

ФЕДЕРАЛЬНОЕ АГЕНТСТВО ПО РЫБОЛОВСТВУ  
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ  
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
«КАМЧАТСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»  
(ФГБОУ ВО «КамчатГТУ»)

Факультет информационных технологий

Кафедра «Системы управления»

УТВЕРЖДАЮ

Декан ФИТ

 /И.А. Рычка/

«17» 03 2021 г.

**РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ**

«Разработка прикладных программ»

направление подготовки

27.03.04 «Управление в технических системах»  
(уровень бакалавриата)

направленность (профиль)

«Управление и информатика в технических системах»

Петропавловск-Камчатский  
2021

Рабочая программа разработана в соответствии с ФГОС ВО по направлению подготовки 27.03.04 «Управление в технических системах», профиль «Управление и информатика в технических системах», и учебного плана ФГБОУ ВО «КамчатГТУ».

Составитель рабочей программы:  
доцент кафедры СУ, к.т.н.



Луковенкова О.О.

Рабочая программа рассмотрена на заседании кафедры «Системы управления»  
Протокол № 8 от « 19 » 02 2021 года.

« 19 » 02 2021 г.



Заведующий кафедрой  
«Системы управления»  
А.А. Марченко

## 1. Цели и задачи учебной дисциплины

Учебная дисциплина «Разработка прикладных программ» относится к обязательной части Блока 1 «Дисциплины (Модули)» основной профессиональной образовательной программы по направлению подготовки 27.03.04 «Управление в технических системах», профиль «Управление и информатика в технических системах», предусмотренной учебным планом ФГОУ ВПО «КамчатГТУ».

### *Цель изучения дисциплины:*

формирование представлений о современных технологиях создания прикладных программ и приобретение практических навыков работы в интегрированных средах. Основное внимание уделяется изучению программирования на языке Python, особенностям разработки прикладных программ на языке Python.

### *Задачи изучения дисциплины:*

- изучение основ программирования на языке Python;
- изучение базовых конструкций и нетривиальных возможностей языка программирования Python;
- изучение возможностей различных пакетов языка Python.

## Компетенции обучающегося, формируемые в результате освоения дисциплины

Дисциплина «Разработка прикладных программ» направлена на освоение следующих компетенций основной профессиональной образовательной программы по направлению 27.03.04 «Управление в технических системах» федерального государственного образовательного стандарта высшего образования:

- способен разрабатывать и использовать алгоритмы и программы, современные информационные технологии (ОПК-6).

Планируемые результаты обучения при изучении дисциплины, соотнесенные с планируемыми результатами освоения образовательной программы, представлены в таблице.

Код компетенции	Планируемые результаты освоения образовательной программы	Код и наименование индикатора достижения компетенции	Планируемый результат обучения по дисциплине	Код показателя освоения
ОПК-6	способность выполнять эксперименты на действующих объектах по заданным методикам и обрабатывать результаты с применением современных информационных технологий и технических средств	ИД-1 <sub>опк-6</sub> Знает основы написания программного кода для работы систем автоматического управления	<i>Знать:</i> <ul style="list-style-type: none"><li>– синтаксис и типы данных языка программирования высокого уровня;</li><li>– теоретические основы объектной модели;</li><li>– библиотеки (пакеты) для научных вычислений.</li></ul> <i>Уметь:</i> <ul style="list-style-type: none"><li>– реализовывать алгоритмы для проведения и обработки результатов вычисли-</li></ul>	3(ОПК-6)1
		ИД-1 <sub>опк-6</sub> Умеет разрабатывать и использовать алгоритмы и программы для функционирования систем автоматического управления		3(ОПК-6)2
		ИД-3 <sub>опк-6</sub> Владеет навыками работы в		3(ОПК-6)3
				У(ОПК-6)1

Код компетенции	Планируемые результаты освоения образовательной программы	Код и наименование индикатора достижения компетенции	Планируемый результат обучения по дисциплине	Код показателя освоения
		системах, связанных с современными информационными технологиями	<p>тельных экспериментов с использованием языка программирования высокого уровня;</p> <ul style="list-style-type: none"> <li>– проектировать собственные типы данных;</li> <li>– использовать библиотеки (пакеты) языков программирования для научных вычислений.</li> </ul> <p><i>Владеть:</i></p> <ul style="list-style-type: none"> <li>– методами разработки программ для проведения и обработки результатов эксперимента на языке программирования высокого уровня;</li> <li>– методами объектно-ориентированного программирования;</li> <li>– инструментами для написания программ, использующих научные вычисления.</li> </ul>	<p>У(ОПК-6)2</p> <p>У(ОПК-6)3</p> <p>В(ОПК-6)1</p> <p>В(ОПК-6)2</p> <p>В(ОПК-6)3</p>

## 2. Место дисциплины в структуре образовательной программы

### Связь с предшествующими дисциплинами

Данная дисциплина изучается в 6-ом семестре. Для успешного освоения данной дисциплины студент должен иметь представление о базовых понятиях информатики и программирования. Вышеперечисленные навыки и умения студенты должны получить в процессе изучения таких дисциплин как «Математика», «Информатика», «Программирование и основы алгоритмизации» и т. д.

### Связь с последующими дисциплинами

Знания, умения и навыки, приобретенные студентами при изучении дисциплины «Разработка прикладных программ», позволят им успешно пройти преддипломную практику, а также выполнить выпускную квалификационную работу.

## 3. Содержание дисциплины

### 3.1. Тематический план дисциплины

**Очная форма обучения**

Наименование разделов и тем	Всего часов	Аудиторные занятия	Контактная работа по видам учебных занятий			Самостоятельная работа	Формы текущего контроля	Итоговый контроль знаний по дисциплине
			ЛК	ПЗ	ЛР			
Тема 1: Основы программирования	18	6	-	-	6	12	РЗ	
Тема 2: Типы данных	27	14	-	-	14	13	РЗ	
Тема 3: Специфические возможности Python	27	14	-	-	14	13	РЗ	
Зачет	-	-	-	-	-	-	РЗ	
<b>Всего</b>	<b>72</b>	<b>34</b>	<b>-</b>	<b>-</b>	<b>34</b>	<b>38</b>	<b>-</b>	

**Заочная форма обучения**

Наименование разделов и тем	Всего часов	Аудиторные занятия	Контактная работа по видам учебных занятий			Самостоятельная работа	Формы текущего контроля	Итоговый контроль знаний по дисциплине
			ЛК	ПЗ	ЛР			
Тема 1: Основы программирования	23	3	-	-	3	20	РЗ	
Тема 2: Типы данных	23	3	-	-	3	20	РЗ	
Тема 3: Специфические возможности Python	22	2	-	-	2	20	РЗ	
Зачет	-	-	-	-	-	-	РЗ	
<b>Всего</b>	<b>72</b>	<b>8</b>	<b>-</b>	<b>-</b>	<b>8</b>	<b>60</b>	<b>-</b>	<b>4</b>

\*ПЗ – практическое задание, РЗ – решение задач, КС – конкретная ситуация

### 3.2. Описание содержания дисциплины

#### Тема 1. Основы программирования

**Лабораторная работа №1.** Ввод-вывод и математические операции в Python. Функции, осуществляющие ввод-вывод на консоль. Использование различных математических операций в Python. Подключение пакетов и импорт функций. Первое консольное приложение.

**Лабораторная работа №2.** Ветвящиеся и циклические алгоритмы. Реализация полного, неполного и множественного ветвлений. Различные виды циклов. Операторы управления циклом. Циклы и else.

## Тема 2. Типы данных

**Лабораторная работа №3.** Тип данных Список. Изменяемый тип данных list. Операторы, функции и методы работы с list. Использование list как массива.

**Лабораторная работа №4.** Тип данных Кортеж. Изменяемый тип данных tuple. Операторы, функции и методы работы с tuple. Отличия tuple и list.

**Лабораторная работа №5.** Тип данных Множество. Неизменяемый тип set. Операторы, функции и методы работы с set. Операции пересечения, объединения, разности и дополнения множеств.

**Лабораторная работа №6.** Тип данных Словарь. Изменяемый тип dict. Пары «ключ-значение». Операторы, функции и методы работы с dict.

**Лабораторная работа №7.** Строки. Тип данных Строка. Методы, функции и операции работы со строками.

## Тема 3. Специфические возможности Python

**Лабораторная работа №8.** Внешние подпрограммы. Функции и процедуры. Реализация функций и процедур в Python. Формальные и фактические параметры функций. Параметры со значениями по умолчанию. Функции с переменным числом параметров. Написание программы использующей внешние функции.

**Лабораторная работа №9.** Работа с текстовыми и бинарными файлами. Определение файла. Отличия текстового и бинарного файлов. Запись и чтение из текстового файла средствами Python. Запись и чтение из бинарного файла средствами Python.

**Лабораторная работа №10.** Обработка исключений в Python. Определения исключения. Виды исключений в Python. Механизмы генерации, ловли и обработки исключений в Python.

**Лабораторная работа №11.** Регулярные выражения в Python. Основы создания и обработки регулярных выражений с использованием Python.

**Лабораторная работа №12.** Префиксные операторы \* и \*\*. Несколько способов использования операторов \* и \*\*: для распаковки итерируемых объектов, для передачи аргументов в функции.

**СРС.** Проработка теоретического материала по следующим темам:

- история появления языка программирования Python;
- области использования языка программирования Python;
- Python – интерпретируемый язык программирования;
- Python – язык программирования с динамической типизацией;
- базовые алгоритмические конструкции в Python;
- составные типы данных Python: списки, кортежи, строки, множества, словари;
- пакеты для научных вычислений Python;
- объектно-ориентированное программирование в Python;
- создание графических интерфейсов для Python-программ.

### 3.3. Учебно-методическое обеспечение для самостоятельной работы обучающихся

В целом внеаудиторная самостоятельная работа обучающегося при изучении курса включает в себя следующие виды работ:

- проработка (изучение) материалов лекций;
- чтение и проработка рекомендованной основной и дополнительной литературы;
- подготовка к лабораторным работам;
- поиск и проработка материалов из Интернет-ресурсов, периодической печати;
- выполнение домашних заданий в форме творческих (проблемно-поисковых, групповых) заданий, докладов;

- подготовка к текущему и итоговому (промежуточная аттестация) контролю знаний по дисциплине.

Основная доля самостоятельной работы обучающихся приходится на подготовку к лабораторным работам, тематика которых полностью охватывает содержание курса. Самостоятельная работа по подготовке к лабораторным работам предполагает умение работать с первичной информацией.

#### 4. Фонд оценочных средств

Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине «Разработка прикладных программ» представлен в приложении к рабочей программе дисциплины и включает в себя:

1. Перечень компетенций с указанием этапов их формирования в процессе освоения образовательной программы;
2. Описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкал оценивания;
3. Типовые контрольные задания или материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций;
4. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций.

#### Примерный перечень вопросов к промежуточной аттестации

1. Встроенный тип str. Методы объекта str.
2. print() и форматирование вывода, считывание данных с помощью input().
3. Работа с файловой системой средствами Python.
4. Работа с файлами. Методы open(), close(), read(), write().
5. Unicode- строки, типы bytes и bytearray.
6. Встроенные типы последовательностей list, tuple, range и их методы.
7. Встроенный объект dict и его методы.
8. Встроенные типы чисел – int, float, complex. Машинное представление чисел с плавающей точкой и целых. Преобразование типов при сравнении чисел.
9. Двоичное представление чисел. Неассоциативность операций в арифметике с плавающей запятой. Целые числа с произвольной точностью.
10. Множества. Встроенные типы set и frozenset.
11. Инструкции if/else/elif, логические операторы и выражения сравнения.
12. Циклы while и for в Python.
13. Функции в Python. Основные понятия.
14. Области видимости и пространство имен в Python.
15. Передача аргументов в функцию. Специальные режимы сопоставления аргументов.
16. Парадигма объектно-ориентированного программирования. Поддержка в Python функционального программирования.
17. Объекты. Динамическая типизация. Инкапсуляция.
18. Генерация объекта class. Новое пространство имен. Объект экземпляра класса.
19. Атрибуты класса. Атрибуты данных. Атрибуты-методы. Параметр self.
20. Специальные методы и атрибуты классов. Методы \_\_init\_\_() и \_\_del\_\_() в Python.
21. Абстрактные методы в Python. Классические классы и классы нового стиля.
22. Наследование. Базовый и производный класс. Построение производного класса.
23. Полиморфизм. Подмена методов в производном классе. Доступ к методам базового класса.

24. Обработка исключений. Инструкция try... except... else... finally. Классы встроенных исключений. Пользовательские исключения. Генерация заданного исключения. Поиск ошибок программирования на стадиях разработки и тестирования.
25. Генераторы Python.
26. Модули и пакеты. Библиотеки сторонних разработчиков (Django, Flask, Pygame, PyQt, twisted).
27. Модули и пакеты. Графический интерфейс.
28. Пакет NumPy.
29. Пакет SciPy.
30. Пакет Matplotlib.

## **5. Рекомендуемая литература**

### **5.1. Основная литература**

1. Марапулец Ю.В. Программирование на языке высокого уровня. Рекомендовано ДВ РУМЦ в качестве учебного пособия. Петропавловск-Камчатский: Издательство КамчатГТУ, 2008 – 189 с. (53)
2. Программирование на языке высокого уровня. Синицын С.В.2010. (15)

### **5.2. Дополнительная литература**

1. Гвоздева В.А. Введение в специальность программиста: учебник, 2007г. (20)
2. Харт Д.М. Системное программирование в среде Windows, 2005г. (5)
3. Проектирование программного обеспечения экономических информационных систем: учебник. Вендров А.2000г. (26)
4. Программирование с использованием Microsoft Windows Forms. Мастер-класс. Петцольд Ч. 2006г. (1)
5. Объектно-ориентированное программирование: учебник/ Г.С. Иванова, Т.Н. Ничушкина, Е.К. Пугачев. Иванова Г.С.2003г. (388)
6. Основы программирования: Учебник для вузов. Иванова Г.С.2002г. (19)
7. Технология программирования: учебник. Иванова Г.С. 2003г. (214)
8. <http://informatics.mccme.ru/course/view.php?id=156>
9. [https://inf5.ru/podgotovka\\_k\\_olympiad/olym\\_zadachi\\_s\\_resheniyami.htm](https://inf5.ru/podgotovka_k_olympiad/olym_zadachi_s_resheniyami.htm)
10. <http://anageorg.ru/olimp/materials>
11. <http://kpolyakov.spb.ru/school/probook/python.htm>
12. <http://dist-olimpiada.krasnogorka.edusite.ru/p4aa1.html>

## **6. Методические указания для обучающихся по освоению дисциплины**

### **Подготовка к лабораторным работам**

Лабораторная работа – это выполнение студентами под руководством преподавателя или по инструкции заданий (решение задач, написание программ) с применением персонального компьютера.

В ходе лабораторных работ студенты воспринимают и осмысливают новый учебный материал. Лабораторные занятия носят систематический характер, регулярно следуя за лекционными занятиями. Лабораторные работы выполняются согласно графику, при этом соблюдается принцип индивидуального выполнения работ.

Обучающийся должен подготовить отчет к каждой лабораторной работе, предусмотренной планом.

При подготовке к лабораторным занятиям необходимо заранее изучить методические рекомендации по его проведению. Обратит внимание на цель занятия, на основные вопросы для подготовки к занятию, на содержание темы занятия.



## **7. Курсовой проект (работа)**

В соответствии с учебным планом курсовое проектирование по дисциплине «Разработка прикладных программ» не предусмотрено.

## **8. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине**

В учебном процессе используются следующие информационные технологии:

- презентации;
- облачные хранилища для обмена информацией;
- электронные учебники;
- информационные ресурсы сети Интернет.

### **Программное обеспечение**

Для формирования отчетов по лабораторным работам

1. MS Word
2. Latex

Для просмотра электронных учебников и документов

1. MS Word
2. MS PowerPoint
3. STDViewer
4. Браузер (Opera, Google Chrome и др.)

Для разработки на Python

1. Python 3.x (<https://www.python.org/>)
2. PyCharm (<https://www.jetbrains.com/pycharm/download/#section=windows>)
3. Anaconda (<https://www.anaconda.com/distribution/>)
4. Wing IDE (<http://www.wingware.com/>)

## **9. Материально-техническое обеспечение дисциплины**

На кафедре имеются аудитории для проведения лекционных занятий и компьютерные классы для проведения лабораторных работ.