерсон П. Функциональное программирование: применение и реализация. М.: Мир, 1983. — 349 с.
6. Братко И. Алгоритмы искусственного интеллекта на языке PROLOG. 3-е изд. — М.: Издательский дом "Вильямс", 2004. — 640 с.

6.3. Перечень интернет ресурсов

1. haskell.org
2. learnyourhaskell.com
3. learnprolognow.com

**7. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ И ИНФОРМАЦИОННОЕ
ОБЕСПЕЧЕНИЕ**

При проведении лабораторных занятий могут компьютерные классы БГТУ им. В.Г.Шухова ауд. 430, ауд. 426.

Для освоения дисциплины могут быть использованы программные средства:

1. Операционная система Microsoft Windows.
2. Интегрированная среда разработки Microsoft Visual Studio.
3. Офисные приложения Microsoft Office.

6. ОСНОВНАЯ И ДОПОЛНИТЕЛЬНАЯ ЛИТЕРАТУРА

6.1. Перечень основной литературы

1. Городняя Л.В. Основы функционального программирования [Электронный ресурс]. — 2-е изд. — М.: Интернет-Университет Информационных Технологий (ИНТУИТ), 2016. — 246 с. — Режим доступа: <http://www.iprbookshop.ru/73703.html>
2. Шрайнер П.А. Основы программирования на языке Пролог [Электронный ресурс]. — М.: Интернет-Университет Информационных Технологий (ИНТУИТ), 2016. — 213 с. — Режим доступа: <http://www.iprbookshop.ru/52194.html>
3. Городняя Л.В. Введение в программирование на Лиспе [Электронный ресурс] / Л.В. Городняя, Н.А. Березин. — 2-е изд. — М.: Интернет-Университет Информационных Технологий (ИНТУИТ), 2016. — 134 с. — Режим доступа: <http://www.iprbookshop.ru/73668.html>
4. Галкина М.Ю. Функциональное и логическое программирование [Электронный ресурс]: практикум. — Новосибирск: Сибирский государственный университет телекоммуникаций и информатики, 2008. — 107 с. — Режим доступа: <http://www.iprbookshop.ru/55464.html>

6.2. Перечень дополнительной литературы

1. Стерлинг Л., Шапиро Э. Искусство программирования на языке Пролог. — М.: Мир, 1990. — 235 с.
2. Непейвода Н.Н. Стили и методы программирования. — М.: ИНТУИТ, 2005. — 320 с.
3. Непейвода Н.Н. Стили и методы программирования [Электронный ресурс]. — 2-е изд. — М.: Интернет-Университет Информационных Технологий (ИНТУИТ), 2016. — 295 с. — Режим доступа: <http://www.iprbookshop.ru/73724.html>
4. Хювёнен Э., Сеппянен Й. Мир Лиспа. — М.: Мир, 1990. — 447 с.
5. Адаменко А.Н., Кучуков А.М. Логическое программирование и Visual Prolog. — СПб.: БХВ-Петербург, 2003. — 992 с.
6. Братко И. Алгоритмы искусственного интеллекта на языке PROLOG. 3-е изд. — М.: Издательский дом "Вильямс", 2004. — 640 с.
7. Рублев В.С. Языки логического программирования [Электронный ресурс]. — 2-е изд. — М.: Интернет-Университет Информационных Технологий (ИНТУИТ), 2016. — 125 с. — Режим доступа: <http://www.iprbookshop.ru/73741.html>
8. Ефимова Е.А. Основы программирования на языке Visual Prolog [Электронный ресурс]. — М.: Интернет-Университет Информационных Технологий (ИНТУИТ), 2016. — 265 с. — Режим доступа: <http://www.iprbookshop.ru/39556.html>

8. УТВЕРЖДЕНИЕ РАБОЧЕЙ ПРОГРАММЫ

Утверждение рабочей программы с изменениями, дополнениями
Рабочая программа с изменениями, дополнениями утверждена на 20~~16~~/20~~17~~
учебный год.

Протокол № 10 заседания кафедры от « 9 » 06 2016г.

Заведующий кафедрой _____ В.М. Поляков
подпись, ФИО

Директор института _____ А. В. Белоусов
подпись, ФИО

6. УТВЕРЖДЕНИЕ РАБОЧЕЙ ПРОГРАММЫ

Утверждение рабочей программы без изменений

Рабочая программа без изменений утверждена на 2017/2018 учебный год.

Протокол № 11 заседания кафедры от «12» 05 2017г.

Заведующий кафедрой .

подпись

Поляков В.М.

Директор института ЭИТУС

Белоусов А.В.

8. УТВЕРЖДЕНИЕ РАБОЧЕЙ ПРОГРАММЫ

Утверждение рабочей программы без изменений
Рабочая программа без изменений утверждена на 2018/2019 учебный год.

Протокол № 10 заседания кафедры от «21» 05 2018 г.

Заведующий кафедрой _____ В.М. Поляков
подпись, ФИО

Директор института _____ А. В. Белоусов

8. УТВЕРЖДЕНИЕ РАБОЧЕЙ ПРОГРАММЫ

Утверждение рабочей программы без изменений
Рабочая программа без изменений утверждена на 2019/2020 учебный
год.

Протокол № 10 заседания кафедры от «18» мая 2019 г.

Заведующий кафедрой _____ В.М. Поляков
подпись, ФИО

Директор института _____ А.В. Белоусов

7. УТВЕРЖДЕНИЕ РАБОЧЕЙ ПРОГРАММЫ³

Рабочая программа утверждена на 20 20 /20 21 учебный год
без изменений / с изменениями, дополнениями⁴

Протокол № 8 заседания кафедры от « 21 » 04 20 20 г.

Заведующий кафедрой _____ (Поляков В.М.)
подпись, ФИО

Директор института _____ (Белоусов А.В.)
подпись, ФИО

³ Заполняется каждый учебный год на отдельных листах

⁴ Нужно подчеркнуть

7. УТВЕРЖДЕНИЕ РАБОЧЕЙ ПРОГРАММЫ¹

Рабочая программа утверждена на 2021/2022 учебный год без изменений²

Протокол № 8 заседания кафедры от « 15 » мая 2021 г.

Заведующий кафедрой _____

подпись, ФИО

Полехов В.М.

Директор института _____

подпись, ФИО

Белюсов А.В.

¹ Заполняется каждый учебный год на отдельных листах

² Нужно подчеркнуть