



Автономная некоммерческая организация  
дополнительного профессионального образования  
(АНО ДПО «Инфосфера»)

Центр профессиональной подготовки  
ИНСТИТУТ ПРОГРАММНЫХ СИСТЕМ

**Рабочая программа дисциплины**  
**«Функциональное и логическое программирование»**

Разработал:  
преподаватель ИПС  
АНО ДПО «Инфосфера»  
С.В. Шамбир

Йошкар-Ола, 2017

## Пояснительная записка

Курс «Функциональное и логическое программирование» предназначен для углубления знаний в области информатики и программирования и ставит **целью** освоение ключевых принципов и абстракций функциональной и логической парадигм программирования.

Содержание курса включает в себя

Основные понятия и принципы функционального программирования излагаются на основе языка программирования Лисп. Особое внимание уделяется технике построения функциональных программ, включая рекурсию и функционалы.

Основные понятия логического программирования изучаются на основе языка программирования Пролог. Затрагиваются вопросы взаимосвязи Пролога и логики предикатов первого порядка.

Основная задача курса – освоение базовых принципов и методов функционального и логического программирования. В результате изучения дисциплины студенты должны:

- Понимать ключевые абстракции и особенности функциональной и логической парадигм программирования, их отличия от традиционной императивной парадигмы;
- Владеть основными средствами и понятиями функционального программирования на языке Лисп и логического программирования на языке Пролог;
- Уметь составлять рекурсивные функциональные программы на языке Лисп и логические программы на языке Пролог;
- Иметь представление о средствах разных функциональных языков программирования;
- Знать области применимости функциональных и логических языков.

### • Планируемые результаты обучения

- Осуществляется предварительная самостоятельная или под руководством разработка алгоритмов с использованием графических средств (блок-схемы, UML-диаграммы и др.). Не требуется взаимодействие с другими программистами, системным аналитиком и архитектором программного обеспечения. Осуществляется решение типовых задач. Полученные результаты представляются руководителю разработки программного обеспечения.
- Осуществляется предварительная самостоятельная или под руководством разработка алгоритмов с использованием графических средств (блок-схемы, UML-диаграммы и др.). Не требуется взаимодействие с другими программистами, системным аналитиком и архитектором программного обеспечения. Осуществляется решение типовых задач. Полученные результаты представляются руководителю разработки программного обеспечения.
- Выполняются самостоятельная разработка процедур сборки модулей и компонент программного обеспечения и верификация выпусков программного продукта. Производится разработка процедур развертывания и обновления программного обеспечения, процедур миграции и преобразования (конвертации) данных и программных интерфейсов с

использованием выбранных программных средств, технологий создания открытых систем. Осуществляется решение различных типов задач проектирования программных комплексов различной сложности, выбор способов реализации взаимодействия программных компонент/модулей. Требуется взаимодействие с программистами-разработчиками модулей, архитектором программного обеспечения. Полученные результаты представляются руководителю разработки программного обеспечения.

- В процессе интеграции программных модулей и компонент и верификации выпусков программного продукта осуществляется сборка модулей и компонент программного обеспечения, производится интеграция с внешней средой. Обеспечивается согласованное функционирование и требуемый уровень качества.
- Проведение интеграции программных модулей и компонент и верификация выпусков программного продукта предполагают определение задач программной интеграции, распределение задач между подчиненными, обеспечение взаимодействия подчиненных сотрудников.
- Программист несет ответственность за результат выполнения работ на уровне группы программистов. В процессе интеграции требуется взаимодействие с архитектором программного обеспечения. Полученные результаты представляются руководителю разработки программного обеспечения.

## Учебно-тематический план

№	Название темы	Лекции	Практика	Всего
1	Введение: парадигмы программирования	2	2	4
2	Основы функционального программирования. Язык Лисп	12	12	24
3	Средства функциональных языков: Функционалы, ленивые вычисления	6	6	12
4	Основы логического программирования. Язык Пролог	10	10	20
5	Приложения языков функционального и логического программирования	4	4	8
		<b>34</b>	<b>38</b>	<b>72</b>

# Содержание курса

## 1. Введение: парадигмы программирования

Основные парадигмы (модели) вычислений: императивная, функциональная, объектная и логическая. Ключевые абстракции и особенности парадигм программирования. Роль переменной в различных парадигмах. Парадигмы и языки программирования. Примеры программ, составленных в разных парадигмах.

## 2. Основы функционального программирования. Язык Лисп

Лямбда-исчисление как теоретическая основа функционального программирования. Функциональная парадигма и функциональные языки. Свойство функциональности, отсутствие побочных эффектов. Достоинства и недостатки функциональной парадигмы.

Язык Лисп. Основные структуры данных: атомы, списки, выражения. Лисп-интерпретатор и лисп-программа. Обычные и особые функции. Встроенные функции базового Лиспа. Условные выражения. Определение новых функций. Представление списков в памяти компьютера. Примеры программ на языке Lisp.

Рекурсивное программирование, его достоинства. Рекуррентные соотношения и индукция как основы рекурсии. Рекурсивные структуры. Накапливающий параметр и его использование. Простая и сложная (косвенная) рекурсия. Параллельная рекурсия. Примеры решения задач с использованием рекурсии.

Дополнительные возможности языка Лисп. Списки свойств атомов и их применение. Сравнение рекурсивного и циклического программирования. Функции ввода и вывода.

## 3. Средства функциональных языков: функционалы, ленивые вычисления

Концепции функциональных языков программирования. Обзор и сравнение функциональных языков: Рефал, Лисп, Scheme, Haskell, Erlang. Функциональное программирование в нефункциональных языках.

Функции высших порядков. Функциональные аргументы. Функционалы с функциональным значением. Отображающие функционалы. Основные встроенные функционалы языка Лисп. Замыкание функционального аргумента. Примеры решения задач с использованием функционалов на языке Лисп.

Понятие ленивого вычисления. Ленивые вычисления в языках Лисп и Haskell. Примеры программ, основанных на ленивых вычислениях.

## 4. Основы логического программирования. Язык Пролог

Основные концепции логического программирования. Предикаты и термы. Унификация термов. Вычисление как логическое доказательство.

Язык Пролог. Основные структуры данных: атомы, термы, списки. Пролог-предикаты и пролог-программа. Виды пролог-предложений: факты, правила, вопросы.

Пролог-интерпретатор, правила выполнения пролог-программ. Сопоставление (унификация) и бектрекинг как базовые механизмы Пролога. Отсечения. Встроенные предикаты Пролога. Примеры пролог-программ.

Язык исчисления предикатов первого порядка и Пролог. Декларативная и процедурная семантика пролог-программ. Реализация логического отрицания в Прологе и гипотеза о замкнутости мира.

## **5. Приложения языков функционального и логического программирования**

Функциональные языки и символьные преобразования. Решение на языке Лисп задачи символьного преобразования полиномов с целыми коэффициентами.

Язык Пролог и задачи искусственного интеллекта. Решение на языке Пролог задачи синтаксического разбора предложений из ограниченного подмножества естественного языка.

## **Пособия по изучению курса**

1. Братко И. Алгоритмы искусственного интеллекта на языке Prolog. пер. с англ. / Иван Братко. - М. [и др.], 2004. - 637 с. : ил.
2. Шрайнер П. А. Основы программирования на языке Пролог : курс лекций : учебное пособие для вузов по специальностям информационных технологий / П. А. Шрайнер. - М., 2005. - 172, [1] с. : ил.
3. Хювенен Э. Мир Лиспа. В 2 т.. Т. 1. введение в язык Лисп и функциональное программирование : [учебное пособие] / Э. Хювёнен, Й. Сеппянен ; пер. с фин. А. А. Рейтсакаса ; под ред. В. Л. Стефанюка. - М., 1990. - 447 с. : ил.
4. Городняя Л. В. Основы функционального программирования. Курс лекций : учебное пособие / Л. В. Городняя ; Интернет ун-т информ. технологий. - М., 2004. - 272 с. - Рекомендовано УМО.
5. Авдеенко Т. В. Введение в искусственный интеллект и логическое программирование : учебное пособие / Т. В. Авдеенко ; Новосиб. гос. техн. ун-т. - Новосибирск, 2007. - 62, [2] с. : ил.

## **Контрольные задания**

Оценка текущей и промежуточной аттестации по курсу осуществляется по результатам выполнения лабораторных и контрольных работ.

Каждая контрольная работа контролирует освоение слушателями определенного раздела изучаемого курса. Итоговый контроль по дисциплине осуществляется по результатам выполнения лабораторных, контрольных работ и сдачи экзамена.