



МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное автономное образовательное учреждение высшего
образования
«Дальневосточный федеральный университет»
(ДВФУ)

ИНСТИТУТ МАТЕМАТИКИ И КОМПЬЮТЕРНЫХ ТЕХНОЛОГИЙ (ШКОЛА)

«СОГЛАСОВАНО»

Руководитель ОП

 Артемьева И.Л.

«УТВЕРЖДАЮ»

И.о. директора департамента



Смагин С.В.
«15» июля 2021 г.



РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ
Современные языки программирования

Направление подготовки 09.03.04 Программная инженерия
(Программная инженерия)
Форма подготовки очная

курс 4 семестр 7
лекции 16 час.
практические занятия 00 час.
лабораторные работы 16 час.
в том числе с использованием МАО лек. 0 / пр. 0 / лаб. 16 час.
всего часов аудиторной нагрузки 32 час.
в том числе с использованием МАО 16 час.
самостоятельная работа 76 час.
в том числе на подготовку к экзамену 45 час.
контрольные работы (количество) не предусмотрены
курсовая работа / курсовой проект не предусмотрены
зачет _ семестр
экзамен 7 семестр

Рабочая программа составлена в соответствии с требованиями Федерального государственного образовательного стандарта по направлению подготовки 09.03.04 Программная инженерия утвержденного приказом Министерства образования и науки РФ от 19 сентября 2017 г. № 920 (с изменениями и дополнениями)

Рабочая программа обсуждена на заседании кафедры прикладной математики, механики, управления и программного обеспечения, протокол № 12.1 от «25» декабря 2019 г.

И.о. директора департамента программной инженерии и искусственного интеллекта Смагин С.В.

Составители: зав. кафедрой ПММУиПО И.Л.Артемьева д.т.н., профессор
старший преподаватель кафедры ПММУиПО А.В.Шуленина

Владивосток
2021

Оборотная сторона титульного листа РПД

I. Рабочая программа пересмотрена на заседании кафедры прикладной математики, механики, управления и программного обеспечения:

Протокол от «09» июля 2021 г. № 7.1

Заведующий кафедрой _____ Артемяева И.Л.
(подпись) (И.О. Фамилия)

II. Рабочая программа пересмотрена на заседании департамента программной инженерии и искусственного интеллекта:

Протокол от «17» сентября 2021 г. № 9.1

И.о. директора департамента _____ Смагин С.В.
(подпись) (И.О. Фамилия)

III. Рабочая программа пересмотрена на заседании департамента:

Протокол от « _____ » _____ 20__ г. № _____

Директор департамента _____
(подпись) (И.О. Фамилия)

IV. Рабочая программа пересмотрена на заседании департамента:

Протокол от « _____ » _____ 20__ г. № _____

Директор департамента _____
(подпись) (И.О. Фамилия)

ЦЕЛИ И ЗАДАЧИ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Цель дисциплины - познакомить студентов с различными современными классами языков и систем программирования, в том числе логическими и функциональными, методами программирования на языках данных классов.

Задачи дисциплины:

- изучение современных классов языков программирования;
- получение навыков программирования на современных языках, на примере языка Python.

Планируемые результаты обучения по данной дисциплине (знания, умения, владения), соотнесенные с планируемыми результатами освоения образовательной программы.

Общепрофессиональные компетенции выпускников и индикаторы их достижения:

Наименование категории (группы) общепрофессиональных компетенций	Код и наименование общепрофессиональной компетенции	Код и наименование индикатора достижения общепрофессиональной компетенции
	ОПК-2. Способен понимать принципы работы современных информационных технологий и программных средств, в том числе отечественного производства, и использовать их при решении задач профессиональной деятельности	ОПК-2.1. Знает понятие программного интерфейса. ОПК-2.2. Умеет определять методы взаимодействия между подсистемами. ОПК-2.3. Имеет навыки применения методов проверки правильности программных систем.
	ОПК-7. Способен применять в практической деятельности основные концепции, принципы, теории и факты, связанные с информатикой	ОПК-7.1. Знает методы поиска информации, методы подготовки текстовых материалов. ОПК-7.2. Умеет выбрать информацию, требуемую для проведения занятий. ОПК-7.3. Владеет технологиями, используемыми при подготовке текстов и презентаций

Профессиональные компетенции выпускников и индикаторы их достижения:

Задача профессиональной деятельности	Объект или область знания	Код и наименование профессиональной компетенции	Код и наименование индикатора достижения профессиональной компетенции	Основание (ПС, анализ иных требований, предъявляемых к выпускникам)
Тип задач профессиональной деятельности: организационно-управленческий				
Участие в проведении переговоров с заказчиком и презентация проектов; участие в организации работ по управлению проектом ИС; участие в организации информационно-телекоммуникационной инфраструктуры и управлении информационной безопасностью ИС; участие в организации и управлении информационными ресурсами и сервисами	Прикладные и информационные процессы. Информационные технологии и. Программное обеспечение	ПК-3. Способность оформления методических материалов и пособий по применению программных систем	ПК-3.1. Знает Методы моделирования объектов и процессов средствами современных языков программирования ПК-3.2. Умеет использовать средства для моделирования объектов и процессов в современных языках при создании приложений ПК-3.3. Владеет методами выбора подходящих средств моделирования при создании приложений	06.022 Системный аналитик
Тип задач профессиональной деятельности: производственно-технологический				
Проведение работ по	Программное	ПК-9. Владение навыками	ПК-9.1. Знает особенности и	06.028 Системный

инсталляции программно о обеспечения автоматизированных систем и загрузки баз данных; настройка параметров ИС и тестирование результатов настройки; ведение технической документации; техническое сопровождение ИС в процессе эксплуатации ; применение Web технологий при реализации удаленного доступа в системах клиент – сервер и распределенных вычислений	обеспечен ие	использования операционных систем, сетевых технологий, средств разработки программного интерфейса, применения языков и методов формальных спецификаций, систем управления базами данных	отличия современных языков программирования ПК-9.2. Умеет использовать конструкции современных языков программирования, подходящие для создаваемых приложений ПК-9.3. Владеет методами сравнения современных языков и систем программирования	программист 06.022 Системный аналитик 06.004 Специалист по тестированию в области информационных технологий 06.001 Программист
		ПК-10. Владение навыками использования различных технологий разработки программного обеспечения	ПК-10.1. Знает методы проектирования программных систем ПК-10.2. Умеет использовать современные языки при разработке программных систем ПК-10.3. Владеет методами проверки корректности и эффективности	

Для формирования вышеуказанных компетенций в рамках дисциплины «Современные языки программирования» применяются следующие методы активного/ интерактивного обучения: метод проектов.

- **СТРУКТУРА И СОДЕРЖАНИЕ ТЕОРЕТИЧЕСКОЙ ЧАСТИ
КУРСА**

Лекционный материал (16 час.)

Раздел 1. Изучение современных языков программирования (4 часа)

Тема 1. Обзор и анализ основных возможностей современных языков программирования, специфики и сферы их применения (Python, Ruby, Go и другие). (2 час.)

Тема 2. Рассмотрение наиболее популярных (согласно рейтингам в открытых источниках) языков программирования и перспектив их использования. (2 час.)

Раздел 2. Язык программирования Python 3.0 (12 часов)

Тема 1. Общие сведения и основные типы данных. (4 час.)

Общие сведения о языке, история, эволюция, версии. Основные типы данных. Реализация управляющих конструкций. Специфика изменяемых и неизменяемых стандартных типов данных. Копирование: глубокое и поверхностное.

Тема 2. Обработка исключительных ситуаций в Python 3.0. (2 час.)

Конструкция try...except. Основы обработки исключений. Возбуждение стандартных исключений. Классы стандартных исключений. Разработка собственных исключений.

Тема 3. Типы коллекций в Python 3.0. (2 час.)

Последовательности: кортежи, именованные кортежи, списки. Множества set, frozen set, словари. Итераторы и функции для работы с итерируемыми объектами.

Тема 4. Модули и пакеты. (2 час.)

Понятие модуль и пакет. Конструкции импорта стандартных модулей. Разработка собственных модулей и пакетов. Доктексты и отладка модулей, запуск модуля в режиме тестирования. Обзор модулей стандартной библиотеки.

Тема 5. Объектно-ориентированное программирование в Python 3.0. (2 час.)

Класс, тип, объект. Ограничение доступа. Структура описания класса. Доступ к полям класса. Реализация наследования. Иерархия классов. Декораторы, их назначение, специфика использования.

- **СТРУКТУРА И СОДЕРЖАНИЕ ПРАКТИЧЕСКОЙ ЧАСТИ КУРСА**

Практические занятия (0 час.)

Практические занятия учебным планом не предусмотрены.

Лабораторные работы (16 час.)

Лабораторная работа 1. Разработка простой программы, знакомство со средой программирования (2 час.)

Лабораторная работа 2. Обработка стандартных классов исключительных ситуаций (2 час.)

Лабораторная работа 3. Работа с типами коллекций (4 часа)

Лабораторная работа 4. Разработка собственных модулей (4 час.)

Лабораторная работа 5. Организация классов, размещение классов в отдельных модулях, реализация наследования (6 час.)

Ш. УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ САМОСТОЯТЕЛЬНОЙ РАБОТЫ ОБУЧАЮЩИХСЯ

Учебно-методическое обеспечение самостоятельной работы обучающихся по дисциплине «Современные языки программирования» представлено в разделе VIII и включает в себя: план-график выполнения самостоятельной работы по дисциплине, в том числе примерные нормы времени на выполнение по каждому заданию; характеристика заданий для самостоятельной работы обучающихся и методические рекомендации по их выполнению; требования к представлению и оформлению результатов самостоятельной работы; критерии оценки выполнения самостоятельной работы.

IV. КОНТРОЛЬ ДОСТИЖЕНИЯ ЦЕЛЕЙ КУРСА

№ п/п	Контролируемые разделы/темы дисциплины	Коды и этапы формирования компетенций	Оценочные средства - наименование		
			текущий контроль	промежуточная аттестация	
1	Раздел 1. Изучение современных языков программирования	ОПК2 ОПК7 ПК3 ПК9 ПК10	Знает	ПР1	Экзамен, вопросы № 17-19
			Умеет, владеет	лабораторные работы и задания для самостоятельного выполнения	
2	Раздел 2. Язык программирования Python 3.0	ОПК2 ОПК7 ПК3 ПК9 ПК10	Знает	ПР1	Экзамен, вопросы № 1-16
			Умеет, владеет	лабораторные работы и задания для самостоятельного выполнения	

Типовые контрольные задания, методические материалы, определяющие процедуры оценивания знаний, умений и навыков и (или) опыта деятельности, а также критерии и показатели, необходимые для оценки знаний, умений, навыков и характеризующие этапы формирования компетенций в процессе освоения образовательной программы, представлены в разделе IX.

V. СПИСОК УЧЕБНОЙ ЛИТЕРАТУРЫ И ИНФОРМАЦИОННО МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

Основная литература

1. Буйначев С.К. Основы программирования на языке Python [Электронный ресурс]: учебное пособие/ Буйначев С.К., Боклаг Н.Ю.— Электрон. текстовые данные.— Екатеринбург: Уральский федеральный университет, ЭБС АСВ, 2014.— 92 с.— Режим доступа: <http://www.iprbookshop.ru/66183.html> .— ЭБС «IPRbooks
2. Шелудько В.М. Язык программирования высокого уровня Python. Функции, структуры данных, дополнительные модули [Электронный ресурс]: учебное пособие/ Шелудько В.М.— Электрон. текстовые данные.— Ростов-на-Дону, Таганрог: Издательство Южного федерального

- университета, 2017.— 107 с.— Режим доступа: <http://www.iprbookshop.ru/87530.html> .— ЭБС «IPRbooks»
3. Сузи Р.А. Язык программирования Python [Электронный ресурс]: учебное пособие/ Сузи Р.А.— Электрон. текстовые данные.— Москва: Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2020.— 350 с.— Режим доступа: <http://www.iprbookshop.ru/97589.html> .— ЭБС «IPRbooks»
 4. Лутц М. Изучаем Python. 4-е издание. Пер. с англ. СПб: Символ-Плюс. 2011. 1280 с.
 5. Саммерфилд М. Программирование на Python 3. Подробное руководство.- Пер. с англ. СПб: Символ-Плюс. 2009. 608 с.
 6. Python на практике / М. Саммерфилд: Пер. с англ. Слинкин А.А. М.: ДМК Пресс, 2014. – 338 с. <https://e.lanbook.com/book/66480>

Дополнительная литература
(печатные и электронные издания)

1. Копырин А. С., Салова Т. Л. Программирование на Python: Учебное пособие для студентов специальности 09.03.03 «Прикладная информатика (в экономике)», 2019 Сочинский государственный университет 48 стр. <https://e.lanbook.com/book/147665>
2. Маккинни У. Python и анализ данных, Издательство "ДМК Пресс", 2020 540 стр. <https://e.lanbook.com/book/131721>
3. Флэнаган Д., Мацумото Ю. Язык программирования Ruby. – СПб:Питер, 2011 – 496 с.
4. Фитцджеральд, М. Изучаем Ruby [Электронный ресурс] / М. Фитцджеральд - СПб.: БХВ-Петербург, 2008. - 336 с.: ил. - ISBN 978-5-9775-0225-2.
5. Ленд Мориц. Python: Непрерывная интеграция и доставка. Краткое руководство с примерами: Практическое пособие, 2020 Издательство: ДМК Пресс, 168 стр.

**Перечень ресурсов информационно-телекоммуникационной сети
«Интернет»**

1. <http://www.intuit.ru/studies/courses/49/49/info> Сузи Р.А. Язык программирования Python: учебное пособие. - 2-е изд., испр. М.: БИНОМ. Лаборатория знаний. Интернет-университет информационных технологий. -2007 – 328 с.
2. <http://window.edu.ru/library/pdf2txt/485/60485/30391> Роганов Е.А., Роганова Н. А. Программирование на языке Ruby. Учебное пособие. — М.: МГИУ, 2008. — 56 с.
3. <http://www.studentlibrary.ru/book/ISBN5940743579.html> Программирование на языке Ruby [Электронный ресурс] / Фултон Х. - М.: ДМК Пресс, 2007. –
4. <https://intuit.ru/studies/courses/49/49/info> Язык программирования Python[Электронный ресурс]

Перечень информационных технологий и программного обеспечения

При осуществлении образовательного процесса студентами и профессорско-преподавательским составом используется следующее программное обеспечение:

1. Microsoft Office (Access, Excel, PowerPoint, Word и т. д).
2. Open Office.
3. Skype.

При осуществлении образовательного процесса студентами и профессорско-преподавательским составом используются следующие информационно-справочные системы:

1. Научная электронная библиотека eLIBRARY.
2. Электронно-библиотечная система издательства «Лань».
3. Электронная библиотека "Консультант студента".
4. Электронно-библиотечная система IPRbooks.
5. Информационная система "ЕДИНОЕ ОКНО доступа к образовательным ресурсам".
6. Доступ к электронному заказу книг в библиотеке ДВФУ, доступ к нормативным документам ДВФУ, расписанию, рассылке писем.

Лабораторные занятия проводятся в специализированном компьютерном классе.

VI. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ

Дисциплина изучается в следующих организационных формах: лекционное занятие; лабораторное занятие; самостоятельное изучение теоретического материала; самостоятельное выполнение индивидуального проекта; индивидуальные и групповые консультации.

Основной формой самостоятельной работы студента является изучение конспекта лекций, их дополнение рекомендованной литературой, выполнение проекта, а также активная работа на лабораторных занятиях.

К прослушиванию лекции следует готовиться, для этого необходимо знать программу курса и рекомендованную литературу. Тогда в процессе лекции легче отделить главное от второстепенного, легче сориентироваться: что записать, что самостоятельно проработать, что является трудным для понимания, а что легко усвоить.

Контроль за выполнением самостоятельной работы студента производится в виде контроля каждого этапа работы, отраженного в документации и защиты проекта.

Студент должен планировать график самостоятельной работы по дисциплине и придерживаться его.

VII. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

Лекции проводятся с использованием проектора и внутренней системы портала ДВФУ. Лабораторные занятия проходят в аудиториях, оборудованных компьютерами типа Lenovo C360G-i34164G500UDK с лицензионными программами Microsoft Office 2013 и аудиовизуальными средствами проектор Panasonic DLPPProjectorPT-D2110XE, плазма LG FLATRON M4716CCBAM4716CJ. Для выполнения самостоятельной работы студенты в жилых корпусах ДВФУ обеспечены Wi-Fi.

VIII. УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ САМОСТОЯТЕЛЬНОЙ РАБОТЫ ОБУЧАЮЩИХСЯ

План-график выполнения самостоятельной работы по дисциплине

№ п/п	Дата/сроки выполнения	Вид самостоятельной работы	Примерные нормы времени на выполнение	Форма контроля
--------------	------------------------------	-----------------------------------	--	-----------------------

1	1-8 недели	Изучение основных принципов работы, типов данных, коллекций, управляющих конструкций в Python	10	Защита индивидуального задания
2	9-11 недели	Организация многомодульных программ	12	Защита индивидуального задания
3	12-16	Составление программ с реализацией объектно-ориентированного подхода	10	Защита индивидуального задания

Рекомендации по самостоятельной работе студентов

Самостоятельная работа (31 час.)

В процессе самостоятельной работы студенты выполняют задания на компьютере по программированию с использованием интерпретатора языка Python.

Примеры задач для самостоятельного решения

1. Напишите программу, которая для рисования цифр использовала бы символ *; соответствующие цифровые символы.

2. Напишите программу, которая в цикле while предлагала бы пользователю ввести число, постепенно накапливая список введенных чисел. Затем, когда пользователь завершит работу с программой (простым нажатием клавиши Enter), она выводила бы числа, введенные пользователем, количество введенных чисел, их сумму, наименьшее и наибольшее число и среднее значение (сумма / количество)

3. Напишите программу, генерирующую тестовый набор. Создайте списки слов, например, артиклей («the», «a» и других), имен существительных («cat», «dog», «man», «woman»), глаголов («sang», «ran», «jumped») и наречий («loudly», «quietly», «well», «badly»). Затем выполните пять циклов и на каждой итерации с помощью функции random.choice() выберите артикль, существительное, глагол и наречие. С помощью функции

`random.randint()` выберите одну из двух структур предложений: артикль, существительное, глагол и наречие, или артикль, существительное и глагол, и выведите предложение.

4. Напишите программу вычисляющую корни квадратного уравнения так, чтобы она не выводила коэффициенты со значением 0.0, а отрицательные коэффициенты выводились бы как $-n$

5. Написать программу, обеспечивающую ввод координат трёх точек (организовать проверку –все точки должны быть различны, значения должны быть числовыми), найти периметр и площадь треугольника, вывести длины всех сторон с указанием единиц измерений, координаты точек и периметр треугольника. 6. Написать функцию, которая строит разность двух списков одинакового типа.

7. Напишите программу, которая считывает слова из файла и выводит их на консоль сортируя их в алфавитном порядке; по частоте встречаемости

8. Напишите программу, которая считывает из файла данные о пользователях и генерирует для каждого уникальный пароль, оформите вывод на консоль в виде таблицы из двух колонок с произвольными разделителями.

9. Напишите интерактивную программу обслуживания списков строк в файлах. При запуске программа должна создать список всех файлов с расширением `.lst` в текущем каталоге. Воспользуйтесь функцией `os.listdir(".')`, чтобы получить список всех файлов, и отфильтруйте из него те файлы, которые не имеют расширения `.lst`. В случае отсутствия таких файлов программа должна попросить пользователя ввести имя файла и добавить расширение `.lst`, если пользователь не сделал этого. Если были найдены один или более файлов `.lst`, программа должна вывести их имена в виде списка пронумерованных строк, начиная с 1. Пользователю должно быть предложено ввести номер желаемого файла или 0; в последнем случае программа должна попросить у пользователя ввести имя нового файла. Если был указан существующий файл, программа должна прочитать его содержимое. Если файл пуст или было указано имя нового файла, программа должна вывести сообщение «no items are in the list» (список не содержит элементов).

10. Организуйте класс «корабль», состав методов и атрибутов определите самостоятельно. Оформите класс в виде отдельного модуля с доктекстами. Протестируйте модуль и разработайте программу для демонстрации работоспособности и возможностей собственного класса.

11. Определить функцию, которая из списка выбирает все элементы – списки чисел и формирует список списков в обратном порядке.

12. Определить функцию, которая к каждому нечетному элементу исходного списка прибавляет заданное число, а затем полученный список упорядочивает по возрастанию элементов.

Критерии оценки отчетов по самостоятельной работе

– 100-86 баллов выставляется, если содержание и составляющие части соответствуют выданному заданию. Продемонстрировано владение навыками подготовки документа по теме. Фактических ошибок, связанных с пониманием структуры и содержания задания нет.

– 85-76 - баллов выставляется, если при выполнении задания допущено не более одной ошибки. Продемонстрировано владение навыками подготовки документа по теме. Фактических ошибок, связанных с пониманием структуры и содержания задания нет.

– 75-61 балл выставляется, если при выполнении задания допущено не более двух ошибок. Продемонстрировано навыки подготовки документа по теме. Допущено не более 2 ошибок, связанных с пониманием структуры и содержания задания.

60-50 баллов - если структура и содержание задания не соответствуют требуемым

Шкала оценивания

Менее 60 баллов	незачтено	неудовлетворительно
От 61 до 75 баллов	зачтено	удовлетворительно
От 76 до 85 баллов	зачтено	хорошо
От 86 до 100 баллов	зачтено	отлично

IX. ФОНДЫ ОЦЕНОЧНЫХ СРЕДСТВ

№ п/п	Контролируемые разделы/темы дисциплины	Коды и этапы формирования компетенций	Оценочные средства - наименование	
			текущий контроль	промежуточная аттестация

1	Раздел 1. Изучение современных языков программирования	ОПК2 ОПК7 ПК3 ПК9 ПК10	Знает	ПР1	Экзамен, вопросы № 17-19
			Умеет, владеет	лабораторные работы и задания для самостоятельного выполнения	
2	Раздел 2. Язык программирования Python 3.0	ОПК2 ОПК7 ПК3 ПК9 ПК10	Знает	ПР1, тест	Экзамен, вопросы № 1-16
			Умеет, владеет	лабораторные работы и задания для самостоятельного выполнения	

Шкала оценивания уровня сформированности компетенций

Код и формулировка компетенции	Этапы формирования компетенции		критерии	показатели
ОПК-2. Способен понимать принципы работы современных информационных технологий и программных средств, в том числе отечественного производства, и использовать их при решении задач профессиональной деятельности	знает (пороговый уровень)	Знает понятие программного интерфейса.	Знание основных понятие программного интерфейса.	Способность дать ответы на вопросы
	умеет (продвинутый)	Умеет определять методы взаимодействия между подсистемами.	Умение определять методы взаимодействия между подсистемами.	Способность определять методы взаимодействия между подсистемами.
	владеет (высокий)	Имеет навыки применения методов проверки правильности программных систем.	Владение навыками применения методов проверки правильности программных систем. документов разных типов	Способность применения методов проверки правильности программных систем. документов разных типов
ОПК-7. Способен применять в практической деятельности основные концепции, принципы,	знает (пороговый уровень)	методы поиска информации, методы подготовки текстовых материалов.	Знание методов поиска информации, методы подготовки текстовых материалов	Способность дать ответы на вопросы
	умеет	выбрать	Умение выбрать	Наличие

теории и факты, связанные с информатикой	(продвинутый)	информацию, требуемую для проведения занятий.	информацию, требуемую для проведения занятий	программ
	владеет (высокий)	технологиями, используемыми при подготовке текстов и презентаций	Владение технологиями, используемыми при подготовке текстов и презентаций	Способность пояснить возможности применяемых технологий
ПК-3. Способность оформления методических материалов и пособий по применению программных систем	знает (пороговый уровень)	Методы моделирования объектов и процессов средствами современных языков программирования	Знание Методы моделирования объектов и процессов средствами современных языков программирования	Способность дать ответы на вопросы
	умеет (продвинутый)	Умеет использовать средства для моделирования объектов и процессов в современных языках при создании приложений	Умение использовать средства для моделирования объектов и процессов в современных языках при создании приложений	Способность использовать средства для моделирования объектов и процессов в современных языках при создании приложений
	владеет (высокий)	Владеет методами выбора подходящих средств моделирования при создании приложений	Владение методами выбора подходящих средств моделирования при создании приложений	Способность выбрать и обосновать выбор подходящих средств моделирования при создании приложений
ПК-9. Владение навыками использования операционных систем, сетевых технологий,	знает (пороговый уровень)	Знает особенности и отличия современных языков программирования	Знание особенности и отличия современных языков программирования	Способность дать ответы на вопросы

средств разработки программного интерфейса, применения языков и методов формальных спецификаций, систем управления базами данных	умеет (продвинутый)	Умеет использовать конструкции современных языков программирования, подходящие для создаваемых приложений	Умение использовать конструкции современных языков программирования, подходящие для создаваемых приложений	Наличие программ
	владеет (высокий)	Владеет методами сравнения современных языков и систем программирования	Владение методами сравнения современных языков и систем программирования	Способность анализировать и применять методы сравнения современных языков и систем программирования
ПК-10. Владение навыками использования различных технологий разработки программного обеспечения	знает (пороговый уровень)	Знает методы проектирования программных систем	Знание методов проектирования программных систем	Способность дать ответы на вопросы
	умеет (продвинутый)	Умеет использовать современные языки при разработке программных систем	Умение использовать современные языки при разработке программных систем	Наличие программ
	владеет (высокий)	Владеет методами проверки корректности и эффективности	Владение методами проверки корректности и эффективности	Способность применять методы проверки корректности и эффективности

Методические рекомендации, определяющие процедуры оценивания результатов освоения дисциплины

Текущая аттестация студентов. Текущая аттестация студентов по дисциплине «Современные языки программирования» проводится в соответствии с локальными нормативными актами ДВФУ и является обязательной.

Текущая аттестация по дисциплине «Современные языки программирования» проводится в форме контрольных мероприятий:

защиты индивидуальных заданий, тестирования.

Объектами оценивания выступают:

- учебная дисциплина (активность на занятиях, своевременность выполнения различных видов заданий, посещаемость всех видов занятий по аттестуемой дисциплине);

- степень усвоения теоретических знаний;

- уровень овладения практическими умениями и навыками по всем видам учебной работы;

- результаты самостоятельной работы.

Критерии оценки (устный ответ)

100-85 баллов - если ответ показывает прочные знания основных процессов изучаемой предметной области, отличается глубиной и полнотой раскрытия темы; владение терминологическим аппаратом; умение объяснять сущность, явлений, процессов, событий, делать выводы и обобщения, давать аргументированные ответы, приводить примеры; свободное владение монологической речью, логичность и последовательность ответа; умение приводить примеры современных проблем изучаемой области.

85-76 - баллов - ответ, обнаруживающий прочные знания основных процессов изучаемой предметной области, отличается глубиной и полнотой раскрытия темы; владение терминологическим аппаратом; умение объяснять сущность, явлений, процессов, событий, делать выводы и обобщения, давать аргументированные ответы, приводить примеры; свободное владение монологической речью, логичность и последовательность ответа. Однако допускается одна - две неточности в ответе.

75-61 - балл - оценивается ответ, свидетельствующий в основном о знании процессов изучаемой предметной области, отличающийся недостаточной глубиной и полнотой раскрытия темы; знанием основных вопросов теории; слабо сформированными навыками анализа явлений, процессов, недостаточным умением давать аргументированные ответы и приводить примеры; недостаточно свободным владением монологической речью, логичностью и последовательностью ответа. Допускается несколько ошибок в содержании ответа; неумение привести пример развития ситуации, провести связь с другими аспектами изучаемой области.

60-50 баллов - ответ, обнаруживающий незнание процессов изучаемой предметной области, отличающийся неглубоким раскрытием темы; незнанием основных вопросов теории, несформированными навыками анализа явлений, процессов; неумением давать аргументированные ответы, слабым владением монологической речью, отсутствием логичности и последовательности. Допускаются серьезные ошибки в содержании ответа; незнание современной проблематики изучаемой области.

Критерии оценки программы по лабораторным работам (проектов)

100-86 баллов выставляется, если содержание и составляющие части соответствуют выданному заданию. Продемонстрировано владение навыками разработки, тестирования программ на языке программирования. Программа правильно работает на всех наборах входных данных. Текст программы содержит комментарии.

85-76 - баллов выставляется, если при выполнении задания допущено не более одной ошибки. Продемонстрировано владение навыками разработки программ на языке программирования. Программа правильно работает не на всех наборах входных данных (90%). Текст программы содержит комментарии.

75-61 балл выставляется, если при выполнении задания допущено не более двух ошибок. Продемонстрировано знание методов разработки программ на языке программирования. Программа правильно работает не на всех наборах входных данных (70%). В тексте программы комментарии отсутствуют.

60- 0 баллов - если структура и содержание задания не соответствуют требуемым

Шкала оценивания

Менее 60 баллов	незачтено	неудовлетворительно
От 61 до 75 баллов	зачтено	удовлетворительно
От 76 до 85 баллов	зачтено	хорошо
От 86 до 100 баллов	зачтено	отлично

Промежуточная аттестация студентов. Промежуточная аттестация студентов по дисциплине «Современные языки программирования» проводится

в соответствии с локальными нормативными актами ДВФУ и является обязательной.

Промежуточная аттестация по дисциплине предусмотрена в виде экзамена в устной форме (устный опрос в форме ответов на вопросы)

Критерии выставления оценки студенту на зачете (экзамене)

Баллы	Оценка зачета/ экзамена	Требования к сформированным компетенциям
86-100	«зачтено»/ «отлично»	Оценка «отлично» выставляется студенту, если он глубоко и прочно усвоил программный материал, исчерпывающе, последовательно, четко и логически стройно его излагает, умеет тесно увязывать теорию с практикой, свободно справляется с задачами, вопросами и другими видами применения знаний, причем не затрудняется с ответом при видоизменении заданий, правильно обосновывает принятое решение, владеет разносторонними навыками и приемами выполнения практических задач.
76-85	«зачтено»/ «хорошо»	Оценка «хорошо» выставляется студенту, если он твердо знает материал, грамотно и по существу излагает его, не допуская существенных неточностей в ответе на вопрос, правильно применяет теоретические положения при решении практических вопросов и задач, владеет необходимыми навыками и приемами их выполнения.
61-75	«зачтено»/ «удовлетворительно»	Оценка «удовлетворительно» выставляется студенту, если он имеет знания только основного материала, но не усвоил его деталей, допускает неточности, недостаточно правильные формулировки, нарушения логической последовательности в изложении программного материала, испытывает затруднения при выполнении практических работ.
0-60	«не зачтено»/ «неудовлетворительно»	Оценка «неудовлетворительно» выставляется студенту, который не знает значительной части программного материала, допускает существенные ошибки, неуверенно, с большими затруднениями выполняет практические работы. Как правило, оценка «неудовлетворительно» ставится студентам, которые не могут продолжить обучение без дополнительных занятий по соответствующей дисциплине.

Оценочные средства для промежуточной аттестации

Вопросы к экзамену (семестр 7)

1. Стандартные типы данных в языке Python.
2. Изменяемые и неизменяемые типы данных. Копирование поверхностное и глубокое.
3. Типы коллекций и множества.
4. Функции, лямбда – функции в языке Python.
5. Словари, словари со значением по умолчанию. Отображения.
6. Основные принципы работы с итерируемыми объектами.
7. Создание собственных классов для обработки исключительных ситуаций.
8. Модуль, варианты импорта, особенности использования разных конструкций для подключения модулей. Исключение конфликтов при подключении модулей .
9. Пакеты, основные принципы организации пакетов, их размещения и импорта.
10. Тестирование модулей. Варианты запуска модуля. Доктексты.
11. Основные особенности реализации объектно-ориентированного подхода при создании программ на языке Python.
12. Атрибуты класса и свойства. Доступ к атрибутам. Декораторы.
13. Возможности средств стандартной библиотеки для работы с файлами и потоками.
14. Возможности средств стандартной библиотеки для работы с сетью Интернет.
15. Библиотека NumPy.
16. Библиотеки Python для работы с «большими» данными.
17. Основные перспективы развития языка Python, сферы применения.
18. Интеграция Python с другими языками программирования, версии языка C Python, J Python и другие.
19. Наиболее востребованные языки программирования

Примеры тестов для текущего контроля знаний.

1. Дан массив `>>> c = array([[1,2], [2,3], [4,5]])`
Чему равен срез `c[:,1]`:
 - а) `array([1, 2, 4])`
 - б) `array([2, 3])`
 - в) `array([2, 3, 5])` +
2. 6. Какие новые имена появятся в текущем модуле после выполнения следующего кода:
`import sre as re`
`from re import compile`
 - а) имена `sre`, `re` и `compile`
 - б) только имена `re` и `compile` +
 - в) только имена `sre` и `compile`
3. Что будет получено в результате вычисления следующего выражения:
`(0 < 5 <= 3) and (0 / 0)`:
 - а) `True` (или 1)
 - б) `False` (или 0) +
 - в) синтаксическая ошибка
4. . Как средствами самого Python определить имена формальных аргументов функции `func()`, если известно, что функция написана на Python:
 - а) `func.func_globals`
 - б) `inspect.getargspec(func)` +
 - в) `func.func_locals`
5. 21. Какие кодировки исходного текста программы поддерживает интерпретатор Python:
 - а) большинство кодировок, распространенных сегодня +
 - б) ASCII, Latin-1, UTF-8
 - в) ASCII
6. Как перевести Unicode-строку `u` в кодировку `koi8-r`:
 - а) `u.decode('koi8-r')`
 - б) `u.encode('koi8-r')` +
 - в) `string.decode(u, 'koi8-r')`
7. Какой модуль стандартной библиотеки Python позволяет работать с WWW на более низком уровне:
 - а) `httplib` +
 - б) `urlparse`
 - в) `urllib2`

8. 30. Скрытие информации о внутреннем устройстве объекта, при котором вся работа с объектом ведется только через общедоступный интерфейс называется:
- а) абстракцией
 - б) инкапсуляцией +
 - в) агрегацией