

Министерство образования и науки Российской Федерации
Федеральное государственное автономное образовательное
учреждение высшего образования
«СЕВЕРО-ВОСТОЧНЫЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ
имени М. К. АММОСОВА»
Институт математики и информатики
Кафедра информационных технологий

УТВЕРЖДАЮ
Директор ИМИ

_____ / В. И. Афанасьева /

« ____ » _____ 20 ____ г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Б1.В.ДВ.3.1 – Программирование на языке Питон

для программы магистратуры
по направлению подготовки
09.04.01 – Информатика и вычислительная техника

ОДОБРЕНО

Заведующий кафедрой
разработчика

_____ / _____ /

ОДОБРЕНО

Заведующий выпускаю-
щей кафедрой ИТ

_____ / _____ /

РЕКОМЕНДОВАНО

Нормоконтроль в составе
ОП пройден

_____ / _____ /

Протокол № ____ от

« ____ » _____ 20 ____ г.

Протокол № ____ от

« ____ » _____ 20 ____ г.

Протокол № ____ от

« ____ » _____ 20 ____ г.

1. АННОТАЦИЯ

к рабочей программе дисциплины
Б1.В.ДВ.3.1 – Программирование на языке Питон
Трудоемкость 2 з. е.

1.1. Цель освоения и краткое содержание дисциплины

Целью изучения дисциплины «Программирование на языке Питон» является: изучение и получение практических навыков использования языка Питон (Python), в том числе для анализа данных.

Краткое содержание дисциплины. Введение в Python. Пакеты и стандартная библиотека. Использование Python как скриптового и интерактивного языка. Веб-приложения на Python. Математика в пакетах NumPy и SciPy. Анализ данных и визуализация при помощи Python..

1.2. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы

Таблица 1. Перечень планируемых результатов обучения

Планируемые результаты освоения программы (содержание и коды компетенций)	Планируемые результаты обучения по дисциплине
ПК-4 : владением существующими методами и алгоритмами решения задач распознавания и обработки данных	<p>В результате изучения дисциплины обучающийся должен:</p> <p><u>знать:</u> основы синтаксиса и основные возможности стандартной библиотеки Python;</p> <p><u>уметь:</u></p> <ul style="list-style-type: none">– создавать несложные консольные приложения на Python;– использовать элементы объектно-ориентированного и функционального программирования;– пользоваться средствами библиотек для анализа данных и визуализации результатов;– концептуально разделять представление, бизнес-логику и модели данных; <p><u>владеть навыками:</u></p> <ul style="list-style-type: none">– установки пакетов средствами пакетного менеджера pip;– чтения документации к стандартной библиотеке и дополнительным пакетам;– использования IPython Notebook (Jupyter Notebook) для интерактивного анализа данных.

1.3. Место дисциплины в структуре образовательной программы

Таблица 2. *Содержательно-логические связи дисциплины*

Индекс дисциплины	Наименование дисциплины	Коды учебных дисциплин, практик	
		на которые опирается содержание дисциплины	для которых содержание дисциплины выступает опорой
Б1.В.ДВ.3.1	Программирование на языке Питон	Б1.В.ОД.2.1 – Объектно-ориентированное программирование	Б1.В.ДВ.7.1 – Визуализация в научных исследованиях

1.4. Язык преподавания

Русский.

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

Таблица 3. Выписка из учебного плана

Код и название дисциплины по учебному плану	Б1.В.ДВ.3.1 – Программирование на языке Питон	
Курс изучения	1	
Семестр(ы) изучения	2	
Форма промежуточной аттестации (зачет/экзамен)	зачет	
Курсовой проект / курсовая работа (указать вид работы при наличии в учебном плане), семестр выполнения		
Трудоемкость (в ЗЕТ)	2 (2)	
Трудоемкость (в часах) (сумма строк №1, 2, 3), в т. ч.:	72	
№ 1. Контактная работа обучающихся с преподавателем (КР), в часах:	Объем аудиторной работы, в часах	В т. ч. с применением ДОТ или ЭО, в часах
Объем работы (в часах) (1.1.+1.2.+1.3.)	37	
1.1. Занятия лекционного типа (лекции)	8	
1.2. Занятия семинарского типа, всего, в т.ч.:		
- семинары (практические занятия, коллоквиумы и т. п.)	–	
- лабораторные работы	26	
- практикумы		
1.3. КСР (контроль самостоятельной работы, консультации)	3	
№ 2. Самостоятельная работа обучающихся (СРС) (в часах)	35	
№ 3. Количество часов на экзамен (при наличии экзамена в учебном плане)	–	

3. Содержание дисциплины, структурированное по темам с указанием отведенного на них количества академических часов и видов учебных занятий

3.1. Распределение часов по темам и видам учебных занятий

Таблица 4

Тема	Всего часов	Контактная работа, в часах									Часы СРС
		Лекции	из них с прим-м ЭО и ДОТ	Семинары (практические занятия, коллоквиумы)	из них с прим-м ЭО и ДОТ	Лабораторные работы	из них с прим-м ЭО и ДОТ	Практикумы	из них с прим-м ЭО и ДОТ	КСР (консультации)	
Тема 1. Введение в Python	18	2	0	0	0	6	0	0	0	1	9
Тема 2. Пакеты. Python скриптовый и интерактивный	12	1	0	0	0	5	0	0	0	1	5
Тема 3. Веб-приложения на Python	16	2	0	0	0	5	0	0	0	1	8
Тема 4. Математика в пакетах NumPy и SciPy	15	2	0	0	0	5	0	0	0	0	8
Тема 5. Анализ данных и визуализация в Питоне	11	1	0	0	0	5	0	0	0	0	5
ВСЕГО ЧАСОВ	72	8	0	0	0	26	0	0	0	3	35

3.2. Содержание тем программы дисциплины

Тема 1. Введение в Python

Основы синтаксиса. Циклы, ветвления. Основные атомарные типы, строки, кортежи, списки, словари. Функции. Области видимости имен. Работа с файлами. Объекты. Аннотации. Пакеты и модули. Ссылочная прозрачность, побочные эффекты и чистые функции. Функциональный аспект языка Питон. Модуль `functools`.

Тема 2. Пакеты. Python скриптовый и интерактивный

Стандартная библиотека. Модули `math`, `os.path`. Менеджер пакетов `pip`. Установка, обновление и удаление пакетов. Портал `PyPI`. Сайты, изоляция при помощи `virtualenv`. `IPython`. `Jupyter Notebook` (`IPython Notebook`).

Тема 3. Веб-приложения на Python

Трехзвенная архитектура веб-приложений. MVC, MVVC. Фреймворк `Django`. Микрофреймворк `Flask`. HTML-шаблоны. Шаблонизатор `Jinja2`. Уровень представления. URL-маршрутизация. Уровень модели. Фреймворк `Pyramid` (`Pylons`).

Тема 4. Математика в пакетах NumPy и SciPy

Библиотека `NumPy`. Матричные операции в `NumPy`. Библиотека `SciPy`. Оптимизация.

Тема 5. Анализ данных и визуализация в Питоне

Библиотека `pandas`. Кадры данных (`dataframes`). Применение функций к кадрам данных. Очистка данных. Многомерные данные. Агрегация. Линейная регрессия. Библиотека

matplotlib. Графики для одномерных данных. Графики для двумерных данных.

3.3. Формы и методы проведения занятий, применяемые учебные технологии

При проведении занятий и организации СРС используются традиционные технологии сообщающего обучения, предполагающие передачу информации в готовом виде: проведение лекционных занятий, самостоятельная работа с источниками. Предусмотрено использование активных и интерактивных форм обучения с целью формирования и развития профессиональных навыков студентов — выполнение практических работ с применением компьютерных технологий.

4. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине

№	Наименование раздела (темы) дисциплины	Вид СРС	Трудоемкость (в часах)	Формы и методы контроля
1	Введение в Python	Прохождение онлайн-курса, покрывающего основы Python	12	Предъявление веб-страницы с информацией о прохождении теста
2	Пакеты. Python скриптовый и интерактивный	Сдача домашнего задания	6	Публикация кода в репозитории на сайте GitHub
3	Веб-приложения на Python	Разработка простого сайта	12	Публикация кода в репозитории на сайте GitHub
4	Математика в пакетах NumPy и SciPy	Прохождение вводной части курса на сайте edx.org	11	Предъявление веб-страницы с информацией о прохождении, публикация кода в репозитории на сайте GitHub
5	Анализ данных и визуализация в Питоне	Построение графиков	6	Публикация блокнота IPython с графиками в репозитории на сайте GitHub
	ИТОГО		47	

5. Методические указания для обучающихся по освоению дисциплины

В связи с небольшим объемом аудиторных часов, важное значение в освоении дисциплины имеет самостоятельная работа. Она предполагает в том числе и сдачу частей онлайн-курсов, некоторые из них на английском языке. Это требует самостоятельности и ответственности.

В диагностическом разделе дисциплины приведены тесты по каждому модулю дисциплины, которые необходимо выполнить для закрепления теоретических знаний.

Последовательное и добросовестное изучение курса является основой для выработки практических навыков использования гибкого и мультипарадигменного языка программирования, который с успехом может быть применен для решения различных задач в областях деятельности, предполагаемых стандартом подготовки по направлению «Информатика и вычислительная техника».

Рейтинговый регламент по дисциплине

Вид выполняемой учебной работы (контролирующие мероприятия)	Количество баллов (min)	Количество баллов (max)
Посещаемость	5	10
Домашние задания, онлайн курсы	25	45
Практические занятия	10	15
Тестирование	20	30
Количество баллов для получения зачета (min-max)	60	100

6. Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине

6.1. Показатели, критерии и шкала оценивания

Коды оцениваемых компетенций	Показатель оценивания (дескриптор) (по п.1.2)	Уровни освоения	Критерий оценивания	Оценка
ПК-4	<u>знать:</u> основы синтаксиса и основные возможности стандартной библиотеки Python; <u>уметь:</u> создавать несложные консольные приложения на Python; использовать элементы объектно-ориентированного и функционального программирования; пользоваться средствами библиотек для анализа данных и визуализации результатов; концептуально разделять представление, бизнес-логику и модели данных; <u>владеть навыками:</u> установки пакетов средствами пакетного менеджера pip; чтения документации к стандартной библиотеке и дополнительным пакетам; использования IPython Notebook (Jupyter Notebook) для интерактивного анализа данных.	освоено	способен писать программы, решающие простые задачи обработки массивов, чтения и записи текстовых данных в/из файла; способен находить справку по функциям стандартной библиотеки и правильно пользоваться ими; способен применять функции map и reduce, пользоваться перечислениями для списков и словарей; способен установить пакет по его описанию на сайте PyPI; способен строить графики функций, заданных значениями на одномерных, двумерных сетках	зачтено
		не освоено	не способен выполнить два и более пунктов из вышеперечисленного	не зачтено

6.2. Типовые контрольные задания (вопросы) для промежуточной аттестации

Коды оцениваемых компетенций	Оцениваемый показатель (ЗУВ)	Тема	Образец типового (тестового или практического) задания (вопроса)

ПК-4	знать основы синтаксиса и основные возможности стандартной библиотеки Python;	Введение в Python	<p>Имеется следующая программа:</p> <pre>iteration = 0 count = 0 while iteration < 5: for letter in "hello, world": count += 1 print('Iteration '+str(iteration)\ +'; count is: ' + str(count)) iteration += 1</pre> <p>Перечислите значения переменной count, которые будут распечатаны при ее исполнении.</p>
ПК-4	уметь создавать несложные консольные приложения на Python;	Пакеты. Python скриптовый и интерактивный	<p>Напишите консольную программу, которая пытается угадать загаданное пользователем целое число от 1 до 1000, показывая пользователю в цикле очередную догадку и запрашивая ответ пользователя (допустимые варианты: больше, меньше, или угадал). При корректной игре пользователя программа не должна делать более 10 попыток.</p>
ПК-4	уметь использовать элементы объектно-ориентированного и функционального программирования;	Введение в Python	<ol style="list-style-type: none"> 1. Получите с использованием map и лямбда-функций список, состоящий из квадратов всех чисел в списке A 2. Запишите в виде вызова нахождения максимума всех нечетных чисел из списка A, используя только reduce и filter. Запишите то же в виде вызова max над перечислением. 3. Какой метод вызывается при создании объекта? <ul style="list-style-type: none"> – self – obj.self – init – __init__ – new

ПК-4	уметь пользоваться средствами библиотек для анализа данных и визуализации результатов;	Математика в Python: NumPy и SciPy; Анализ данных в pandas и визуализация в matplotlib	<p>1. Какое из выражений станет после выполнения присваивания $A = \text{np.arcsin}(\text{np.array}(\text{np.arange}(0.0, 2.0, 0.1), \text{dtype=float})) / \text{np.pi} * 180$ равным 30 с точностью, лучшей 0.001?</p> <ul style="list-style-type: none"> – $A[0]$; – $A[5]$; – $A[6]$; – $A[30]$; – никакое, все элементы массива A меньше π; – никакое, попытка выполнить такого присваивания приведет к ошибке и массив A сформирован не будет. <p>2. Найдите в SciPy минимум функции</p> $\frac{1}{2}(1-x)^2 + (y-x^2)^2$ <p>3. Постройте цветной контурный график для</p> $\frac{1}{2}(1-x)^2 + (y-x^2)^2$ <p>в области $[0, 1.5] \times [0, 2]$.</p>
ПК-4	уметь концептуально разделять представление, бизнес-логику и модели данных;	Веб-приложения на Python	Постройте модели и перечислите представления, необходимые для веб-приложения ведения учета оценок, полученных студентами нескольких групп у одного преподавателя по одной и той же дисциплине за контрольные работы и тесты по различным темам этой дисциплины. Приложение должно предоставлять возможность студенту видеть все свои оценки, а преподавателю добавлять группы и студентов, а также контрольные мероприятия и оценки за них, просматривать отчет по всем мероприятиям в данной группе, а также средние баллы групп за каждое контрольное мероприятие.
ПК-4	владеть навыками установки пакетов средствами пакетного менеджера pip;	Пакеты. Python скриптовый и интерактивный	Установите пакеты numpy и matplotlib. Установите django в новое виртуальное окружение Python 3.x в папке c:\Users\student\mysite

ПК-4	владеть навыками чтения документации к стандартной библиотеке и дополнительным пакетам;	Пакеты. Python скриптовый и интерактивный	Прочитайте документацию функции <code>itertools.starmap</code> . Напишите нужную функцию и получите при помощи <code>starmap</code> суммы первых 1,2,3, ...100 натуральных чисел.
ПК-4	владеть навыками использования IPython Notebook (Jupyter Notebook) для интерактивного анализа данных.	Python скриптовый и интерактивный; Анализ данных в <code>pandas</code> и визуализация в <code>matplotlib</code>	Пройдите Titanic: Machine Learning from Disaster на сайте Kaggle.com

6.3. Методические материалы, определяющие процедуры оценивания

Форма промежуточной аттестации: зачет

Данный вид комплексного испытания предполагает последовательное выполнение всех форм текущего контроля, таких, как тесты, прохождение онлайн-курсов и выполнение практических заданий.

Тестирование. Данная форма контроля направлена на оценку основных теоретических знаний обучающегося по мере освоения основных разделов дисциплины.

Контрольные работы. В этой форме промежуточного контроля проверяются способности обобщенного анализа имеющихся теоретических знаний и умение пользоваться специальной литературой. Во время выполнения контрольной работы разрешается пользоваться справочной литературой.

7. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины

Перечень литературы

№	Автор, название, место издания, издательство, год издания учебной литературы, вид и характеристика иных информационных ресурсов	Наличие грифа, вид грифа	НБ СВФУ, кафедральная библиотека и кол-во экземпляров	Электронные издания: точка доступа к ресурсу (наименование ЭБС, ЭБ СВФУ)
Основная литература				
1	Степанов, Ю.А. Алгоритмизация и программирование. [Электронный ресурс] — НФИ КемГУ, 2013.		—	ЭБС «Лань»: https://e.lanbook.com/
Дополнительная литература				
1	Соловьев И.А., Червяков А.В., Репин А.Ю. Вычислительная математика на смартфонах, коммуникаторах и ноутбуках с использованием программных сред Python. [Электронный ресурс] М.: Лань, 2011.		—	ЭБС «Лань»: https://e.lanbook.com/

8. Перечень ресурсов информационно-телекоммуникационной сети «Интернет» (далее сеть-Интернет), необходимых для освоения дисциплины

- Онлайн-курс Using Python for Research.
Режим доступа: <https://www.edx.org/course/using-python-research-harvardx-ph526x>.
- Компания Microsoft. Онлайн-курс Introduction to Python for Data Science.
Режим доступа: <https://www.edx.org/course/introduction-python-data-science-microsoft-dat208x-3>.
- Компания Microsoft. Онлайн-курс Programming with Python for Data Science.
Режим доступа: <https://www.edx.org/course/programming-python-data-science-microsoft-dat210x-1>.
- В. Дронов. Django: Практика создания Web-сайтов на Python.
Режим доступа: <http://www.litres.ru/vladimir-dronov/django-praktika-sozdaniya-web-saytov-na-python-19213409>

9. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине

Для проведения лекционных занятий требуется аудитория, оборудованная доской, мультимедийным проектором с экраном. Для проведения лабораторных занятий требуется компьютерный класс с подключением к интернету.

10. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень программного обеспечения

10.1. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине

При осуществлении образовательного процесса по дисциплине используются следующие информационные технологии:

- использование на занятиях электронных изданий (чтение лекций с использованием слайд-презентаций);
- ведение учета посещаемости и выполнения учебных заданий в системе Google Docs;
- разработка обучающимися программ на языке Python;
- организация взаимодействия с обучающимися посредством электронной почты, специализированного образовательного форума Piazza;
- компьютерное тестирование.

10.2. Перечень программного обеспечения

При осуществлении образовательного процесса по дисциплине используются следующее программное обеспечение:

- язык Python версии 3.4 и новее;
- менеджер пакетов pip для Python;
- среда разработки JetBrains PyCharm;
- менеджер версий Git;
- интернет-браузер.

ЛИСТ АКТУАЛИЗАЦИИ РАБОЧЕЙ ПРОГРАММЫ ДИСЦИПЛИНЫ

Б1.В.ДВ.3.1 — Программирование на языке Питон

[illegible]

В таблице указывается только характер изменений (например, изменение темы, списка источников по теме или темам, средств промежуточного контроля) с указанием пунктов рабочей программы. Само содержание изменений оформляется приложением по сквозной нумерации.

Содержание

1	АННОТАЦИЯ	2
1.1	Цель освоения и краткое содержание дисциплины	2
1.2	Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы	2
1.3	Место дисциплины в структуре образовательной программы	3
1.4	Язык преподавания	3
2	Объем дисциплины в зачетных единицах с указанием количества академических часов, выделенных на контактную работу обучающихся с преподавателем (по видам учебных занятий) и на самостоятельную работу обучающихся	4
3	Содержание дисциплины, структурированное по темам с указанием отведенного на них количества академических часов и видов учебных занятий	5
3.1	Распределение часов по темам и видам учебных занятий	5
3.2	Содержание тем программы дисциплины	5
3.3	Формы и методы проведения занятий, применяемые учебные технологии . .	6
4	Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине	6
5	Методические указания для обучающихся по освоению дисциплины	7
6	Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине	8
6.1	Показатели, критерии и шкала оценивания	8
6.2	Типовые контрольные задания (вопросы) для промежуточной аттестации . .	8
6.3	Методические материалы, определяющие процедуры оценивания	11
7	Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины	12
8	Перечень ресурсов информационно-телекоммуникационной сети «Интернет» (далее сеть-Интернет), необходимых для освоения дисциплины	12
9	Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине	13
10	Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень программного обеспечения	13
10.1	Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине	13
10.2	Перечень программного обеспечения	13