

Аннотация рабочей программы дисциплины

**«Функциональное программирование»**

Направление подготовки

*02.03.02 Фундаментальная информатика и информационные технологии*

Направленность (профиль) образовательной программы

*Сетевые технологии*

## **Цели и задачи освоения дисциплины**

Целью дисциплины является изучение и практическое освоение средств функционального и логического программирования для решения научных и прикладных задач. В качестве инструментальных средств изучаются языки Scheme, F#, а также функциональный подход в традиционных языках программирования C++ и C#. Рассматриваются теоретические и прикладные аспекты использования данных языков программирования для решения задач искусственного интеллекта.

Материал курса является инструментальной основой при изучении студентами дисциплин учебного плана, связанных с созданием программных средств искусственного интеллекта.

Задача курса – дать студентам теоретические знания об методах и алгоритмах функционального и логического программирования, технологий интеллектуальных систем.

### **Задачи дисциплины:**

1. Изучение математических основ, основных концепций и приемов функционального программирования;
2. Изучение базовых языков функционального и логического программирования;
3. Получение практических навыков разработки и реализации алгоритмов с использованием языка функционального программирования.

## **Место дисциплины в структуре ОПОП ВО**

Дисциплина «Функциональное программирование» является дисциплиной по выбору вариативной части учебного плана по направлению 02.03.02 Фундаментальная информатика и информационные технологии, изучается в 6 семестре (3 курс), форма промежуточного контроля – зачет с оценкой.

Перечень дисциплин с указанием разделов (тем), усвоение которых студентами необходимо для изучения дисциплины:

- Объектно-ориентированное программирование (все);
- Структуры и алгоритмы обработки данных (все);

**Планируемые результаты обучения по дисциплине (модулю),  
соотнесенные с планируемыми результатами освоения  
образовательной программы (компетенциями выпускников)**

После изучения данной дисциплины студенты приобретают знания, умения и владения (навыки), соответствующие результатам основной профессиональной образовательной программы.

<b>Формируемые компетенции</b>	<b>Планируемые результаты обучения по дисциплине (модулю), характеризующие этапы формирования компетенций</b>
ОПК-2. Способность применять в профессиональной деятельности современные языки программирования и языки баз данных, методологии системной инженерии, системы автоматизации проектирования, электронные библиотеки и коллекции, сетевые технологии, библиотеки и пакеты программ, современные профессиональные стандарты информационных технологий	32 (ОПК-2) Иметь представление о роли и месте знаний современных языков программирования и языков баз данных, библиотек и пакетов программ при освоении смежных дисциплин и в сфере профессиональной деятельности *) 36 (ОПК-2) Знать методы построения и оценки эффективности алгоритмов и их программной реализации
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	У3 (ОПК-3) Умение применять в профессиональной деятельности современные языки программирования для разработки алгоритмических и программных решений, в том числе параллельных вычислений на базе технологий MPI и OpenMP *) В3 (ОПК-3) Владеть современными средами и средствами разработки программного обеспечения *) В4 (ОПК-3) Владеть методами проектирования и конструирования программного обеспечения
ПК-3. Способность использовать современные инструментальные и вычислительные средства	32 (ПК-3) Знать технологии создания информационно-вычислительных систем У2 (ПК-3) Уметь выбирать алгоритмы и осуществлять их программную реализацию для решения типовых задач предметной области У3 (ПК-3) Уметь планировать и выполнять верификацию и валидацию программного решения типовых задач предметной области В2 (ПК-3) Владеть современными средами и средствами разработки программного обеспечения В3 (ПК-3) Владеть методами проектирования и конструирования программного обеспечения

Формируемые компетенции	Планируемые результаты обучения по дисциплине (модулю), характеризующие этапы формирования компетенций
	В6 (ПК-3) Владеть средствами и методами тестирования программного обеспечения *)

\*) Результат обучения сформулирован на основании требований профессионального стандарта: 06.015 СПЕЦИАЛИСТ ПО ИНФОРМАЦИОННЫМ СИСТЕМАМ, утв. Приказом Минтруда России от 18.11.2014, № 896н (ред. от 12.12.2016, № 727н) , регистрационный номер - 153.

**Объем дисциплины (модуля) в зачетных единицах с указанием количества академических или астрономических часов, выделенных на контактную работу обучающихся с преподавателем (по видам учебных занятий) и на самостоятельную работу обучающихся**

Объем дисциплины (модуля) составляет 3 зачетных единиц, всего 108 часов, из которых:

- **50 час составляет контактная работа обучающегося с преподавателем<sup>1</sup>:**
  - 18 часов – лекции,
  - 32 часов – практические занятия
- **58 часа составляет самостоятельная работа обучающегося**
- **мероприятия промежуточного контроля (зачет с оценкой в 6 семестре)**

---

<sup>1</sup> Перечень видов учебных занятий уточняется в соответствии с учебным планом.

**Содержание дисциплины (модуля), структурированное по темам (разделам) с указанием отведенного на них количества академических или астрономических часов и виды учебных занятий**

Наименование и краткое содержание разделов и тем дисциплины (модуля) Форма промежуточной аттестации по дисциплине (модулю)	Всего (часы)	Контактная работа (работа во взаимодействии)					Всего	Самостоятельная работа обучающегося,		
		Лекционные	Семинары	Практические	Лабораторные	Учебные занятия, направленные на проведение текущего контроля		Выполнение	Подготовка	Всего
Введение в функциональное программирование. Классификация языков программирования. Общее представление о ФП и его применении. Математические основы функционального программирования – лямбда-исчисление Черча. История создания и развития Лиспа.	8	2		2		ПР-1	4	4		4
Базовые средства символьной обработки данных. Структуры данных: атомы и списки. Списочные ячейки. Понятие точечной пары и S-выражения. Соответствие между списочной и точечной нотациями. Базовые функции работы со списками.	12	2		4		ПР-1	6	6		6
Основные понятия: программа, функция, выражение. Определение функций. Композиция функций. Рекурсивные функции: определение и исполнение. Введение в теорию рекурсивных функций. Простая рекурсия. Рекурсивные функции работы со списками.	14	2		4		ПР-1	6	8		8
Внутреннее представление списков. Память и ссылочные ячейки. Указатели car и cdr. Логическое и физическое равенство списков	12	2		2		ПР-1	4	8		8
Математические основы функционального программирования. Лямбда-исчисление как формальная система. Синтаксис и семантика лямбда-исчисления. Вычисление лямбда-выражений. Подстановка, конверсия, равенство. Порядок редукций и нормальные формы. Лямбда-редукция и проблема конфликта имен. Рекурсивные выражения. Комбинаторы. Чистое лямбда -исчисление. Лямбда-выражения в Scheme.	12	2		4		ПР-1	6	6		6

2 Перечень видов учебных занятий уточняется в соответствии с учебным планом.

Наименование и краткое содержание разделов и тем дисциплины (модуля) Форма промежуточной аттестации по дисциплине (модулю)	Всего (часы)	Контактная работа (работа во взаимодействии)					Самостоятельная работа обучающегося,			
		Лекционные	Семинары	Практические	Лабораторные	Учебные занятия, направленные на проведение текущего контроля	Всего	Выполнение	Подготовка	Всего
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.	12	2		4		ПР-1	6	6		6
Функционалы. Функциональный аргумент, функциональное значение функции. Способы композиции функций. Функции более высокого порядка. Применяющие функционалы. Отображающие функционалы. Композиция функционалов. Функциональное замыкание - функция и контекст ее определения. Абстрактный подход - обобщение функций, имеющих одинаковый вид. Параметризованное определение функций. Автофункции. Автоаппликация и авторепликация. Порядок и тип функции. Проблемы абстрактного подхода.	12	2		4		ПР-1	6	6		6
Функциональное программирование в .Net. Язык программирования F# и его особенности. «Ленивые» вычисления в .Net. Сравнение со Scheme.	12	2		4		ПР-1	6	6		6
Функциональное программирование в языках C# и C++. Функции высших порядков (делегаты), лямбда-выражения, «ленивые» вычисления.	14	2		4		ПР-1	6	8		8
Итого	108	18		32			50	58		58

\*Опрос (ПР-1), Практические работы (ПР-2), Реферат (ПР-3), \*Экзамен (УО-4). Текущий контроль проводится за счет времени, отведенного на аудиторные занятия