

Майнор «Интеллектуальный анализ данных»
Программа учебной дисциплины «Введение в программирование»

Утверждена
Академическим советом ОП
Протокол № от _____._____.2019

Автор	Бычков И.С. Гречихин И.С. Казаков М.А.
Число кредитов	5
Контактная работа (час.)	60
Самостоятельная работа (час.)	130
Курс	2
Формат изучения дисциплины	Без использования онлайн курса

I. ЦЕЛЬ, РЕЗУЛЬТАТЫ ОСВОЕНИЯ ДИСЦИПЛИНЫ И ПРЕРЕКВИЗИТЫ

Изучение дисциплины «Введение в программирование» нацелено на освоение языка программирования Python и восполнение пробелов в знаниях по информатике, полученных в школе. Курс является первым в майноре «Интеллектуальный анализ данных». Его задача обучить студентов основам программирования на языке Python. Целью курса также является ознакомление студентов с основными принципами проектирования и анализа алгоритмов и структур данных, развитие навыков оценки сложности алгоритмов, их практической реализации. Предполагается, что в результате освоения курса студенты будут способны самостоятельно разработать и реализовать на языке программирования Python несложные алгоритмы.

В результате освоения дисциплины студент должен:

знать:

- синтаксис и семантику основных конструкций языка программирования Python.

уметь:

- разрабатывать несложные приложения на языке Python
- эффективно использовать структуры данных и алгоритмы языка Python для решения прикладных задач
- применять наиболее известные внешние библиотеки
- проводить простую оценку эффективности разработанных алгоритмов

владеть:

- навыками работы с интерпретатором языка Python и в интегрированной среде разработки PyCharm

Изучение дисциплины базируется на дисциплинах математического и информационного блоков основных образовательных программ студентов. :

Для освоения учебной дисциплины студенты должны владеть следующими знаниями и компетенциями:

- Математика и информатика в объеме программы общих курсов 1-го года обучения в ВУЗе и программы средней школы.

Основные положения дисциплины должны быть использованы в дальнейшем при изучении следующих дисциплин майнора:

- 1 Введение в анализ данных
- 2 Современные методы машинного обучения
- 3 Прикладные задачи анализа данных

II. СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ

Тема (раздел дисциплины)	Объем в часах лк/см/ср	Планируемые результаты обучения (ПРО), подлежащие контролю	Формы контроля
1. Обзор курса. Язык программирования Python. Особенности и области применения языка Python. Запуск программ на языке Python с помощью интерпретатора и с использованием IDE.	2 / 4 / 6	Базовые знания языка Python	Текущий опрос, лаб 1, экзамен
2. Обзор базовых типов данных языка Python. Функции type, dir, help. Неизменяемые типы данных. Числовые типы. Встроенные функции и модули для работы с числовыми типами. Строковый тип данных. Встроенные функции и модули для работы со строками.	2 / 4 / 6	Базовые знания языка Python	Текущий опрос, лаб 1, экзамен
3. Концепция неизменяемости в типах данных языка Python. Создание переменных (имен). Перекрестные ссылки. Изменяемые типы данных. Списки. Кортежи.	2 / 4 / 6	Базовые знания языка Python	Текущий опрос, лаб 1, экзамен
4. Изменяемые типы данных. Словари. Множества. Файлы. Bool. NoneType. Пример простой програм-	2 / 4 / 6	Базовые знания языка Python	Текущий опрос, лаб 1, экзамен

мы на языке Python для визуализации дружб в социальной сети VK и проверки теории шести рукопожатий.			
5. Выражения и синтаксис языка Python. Условный оператор if. Оператор присваивания и его формы. Присваивание последовательностей. Дополнительные варианты оператора присваивания.	2 / 4 / 6	Базовые знания языка Python	Текущий опрос, лаб 1, экзамен
6. Выражения и синтаксис языка Python. Операторы циклов for и while. Вложенные циклы. Инструкции break и continue. Функции range, enumerate.	2 / 4 / 6	Простой анализ данных, работа с библиотеками Python	Текущий опрос, лаб 2, экзамен
7. Функции в языке Python. Ключевое слово def. Области видимости. Локальные переменные, ключевые слова global, nonlocal. Передача аргументов. Упаковка и распаковка аргументов.	2 / 4 / 6	Простой анализ данных, работа с библиотеками Python	Текущий опрос, лаб 2, экзамен
8. Функции в языке Python. Функциональное программирование и ООП. Анонимные функции, ключевое слово lambda. Элементы функционального программирования - map, reduce, filter. Итераторы и генераторы. Модуль itertools.	2 / 4 / 6	Простой анализ данных, работа с библиотеками Python	Текущий опрос, лаб 2, экзамен
9. Объектно-ориентированное программирование в Python. Классы и ключевое слово class. Наследование. Перегрузка операторов. Исключения в языке Python.	2 / 4 / 6	Простой анализ данных, работа с библиотеками Python	Текущий опрос, лаб 2, экзамен

Ключевые слова try/except/finally. Инструкции raise/assert.			
10. Полезные модули для языка Python. Модули itertools, functools. Модули NumPy, SciPy. Модуль Matplotlib. Модуль nltk.	2 / 4 / 6	Простой анализ данных, работа с библиотеками Python	Текущий опрос, лаб 2, экзамен

Формы учебных занятий:

лк – лекции в аудитории;

см - семинары/ практические занятия/ лабораторные работы в аудитории;

ср – самостоятельная работа студента.

Тема 1. Обзор курса. Язык программирования Python. Особенности и области применения языка Python. Запуск программ на языке Python с помощью интерпретатора и с использованием IDE.

Обзор структуры курса, систем оценки. Обзор языка программирования Python. Литература, обучающие ресурсы. История языка Python. Где и кем используется язык Python. Сильные и слабые стороны языка Python. Две версии языка 2.* и 3.*. Интерпретатор языка Python: как выполняются и запускаются программы. IDE для языка Python. Zen of Python.

Тема 2. Обзор базовых типов данных языка Python. Функции type, dir, help. Неизменяемые типы данных. Числовые типы. Встроенные функции и модули для работы с числовыми типами. Строковый тип данных. Встроенные функции и модули для работы со строками.

Первая программа на языке Python. Функция print. Вторая программа на языке Python. Функция input, инструкция import. Несколько слов о стандартной библиотеке языка Python. Обзор базовых типов данных языка Python. Функции type, dir, help. Числовые типы данных. Арифметические операторы, операторы бинарных сдвигов. Функции pow, abs, round и т.д. Работа с системами счисления. Модули для работы с числами: random, math. Сравнения. Смешивание числовых типов. Типы Decimal и Fraction. Строковый тип данных. Экранирование. Базовые операции со строками: len, slicing, in, конкатенация и повторение. Преобразования, функции ord и chr. Методы строкового типа для работы с регистром: lower(), upper(), casefold(), swapcase() и т.д.

Методы строкового типа для работы с подстроками: startswith(), endswith(), count() и т.д.

Методы строкового типа для простого анализа строк: isalpha(), isnum(), islower() и т.д.

Форматирование строк.

Тема 3. Концепция неизменяемости в типах данных языка Python. Создание переменных (имен). Перекрестные ссылки. Изменяемые типы данных. Списки. Кортежи.

Концепция неизменяемости в типах данных языка Python. Процесс создания переменных (имен). Перекрестные ссылки. Модуль dis. Списки. Литералы списков, создание списков. Базовые операции со списками: len, конкатенация, повторение, in, slicing. In-place изменения в списках. Методы append(), extend(), insert(), remove(), перегруженные операторы. Инструкция del. Сортировка списков. Метод sort и функция sorted(). Методы reverse(), in-

dex(), count(). List comprehensions. Кортежи. Литералы кортежей, создание кортежей. . Базовые операции с кортежами.

Тема 4. Изменяемые типы данных. Словари. Множества. Файлы. Bool. NoneType. Пример простой программы на языке Python для визуализации дружб в социальной сети VK и проверки теории шести рукопожатий.

Изменяемые типы данных. Словари. Литералы словарей, создание словарей. Базовые операции со словарями: создание элемента словаря, получение по ключу, len, in. Создание, изменение записей в словарях. Инструкция del. Методы словарей: keys(), values(), items(), get(), update(), pop(). Dict comprehensions. Множества. Базовые операции с множествами. Методы add(), remove(), update(). Файлы. Функция open(). Методы write(), close(). Контекстные менеджеры. Bool. NoneType. Графы дружб на примере Facebook. Small-world experiment. Six degrees of separation. Эксперимент дружб участников майнора с известными людьми России через социальную сеть VK.

Тема 5. Выражения и синтаксис языка Python. Условный оператор if. Оператор присваивания и его формы. Присваивание последовательностей. Дополнительные варианты оператора присваивания.

Подробный разбор программы по проверке дружб в социальной сети VK. Выражения и синтаксис языка Python. Различия и преимущества синтаксиса языка Python. Правила именования переменных. Присваивания. Формы присваивания. Присваивания последовательностей. Дополнительные формы присваивания последовательностей. Множественные присваивания. Составные операции присваивания. Условный оператор if. Имитация множественного выбора в языке Python. Истинность выражений и объектов в Python.

Тернарный оператор.

Тема 6. Выражения и синтаксис языка Python. Операторы циклов for и while. Вложенные циклы. Инструкции break и continue. Функции range, enumerate.

Циклы в программировании. Циклы while. Общий вид цикла while. Бесконечный цикл. Вспомогательные инструкции в циклах – break, continue, else блок. Ключевое слово break. Ключевое слово continue. Опциональный else блок. Циклы for. Общий вид цикла for. Присваивание последовательностей в цикле for. Вложенные циклы. Функция range. Функция enumerate.

Тема 7. Функции в языке Python. Ключевое слово def. Области видимости. Локальные переменные, ключевые слова global, nonlocal. Передача аргументов. Упаковка и распаковка аргументов.

Функция как элемент программы. Функции в языке Python. Ключевое слово def. Объявление функций с помощью ключевого слова def. Инструкция return. Определение функции и вызов функции. Функции как объекты. Локальные переменные. Области видимости. Правило LEGB. Соккрытие имен. Ключевые слова global и nonlocal. Передача аргументов в функции. Аргументы и изменяемые объекты. Возвращение результатов функции. Позиционные аргументы и keywords аргументы. Аргументы по умолчанию. Упаковка и распаковка аргументов.

Тема 8. Функции в языке Python. Функциональное программирование и ООП. Анонимные функции, ключевое слово lambda. Элементы функционального программирования - map, reduce, filter. Итераторы и генераторы. Модуль itertools.

Две популярные парадигмы программирования – функциональное программирование и объектно-ориентированное. Особенности функциональной парадигмы. Обзор элементов функционального программирования в Python. Анонимные функции в языке Python. Опе-

ратор lambda. Функция map. Функция filter. Функция reduce. Итераторы и генераторы. Функции iter() и next(). Генераторы ключевое слово yield. Выражения-генераторы. Модуль itertools.

Тема 9. Объектно-ориентированное программирование в Python. Классы и ключевое слово class. Наследование. Перегрузка операторов. Исключения в языке Python. Ключевые слова try/except/finally. Инструкции raise/assert.

Основы объектно-ориентированного программирования. Классы и объекты. Проектирование класса. Ключевое слово class. Метод __init__(), аргумент self. Наследование. Перегрузка операторов. Исключения в языке Python. Обзор инструкций, связанных с обработкой исключений. Встроенные исключения на примере IndexError. Иерархия исключений. Общий вид блока try/except. Блок finally. Инструкции raise/assert.

Тема 10. Полезные модули для языка Python. Модули itertools, functools. Модули NumPy, SciPy. Модуль Matplotlib. Модуль nltk.

Модуль itertools: count, cycle, chain, combinations, product, permutations. Модуль functools: partial. Модули numpy/scipy. Модуль matplotlib. Модуль nltk.

III. ОЦЕНИВАНИЕ

Критерии оценки знаний, навыков

Оценки по всем формам текущего контроля выставляются по 10-ти балльной шкале.

Порядок формирования оценок по дисциплине

с округлением до ближайшего целого числа традиционным способом (например, 3,4 округляется до 3, а 3,5 до 4).

Итоговая оценка за дисциплину рассчитывается следующим образом:

$$\text{Итоговая} = 0.5 * (0.4 * \text{Лаб1} + 0.6 * \text{Лаб2} + \text{Бонус}) + 0.5 * \text{Экзамен}$$

На экзамене каждый студент получает два теоретических вопроса.

Важные примечания.

В случае болезни или отсутствия на контрольной работе по уважительной причине, студенту нужно предупредить преподавателя, проводящего контрольную работу, до ее начала.

Повторное проведение контрольных работ для студентов, пропустивших их **по уважительной причине не производится**, но вес контрольной работы переносится на проектную работу. В случае пропуска контрольной работы по неуважительной причине и/или отсутствия соответствующих документов повторное проведение контрольной работы не производится и в качестве оценки выставляется 0 баллов.

Пересдача элементов текущего контроля (контрольных работ) не производится.

Блокирующие элементы не предусмотрены.

IV. ПРИМЕРЫ ОЦЕНОЧНЫХ СРЕДСТВ

Образец экзаменационного задания

Билет 1.

Вопрос 1. Структура данных список. Основные операции со списками. Примеры.

Вопрос 2. Передача аргументов в функцию. Упаковка и распаковка аргументов.

Билет 2.

Вопрос 1. Модуль itertools. Функции модуля и примеры их применения.

Вопрос 2. Элементы ООП в языке Python. Ключевое слово class.

Образец лабораторной работы

Лабораторная работа №1

В данной лабораторной работе вам необходимо реализовать простой телефонный справочник.

Телефонным справочником является файл, хранящийся на жестком диске, который содержит информацию, соотносящую идентификатор человека (например, имя) с некоторыми данными о нем: номер телефона, дата рождения и т.д.

Необходимо разработать программу, которая будет предоставлять консольный интерфейс для работы с этим файлом.

Каждая запись в справочнике должна обладать следующими полями:

- Имя (только латинские буквы, цифры и пробелы, первая буква – заглавная)
- Фамилия (только латинские буквы, цифры и пробелы, первая буква – заглавная)
NB: Пара (Имя, Фамилия) является уникальным идентификатором записи в справочнике.
- Номер телефона (состоит из 11 цифр без знака +)
NB: при заполнении обязательна проверка на корректность номера и автозамена «+7» в начале номера на «8».
- Дата рождения (полная – число, месяц и год). Поле является опциональным, т.е. его заполнение не обязательно, оно может оставаться пустым.
NB: при заполнении обязательна проверка на корректность даты.

Консольный интерфейс:

- Под консольным интерфейсом понимается следующее: в интерпретаторе/командной строке/терминале запускается на выполнение программа (скрипт), которая предлагает пользователю конечный набор команд для работы со справочником.
- При вводе определенной команды (например, «добавить новую запись») программа выполняет соответствующие действия, необходимые для реализации этой команды.
- После выполнения введенной команды программа снова предлагает пользователю выбрать команду и ожидает ввода.
- Программа должна заканчивать свое выполнение только после ввода специальной команды (например, «quit»).

Требования к программе:

- 1) При запуске программы (скрипта), она должна считывать содержимое телефонного справочника из файла, хранящегося на жестком диске. Файл может как содержать записи (в корректном виде), так и быть пустым.
- 2) При завершении работы программы и/или при завершении выполнении команды, изменяющей справочник, все изменения в справочнике должны быть сохранены в файл (тот, из которого справочник считывается в начале).
- 3) Минимальный набор поддерживаемых команд для работы со справочником:

- Просмотр всех записей справочника: вывод всего справочника так, чтобы было читабельно.
 - Поиск по справочнику.
Поиск может осуществляться по любому из полей, а также по нескольким полям одновременно (например, найти запись с именем «А» и фамилией «Б»).
 - В результате поиска должны быть выведены найденные записи со значением полей.
 - Добавление новой записи.
NB: Обязательна проверка на то, что такая запись не содержится в справочнике (по уникальному идентификатору).
Если такая запись уже содержится в справочнике, сообщить об этом пользователю и предложить: изменить существующую запись, изменить (Имя, Фамилия) новой записи или вернуться к выбору команды.
NB: При вводе Имени и Фамилии обязательна автозамена первой буквы на заглавную.
 - Удаление записи из справочника по Имени и Фамилии
 - Изменение любого поля в определенной записи справочника
 - Вывод возраста человека (записи) по Имени и Фамилии
- 4) Если пользователь неправильно ввел команду, программа должна проинформировать его о некорректном вводе и вернуться в состояние ожидания ввода команды.
- 5) Ни одна операция не должна заканчиваться ошибкой или вылетом программы при поступлении любых входных данных – либо операция должна быть корректно обработана, либо должно быть выведено сообщение об ошибке (желательно информативное), и программа возвращена в состояние ожидания ввода.

В случае идеального выполнения всех обязательных требований ставится оценка 7.

Дополнительные баллы можно получить за следующие пункты:

- Качественный и понятный для чтения код.
- Наличие объясняющей документации к программе: описание реализованных функций и возможностей программы, примеры использования (в формате .docx, .pdf или .pptx).
- Удобный интерфейс для взаимодействия, в котором довольно просто ориентироваться.
- Реализация дополнительных возможностей программы:
 - Возможность добавления несколько номеров телефона для одной записи с категоризацией: мобильный, рабочий, домашний и т.д.
 - Удаление по номеру телефона.
NB: обязательна проверка на наличие нескольких записей с таким номером. Предоставить выбор пользователю, какую именно запись или несколько удалить.
 - Поиск и вывод записей по дате рождения (день и месяц, год не учитывается).
 - Просмотр всех записей (только поля Имя и Фамилия), у которых день рождения в ближайший месяц (30 дней).

- Просмотр всех записей, которые старше / младше / ровно N лет (N задаётся пользователем).
- Дополнительные возможности программы и расширение функционала. От добавления новых полей и функций работы с ними, до графического интерфейса и использования базы данных для хранения словаря.

Успешное выполнение первых 3-х пунктов оценивается в 0.5 балла за каждый пункт. Реализация дополнительных возможностей программы оценивается в 1 балл за каждый пункт, кроме последнего. Ваши личные идеи будут оценены в зависимости от сложности задачи и качества реализации.

V. РЕСУРСЫ

5.1 Основная литература

1. Stephenson, Ben. The Python Workbook [Электронный ресурс]: A Brief Introduction with Exercises and Solutions / Ben Stephenson. БД Springer . Springer International Publishing Switzerland 2014, ISBN 978-3-319-14239-5 ISBN 978-3-319-14240-1 (eBook). Режим доступа: <https://proxylibrary.hse.ru:2184/book/10.1007/978-3-319-14240-1> - загл. с экрана.

5.2 Дополнительная литература

1. Sarkar, Dipanjan. Text Analytics with Python [Электронный ресурс] / Dipanjan Sarkar, БД Springer . Springer Science+Business Media New York 2016. ISBN-13 (pbk): 978-1-4842-2387-1 ISBN-13 (electronic): 978-1-4842-2388-8. Режим доступа: <https://proxylibrary.hse.ru:2184/book/10.1007/978-1-4842-2388-8> - загл. с экрана.

Дополнительная литература для самостоятельного изучения

1. Марк Лутц. Изучаем Python. 4-е издание. – Пер. с англ. – СПб.: Символ-Плюс, 2011.
2. Марк Лутц. Python. Карманный справочник. – Вильямс, 2015.
3. Марк Лутц. Изучаем Python. 4-е издание. – Пер. с англ. – СПб.: Символ-Плюс, 2011.
4. Mark Pilgrim. Dive Into Python 3. 2009
5. Zed A. Shaw. Learn Python the Hard Way: A Very Simple Introduction to the Terrifyingly Beautiful World of Computers and Code, 3rd Edition. 2013
6. <http://www.swaroopch.com/notes/python/>
7. <http://pythontutor.ru/>
8. <https://ru.coursera.org/learn/python-osnovy-programmirovaniya>
9. <https://leetcode.com/>

5.3 Программное обеспечение

№ п/п	Наименование	Условия доступа
1.	Python Software Foundation Python	свободное лицензионное соглашение

**5.4 Профессиональные базы данных, информационные справочные системы,
интернет-ресурсы (электронные образовательные ресурсы)**

№ п/п	Наименование	Условия доступа
	<i>Профессиональные базы данных, информационно-справочные системы</i>	
1.	Электронно-библиотечная система Юрайт	Договор возмездного оказания услуг по подключению и обеспечению доступа к базе(-ам) данных № 77/3.1-04-19 от 18.04.2019
	<i>Интернет-ресурсы (электронные образовательные ресурсы)</i>	
1.	Открытое образование	URL: https://openedu.ru/

5.5 Материально-техническое обеспечение дисциплины

Для лекционных и семинарских занятий используется проектор.

6. Особенности организации обучения для лиц с ограниченными возможностями здоровья и инвалидов

В случае необходимости, обучающимся из числа лиц с ограниченными возможностями здоровья (по заявлению обучающегося) а для инвалидов также в соответствии с индивидуальной программой реабилитации инвалида, могут предлагаться следующие варианты восприятия учебной информации с учетом их индивидуальных психофизических особенностей, в том числе с применением электронного обучения и дистанционных технологий:

6.1.1. для лиц с нарушениями зрения: в печатной форме увеличенным шрифтом; в форме электронного документа; в форме аудиофайла (перевод учебных материалов в аудиоформат); в печатной форме на языке Брайля; индивидуальные консультации с привлечением тифлосурдопереводчика; индивидуальные задания и консультации.

6.1.2. для лиц с нарушениями слуха: в печатной форме; в форме электронного документа; видеоматериалы с субтитрами; индивидуальные консультации с привлечением сурдопереводчика; индивидуальные задания и консультации.

6.1.3. для лиц с нарушениями опорно-двигательного аппарата: в печатной форме; в форме электронного документа; в форме аудиофайла; индивидуальные задания и консультации.