

**ФИНАНСОВЫЙ УНИВЕРСИТЕТ ПРИ ПРАВИТЕЛЬСТВЕ
РОССИЙСКОЙ ФЕДЕРАЦИИ»
(Финансовый университет)**

Курский филиал федерального государственного образовательного бюджетного
учреждения высшего образования
Кафедра «Менеджмента и информационных технологий»

УТВЕРЖДАЮ

Директор филиала

Ю.В. Вертакова
«__31__»_08_____20__21__ г.

Приложение к рабочей программе дисциплины
Алгоритмы и структуры данных в языке Python

Направление подготовки 38.03.05 «Бизнес-информатика»

Профиль «ИТ- менеджмент в бизнесе»

Год утверждения программы: 2021

Одобрено кафедрой «Менеджмент и информационные технологии»

Протокол от «__31__»_08_____2021____. №_1_

Курск 2021

Содержание

1. Наименование дисциплины	4
2. Перечень планируемых результатов освоения образовательной программы (перечень компетенции) с указанием индикаторов их достижения и планируемых результатов обучения по дисциплине	4
3. Место дисциплины в структуре образовательной программы	6
4. Объем дисциплины(модуля) в зачетных единицах и в академических часах с выделением объема аудиторной (лекции, семинары) и самостоятельной работы обучающихся	7
5. Содержание дисциплины, структурированное по темам (разделам) дисциплины с указанием их объемов (в академических часах) и видов учебных занятий	7
5.1 Содержание дисциплины	7
5.2 Учебно-тематический план	12
5.3 Содержание практических занятий	13
6 Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине	14
6.1 Формы внеаудиторной самостоятельной работы	14
6.2 Перечень вопросов, заданий, тем для подготовки к текущему контролю .	15
7. Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине	17
7.1 Перечень компетенций с указанием индикаторов их достижения ¹ в процессе освоения образовательной программы	17
7.2 Типовые контрольные задания или иные материалы, необходимые для оценки индикаторов достижения компетенций, умений и знаний	18
7.3 Соответствующие приказы, распоряжения ректората о контроле уровня освоения дисциплин и сформированности компетенций студентов	21
8. Перечень основной и дополнительной учебной литературы, необходимой для	

освоения дисциплины	22
9. Перечень ресурсов информационно-телекоммуникационной сети.....	23
«Интернет», необходимых для освоения дисциплины.....	23
10. Методические указания для обучающихся по освоению дисциплины	24
11. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень необходимого программного обеспечения и информационных справочных систем	25
12 Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине.....	25

1. Наименование дисциплины

«Методы принятия управленческих решений»

2. Перечень планируемых результатов освоения образовательной программы (перечень компетенции) с указанием индикаторов их достижения и планируемых результатов обучения по дисциплине

Направление подготовки: 38.03.05 Бизнес-информатика. Профиль: ИТ-менеджмент в бизнесе. Форма обучения: очно-заочная. 2021 год приема

Таблица 1

Код компетенции	Наименование компетенции	Индикаторы достижения компетенции	Результаты обучения (владения, умения и знания), соотнесенные с компетенциями/индикаторами достижения
УК-4	Способность использовать прикладное программное обеспечение при решении профессиональных задач.	1. Использует основные методы и средства получения, представления, хранения и обработки данных.	Знать: Основные источники информации по программированию на языке Python, в том числе литературу и ресурсы в сети Интернет; Уметь: Использовать официальную документацию по языку и библиотекам Python; Владеть: Навыками анализа программного кода, выявления и исправления в нем ошибок
		2. Демонстрирует владение профессиональными пакетами прикладных программ.	Знать: Методику, правила установки и использования специализированных информационных систем и баз данных, при их техническом сопровождении. Уметь: Использовать специализированные информационные системы и базы данных Владеть: Практическими навыками работы с прикладными информационными системами.
		3. Выбирает необходимое прикладное программное обеспечение в зависимости от решаемой задачи.	Знать: Возможности современных и перспективных средств разработки программных продуктов, инструментальных сред автоматизации. Уметь: Использовать возможности существующей программно-

			технической архитектуры, использовать возможности современных и перспективных средств разработки программных продуктов. Владеть: Анализом возможностей реализации требований к программному обеспечению, оценка времени и трудоемкости реализации требований к программному обеспечению.
		4. Использует прикладное программное обеспечение для решения конкретных прикладных задач.	Знать: инструкции языка программирования; типы и структуры данных, используемые в языке; основные понятия процедурного, объектно-ориентированного и функционального программирования. Уметь: создавать собственные функции, классы, модули; Владеть: навыками использования интегрированных сред для создания программ; навыками решения практических задач с использованием высокоуровневых структур данных
УК-10	Способность осуществлять поиск, критически анализировать, обобщать и систематизировать информацию, использовать системный подход для решения поставленных задач	1. Четко описывает состав и структуру требуемых данных и информации, грамотно реализует процессы их сбора, обработки и интерпретации.	Знать: Методы и приемы формализации и алгоритмизации задач. Уметь: Использовать алгоритмы и структуры данных при решении профессиональных задач. Владеть: Навыками алгоритмизации решения профессиональных задач.
		2. Обосновывает сущность происходящего, выявляет закономерности, понимает природу variability.	Знать: Методы обоснования сущности происходящего. Уметь: Выявлять закономерности. Владеть: Пониманием природы variability.
		3. Формулирует признак классификации, выделяет соответствующие ему группы	Знать: Признаки классификации, общие свойства элементов групп и их

		однородных «объектов», идентифицирует общие свойства элементов этих групп, оценивает полноту результатов классификации, показывает прикладное назначение классификационных групп.	назначение. Уметь: Формулировать признак классификации, выделять соответствующие ему группы однородных «объектов». идентифицирует общие свойства элементов этих групп, оценивает полноту результатов классификации, показывает прикладное назначение классификационных групп. Владеть: Методикой идентифицирования общих свойств элементов этих групп, методикой оценки полноты результатов классификации, навыком поиска прикладного назначения классификационных групп.
		4. Грамотно, логично, аргументировано формирует собственные суждения и оценки. Отличает факты от мнений, интерпретаций, оценок и т.д. в рассуждениях других участников деятельности.	Знать: Понятия фактов, мнений, интерпретаций, оценок и их отличия. Уметь: Грамотно, логично, аргументировано формировать собственные суждения и оценки. Отличать факты от мнений, интерпретаций, оценок и т.д. в рассуждениях других участников деятельности. Владеть: Навыками поиска фактов, анализа мнений, интерпретаций и субъективных оценок.
		Аргументированно и логично представляет свою точку зрения посредством и на основе системного описания.	Знать: Методы системного описания языка Python. Уметь: Аргументированно и логично представлять свою точку зрения с помощью системного описания языка Python. Владеть: Навыком системного описания языка Python.

3. Место дисциплины в структуре образовательной программы

Дисциплина «Алгоритмы и структуры данных в языке Python» относится к циклу математики и информатики направления подготовки, образовательной

программы по направлению 38.03.05 Бизнес-информатика. Профиль: ИТ-менеджмент в бизнесе.

4. Объем дисциплины(модуля) в зачетных единицах и в академических часах с выделением объема аудиторной (лекции, семинары) и самостоятельной работы обучающихся

2021 год приема, очно-заочная форма обучения

Таблица 2

Вид учебной работы по дисциплине	Всего (в з/е и часах)	Семестр 1 (в часах)	Семестр 2 (в часах)
Общая трудоемкость дисциплины	6 з.е., 216 ч з.е	3 з.е., 216 ч з.е	3 з.е., 216 ч з.е
Контактная работа - Аудиторные занятия	28	12	16
<i>Лекции</i>	8	4	4
<i>Семинары, практические занятия</i>	20	8	12
Самостоятельная работа	188	96	92
Вид текущего контроля	<i>Контрольная работа (2)</i>	<i>Контрольная работа</i>	<i>Контрольная работа</i>
Вид промежуточной аттестации	<i>Зачет, экзамен</i>	<i>зачет</i>	<i>экзамен</i>

5. Содержание дисциплины, структурированное по темам (разделам) дисциплины с указанием их объемов (в академических часах) и видов учебных занятий

5.1 Содержание дисциплины

Тема 1. Языки и технологии программирования

Понятие языка программирования. Синтаксис и семантика языка. Способы реализации языков: компиляция, интерпретация, смешанный подход.

Обзор современных языков программирования, классификация и сравнительный анализ наиболее распространенных языков программирования. Основные парадигмы программирования (императивное программирование, декларативное программирование, объектно-ориентированное программирование, функциональное программирование) и их сравнительный анализ. Анализ сфер применения наиболее распространенных языков программирования.

Классификация и специфика применения языка программирования Python.

Язык программирования Python и его место среди других языков программирования. Версии языка программирования Python, дистрибутивы языка Python.

Установка Python, установка дистрибутива Anaconda. Работа в интерактивном режиме интерпретатора. Интерактивная оболочка IPython notebook: принципы работы и применения. Среда программирования. Использование документации.

Тема 2. Типы данных и операции

Работа с числами. Базовые числовые типы `int` и `float`. Числовые литералы. Операторы для работы с числовыми объектами. Форматы чисел. Встроенные функции и модули для работы с числами. Преобразование и смешивание в выражениях значений разных типов.

Строки. Литералы строк. Типы `str`, `bytes`, `bytearray`. Операции над строками: конкатенация, повторение, доступ по индексу, получение подстроки, проверка вхождения. Форматирование строк. Функции и методы для работы со строками. Регулярные выражения.

Списки. Создание списка. Операции над списками. Перебор элементов списка. Многомерные списки. Методы списков. Кортежи.

Словари. Создание словаря. Операции над словарями. Перебор элементов словаря. Методы для работы со словарями. Множества.

Переменные. Правила именования переменных. Присваивание значения переменным. Динамическая типизация. Понятие о счетчике ссылок и сборке мусора. Проверка и преобразование типов данных. Удаление переменной.

Структура программы. Комментарии. Блок. Правила оформления отступов. Инструкция присваивания. Групповое присваивание. Комбинированные инструкции присваивания. Инструкции выражений. Функция `print`. Операторы сравнения. Логические операторы. Инструкция ветвления `if...else`. Инструкция цикла `while`. Инструкции `break`, `continue`, `pass`, `else`. Инструкция цикла `for`.

Тема 3. Функции, модули, исключения

Функции. Создание функций. Область видимости переменной. Передача аргументов в функцию. Инструкция return. Вызов функции.

Необязательные параметры и сопоставление по ключам. Переменное число параметров в функции.

Лямбда-функции. Функции-генераторы. Инструкция yield. Вложенные функции. Аннотации функций.

Модули. Инструкции import и from. Повторная загрузка модулей.

Исключения. Инструкция try...except...else...finally. Получение информации об исключении. Классы встроенных исключений. Пользовательские исключения. Инструкции raise и assert.

Тема 4. Стандартная библиотека, написание сценариев

Обзор стандартной библиотеки Python. Встроенные функции Python. Работа с файлами: чтение, запись, операции с файлами и каталогами. Модули для взаимодействия с операционной системой: sys, os. Модули для математических операций и работы со случайными числами: math, random. Модули для работы с датой и временем: time, datetime, calendar.

Обзор библиотек для решения математических и научных задач (SymPy, NumPy, Matplotlib).

Организация файла сценария, переменная name. Контекст выполнения сценариев. Текущий рабочий каталог, файлы и путь поиска модулей. Анализ аргументов командной строки. Стандартные потоки ввода-вывода и перенаправление потоков ввода-вывода.

Тема 5. Объектно-ориентированное программирование

Базовые принципы объектно-ориентированного программирования: инкапсуляция, наследование, полиморфизм.

Классы в языке Python. Инструкция class. Создание экземпляра класса. Атрибуты класса и экземпляра класса. Методы класса. Конструктор и деструктор. Статические методы. Закрытые методы и атрибуты. Свойства.

Наследование. Множественное наследование. Абстрактные методы. Переопределение методов. Итераторы, контейнеры, перечисления.

Тема 6. Разработка графических интерфейсов

Событийно-ориентированное программирование. Событие. Обработчик события. Цикл обработки событий.

Инструменты для создания графических интерфейсов пользователя (GUI). Создание базового окна. Элементы графического интерфейса (виджеты). Создание и конфигурирование виджета. Менеджер размещения. Использование элементов Кнопка, Надпись, Текстовое поле, Флажок, Переключатель.

Тема 7. Элементы функционального и параллельного программирования

Функциональное программирование. Основные атрибуты функционального программирования и их поддержка в языке программирования Python.

Причины развития технологий параллельного программирования. Современное аппаратное обеспечение, закон Мура и актуальность технологий параллельного программирования. Ограничения в реализации параллельных вычислений, закон Амдала.

Способы синхронизации параллельного взаимодействия: взаимодействие через разделяемую память, взаимодействие с помощью передачи сообщений. Поддержка параллельных вычислений на уровне операционных систем: потоки и процессы.

Поддержка параллельных вычислений на уровне библиотек, языков программирования и их инфраструктуры. Низкоуровневые и высокоуровневые абстракции для параллельной обработки данных, модель распределенных вычислений map-reduce. Преимущества функционального программирования для построения программ с параллельными вычислениями.

Конструкция map-reduce в языке программирования Python: функции map(), filter(), reduce(). Модуль functools. Конструкции генераторов в Python, как элементы функционального программирования.

Проблема Global Interpreter Lock (GIL) в Python, пути ее решения. Модули поддержки многопоточные вычислений в Python: threading, thread, Queue.

Тема 8. Технологии обработки данных

Хранение и обмен структурированной информацией в виде документов или сообщений. Форматы представления переносимой структурированной информации. Сравнение различных принципов представления структурированной информации: закрытые и открытые форматы, бинарное и текстовое представление данных.

Универсальные форматы хранения структурированной информации (разметки документов): CSV, XML, HTML (XHTML), JSON. Язык разметки XML: основные принципы построения и специфика использования.

Построение схемы документа с помощью XML DTD или XML Schema. HTML (XHTML) - отличие от XML, специфика использования. Формат представления структурированной информации JSON: принципы построения, специфика использования.

Модули Python для хранения и обработки данных в форматах CSV, XML и JSON. Знакомство с библиотеками numpy и pandas и решением базовых задач подготовительных операций для выполнения анализа данных с помощью этих библиотек.

Тема 9. Технологии построения web-ориентированных систем

Знакомство с различными технологиями построения распределенных информационных систем: клиент-серверная архитектура, веб-ориентированные распределенные ИС, веб-сервисы, объектные распределенные системы, агентные технологии, технологии одноранговых (P2P) сетей, Grid технологии, облачные вычисления (PaaS, IaaS, SaaS(SOA)). Сравнительный анализ и области применения различных технологий построения распределенных информационных систем.

Стек технологий на базе языка программирования Python для построения веб-ориентированных распределенных ИС. Модули Python для создания веб-

сервисов и сервисов REST. Логика поведения и реализация распределенных информационных систем с использованием веб-технологий.

Определение PaaS, IaaS, SaaS(SOA), облачные вычисления. Сектор применения. Положительные и отрицательные стороны, экономические и организационные аспекты использования облачных вычислений. Облачные решения для разворачивания приложений на языке Python.

5.2 Учебно-тематический план

2021 г. приема, очно-заочная

Таблица 3

№ п/ п	Наименование тем (разделов) дисциплины	Трудоемкость в часах						Формы текущего контроля успеваемости
		Всего	Аудиторная работа				Самост- оятельная работа	
			Общая, в т.ч.:	Лекции	Семина ры, практич еские занятия	Занятия в интеракт ивных формах		
1	Языки и технологии программирования	27	3	1	2	-	24	УО, ППЗ
2	Типы данных и операции	27	3	1	2	-	24	УО, ППЗ
3	Функции, модули, исключения	27	3	1	2	-	24	УО, ППЗ, КР
4	Стандартная библиотека, написание сценариев	27	3	1	2	-	24	УО, ППЗ
-	ИТОГО (1й семестр)	108	12	4	8	-	96	
5	Объектно- ориентированное программирование	25	5	1	4	-	20	УО, ППЗ
6	Разработка графических интерфейсов	23	3	1	2	-	20	УО, ППЗ
7	Элементы функционального и параллельного программирования	22,5	2,5	0,5	2	-	20	УО, ППЗ, КР
8	Технологии обработки данных	23	3	1	2	-	20	УО, ППЗ
9	Технологии построения web- ориентированных систем	14,5	2,5	0,5	2	-	12	УО, ППЗ
	ИТОГО (2й семестр)	108	16	4	12	-	92	
	В целом по дисциплине	216		8	20	-	188	Контрольные работы
	Итого в %					-		

5.3 Содержание практических занятий

Целью проведения практических занятий является приобретение студентами навыков программирования на языке Python.

Темы практических занятий приведены в табл. 4.

В качестве интерактивной формы обучения используется метод мозгового штурма, дерево решений (для выбора наилучшего варианта), обсуждение в группах (группа из двух студентов).

Таблица 4

Наименование темы (раздела) дисциплины	Перечень вопросов для обсуждения на семинарских, практических занятиях, рекомендуемые источники из разделов 8,9	Формы проведения занятий
Семестр 1		
Тема 1. Языки и технологии программирования	Знакомство с интерактивной оболочкой IPython notebook. Изучение принципов работы в оболочке. [9.3]	Индивидуальное выполнение заданий, групповой разбор результатов выполнения заданий (30% времени на интерактивные технологии)
Тема 2. Типы данных и операции	Регулярные выражения. Картежи (создание, операции, итерация). Множества (создание, операции, итерация). [8.1; 8.2]	Индивидуальное выполнение заданий, групповой разбор результатов выполнения заданий (30% времени на интерактивные технологии)
Тема 3. Функции, модули, исключения	Необязательные параметры и сопоставление по ключам. Переменное число параметров в функции. Инструкция yield. Вложенные функции. Аннотации функций. Классы встроенных исключений. Пользовательские исключения. Инструкции raise и assert. [8.1; 8.2]	Индивидуальное выполнение заданий, групповой разбор результатов выполнения заданий (30% времени на интерактивные технологии)
Тема 4. Стандартная библиотека, написание сценариев	Библиотеки для решения математических и научных задач (SymPy, NumPy, Matplotlib). Модули для работы с датой и временем. Контекст выполнения сценариев. [9.2; 9.12; 9.13]	Индивидуальное выполнение заданий, групповой разбор результатов выполнения заданий (30% времени на интерактивные технологии)

Семестр 2		
Тема 5. Объектно-ориентированное программирование	Статические методы. Закрытые методы и атрибуты. Свойства. Множественное наследование. Абстрактные методы. Переопределение методов. [8.1; 8.2]	Индивидуальное выполнение заданий, групповой разбор результатов выполнения заданий (30% времени на интерактивные технологии)
Тема 6. Разработка графических интерфейсов	Менеджер размещения. Использование элементов Кнопка, Надпись, Текстовое поле, Флажок, Переключатель. [9.4; 9.11]	Индивидуальное выполнение заданий, групповой разбор результатов выполнения заданий (30% времени на интерактивные технологии)
Тема 7. Элементы функционального и параллельного программирования	Способы синхронизации параллельного взаимодействия: взаимодействие через разделяемую память, взаимодействие с помощью передачи сообщений. Поддержка параллельных вычислений на уровне операционных систем: потоки и процессы. Модули поддержки многопоточных вычислений в Python: threading, thread, Queue. [8.1; 8.2]	Индивидуальное выполнение заданий, групповой разбор результатов выполнения заданий (30% времени на интерактивные технологии)
Тема 8. Технологии обработки данных	HTML (XHTML), JSON. Модули Python для обработки данных в формате JSON. [9.3; 9.11]	Индивидуальное выполнение заданий, групповой разбор результатов выполнения заданий (30% времени на интерактивные технологии)
Тема 9. Технологии построения web-ориентированных систем	Модули Python для создания веб-сервисов и сервисов REST. [9.3; 9.11]	Индивидуальное выполнение заданий, групповой разбор результатов выполнения заданий (30% времени на интерактивные технологии)

6 Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине

6.1 Формы внеаудиторной самостоятельной работы

Таблица 5

Наименование тем (разделов) дисциплины	Перечень вопросов, отводимых на самостоятельное освоение	Формы внеаудиторной самостоятельной работы
Языки и технологии программирования	1) Знакомство с интерактивной оболочкой IPython notebook. Изучение принципов работы в оболочке.	РЛ, РЭИ, РАП

Типы данных и операции	1) Регулярные выражения. 2) Картежи (создание, операции, итерация). 3) Множества (создание, операции, итерация).	РЛ, РЭИ, РАП
Функции, модули, исключения	1) Необязательные параметры и сопоставление по ключам. Переменное число параметров в функции. 2) Инструкция yield. Вложенные функции. Аннотации функций. 3) Классы встроенных исключений. Пользовательские исключения. Инструкции raise и assert.	РЛ, РЭИ, РАП
Стандартная библиотека, написание сценариев	1) Библиотеки для решения математических и научных задач (SymPy, NumPy, Matplotlib). 2) Модули для работы с датой и временем. 3) Контекст выполнения сценариев.	РЛ, РЭИ, РАП
Объектно-ориентированное программирование	1) Статические методы. Закрытые методы и атрибуты. Свойства. 2) Множественное наследование. Абстрактные методы. Переопределение методов.	РЛ, РЭИ, РАП
Разработка графических интерфейсов	1) Менеджер размещения. Использование элементов Кнопка, Надпись, Текстовое поле, Флажок, Переключатель.	РЛ, РЭИ, РАП
Элементы функционального и параллельного программирования	1) Способы синхронизации параллельного взаимодействия. 2) Модули поддержки многопоточных вычислений в Python: threading, thread, Queue.	РЛ, РЭИ, РАП
Технологии обработки данных	1) HTML (XHTML), JSON. 2) Модули Python для обработки данных в формате JSON.	РЛ, РЭИ, РАП
Технологии построения web-ориентированных систем	1) Модули Python для создания веб-сервисов и сервисов REST.	РЛ, РЭИ, РАП

* Сокращения в таблице: **НТ** – Номера тем; **ФВСТ** – Формы внеаудиторной самостоятельной работы; **Т/Ч** – Трудоемкость в часах; **РЛ** – работа с литературой; **РЭИ** – работа с электронными источниками; **РАП** – разработка алгоритмов и программ.

6.2 Перечень вопросов, заданий, тем для подготовки к текущему контролю

Контрольная работа № Перечень вопросов к контрольной работе

1. Инструкции языка Python: присваивание, ветвление, циклы.
2. Структуры данных строка, список, словарь.
3. Создание и использование функций.
4. Текстовые файлы.

Пример задания

В текстовом редакторе создайте текстовый файл Товары.txt содержащий строки вида «Яблоки, 20, 95», где 20 – это количество товара, 95 – цена единицы товара. Используя данные этого файла создайте словарь, у которого ключом является название товара, значением – кортеж из двух чисел (количество и цена товара).

Напишите функцию, которая вычисляет стоимость заказа. Параметрами функции являются: список кортежей из двух элементов (товар, количество) и словарь, содержащий информацию о товарах. В случае наличия достаточного количества товара функция возвращает стоимость заказа и уменьшает общее количество товара (в словаре) на количество товара в заказе. При отсутствии товара или его недостаточного количества функция не изменяет общее количество товара и возвращает 0.

Продемонстрируйте работу с функцией для 3 заказов. Выведите на экран значения словаря после его формирования и в конце после обработки заказов.

Контрольная работа № Перечень вопросов к контрольной работе

1. Создание классов.
2. Специальные методы класса.
3. Реализация связанных списков.
4. Понятие стека и очереди.

Пример задания

Создайте класс Queue, реализующий работу с очередью. Для хранения элементов очереди используйте связный список. Элемент связного списка хранится в виде списка. В классе должны быть:

- конструктор с параметрами. При создании очереди ее можно инициализировать произвольным количеством значений (как минимум одним).

- метод `push(self, *value)` – добавляет указанные элементы в очередь (как минимум одно);

- метод `pop(self)` – удаляет элемент из очереди и возвращает его значение. Если очередь пуста, то возвращает `None`;

- метод `top()` – возвращает элемент, находящийся в начале очереди, элемент из очереди не удаляется. Если очередь пуста, то возвращает `None`;

- метод `empty()` – удаляет все элементы из очереди;

- методы `__next__(self)` и `__iter__(self)`;

- метод `__str__(self)` – возвращает строку вида `s1<-s2<-...<-sn`;

- свойство `size` – возвращает количество элементов в очереди;

- свойство `isempty` – возвращает `True`, если очередь не содержит элементов.

Продемонстрируйте работу с классом, создав объект и вызвав все методы. Файл с текстом программы должен содержать необходимые комментарии.

Критерии балльной оценки различных форм текущего контроля успеваемости содержатся в соответствующих документах Департамента анализа данных, принятия решений и финансовых технологий.

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

7.1 Перечень компетенций с указанием индикаторов их достижения¹ в процессе освоения образовательной программы

Перечень компетенций, формируемых в процессе освоения дисциплины содержится в разделе 2.

7.2 Типовые контрольные задания или иные материалы, необходимые для оценки индикаторов достижения компетенций, умений и знаний

Наименований компетенций	Примеры типовых заданий для оценки уровня сформированности компетенций
УК-4 Способность использовать основные законы естественнонаучных дисциплин и современные информационно-коммуникационные технологии в профессиональной деятельности	Задание 1 Напишите функцию, которая находит максимум функции $f(x)$ в точках отрезка $[a, b]$ с постоянным шагом h. Параметрами функции являются f, a, b, h. Параметры a, b, h – необязательные, по умолчанию $a=0, b=1, h=0.1$. Используя эту функцию, найдите максимум функции $(2-x)\sin(x/2)$ на отрезке $[0, 4]$. Задание 2 Сформируйте N различных паролей длиной M символов, состоящих из строчных и прописных латинских букв, и цифр, кроме тех, которые легко перепутать между собой: «l» (L маленькое), «I» (i большое), «1» (цифра), «o» и «O» (большая и маленькая буквы) и «0» (цифра). В каждом пароле обязательно должна присутствовать хотя бы одна цифра и как минимум по одной букве в верхнем и нижнем регистре. Указание: воспользуйтесь функциями библиотеки <code>random</code>. Задание 3 Создайте класс <code>Ведомость</code>, имеющий свойства: • дисциплина, • группа. методы: • <code>put</code> – добавляет в ведомость информацию об оценке студента (фамилия, оценка – параметры метода). Для хранения данных внутри класса используйте словарь, в котором ключом является фамилия студента. Возможные оценки – «отлично», «хорошо», «удовл.», «неудовл.», «н/я»; • <code>get</code> – возвращает оценку, полученную студентом (фамилия студента – параметр метода); • <code>change</code> – изменяет оценку, полученную студентом (фамилия студента и новая оценка – параметры метода); • <code>del</code> – удаляет информацию о студенте из ведомости (фамилия студента – параметр метода); • <code>result</code> – возвращает кортеж из 5 чисел (количество каждого вида оценок в ведомости); • <code>__init__</code> – конструктор; • <code>__str__</code> – возвращает строку, содержащую заголовки (название экзамена, группа) и результаты экзамена в виде таблицы; • <code>count</code> – возвращает количество студентов в ведомости; • <code>names</code> – возвращает список фамилий, имеющих в ведомости. Продемонстрируйте работу с классом.
УК-10 Способность решать стандартные задачи профессиональной деятельности на основе информационной и библиографической	Задание 1 Установите на домашний компьютер пакет <code>Anaconda</code>. Используя документацию, научитесь создавать и запускать программы в среде <code>Jupyter Notebook</code>. Задание 2 Используя документацию, научитесь создавать и запускать программы в среде <code>Spyder</code>.

культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности	Задание 3 Изучите возможности по отладке программ в среде Spyder, Задание 4 Используя отладчик, найдите ошибку в программе, которая вычисляет максимальный отрицательный элемент списка: <pre>num = [1, -7, -1, 3, 5, -2] mx = num[0] for x in num: if x<0 and x>mx: mx = x print(mx)</pre>
---	---

Примеры тестовых заданий

1. Преобразование строки B='1+2+7' в список [1, 2, 7] можно выполнить с помощью выражения

- a) int(B.split('+'))
- b) map(int, B.split('+'))
- c) list(map(int, B.split('+')))
- d) list(map(int, '+' .join(B)))

2. Отметьте правильные операторы, в которых переменной D в качестве значения присваивается словарь a) D = { 1 : "a", 2 : ("ab", "ac") }

- a) D = { [1] : "a", [2] : "b" }
- b) D = tuple()
- c) D = { }

3. Инструкция

s = fl.read() считает из файла

- a) один символ
- b) одну строку
- c) все символы до конца файла
- d) инструкция содержит ошибку, поэтому выполнена не будет

Теоретические вопросы для подготовки к зачету, экзамену

1. Компилятор и интерпретатор. Достоинства и недостатки.

2. Назовите и дайте краткую характеристику основных классов языков программирования.

3. Встроенные числовые типы языка Python.
4. Списки. Создание, основные операции.
5. Основные методы списка.
6. Кортежи. Создание, основные методы и операции.
7. Словари. Создание, основные операции.
8. Методы для работы со словарями.
9. Множества. Создание, основные методы и операции.
10. Переменные. Правила именования переменных.
11. Динамическая типизация.
12. Операторы сравнения и логические операторы.
13. Инструкция if...else.
14. Инструкция цикла while.
15. Инструкция цикла for.
16. Создание и вызов функции.
17. Передача аргументов функцию.
18. Функции-генераторы.
19. Лямбда-функции.
20. Модули. Инструкции import и from.
21. Базовые принципы объектно-ориентированного программирования.
22. Класс, метод класса, атрибут класса. Определение класса и создание экземпляра класса.
23. Конструктор и деструктор.
24. Наследование.
25. Абстрактные методы класса.
26. Статические методы класса.
27. Свойства класса.
28. Исключения. Обработка исключений.

- 29. Пользовательские исключения.
- 30. Событие. Обработчик события. Цикл обработки событий.
- 31. Элемент Кнопка. Создание и настройка.
- 32. Элемент Кнопка. Создание обработчика события.
- 33. Элементы Надпись и Текстовое поле. Создание и настройка. Метод `get()`.
- 34. Элемент Флажок. Создание, настройка, получение статуса флажка.
- 35. Элемент переключатель. Создание, настройка, доступ к значению.
- 36. Классы `date`, `time` и `datetime`.
- 37. Возможности библиотеки `SymPy`. 38. Возможности библиотеки `NumPy`.

Пример экзаменационного билета

Экзаменационный билет №

Присвоение по ссылке и по значению. Специфика создания объектов и присвоения в Python. Динамическая типизация в Python. (20 баллов)

Вычислите значение числа e с точностью 10^{-4} , используя разложение в ряд: $e = 2 + 1/2! + 1/3! + 1/4! + \dots$

Выведите на экран полученное значение и значение числа e из стандартной библиотеки с 5 знаками после запятой. (20 баллов)

При помощи механизма `map/filter/reduce` превратить список целых чисел в строку, содержащую строковое представление этих чисел, разделенное пробелами. (20 баллов)

7.3 Соответствующие приказы, распоряжения ректората о контроле уровня освоения дисциплин и сформированности компетенций студентов

Промежуточная аттестация по дисциплине «Алгоритмы и структуры данных в языке Python» проводится в форме зачета в первом семестре и экзамена во втором

семестре и оценивается по 100-бальной шкале в соответствии с Балльно-рейтинговой системой Финансового университета (приложение 1 к приказу от 2 октября 2013 г. № 1616/0).

Ориентировочное распределение максимальных баллов по видам работы приведено в таблице 8.

Таблица 8

№ п/п	Вид отчетности	Баллы
1	<i>Работа в семестре 1</i>	40
1.1	- выполнение и защита контрольной работы	30
1.2	- активное участие в дискуссии, ролевой игре, дебатах, защите экспертных заключений (не в качестве защищающегося)	10
2	<i>Зачет</i>	60
3	Итого:	100
1	<i>Работа в семестре 2</i>	40
1.1	- выполнение и защита контрольной работы	30
1.2	- активное участие в дискуссии, ролевой игре, дебатах, защите экспертных заключений (не в качестве защищающегося)	10
2	<i>Экзамен</i>	60
3	Итого:	100

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

1. Сузи Р.А. Язык программирования Python [Электронный ресурс]: курс / Р.А. Сузи. - 2-е изд., испр. - Москва: Интернет-Университет Информационных Технологий, 2007. - 327 с. - Режим доступа: <http://biblioclub.ru/index.php?page=book&id=233288>.

2. Северенс Ч. Введение в программирование на Python [Электронный ресурс] / Ч. Северенс. - 2-е изд., испр. - Москва: Национальный Открытый

Университет «ИНТУИТ», 2016. - 231 с. [Электронный ресурс]. - Режим доступа: <http://biblioclub.ru/index.php?page=book&id=429184>.

9. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины

1. PyLru 1.0.9 [Электронный ресурс]: сайт. – Режим доступа: <https://pypi.python.org/pypi/pylru>
2. Python Data Analysis Library [Электронный ресурс]: сайт. – Режим доступа: <http://pandas.pydata.org/>
3. Python Documentation [Электронный ресурс]: сайт. – Режим доступа: <http://python.org/doc/>
4. The Python Tutorial // <https://docs.python.org/3/tutorial/index.html>.
5. The Python Standard Library //
6. <https://docs.python.org/3/library/index.html>.
7. Python Standard Library [Электронный ресурс]: сайт. – Режим доступа: <https://docs.python.org/2/library/>
8. Scikit-learn Machine Learning in Python [Электронный ресурс]: сайт. – Режим доступа: <http://scikit-learn.org>
9. Официальный сайт продукта <https://www.python.org/>
10. Портал Финансового университета <http://www.fa.ru/>
11. Каталог курсов Интернет Университета Информационных Технологий <http://www.intuit.ru/>
12. SciPy // <http://docs.scipy.org/doc/scipy/reference/>
13. NumPy User Guide // <http://docs.scipy.org/doc/numpy/user/index.html>
14. Электронная библиотека Финансового университета (ЭБ) <http://elib.fa.ru/> (<http://library.fa.ru/files/elibfa.pdf>)
15. Электронно-библиотечная система BOOK.RU <http://www.book.ru>
16. Электронно-библиотечная система «Университетская

библиотека ОНЛАЙН» <http://biblioclub.ru/>

17. Электронно-библиотечная система Znanium
<http://www.znaniy.com>

18. «Деловая онлайн библиотека» издательства «Альпина Паблишер» <http://lib.alpinadigital.ru/en/library>

19. Электронно-библиотечная система издательства «Лань»
<https://e.lanbook.com/>

20. Электронно-библиотечная система издательства «ЮРАЙТ»
<https://www.biblio-online.ru/>

21. Научная электронная библиотека eLibrary.ru <http://elibrary.ru>

10. Методические указания для обучающихся по освоению дисциплины

10.1. Весь лекционный материал содержится на преподавательском диске. Однако при изложении лекции используется проблемный подход, что значительно расширяет предоставленный материал. Кроме текстов лекций, на преподавательском диске также находятся материалы практических занятий, разбитых по темам. Там же приведены постановки задач, образцы программ решения типовых задач и справочные материалы.

Для получения доступа к облачному хранилищу студенты должны получить соответствующую ссылку от преподавателя.

10.2. При переходе к новой теме проводится тестирование, направленное на оценивание теоретических знаний. Помимо тестирования, может проводиться выборочный устный опрос студентов. Полученные оценки участвуют в формировании итоговой оценки по дисциплине.

10.3. Практические навыки оцениваются путем разработки прикладных программ. Студенты должны самостоятельно и вовремя решать поставленные преподавателем задачи. Преподаватель должен отмечать и поощрять наиболее исполнительных студентов.

11. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень необходимого программного обеспечения и информационных справочных систем

11.1 Комплект лицензионного программного обеспечения:

Windows Microsoft Office,

Антивирус ESET Endpoint Security

11.2 Современные профессиональные базы данных и информационные справочные системы:

Справочная правовая система «КонсультантПлюс» (<http://www.consultant.ru>).

Справочная правовая система «Гарант» (<http://www.garant.ru>).

Информационно-образовательный портал Финансового университета. - <http://portal.ufrf.ru>.

11.3 Сертифицированные программные и аппаратные средства защиты информации – не предусмотрено.

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

Лекционные и практические занятия проводятся в аудиториях, оборудованных мультимедийной системой. Для проведения семинарских занятий, предполагающих свободный доступ студентов к ресурсам сети Интернет, необходимо наличие в аудитории стандартного беспроводного доступа в Интернете.