

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное образовательное учреждение высшего образования
"Казанский (Приволжский) федеральный университет"
Институт вычислительной математики и информационных технологий



УТВЕРЖДАЮ

Проректор по образовательной деятельности КФУ
проф. Таюрский Д.А.

"__" _____ 20__ г.

Программа дисциплины

Язык Python Б1.В.ОД.3

Направление подготовки: 01.04.02 - Прикладная математика и информатика

Профиль подготовки: Анализ данных и его приложения

Квалификация выпускника: магистр

Форма обучения: очное

Язык обучения: русский

Год начала обучения по образовательной программе: 2017

Автор(ы): Абдуллин А.И.

Рецензент(ы): Андреанова А.А.

СОГЛАСОВАНО:

Заведующий(ая) кафедрой: Миссаров М. Д.

Протокол заседания кафедры No ___ от "___" _____ 20__ г.

Учебно-методическая комиссия Института вычислительной математики и информационных технологий:

Протокол заседания УМК No ___ от "___" _____ 20__ г.

Содержание

1. Перечень планируемых результатов обучения по дисциплине (модулю), соотнесенных с планируемыми результатами освоения образовательной программы
2. Место дисциплины в структуре основной профессиональной образовательной программы высшего образования
3. Объем дисциплины (модуля) в зачетных единицах с указанием количества часов, выделенных на контактную работу обучающихся с преподавателем (по видам учебных занятий) и на самостоятельную работу обучающихся
4. Содержание дисциплины (модуля), структурированное по темам (разделам) с указанием отведенного на них количества академических часов и видов учебных занятий
 - 4.1. Структура и тематический план контактной и самостоятельной работы по дисциплине (модулю)
 - 4.2. Содержание дисциплины
5. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине (модулю)
6. Фонд оценочных средств по дисциплине (модулю)
 - 6.1. Перечень компетенций с указанием этапов их формирования в процессе освоения образовательной программы и форм контроля их освоения
 - 6.2. Описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкал оценивания
 - 6.3. Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций в процессе освоения образовательной программы
 - 6.4. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций
7. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины (модуля)
 - 7.1. Основная литература
 - 7.2. Дополнительная литература
8. Перечень ресурсов информационно-телекоммуникационной сети "Интернет", необходимых для освоения дисциплины (модуля)
9. Методические указания для обучающихся по освоению дисциплины (модуля)
10. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине (модулю), включая перечень программного обеспечения и информационных справочных систем (при необходимости)
11. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине (модулю)
12. Средства адаптации преподавания дисциплины к потребностям обучающихся инвалидов и лиц с ограниченными возможностями здоровья

Программу дисциплины разработал(а)(и) Абдуллин А.И.

1. Перечень планируемых результатов обучения по дисциплине (модулю), соотнесенных с планируемыми результатами освоения образовательной программы

Выпускник, освоивший дисциплину, должен обладать следующими компетенциями:

Шифр компетенции	Расшифровка приобретаемой компетенции
ОК-1	способность к абстрактному мышлению, анализу, синтезу
ОК-3	готовность к саморазвитию, самореализации, использованию творческого потенциала
ОПК-3	способностью самостоятельно приобретать с помощью информационных технологий и использовать в практической деятельности новые знания и умения, в том числе, в новых областях знаний, непосредственно не связанных со сферой деятельности, расширять и углублять своё научное мировоззрение
ОПК-4	способностью использовать и применять углубленные знания в области прикладной математики и информатики
ПК-1	способность проводить научные исследования и получать новые научные и прикладные результаты самостоятельно и в составе научного коллектива
ПК-10	способностью разрабатывать учебно-методические комплексы для электронного и мобильного обучения
ПК-8	способностью разрабатывать корпоративные стандарты и профили функциональной стандартизации приложений, систем, информационной инфраструктуры

Выпускник, освоивший дисциплину:

Должен знать:

- синтаксис и управляющие конструкции языка Python;
- основные стандартные модули и библиотеки в Python.
- особенности разработки прикладных программ на языке Python;
- специализированные модули и приложения;
- принципы разработки собственных модулей и библиотек.

Должен уметь:

- разрабатывать прикладные программы на языке программирования Python;
- использовать как стандартные, так и дополнительные модули, расширения и пакеты;
- создавать собственные модули и импортировать их
- создавать собственные приложения и библиотеки.

Должен владеть:

- навыками разработки приложений с внешними источниками данных (текстовыми файлами, xml-файлами, csv-файлами);
- практическими навыками использования набора библиотек языка Python для различных прикладных задач;
- навыками разработки приложений и собственных библиотек на языке Python.

Должен демонстрировать способность и готовность:

- применять полученные знания в своей дальнейшей профессиональной деятельности.

2. Место дисциплины в структуре основной профессиональной образовательной программы высшего образования

Данная учебная дисциплина включена в раздел "Б1.В.ОД.3 Дисциплины (модули)" основной профессиональной образовательной программы 01.04.02 "Прикладная математика и информатика (Анализ данных и его приложения)" и относится к обязательным дисциплинам.

Осваивается на 1 курсе в 1 семестре.

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

Общая трудоемкость дисциплины составляет 4 зачетных(ые) единиц(ы) на 144 часа(ов).

Контактная работа - 28 часа(ов), в том числе лекции - 0 часа(ов), практические занятия - 0 часа(ов), лабораторные работы - 28 часа(ов), контроль самостоятельной работы - 0 часа(ов).

Самостоятельная работа - 116 часа(ов).

Контроль (зачёт / экзамен) - 0 часа(ов).

Форма промежуточного контроля дисциплины: зачет в 1 семестре.

4. Содержание дисциплины (модуля), структурированное по темам (разделам) с указанием отведенного на них количества академических часов и видов учебных занятий

4.1 Структура и тематический план контактной и самостоятельной работы по дисциплине (модулю)

N	Разделы дисциплины / модуля	Семестр	Виды и часы контактной работы, их трудоемкость (в часах)			Самостоятельная работа
			Лекции	Практические занятия	Лабораторные работы	
1.	Тема 1. Введение в программирование на языке Python	1	0	0	4	14
2.	Тема 2. Синтаксис и управляющие конструкции языка Python.	1	0	0	4	20
3.	Тема 3. Модули и пакеты в Python.	1	0	0	4	16
4.	Тема 4. Файлы и исключения. Работа с внешними источниками данных.	1	0	0	4	14
5.	Тема 5. Функциональное программирование. Функции lambda, map, filter. Генераторы, декораторы, рекурсия.	1	0	0	4	16
6.	Тема 6. Объектно-ориентированное программирование.	1	0	0	4	18
7.	Тема 7. Регулярные выражения.	1	0	0	4	18
	Итого		0	0	28	116

4.2 Содержание дисциплины

Тема 1. Введение в программирование на языке Python

Компилируемые и интерпретируемые языки программирования. Примеры и сравнения.

Общие сведения о языке Python и особенности его стиля программирования

Переменные, значения и их типы. Присваивание значения. Ввод значений с клавиатуры.

Встроенные операции и функции.

Краткое изложение стандартов оформления кода на Python (PEP8).

Тема 2. Синтаксис и управляющие конструкции языка Python.

Основные алгоритмические конструкции.

Условный оператор. Множественное ветвление. Условия равенства/неравенства.

Циклы и счетчики.

Определение функций. Параметры и аргументы. Вызовы функций. Оператор возврата return.

Списки, кортежи и словари.

Операторы общие для всех типов последовательностей.

Специальные операторы и функции для работы со списками.

Работа со словарями. Методы словарей.

Вложенные списки.

Тема 3. Модули и пакеты в Python.

Основные стандартные модули и пакеты в Python.

Импортирование модулей. Пакет `importlib`. Создание собственных модулей и их импортирование.

Стандартная библиотека модулей Python.

Специализированные модули и приложения.

Модуль `math`.

Модуль `copy` - поверхностное и глубокое копирование объектов. Модуль `collections` - специализированные типы данных, на основе словарей, кортежей, множеств, списков.

Тема 4. Файлы и исключения. Работа с внешними источниками данных.

Исключения, обработка исключений, вызов исключений (`try-except-finally`, `raise exception`)

Утверждения (`assert`).

Работа с данными из внешних источников.

Открытие, чтение, запись (инструкция `open`, менеджер контекста `with`)

Модуль `os.path`. Модуль `json`.

Работа с текстовыми файлами, `xml` и `csv` - файлами.

Тема 5. Функциональное программирование. Функции `lambda`, `map`, `filter`. Генераторы, декораторы, рекурсия.

Модуль `functools` - сборник функций высокого уровня: взаимодействующих с другими функциями или возвращающие другие функции.

Использование лямбда-функций. Функции `map filter`. Модификация функций с помощью декораторов.

Итерируемые объекты. Итераторы. Модуль `itertools`. Использование генераторов (`yield`). Использование рекурсии.

Тема 6. Объектно-ориентированное программирование.

Объектно-ориентированное программирование. Классы и объекты Python. Определение класса. Методы класса. Объекты, экземпляры и методы. Атрибутами класса - поля и методы, атрибуты экземпляра. Методы и значение `self`. Перегрузка операторов. Метод `__init__`. Обращения к атрибутам - специальный метод `__getattr__`.

Тема 7. Регулярные выражения.

Использование регулярных выражений.

Регулярные выражения в Python.

Шаблоны, соответствующие одному символу. Квантификаторы (указание количества повторений).

Модуль `re`.

`re.match()`

`re.search()`

`re.findall()`

`re.split()`

`re.sub()`

`re.compile()`

Скобочные группы и перечисления.

Шаблоны, соответствующие позиции.

5. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине (модулю)

Самостоятельная работа обучающихся выполняется по заданию и при методическом руководстве преподавателя, но без его непосредственного участия. Самостоятельная работа подразделяется на самостоятельную работу на аудиторных занятиях и на внеаудиторную самостоятельную работу. Самостоятельная работа обучающихся включает как полностью самостоятельное освоение отдельных тем (разделов) дисциплины, так и проработку тем (разделов), осваиваемых во время аудиторной работы. Во время самостоятельной работы обучающиеся читают и конспектируют учебную, научную и справочную литературу, выполняют задания, направленные на закрепление знаний и отработку умений и навыков, готовятся к текущему и промежуточному контролю по дисциплине.

Организация самостоятельной работы обучающихся регламентируется нормативными документами, учебно-методической литературой и электронными образовательными ресурсами, включая:

Порядок организации и осуществления образовательной деятельности по образовательным программам высшего образования - программам бакалавриата, программам специалитета, программам магистратуры (утвержден приказом Министерства образования и науки Российской Федерации от 5 апреля 2017 года №301).

Письмо Министерства образования Российской Федерации №14-55-996ин/15 от 27 ноября 2002 г. "Об активизации самостоятельной работы студентов высших учебных заведений".

Положение от 29 декабря 2018 г. № 0.1.1.67-08/328 "О порядке проведения текущего контроля успеваемости и промежуточной аттестации обучающихся федерального государственного автономного образовательного учреждения высшего образования "Казанский (Приволжский) федеральный университет".

Положение № 0.1.1.67-06/241/15 от 14 декабря 2015 г. "О формировании фонда оценочных средств для проведения текущей, промежуточной и итоговой аттестации обучающихся федерального государственного автономного образовательного учреждения высшего образования "Казанский (Приволжский) федеральный университет".

Положение № 0.1.1.56-06/54/11 от 26 октября 2011 г. "Об электронных образовательных ресурсах федерального государственного автономного образовательного учреждения высшего профессионального образования "Казанский (Приволжский) федеральный университет".

Регламент № 0.1.1.67-06/66/16 от 30 марта 2016 г. "Разработки, регистрации, подготовки к использованию в учебном процессе и удаления электронных образовательных ресурсов в системе электронного обучения федерального государственного автономного образовательного учреждения высшего образования "Казанский (Приволжский) федеральный университет".

Регламент № 0.1.1.67-06/11/16 от 25 января 2016 г. "О балльно-рейтинговой системе оценки знаний обучающихся в федеральном государственном автономном образовательном учреждении высшего образования "Казанский (Приволжский) федеральный университет".

Регламент № 0.1.1.67-06/91/13 от 21 июня 2013 г. "О порядке разработки и выпуска учебных изданий в федеральном государственном автономном образовательном учреждении высшего профессионального образования "Казанский (Приволжский) федеральный университет".

6. Фонд оценочных средств по дисциплине (модулю)

6.1 Перечень компетенций с указанием этапов их формирования в процессе освоения образовательной программы и форм контроля их освоения

Этап	Форма контроля	Оцениваемые компетенции	Темы (разделы) дисциплины
Семестр 1			
	Текущий контроль		
1	Лабораторные работы	ОК-1 , ОК-3 , ОПК-3 , ПК-1 , ПК-10	1. Введение в программирование на языке Python 2. Синтаксис и управляющие конструкции языка Python.
2	Контрольная работа	ОК-1 , ОК-3 , ОПК-3 , ОПК-4 , ПК-1 , ПК-10 , ПК-8	3. Модули и пакеты в Python.
3	Компьютерная программа	ОК-1 , ОПК-3 , ОПК-4 , ПК-1 , ПК-10 , ПК-8	4. Файлы и исключения. Работа с внешними источниками данных. 5. Функциональное программирование. Функции lambda, map, filter. Генераторы, декораторы, рекурсия. 6. Объектно-ориентированное программирование. 7. Регулярные выражения.
	Зачет	ОК-1, ОК-3, ОПК-3, ОПК-4, ПК-1, ПК-10, ПК-8	

6.2 Описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкал оценивания

Форма контроля	Критерии оценивания				Этап
	Отлично	Хорошо	Удовл.	Неуд.	
Семестр 1					
Текущий контроль					

Форма контроля	Критерии оценивания				Этап
	Отлично	Хорошо	Удовл.	Неуд.	
Лабораторные работы	Оборудование и методы использованы правильно. Проявлена превосходная теоретическая подготовка. Необходимые навыки и умения полностью освоены. Результат лабораторной работы полностью соответствует её целям.	Оборудование и методы использованы в основном правильно. Проявлена хорошая теоретическая подготовка. Необходимые навыки и умения в основном освоены. Результат лабораторной работы в основном соответствует её целям.	Оборудование и методы частично использованы правильно. Проявлена удовлетворительная теоретическая подготовка. Необходимые навыки и умения частично освоены. Результат лабораторной работы частично соответствует её целям.	Оборудование и методы использованы неправильно. Проявлена неудовлетворительная теоретическая подготовка. Необходимые навыки и умения не освоены. Результат лабораторной работы не соответствует её целям.	1
Контрольная работа	Правильно выполнены все задания. Продемонстрирован высокий уровень владения материалом. Проявлены превосходные способности применять знания и умения к выполнению конкретных заданий.	Правильно выполнена большая часть заданий. Присутствуют незначительные ошибки. Продемонстрирован хороший уровень владения материалом. Проявлены средние способности применять знания и умения к выполнению конкретных заданий.	Задания выполнены более чем наполовину. Присутствуют серьезные ошибки. Продемонстрирован удовлетворительный уровень владения материалом. Проявлены низкие способности применять знания и умения к выполнению конкретных заданий.	Задания выполнены менее чем наполовину. Продемонстрирован неудовлетворительный уровень владения материалом. Проявлены недостаточные способности применять знания и умения к выполнению конкретных заданий.	2
Компьютерная программа	Высокий уровень умений и навыков программирования, в том числе моделирования, алгоритмизации, использования языка программирования. Поставленная задача полностью решена.	Хороший уровень умений и навыков программирования, в том числе моделирования, алгоритмизации, использования языка программирования. Поставленная задача в основном решена.	Удовлетворительный уровень умений и навыков программирования, в том числе моделирования, алгоритмизации, использования языка программирования. Поставленная задача решена частично.	Недостаточный уровень умений и навыков программирования, в том числе моделирования, алгоритмизации, использования языка программирования. Поставленная задача не решена.	3
	Зачтено		Не зачтено		
Зачет	Обучающийся обнаружил знание основного учебно-программного материала в объеме, необходимом для дальнейшей учебы и предстоящей работы по специальности, справился с выполнением заданий, предусмотренных программой дисциплины.		Обучающийся обнаружил значительные пробелы в знаниях основного учебно-программного материала, допустил принципиальные ошибки в выполнении предусмотренных программой заданий и не способен продолжить обучение или приступить по окончании университета к профессиональной деятельности без дополнительных занятий по соответствующей дисциплине.		

6.3 Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций в процессе освоения образовательной программы

Семестр 1

Текущий контроль

1. Лабораторные работы

Темы 1, 2

Введение в программирование на языке Python. Особенности языка, основные библиотеки.

Изучение синтаксиса и управляющих конструкций языка Python. Приоритет операций. Работа со списками. Циклы for, while. Функция range. Глубокое и поверхностное копирование. id объекта.

Импорт модулей. Создание собственных модулей минимум из 3-х взаимосвязанных функций.

2. Контрольная работа

Тема 3

Теоретические вопросы и практическое задание по вариантам. Создать базу данных, удалить записи по критерию, вывести результирующую базу в файл или на печать. Импорт - экспорт данных в формате xml и csv.

Примерные темы заданий:

Как читать конструкцию `not A or B and C` (каков приоритет операций)?

Как вызвать исключение `KeyError` с текстом "my exception"?

Как удалить элемент 'val1' из словаря `my_dict`?

Какой модуль содержит инструменты для обработки времени и даты разными способами?

В чем отличие методов `datetime.now()` и `datetime.utcnow()`?

Какой модуль стандартной библиотеки содержит "Сортированный словарь"?

Какой модуль и какая переменная в нем содержит список аргументов командной строки?

Что означает: `@classmethod`, `@staticmethod`, `@property`?

3. Компьютерная программа

Темы 4, 5, 6, 7

Разработать компьютерную программу и реализовать на языке Python с использованием функционального и объектно-ориентированного программирования. Использовать лямбда-функцию. Построение генераторов, итераторов, декораторов.

Разбор текста с использованием регулярных выражений.

Задания по вариантам.

Зачет

Вопросы к зачету:

1. Язык Python и особенности его стиля программирования. Интерактивный режим Python.

`ipython`. `Jupyter Notebook`.

2. Синтаксис и управляющие конструкции языка Python. Переменные, значения и их типы.

Типы данных в Python.

3. Встроенные операции и функции. Основные алгоритмические конструкции.

4. Условный оператор. Множественное ветвление.

5. Циклы и счетчики.

6. Определение функций. Параметры и аргументы. Вызовы функций. Оператор возврата.

Конструкции `*args`, `**kwargs`.

7. Списки, кортежи и словари.

8. Операторы общие для всех типов последовательностей.

9. Специальные операторы и функции для работы со списками. Срезы.

10. Работа со словарями. Методы словарей.

11. Случайные числа. `random`, `randrange`, `choice`.

12. Функции обработки строк. `join`, `replace`, `split`.

13. Стандартная библиотека `re`. Модули и пакеты в Python. Основные стандартные модули.

14. Импортирование модулей. Создание собственных модулей и их импортирование.

Специализированные модули и приложения.

15. Файлы и исключения. Работа с внешними источниками данных.

16. Исключения, обработка исключений, вызов исключений (`try-except-finally`).

17. Утверждения (`assert`). Открытие, чтение, запись. (`open`, инструкция `with`).

18. Работа с текстовыми файлами, `xml` и `csv` - файлами.

19. Функциональное программирование. Лямбда-функции.

20. Использование функций `map`, `filter`, `reduce`, `zip`.

21. Генераторы, декораторы, рекурсия.

22. Модификация функций с помощью декораторов.

23. Итерируемые объекты. Использование генераторов (`yield`).

24. ООП в Python. Классы, объекты и экземпляры классов. Наследование.

25. Магические методы. Переопределение операторов. Методы классов.

26. Инкапсуляция. Условно частные и строго частные методы.

27. Регулярные выражения. Использование регулярных выражений. Пакет `re`.

28. Наука о данных и Python. Библиотеки: `NumPy`, `pandas`, `matplotlib`, `SciPy`.

29. Основы `NumPy`: массивы и векторные вычисления.

30. Инструменты визуализации данных для Python.

31. Введение в API библиотеки `matplotlib`.

6.4 Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций

В КФУ действует балльно-рейтинговая система оценки знаний обучающихся. Суммарно по дисциплине (модулю) можно получить максимум 100 баллов за семестр, из них текущая работа оценивается в 50 баллов, итоговая форма контроля - в 50 баллов.

Для зачёта:

56 баллов и более - "зачтено".

55 баллов и менее - "не зачтено".

Для экзамена:

86 баллов и более - "отлично".

71-85 баллов - "хорошо".

56-70 баллов - "удовлетворительно".

55 баллов и менее - "неудовлетворительно".

Форма контроля	Процедура оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций	Этап	Количество баллов
Семестр 1			
Текущий контроль			
Лабораторные работы	В аудитории, оснащённой соответствующим оборудованием, обучающиеся проводят учебные эксперименты и тренируются в применении практико-ориентированных технологий. Оцениваются знание материала и умение применять его на практике, умения и навыки по работе с оборудованием в соответствующей предметной области.	1	15
Контрольная работа	Контрольная работа проводится в часы аудиторной работы. Обучающиеся получают задания для проверки усвоения пройденного материала. Работа выполняется в письменном виде и сдаётся преподавателю. Оцениваются владение материалом по теме работы, аналитические способности, владение методами, умения и навыки, необходимые для выполнения заданий.	2	15
Компьютерная программа	Обучающиеся самостоятельно составляют программу на определённом языке программирования в соответствии с заданием. Программа сдаётся преподавателю в электронном виде. Оценивается реализация алгоритмов на языке программирования, достижение заданного результата.	3	20
Зачет	Зачёт нацелен на комплексную проверку освоения дисциплины. Обучающийся получает вопрос (вопросы) либо задание (задания) и время на подготовку. Зачёт проводится в устной, письменной или компьютерной форме. Оценивается владение материалом, его системное освоение, способность применять нужные знания, навыки и умения при анализе проблемных ситуаций и решении практических заданий.		50

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

7.1 Основная литература:

- Сузи Р. А. Python: Пособие / Сузи Р.А. - СПб:БХВ-Петербург, 2015. - 759 с. ISBN 978-5-9775-1417-0 - Режим доступа: <http://znanium.com/catalog/product/939857>
- Python 3. Самое необходимое: Пособие / Прохоренок Н.А., Дронов В.А. - СПб:БХВ-Петербург, 2016. - 464 с. ISBN 978-5-9775-3631-8 - Режим доступа: <http://znanium.com/catalog/product/944129>
- Гуриков С. Р. Основы алгоритмизации и программирования на Python: учеб. пособие / С.Р. Гуриков. ? М. : ФОРУМ : ИНФРА-М, 2018. ? 343 с. ? (Высшее образование). - Режим доступа: <http://znanium.com/catalog/product/924699>

7.2. Дополнительная литература:

- Прохоренок Н. А. Python 3 и PyQt 5. Разработка приложений: Пособие - СПб: Издательство 'БХВ-Петербург', 2016 - 833с. - Режим доступа <http://znanium.com/go.php?id=944629>
- Саммерфилд М., Python на практике [Электронный ресурс] / Марк Саммерфилд - М. : ДМК Пресс, 2014. - 338 с. - ISBN 978-5-97060-095-5 - Режим доступа: <http://www.studentlibrary.ru/book/ISBN9785970600955.html>
- Python. Самое необходимое: Практическое руководство / Прохоренок Н.А. - СПб:БХВ-Петербург, 2010. - 414 с.: ISBN 978-5-9775-0614-4 - Режим доступа: <http://znanium.com/catalog/product/354989>

8. Перечень ресурсов информационно-телекоммуникационной сети "Интернет", необходимых для освоения дисциплины (модуля)

дистрибутивы и документация для языка программирования Python - www.python.org

Онлайн Python интерпретатор - <https://repl.it/languages/python3>

Онлайн учебник и интерпретатор Python - tutorialspoint.com/python3

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

Вид работ	Методические рекомендации
лабораторные работы	Чтение конспекта лекций и учебника должно сопровождаться практическим решением задач на основании теоретических положений дисциплины, для чего рекомендуется завести специальную тетрадь. Если студент видит несколько путей для решения задачи, то он должен сравнить их и выбрать из них самый оптимальный (минимальное количество строк кода). Полезно до начала программирования составить блок-схему. Решения задач и примеров следует излагать подробно, обосновывать каждый этап решения, исходя из теоретических положений курса. Программный код снабжать комментариями, придерживаться рекомендации PEP8.
самостоятельная работа	Самостоятельная работа над учебным материалом является составной частью обучения студента. По математическим курсам она складывается из чтения конспекта лекций и учебника, решения практических задач, самопроверки и выполнения контрольных заданий. Кроме этого, студент может обращаться с вопросами к преподавателю для получения устной или письменной консультации.
контрольная работа	В процессе изучения дисциплины студент должен выполнить контрольную работу, главная цель которой ? оказать студенту помощь в его самостоятельной работе. Оценка и разбор этой работы позволяют студенту судить о степени усвоения им соответствующего раздела курса; указывают на имеющиеся у него пробелы, на желательное направление дальнейшей работы.
компьютерная программа	Наличие хотя бы одной типичной ошибки в программе дает право преподавателю считать работу не выполненной: - Наличие синтаксических ошибок - используются конструкции, несуществующие в языке - Код написан для Python 2.x (используйте 3.x) - Использование счетчика в цикле, когда достаточно итерации по элементам. Пример <code>for i in range(len(lst))</code> вместо <code>for lst_elem in lst</code> - Открытие файлов (<code>open(path, mode)</code>) с излишним <code>mode</code> - не открывайте файл на чтение и запись, если достаточно только для чтения - Не закрытие файла (функция <code>close()</code> у переменной файла) после завершения работы с ним - Все данные, необходимые для работы функции, должны передаваться ей в виде параметров. - В коде программы не должно быть более 2х скопированных (одинаковых) строк кода. Выносите одинаковые строчки в отдельные функции. - Весь код (как правило тестовый), непосредственно не решающий задачу должен помещаться под условное исполнение <code>"if __name__ == '__main__':"</code> .
зачет	Чтение конспекта лекций и учебника. Наличие хотя бы одной типичной ошибки в программе дает право преподавателю считать работу не выполненной: - Наличие синтаксических ошибок - используются конструкции, несуществующие в языке - Код написан для Python 2.x (используйте 3.x) - Использование счетчика в цикле, когда достаточно итерации по элементам. Пример <code>for i in range(len(lst))</code> вместо <code>for lst_elem in lst</code> - Открытие файлов (<code>open(path, mode)</code>) с излишним <code>mode</code> - не открывайте файл на чтение и запись, если достаточно только для чтения - Не закрытие файла (функция <code>close()</code> у переменной файла) после завершения работы с ним - Все данные, необходимые для работы функции, должны передаваться ей в виде параметров. - В коде программы не должно быть более 2х скопированных (одинаковых) строк кода. Выносите одинаковые строчки в отдельные функции. - Весь код (как правило тестовый), непосредственно не решающий задачу должен помещаться под условное исполнение <code>"if __name__ == '__main__':"</code> .

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

Освоение дисциплины "Язык Python" предполагает использование следующего программного обеспечения и информационно-справочных систем:

Браузер Mozilla Firefox

Adobe Reader XI

Учебно-методическая литература для данной дисциплины имеется в наличии в электронно-библиотечной системе Издательства "Лань", доступ к которой предоставлен обучающимся. ЭБС Издательства "Лань" включает в себя электронные версии книг издательства "Лань" и других ведущих издательств учебной литературы, а также электронные версии периодических изданий по естественным, техническим и гуманитарным наукам. ЭБС Издательства "Лань" обеспечивает доступ к научной, учебной литературе и научным периодическим изданиям по максимальному количеству профильных направлений с соблюдением всех авторских и смежных прав.

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

Освоение дисциплины "Язык Python" предполагает использование следующего материально-технического обеспечения:

Мультимедийная аудитория, вместимостью более 60 человек. Мультимедийная аудитория состоит из интегрированных инженерных систем с единой системой управления, оснащенная современными средствами воспроизведения и визуализации любой видео и аудио информации, получения и передачи электронных документов. Типовая комплектация мультимедийной аудитории состоит из: мультимедийного проектора, автоматизированного проекционного экрана, акустической системы, а также интерактивной трибуны преподавателя, включающей тач-скрин монитор с диагональю не менее 22 дюймов, персональный компьютер (с техническими характеристиками не ниже Intel Core i3-2100, DDR3 4096Mb, 500Gb), конференц-микрофон, беспроводной микрофон, блок управления оборудованием, интерфейсы подключения: USB, audio, HDMI. Интерактивная трибуна преподавателя является ключевым элементом управления, объединяющим все устройства в единую систему, и служит полноценным рабочим местом преподавателя. Преподаватель имеет возможность легко управлять всей системой, не отходя от трибуны, что позволяет проводить лекции, практические занятия, презентации, вебинары, конференции и другие виды аудиторной нагрузки обучающихся в удобной и доступной для них форме с применением современных интерактивных средств обучения, в том числе с использованием в процессе обучения всех корпоративных ресурсов. Мультимедийная аудитория также оснащена широкополосным доступом в сеть интернет. Компьютерное оборудование имеет соответствующее лицензионное программное обеспечение.

Компьютерный класс, представляющий собой рабочее место преподавателя и не менее 15 рабочих мест студентов, включающих компьютерный стол, стул, персональный компьютер, лицензионное программное обеспечение. Каждый компьютер имеет широкополосный доступ в сеть Интернет. Все компьютеры подключены к корпоративной компьютерной сети КФУ и находятся в едином домене.

12. Средства адаптации преподавания дисциплины к потребностям обучающихся инвалидов и лиц с ограниченными возможностями здоровья

При необходимости в образовательном процессе применяются следующие методы и технологии, облегчающие восприятие информации обучающимися инвалидами и лицами с ограниченными возможностями здоровья:

- создание текстовой версии любого нетекстового контента для его возможного преобразования в альтернативные формы, удобные для различных пользователей;
- создание контента, который можно представить в различных видах без потери данных или структуры, предусмотреть возможность масштабирования текста и изображений без потери качества, предусмотреть доступность управления контентом с клавиатуры;
- создание возможностей для обучающихся воспринимать одну и ту же информацию из разных источников - например, так, чтобы лица с нарушениями слуха получали информацию визуально, с нарушениями зрения - аудиально;
- применение программных средств, обеспечивающих возможность освоения навыков и умений, формируемых дисциплиной, за счёт альтернативных способов, в том числе виртуальных лабораторий и симуляционных технологий;
- применение дистанционных образовательных технологий для передачи информации, организации различных форм интерактивной контактной работы обучающегося с преподавателем, в том числе вебинаров, которые могут быть использованы для проведения виртуальных лекций с возможностью взаимодействия всех участников дистанционного обучения, проведения семинаров, выступления с докладами и защиты выполненных работ, проведения тренингов, организации коллективной работы;
- применение дистанционных образовательных технологий для организации форм текущего и промежуточного контроля;
- увеличение продолжительности сдачи обучающимся инвалидом или лицом с ограниченными возможностями здоровья форм промежуточной аттестации по отношению к установленной продолжительности их сдачи;
- продолжительности сдачи зачёта или экзамена, проводимого в письменной форме, - не более чем на 90 минут;
- продолжительности подготовки обучающегося к ответу на зачёте или экзамене, проводимом в устной форме, - не более чем на 20 минут;

- продолжительности выступления обучающегося при защите курсовой работы - не более чем на 15 минут.

Программа составлена в соответствии с требованиями ФГОС ВО и учебным планом по направлению 01.04.02 "Прикладная математика и информатика" и магистерской программе Анализ данных и его приложения .