

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

ТОМСКИЙ ГОСУДАРСТВЕННЫЙ ПЕДАГОГИЧЕСКИЙ УНИВЕРСИТЕТ
(ТГПУ)

«УТВЕРЖДАЮ»

Декан физико-математического факультета



к.п.н, доцент Е.Г. Пьяных

« 26 » ноя 2016 года

РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ (МОДУЛЯ)
ФУНКЦИОНАЛЬНОЕ ПРОГРАММИРОВАНИЕ

Направление подготовки 01.04.02 Прикладная математика и информатика

Направленность (профиль) Прикладная информатика

Форма обучения очная, заочная

1. Место учебной дисциплины (модуля) в структуре образовательной программы

Дисциплина входит в вариативную часть блока 1 (обязательная дисциплина).

Для освоения данной дисциплины необходимо изучить дисциплину «Современные проблемы прикладной математики и информатики».

2. Перечень планируемых результатов обучения по дисциплине (модулю), соотнесенных с планируемыми результатами освоения ОП

Компетенции, формируемые учебной дисциплиной «Функциональное программирование»:

способность разрабатывать и применять математические методы, системное и прикладное программное обеспечение для решения задач научной и проектно-технологической деятельности (ПК-3);

способность к преподаванию математических дисциплин и информатики в общеобразовательных организациях, профессиональных образовательных организациях и образовательных организациях высшего образования (ПК-9).

В результате освоения дисциплины обучающийся должен:

- ✓ иметь целостное представление об основных типах языков программирования
- ✓ знать основные типы данных и функций в языке программирования функционального типа.

Обучающийся должен владеть:

- ✓ навыками написания учебных программ на языке функционального программирования.

Должен уметь ориентироваться в языках программирования,

- ✓ пользоваться операционной системой Линукс, интерпретаторами языка.

3. Содержание учебной дисциплины (модуля)

1. Функциональная парадигма в программировании.

Стили программирования. Величины. Функции. Композиции функций. Виды вычислений. Ленивые и энергичные вычисления. Строгие функции. Язык программирования.

2. Знакомство с языком функционального программирования.

Начало работы с интерпретатором. Команды интерпретатора. Сессии и скрипты. Символьный стиль. Базовые типы языка. Функции. Числа. Логические величины. Символы. Списки. Упорядоченные множества. Задание функций. Комбинация функций. Частные определения. Определения с альтернативой. Охраняемые выражения. Сопоставление с образцом. Определение рекурсией или индукцией. Двумерный синтаксис.

3. Функции в языке.

Полиморфизм и перегрузка функций. Операторы. Префиксная и инфиксная запись. Приоритет операций. Карринг. Частная параметризация. Скобки в функциональной записи. Операторные секции.

4. Функции высших порядков.

Функции на списках. Итерации. Композиция функций. Лямбда функции.

5. Числовые функции.

Работа с целыми числами. Получение списка простых чисел. Определение дней недели. Стратегия разработки программы. Численные выражения. Численное дифференцирование. Вычисление квадратного корня. Нули функции. Обратные функции.

6. Структуры данных.

Списки. Определение списка. Функции на списках. Сортировка списка. Функции высшего порядка на списках. Абстракция списков. Бесконечные списки. Создание функции на бесконечных списках. Примеры применения бесконечных списков. Кorteжи. Использование кортежей. Кorteжи и списки. Кorteжи и карринг. Синоним. Рациональные числа.

7. Объектно-ориентированное программирование.

Инкапсуляция и наследование. Классы и перегрузки. Встроенные классы. Класс Ord. Классы Show и Read. Класс Enum. Классы чисел. Алгебраические типы данных. Определение новых типов. Перечисляемые типы. Полиморфные типы данных. Объявление рекурсивных типов данных. Метки полей. Строгий конструктор в объявлениях данных. Новые типы: объявление newtypes. Примеры структурирования данных. Тип Angle. Тип Day. Векторы и матрицы. Инкапсуляция данных в модулях. Необходимость сокрытия данных. Модули: экспорт и импорт. Контроль пространства имен. Модуль как способ реализации АТД. Библиотеки программ.

8. Ввод и вывод информации.

Интерактивный ввод и вывод. Хранение информации в файлах. Графический интерфейс (GUI).

4. Трудоемкость дисциплины (модуля) по видам учебных занятий, самостоятельной работы обучающихся и формам контроля

4.1. Очная форма обучения

Объем в зачетных единицах: 3.

4.1.1. Виды учебных занятий, самостоятельная работа обучающихся, формы контроля (в академических часах)

Вид учебной работы	Всего часов	Распределение по семестрам (в академических часах)	
		3	
Аудиторные занятия	45	45	
Лекции	15	15	
Лабораторные работы			
Практические занятия (семинары)	30	30	
Самостоятельная работа	63	63	
Курсовая работа			
Другие виды занятий			
Формы текущего контроля		тест	
Формы промежуточной аттестации		зачет	
Итого часов	108	108	

4.1.2. Содержание учебной дисциплины (модуля), структурированное по темам (разделам)

№п/п	Наименование темы (раздела) дисциплины	Всего часов	Аудиторные занятия (в часах)			Самостоятельная работа (в часах)
			Лекции	Практические занятия (семинары)	Лабораторные работы	
1	Функциональная парадигма в программировании	10	1	2		7
2	Знакомство с языком функционального программирования	14	2	4		8
3	Функции в языке	14	2	4		8
4	Функции высшего порядка	14	2	4		8
5	Числовые функции	14	2	4		8
6	Структуры данных	14	2	4		8
7	Объектно-ориентированное программирование	14	2	4		8
8	Ввод и вывод информации	14	2	4		8
	Итого:	108	15	30		63

4.2. Заочная форма обучения Объем в зачетных единицах: 3.

4.2.1. Виды учебных занятий, самостоятельная работа обучающихся, формы контроля (в академических часах)

Вид учебной работы	Всего часов	Распределение по семестрам (в академических часах)	
		2	
Аудиторные занятия	12	12	
Лекции	2	2	
Лабораторные работы	10	10	
Практические занятия (семинары)			
Самостоятельная работа	92	92	
Курсовая работа			
Другие виды занятий			
Формы текущего контроля		тест	
Формы промежуточной аттестации	4	4 (зачет)	
Итого часов	108	108	

4.2.2. Содержание учебной дисциплины (модуля), структурированное по темам (разделам)

№п/п	Наименование темы (раздела) дисциплины	Всего часов	Аудиторные занятия (в часах)			Самостоятельная работа (в часах)
			Лекции	Практические занятия	Лабораторные работы	

				(семинары)		
1	Функциональная парадигма в программировании	11	1			10
2	Знакомство с языком функционального программирования	12	1		1	10
3	Функции в языке	14			2	12
4	Функции высших порядков	13			1	12
5	Числовые функции	13			1	12
6	Структуры данных	14			2	12
7	Объектно-ориентированное программирование	14			2	12
8	Ввод и вывод информации	13			1	12
	Итого:	104	2		10	92

4.2.3. Лабораторный практикум

Наименование темы (раздела) дисциплины	Наименование лабораторной работы
2	Знакомство с языком функционального программирования
3	Функции в языке
4	Функции высших порядков
5	Числовые функции
6	Структуры данных
7	Объектно-ориентированное программирование
8	Ввод и вывод информации

5. Перечень учебно-методического обеспечения по дисциплине (модулю)

5.1. Основная учебная литература:

1. Сергиевский, Г. М. Функциональное и логическое программирование [Текст]: учебное пособие для вузов/Г. М. Сергиевский, Н. Г. Волченков.-М.:Академия,2010.-317 с.

5.2. Дополнительная литература:

1. Островский С. Л. Функциональное программирование на языке Лисп. – М.: Чистые пруды, 2009.

5.3. Перечень ресурсов информационно-коммуникационной сети Интернет, необходимых для освоения дисциплины (модуля)

В процессе изучения дисциплины, магистрант работает с многочисленными информационными источникам в сети Интернет.

В качестве примеров ссылок на интернет-источники можно привести:

<http://intuit.ru>

<http://lib.ru>

5.4. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине (модулю), включая перечень программного обеспечения и информационных справочных систем (при необходимости)

№п/п	Номер раздела (темы) учебной дисциплины	Наименование материалов обучения, пакетов программного обеспечения	Наименование технических и аудиовизуальных средств, используемых с целью демонстрации материалов
1	1	Компилятор для Lisp	проектор
2	1-8	пакет Hugs для языка Haskell	проектор

6. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине (модулю)

Лекционная аудитория, кабинет программирования и информационных систем

7. Методические указания для обучающихся по освоению дисциплины (модуля)

Самостоятельная работа обучающегося предполагает различные формы индивидуальной учебной деятельности: конспектирование научной литературы, сбор и анализ практического материала в СМИ, проектирование, выполнение тематических и творческих заданий и пр. Выбор форм и видов самостоятельной работы определяется индивидуально-личностным подходом к обучению совместно преподавателем и обучающимся.

8. Фонд оценочных средств для проведения текущего контроля успеваемости и промежуточной аттестации обучающихся по дисциплине (модулю)

Представлен в виде отдельного документа (приложение к рабочей программе учебной дисциплины (модуля)).

Рабочая программа учебной дисциплины (модуля) составлена в соответствии с учебным планом, федеральным государственным образовательным стандартом высшего образования по направлению подготовки **01.04.02 – Прикладная математика и информатика**

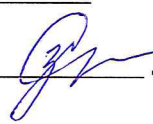
Рабочая программа учебной дисциплины составлена д.ф.-м.н., профессором кафедры информатики Л.В. Горчаковым

Рабочая программа учебной дисциплины утверждена на заседании кафедры информатики протокол № 10 от «26» мая 2016 г.

Зав. кафедрой информатики  к.т.н, А.Н. Стась

Рабочая программа учебной дисциплины одобрена методической комиссией физико-математического факультета

протокол № 9 от «26» мая 2016 г.

Председатель учебно-методической комиссии  д.п.н, профессор З.А. Скрипко