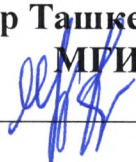


**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ ИНСТИТУТ
МЕЖДУНАРОДНЫХ ОТНОШЕНИЙ (УНИВЕРСИТЕТ)
МИНИСТЕРСТВА ИНОСТРАННЫХ ДЕЛ
РОССИЙСКОЙ ФЕДЕРАЦИИ»
ТАШКЕНТСКИЙ ФИЛИАЛ**

УТВЕРЖДАЮ

**Директор Ташкентского филиала
МГИМО МИД России**

М.Т. Бакоев
«31» мая 2021 г.

Рабочая программа дисциплины

**РАЗРАБОТКА ИНФОРМАЦИОННО-АНАЛИТИЧЕСКИХ СИСТЕМ НА
ОСНОВЕ ЯЗЫКА PYTHON**

**Направление подготовки
38.04.05 Бизнес-информатика**

**Направление (профиль) подготовки
Системный анализ в международном бизнесе
)**

Квалификация – *Магистр*

Форма обучения – *очная*

Ташкент - 2021

Рабочая программа по дисциплине «Разработка информационно-аналитических систем на основе языка Python» составлена в соответствии с требованиями образовательного стандарта высшего образования МГИМО по направлению подготовки 38.04.05 Бизнес-информатика.

Автор программы: Погребняк Евгений Владимирович, доцент, кандидат экономических наук

Библиотекарь: _____



С.К. Атаханова

Содержание

1. Перечень планируемых результатов обучения по дисциплине (модулю), соотнесенных с планируемыми результатами освоения образовательной программы	4
2. Место дисциплины в структуре образовательной программы	8
3. Объем дисциплины (модуля) в зачетных единицах с указанием количества академических часов, выделенных на контактную работу обучающихся с преподавателем (по видам занятий) и на самостоятельную работу обучающихся	8
4. Содержание дисциплины (модуля), структурированное по темам (разделам) с указанием отведенного на них количества академических часов и видов учебных занятий	9
5. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине (модулю).....	12
6. Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине (модулю).....	13
7. Перечень основной и дополнительной литературы, необходимой для освоения дисциплины (модуля)	31
8. Методические указания для обучающихся по освоению дисциплины (модуля).....	33
9. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине (модулю), включая перечень программного обеспечения и информационных справочных систем (при необходимости)	33
10. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине (модулю).....	33
11. Иные сведения и материалы.....	34
12. Лист регистрации внесенных изменений.....	35

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

Цель дисциплины:

- приобретение комплекса теоретических знаний и методологических основ в области применения методов и моделей для проектирования, разработки и эксплуатации информационно-аналитических систем, обеспечивающих информационную поддержку принятия решений стратегического, тактического и операционного уровней на основе языка Python 3.x;
- практическое освоение слушателями сущности концепции управления эффективностью СРМ (Corporate Performance Management) и вопросов, связанных с автоматизацией процессов стратегического управления на основе сбалансированной системы показателей эффективности, процессов планирования, бюджетирования, консолидации и корпоративной отчетности, а также реализация проектов в области СРМ на базе современных паттернов на языке Python 3.x.

Задачи дисциплины:

- изучение современных технологий, систем и платформ, позволяющих внедрить в систему менеджмента предприятия инновационные методологии управления на основе языка Python 3.x;
- привитие навыков разработки архитектуры информационно-аналитических систем, разработки и применения соответствующих инструментальных средств на языке Python 3.x.

В результате освоения образовательной программы обучающийся должен овладеть следующими результатами обучения по дисциплине (модулю):

Коды компетенции	Содержание компетенций	Индикаторы достижения компетенций	Перечень планируемых результатов обучения по дисциплине
ПК-9	Способен к организации исследования лучших мировых практик и бизнес-процессов в области управления продуктами в ИТ-инфраструктуре международного бизнеса	ПК-9.1: Способен ставить задачи по методическому описанию бизнес-процессов в области управления продуктами в ИТ-инфраструктуре международного бизнеса.	Знания: <ul style="list-style-type: none"> - методы анализа и моделирования бизнес-процессов; - теорию бизнес-процессов; - принципы структурирования организации; - методологии структурного анализа и современные методологии моделирования; - инструментальные системы, используемые для описания бизнес-процессов; - основные принципы анализа бизнес-процессов; - информационные системы документооборота в ИТ-инфраструктуре международного бизнеса. Умения: <ul style="list-style-type: none"> - оформлять аналитические материалы по результатам исследований, проводимых в области ИКТ. Навыки: <ul style="list-style-type: none"> - ставить задачи по методическому описанию бизнес-процессов в области управления продуктами в ИТ-инфраструктуре международного бизнеса.

		<p>ПК-9.2: Способен организовать командную работу в области управления продуктами в ИТ-инфраструктуре международного бизнеса.</p>	<p>Знания:</p> <ul style="list-style-type: none"> - современные концепции управления бизнес-процессами; - основы организационной структуры; - основные математические методы и модели, используемые при организационном управлении бизнес-процессами; - последствия принимаемых организационно-управленческих решений; - факторы, влияющие на силу конкурентной борьбы; - принципы и методику проектирования процессов межличностных, групповых и организационных коммуникаций. <p>Умения:</p> <ul style="list-style-type: none"> - оценивать эффективность вариантов построения процессов межличностных, групповых и организационных коммуникаций в области управления продуктами в ИТ-инфраструктуре международного бизнеса; - обобщать полученные данные в своей научной и практической деятельности. <p>Навыки:</p> <ul style="list-style-type: none"> - применение систем поддержки принятия решений для организации командной работы в области управления продуктами в ИТ-инфраструктуре международного бизнеса; - разработка организационно-управленческих решений по проектированию коммуникативных бизнес-процессов в организации.
		<p>ПК-9.3: Оценивает и контролирует качество процессов управления ИТ-инфраструктурой.</p>	<p>Знания:</p> <ul style="list-style-type: none"> - принципы сбора, анализа и обработки данных, необходимых для управления ИТ-инфраструктурой. <p>Умения:</p> <ul style="list-style-type: none"> - выполнять технико-экономическое обоснование проектов по совершенствованию и регламентации управления ИТ-инфраструктурой организации. <p>Навыки:</p> <ul style="list-style-type: none"> - оценивает и контролирует качество процессов управления ИТ-инфраструктурой.
		<p>ПК-9.4: Способен контролировать и оптимизировать процесс управления инфраструктурой ИТ.</p>	<p>Знания:</p> <ul style="list-style-type: none"> - методы моделирования, контроля, оптимизации и совершенствования бизнес-процессов для повышения эффективности управления инфраструктурой ИТ организации. <p>Умения:</p> <ul style="list-style-type: none"> - консультировать заказчиков по совершенствованию бизнес-процессов и ИТ-инфраструктуры предприятия. <p>Навыки:</p> <ul style="list-style-type: none"> - разрабатывает мероприятия по проектированию процессов контроля и оптимизации управления инфраструктурой ИТ.

ПК-12	Способен подготавливать, разрабатывать, сопровождать техническое задание на проектирование и внедрение информационной системы для автоматизации операций в сетевой платежной инфраструктуре на базе инфокоммуникационных технологий	ПК-12.1: Способен определять назначение и цели создания информационной системы в платежной инфраструктуре.	<p>Знания:</p> <ul style="list-style-type: none"> - децентрализованные приложения на блокчейне для различных информационных платформ; - теории и практики создания и функционирования криптовалют и технологии блокчейн; - тенденции развития технологий блокчейн и криптовалют и потенциальные сферы их применения; - основные понятия финансовых технологий; - роли и места платежных систем в экономике и денежно-кредитной политике государства; - теоретические основы и предпосылки создания платежных систем; - проблемы и перспективы функционирования международных платежных и телекоммуникационных систем. <p>Умения:</p> <ul style="list-style-type: none"> - обосновывать выбор приоритетов при формировании информационно-технологической стратегии платежной инфраструктуры. <p>Навыки:</p> <ul style="list-style-type: none"> - определение назначения и цели создания информационной системы в платежной инфраструктуре.
		ПК-12.2: Формулирует требования к платежной информационной системе и видам её обеспечений.	<p>Знания:</p> <ul style="list-style-type: none"> - сущность и принципы организации платежных систем; - формы и инструменты банковских расчетов; - формы и инструменты по внешнеторговым операциям; - безналичные расчеты в розничном секторе экономики; - предоставляемые услуги банков населению для проведения безналичных расчетов и платежей; - финансовые услуги финтеха; - принципы и механизмы функционирования платежных систем; - показатели эффективности работы платежных систем. <p>Умения:</p> <ul style="list-style-type: none"> - формулировать требования к платежной информационной системе и видам её обеспечений. <p>Навыки:</p> <ul style="list-style-type: none"> - самостоятельная аналитическая работа с платежными информационными системами с использованием современного инструментария и компьютерных технологий.
		ПК-12.3: Определяет состав и содержание работ по созданию и внедрению сетевой платежной инфраструктуры на базе инфокоммуникационных технологий.	<p>Знания:</p> <ul style="list-style-type: none"> - перспективы и риски использования криптовалют в платежной сфере; - методологические основы в области применения методов и моделей для проектирования, разработки и эксплуатации информационно-аналитических систем, обеспечивающих информационную поддержку принятия решений стратегического, тактического и операционного уровней на основе объектно-ориентированного языка программирования; - основные принципы использования теории и методов искусственного интеллекта и нейроинформатики в построении современных платежных автоматизированных систем; - паттерны проектирования сетевой платежной

			<p>инфраструктуры на базе инфокоммуникационных технологий.</p> <p>Умения:</p> <ul style="list-style-type: none"> - моделировать и оценивать эффективности проектов коллективного инвестирования; - разрабатывать смарт-контракты; - определять состав и содержание платформы для смарт-контрактов; - разрабатывать блокчейны для консорциума; - решать практические задачи обеспечения функционирования платежных систем. <p>Навыки:</p> <ul style="list-style-type: none"> - разработка архитектуры информационно-аналитических и платежных систем на объектно-ориентированном языке программирования.
		<p>ПК-12.4: Определяет порядок контроля и приемки платежной информационной системы.</p>	<p>Знания:</p> <ul style="list-style-type: none"> - управление платежами на основе финтех; - технологии машинного зрения; - организационную и функциональную инфраструктуру платежных систем; - задачи и роли международных и национальных регулирующих органов в развитии платежных систем. <p>Умения:</p> <ul style="list-style-type: none"> - применять инфокоммуникационные инструменты для анализа эффективности деятельности платежных систем; - интерпретировать количественные и качественные показатели деятельности платежных систем. <p>Навыки:</p> <ul style="list-style-type: none"> - оценивать возможности использования смарт-контрактов в экономической деятельности; - расчет показателей эффективности и оценки работы платежной системы.
		<p>ПК-12.5: Определяет требования к документированию платежной информационной системы.</p>	<p>Знания:</p> <ul style="list-style-type: none"> - концепции управления эффективностью CPM (Corporate Performance Management) и вопросы, связанные с автоматизацией процессов стратегического управления на основе сбалансированной системы показателей эффективности, процессов планирования, бюджетирования, консолидации и корпоративной отчетности, а также реализации проектов в области CPM на базе современных паттернов, написанных на объектно-ориентированном языке программирования; - стандарты обмена финансовыми данными и их роли для функционирования платежных систем (Стандарт ISO 20022); - роль и функции центрального банка страны в развитии платежных систем. <p>Умения:</p> <ul style="list-style-type: none"> - устанавливать требования к документированию платежной информационной системы. <p>Навыки:</p> <ul style="list-style-type: none"> - разработка методического обеспечения к документированию платежной информационной системы.
		<p>ПК-12.6: Оформляет платежные документы с использованием современных информационных технологий.</p>	<p>Знания:</p> <ul style="list-style-type: none"> - принципы функционирования нейронных сетей в задачах распознавания образов; - принципы построения экспертных систем и систем, основанных на знаниях;

			<ul style="list-style-type: none"> - подходы к представлению знаний в интеллектуальных системах; - методы разбора и понимания естественного языка, а также машинного перевода; - стандарты обмена финансовыми данными и их роли для функционирования платежных систем (Стандарт ISO 20022); - нормативные документы регулирующие платежи и платежные системы. <p>Умения:</p> <ul style="list-style-type: none"> - анализировать и интерпретировать финансовую, бухгалтерскую и иную информацию, содержащуюся в документальной отчетности банковских и небанковских платежных систем с использованием современных информационных технологий. <p>Навыки:</p> <ul style="list-style-type: none"> - оформление платежных документов с использованием современных информационных технологий.
--	--	--	---

2. Место дисциплины в структуре образовательной программы

Дисциплина относится к вариативной части образовательного цикла (Б1.В.04).

Дисциплина изучается на 2 курсе в 4-м семестре.

Освоение курса дисциплины основано на знаниях, умениях и навыках, полученных на следующих дисциплинах: «Теория систем и системный анализ», «Эконометрика (продвинутый уровень)», «Программные пакеты экономико-математического моделирования», «Стратегический менеджмент международных компаний», «Микроэкономика (продвинутый уровень)», «Информационные технологии принятия решений в международном бизнесе», «Макроэкономика (продвинутый уровень)», «Прогностическая аналитика», «Эволюция фин-тех: финансовые инновации и будущее денег».

Умения и навыки, полученные в ходе прохождения практик: «Технологическая (проектно-технологическая) практика», «Эксплуатационная практика».

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

Общая трудоемкость (объем) дисциплины составляет 2 зачетные единицы, 72 академических часа.

3.1. Объём дисциплины по видам учебных занятий (в часах)

Вид работы	Трудоемкость	
	2 семестр	
	Академические часы	Зачетные единицы
Общая трудоемкость	72	2
Аудиторная работа, всего:	28	
в том числе:		
Лекции	8	
Практические занятия/семинары:	20	
Самостоятельная работа, всего:	44	
в том числе:		
Самоподготовка (самостоятельное изучение лекционного материала и материала учебников, подготовка к практическим занятиям, текущему контролю и т.д.)	44	
Вид промежуточной аттестации	зачет	

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

4.1. Разделы дисциплины (модуля) и трудоемкость по видам учебных занятий (в академических часах)

№ п/п	Раздел/тема Дисциплины	Общая трудоёмкость (в часах)	Виды учебных занятий, включая самостоятельную работу обучающихся и трудоёмкость (в часах)			Формы текущего контроля успеваемости
			аудитор- ные учеб- ные заня- тия		самостоятельная работа обучающихся	
		всего	лекции	семинары, практические занятия		
1.	Тема 1. Модули и пакеты в Python	32	4	8	20	Устный опрос, решение практических задач. Текущий срез №1 (контрольная работа 1).

№ п/п	Раздел/тема Дисциплины	Общая трудоёмкость (в часах)	Виды учебных занятий, включая самостоятельную работу обучающихся и трудоёмкость (в часах)			Формы текущего контроля успеваемости
			аудитор- ные учеб- ные заня- тия		самостоятельная работа обучающихся	
		всего	лекции	семинары, практические занятия		
2.	Тема 2. Проектирование ИС	20	2	6	12	Устный опрос, решение прак- тических задач. Текущий срез №2 (контроль- ная работа 2).
3.	Тема 3. Паттерны проектиро- вания	20	2	6	12	Устный опрос, решение прак- тических задач. Текущий срез №3 (контроль- ная работа 3).
ИТОГО:		72	8	20	44	

4.2 Содержание дисциплины (модуля), структурированное по разделам (темам)

Тема 1. Модули и пакеты в Python.

Модули в Python. Архитектура программы на Python. NumPy. Matplotlib. Модуль math. Модуль fractions. Модуль cmath. Модуль struct. Файлы CSV. Модуль shelve. Модуль locale. Модуль os.path. Модуль logging. Создание GUI на Python с помощью библиотеки Tkinter.

Тема 2. Проектирование ИС.

Работа с файлами и каталогами (Модуль bz2. Модуль filecmp. Модуль fnmatch. Модуль glob. Модуль gzip. Модуль shutil. Модуль tarfile. Модуль tempfile. Модуль zipfile. Модуль zlib).

Службы операционной системы (Модуль commands. Модули ConfigParser и configparser. Модуль datetime. Модуль errno. Модуль fcntl. Модуль io. Модуль logging. Модуль mmap. Модуль msvcrt. Модуль optparse. Модуль os. Модуль os.path. Модуль signal. Модуль subprocess. Модуль time. Модуль winreg).

Потоки и многозадачность (Параллельное программирование и Python. Модуль multiprocessing. Модуль threading. Модуль queue (Queue). Сопрограммы и микропотоки).

Работа с сетью и сокеты (Основы разработки сетевых приложений. Модуль asynchat. Модуль asyncore. Модуль select. Модуль socket. Модуль ssl. Мо-

дуль SocketServer).

Разработка интернет-приложений (Модуль ftplib. Пакет http. Модуль smtplib. Пакет urllib. Пакет xmlrpc).

Веб-программирование (Модуль cgi. Модуль cgitb. Поддержка WSGI. Пакет wsgiref).

Обработка и представление данных в Интернете (Модуль base64. Модуль binascii. Модуль csv. Пакет email. Модуль hashlib. Модуль hmac. Модуль HTMLParser. Модуль json. Модуль mimetypes. Модуль quopri. Пакет xml).

Различные библиотечные модули (Службы интерпретатора Python. Обработка строк. Модули для доступа к службам операционной системы. Сети. Обработка и представление данных в Интернете. Интернационализация. Мультимедийные службы. Различные модули).

Расширение и встраивание интерпретатора Python (Модули расширений. Встраивание интерпретатора Python. Модуль ctypes. Дополнительные возможности расширения и встраивания. Jython и IronPython).

Тема 3. Паттерны проектирования.

Структурные паттерны проектирования классов/объектов (Адаптер (Adapter). Декоратор (Decorator) или Оболочка (Wrapper). Заместитель (Proxy) или Суррогат (Surrogate). Компонировщик (Composite). Мост (Bridge). Приспособленец (Flyweight). Фасад (Facade)).

Паттерны проектирования поведения классов/объектов.

Интерпретатор (Interpreter). Итератор (Iterator). Команда (Command). Наблюдатель (Observer). Посетитель (Visitor). Посредник (Mediator). Состояние (State). Стратегия (Strategy). Хранитель (Memento). Цепочка обязанностей (Chain of Responsibility). Шаблонный метод (Template Method).

Порождающие паттерны проектирования (Абстрактная фабрика (Abstract Factory, Factory). Одиночка (Singleton). Прототип (Prototype). Строитель (Builder). Фабричный метод (Factory Method)).

Паттерны интеграции корпоративных информационных систем (Взаимодействие "точка - точка". Взаимодействие "звезда" (интегрирующая среда). Смешанный способ взаимодействия).

Практические занятия и семинары.

1. NumPy. Matplotlib.
2. Модуль math.
3. Модуль fractions.
4. Модуль cmath.
5. Модуль struct.
6. Модуль shelve.
6. Модуль locale.
7. Модуль os.path.
8. Модуль logging.
9. Создание GUI на Python с помощью библиотеки Tkinter.

Тема 2. Проектирование ИС.

1. Работа с файлами и каталогами.
2. Службы операционной системы.

3. Потоки и многозадачность.
4. Работа с сетью и сокеты.
5. Разработка интернет-приложений.
6. Веб-программирование.
7. Обработка и представление данных в Интернете.

Тема 3. Паттерны проектирования.

1. Структурные паттерны проектирования классов/объектов.
2. Паттерны проектирования поведения классов/объектов.
3. Порождающие паттерны проектирования.
4. Паттерны интеграции корпоративных информационных систем.

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

Содержание самостоятельной работы	Кол-во часов	Формы самостоятельной работы
Тема 1. Модули и пакеты в Python	20	Самостоятельное освоение теоретического материала по отдельным вопросам, чтение и проработка дополнительной литературы. Подготовка к практическим работам
Тема 2. Проектирование ИС	12	Самостоятельное освоение теоретического материала по отдельным вопросам, чтение и проработка дополнительной литературы. Подготовка к практическим работам
Тема 3. Паттерны проектирования	12	Самостоятельное освоение теоретического материала по отдельным вопросам, чтение и проработка дополнительной литературы. Подготовка к практическим работам

Основная часть самостоятельной работы должна включать самоподготовку студентов с использованием учебной литературы согласно списку литературы, приведенному в рабочей программе по указанной дисциплине.

Студент должен самостоятельно освоить разделы, указанные в рабочей программе для самостоятельной работы. Как правило, эти разделы включают в себя темы дисциплины, на которые в курсе читаемых лекций уделялось недостаточное внимание, либо эти разделы не включены в курс лекций, а должны осваиваться студентом самостоятельно. В разделы самостоятельной работы студентов также включаются наиболее сложные для понимания части дисциплины, требующие более детального и углубленного изучения и осмысления.

Студент должен найти в учебной литературе соответствующую тему, прочитать ее и попытаться изложить устно или письменно основные положения или идеи прочитанного раздела.

Далее студент должен составить сам письменно вопросы, отражающие основные положения разбираемой темы и устно (или письменно) ответить на них.

Во многих рекомендуемых учебных пособиях в конце каждого раздела

имеются тесты или уже сформулированные вопросы, на которые студент должен самостоятельно ответить.

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

6.1 Паспорт фонда оценочных средств по дисциплине (модулю)

1. Перечень компетенций с указанием этапов их формирования в процессе освоения образовательной программы

№ п/п	Контролируемые темы дисциплины	Код контролируемой компетенции и ее формулировка	Индикаторы достижения компетенции	Наименование оценочного средства
1.	Тема 1. Модули и пакеты в Python	ПК-9: способен к организации исследования лучших мировых практик и бизнес-процессов в области управления продуктами в ИТ-инфраструктуре международного бизнеса.	ИПК-9.1: Способен ставить задачи по методическому описанию бизнес-процессов в области управления продуктами в ИТ-инфраструктуре международного бизнеса.	Устный опрос, решение практических задач. Текущий срез №1 (контрольная работа 1).
2.	Тема 2. Проектирование ИС	ПК-12: способен подготавливать, разрабатывать, сопровождать техническое задание на проектирование и внедрение информационной системы для автоматизации операций в сетевой платежной инфраструктуре на базе инфокоммуникационных технологий.	ИПК-9.2: Способен организовать командную работу в области управления продуктами в ИТ-инфраструктуре международного бизнеса. ИПК-9.3: Оценивает и контролирует качество процессов управления ИТ-инфраструктурой.	Устный опрос, решение практических задач. Текущий срез №2 (контрольная работа 2).
3.	Тема 3. Паттерны проектирования		ИПК-9.4: Способен контролировать и оптимизировать процесс управления инфраструктурой ИТ. ИПК-12.1: Способен определять назначение и цели создания информационной системы в платежной инфраструктуре. ИПК-12.2: Формулирует требования к платежной информационной системе и видам её обеспечений. ИПК-12.3: Определяет состав и содержание работ по созданию и внедрению сетевой платежной инфраструктуры на базе инфокоммуникационных технологий. ИПК-12.4: Определяет порядок контроля и приемки платежной информационной системы. ИПК-12.5: Определяет требования к документированию платежной информационной системы.	Устный опрос, решение практических задач. Текущий срез №3 (контрольная работа 3).

			ИПК-12.6: Оформляет платежные документы с использованием современных информационных технологий.	
--	--	--	---	--

2а) Описание показателей и критериев оценивания компетенций на различных этапах их формирования

№ п/п	Наименование оценочного средства	Краткая характеристика оценочного средства	Представление оценочного средства в фонде
1.	Устный опрос	Продукт самостоятельной работы обучающегося, представляющий собой публичное выступление по представлению полученных результатов решения определённой учебно-практической, учебно-исследовательской или научной темы.	Перечень вопросов для обсуждения
2.	Решение практических задач	Проблемное задание, в котором обучающемуся предлагают осмыслить реальную профессионально-ориентированную ситуацию, необходимую для решения данной проблемы.	Практические ситуации по теме семинара
3.	Контрольная работа	Письменная работа, состоящая из нескольких вопросов.	Список вопросов для контрольной работы

2б) Описание шкал оценивания

Общий критерий оценки контрольной работы	A (90-100%)	Работа (письменный ответ) полностью отвечает целям/задачам обучения по данному курсу
	B (82-89%)	Работа (письменный ответ) в основном отвечает целям/задачам обучения по данному курсу
	C (75-81%)	Работа (письменный ответ) отвечает отдельным целям/задачам обучения по данному курсу, однако имеет серьезные недостатки в отношении остальных целей/задач
	D (67-74%)	Работа (письменный ответ) не отвечает большинству или всем целям/задачам обучения по данному курсу
	E (60-66%)	Работа (письменный ответ) совершенно не соответствует/противоречит целям данного курса; и/или не достигла их
Устный ответ	A (90-100%)	Самостоятельное и оригинальное осмысление материала; ясное и убедительное рассуждение; мощный и убедительный анализ
	B (82-89%)	Четкость логики и анализа, некоторая оригинальность в осмыслении материала, в целом работа хорошо аргументирована и убедительна
	C (75-81%)	Удовлетворительные построение и анализ при отсутствии оригинальности или критического осмысления материала
	D (67-74%)	Логика слабая, оригинальность отсутствует и/или материал недостаточно критически осмыслен
	E (60-66%)	Логика крайне слабая, отсутствует или неадекватна выбранной теме

Решение практических задач	A (90-100%)	Обучающийся решил задачу верно, без логических и арифметических ошибок, ответ обосновал и исчерпывающе аргументировал
	B (82-89%)	Обучающийся решил задачу, однако допустил некоторые арифметические ошибки, ответ обосновал
	C (75-81%)	Обучающийся решил задачу, однако допустил некоторые логические и арифметические ошибки, ответ недостаточно обоснован
	D (67-74%)	Обучающийся решил задачу неверно, допустил серьезные логические и арифметические ошибки, ответ попытался обосновать
	E (60-66%)	Обучающийся задачу не решил
Работа на занятиях	A (90-100%)	На занятиях оцениваются индивидуальные устные ответы на вопросы у доски или с места и письменные опросы. Точные, логичные ответы, быстрое и безошибочное выполнение заданий. Активен.
	B (82-89%)	Хорошо формулирует свои мысли, достаточно быстро и правильно выполняет текущие задания. Активен.
	C (75-81%)	То же, что и предыдущем пункте, только не столь безошибочно и не так быстро. Недостаточно инициативен.
	D (67-74%)	В ответах на вопросы допускает ошибки. Задания выполняет с ошибками. На занятиях неактивен.
	E (60-67%)	В ответах на вопросы допускает грубые ошибки. Невнятно излагает свои мысли. Пассивен на занятиях.
Общие умения	A (90-100%)	В ответах на вопросы допускает грубые ошибки либо отказывается от ответа. Пассивен на занятиях либо пропускает их.
	B (82-89%)	Проявлено владение достаточно широким спектром соответствующих умений
	C (75-81%)	Проявлено владение удовлетворительным спектром соответствующих умений
	D (67-74%)	Использованы отдельные общие умения, они применяются слабо или неадекватно
	E (60-67%)	Работа показывает недостаточную компетентность в области общих умений, крайне слабая работа на занятиях

3. Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков в ходе проведения промежуточной аттестации.

1. Подключение модуля из стандартной библиотеки.
2. Использование псевдонимов в модуле.
3. Инструкция from в модуле.
4. Местонахождение модулей в Python.
5. Получение списка всех модулей Python установленных на компьютере.
6. Создание своего модуля в Python.
7. Функция dir().
8. Архитектура программы на Python.

9. Пакеты модулей в Python.
10. Установка NumPy.
11. Наиболее важные атрибуты NumPy.
12. Создание массивов посредством NumPy.
13. Печать массивов посредством NumPy.
14. NumPy, базовые операции над массивами.
15. Список математических функций пакета NumPy
16. Индексы, срезы, итерации NumPy.
17. Манипуляции с формой в NumPy.
18. Объединение массивов посредством NumPy
19. Разбиение массива посредством NumPy.
20. Копии и представления посредством NumPy.
21. Глубокая копия посредством NumPy.
22. NumPy случайные числа.
23. Списки в массивы посредством NumPy.
24. Модуль numpy.random.
25. Создание массивов со случайными элементами посредством NumPy.
26. Выбор и перемешивание посредством NumPy.
27. Инициализация генератора случайных чисел в NumPy.
28. NumPy linalg – операции линейной алгебры.
29. Возведение в степень в NumPy.
30. Разложения в NumPy.
31. Системы уравнений в NumPy.
32. Matplotlib.
33. Модуль math.
34. Модуль fractions.
35. Модуль cmath.
36. Модуль struct -упаковка данных в бинарный файл.
37. Файлы CSV. Работа со словарями.
38. Модуль shelve. Обновление данных. Удаление данных.
39. Модуль locale. Создание и удаление папки. Переименование файла.
- Удаление файла. Существование файла. Работа с операционной системой.
40. Модуль os.path. Класс date. Класс time. Класс datetime.
41. Преобразование из строки в дату.
42. Операции с датами.
43. Форматирование дат и времени. Сложение и вычитани дат и време-
ни.
44. Свойства timedelta.
45. Сравнение дат.
46. Модуль logging.
47. Создание GUI на Python с помощью библиотеки Tkinter.
48. Создание главного окна в Tkinter.
49. Создание виджет в Tkinter. Установка свойств виджет.
50. Определение событий и их обработчиков в Tkinter.
51. Размещение виджет посредством Tkinter.

52. Отображение главного окна в Tkinter.
53. Виджеты (графические объекты) и их свойства: Кнопки; Метки; Однострочное текстовое поле; Многострочное текстовое поле; Радиокнопки (переключатели); Флажки; Списки.
54. Виджеты (графические объекты) и их свойства: Frame (рамка); Scale (шкала); Scrollbar (полоса прокрутки); Toplevel (окно верхнего уровня); Метод bind модуля Tkinter.
55. Программирование событий в Tkinter: Типы событий; Способ записи; События, производимые мышью; События, производимые с помощью клавиатуры.
56. Переменные Tkinter.
57. Объект Меню (Menu) в GUI: Что такое меню; Создание меню в Tkinter; Привязка функций к меню.
58. Диалоговые окна в Tkinter
59. Геометрические примитивы графического элемента Canvas (холст) модуля Tkinter
60. Canvas (холст): методы, идентификаторы и теги.
61. Особенности работы с виджетом Text модуля Tkinter.
62. Модуль bz2.
63. Модуль filecmp.
64. Модуль fnmatch.
65. Модуль glob.
66. Модуль gzip.
67. Модуль shutil.
68. Модуль tarfile.
69. Модуль tempfile.
70. Модуль zipfile.
71. Модуль zlib.
72. Модуль commands.
73. Модули ConfigParser и configparser.
74. Модуль errno.
75. Модуль fcntl.
76. Модуль io.
77. Модуль logging.
78. Модуль mmap.
79. Модуль msvcrt.
80. Модуль optparse.
81. Модуль signal.
82. Модуль subprocess.
83. Модуль time.
84. Модуль winreg.
85. Параллельное программирование и Python.
86. Модуль multiprocessing.
87. Модуль threading.
88. Модуль queue (Queue).

89. Сопрограммы и микропотoki.
90. Основы разработки сетевых приложений.
91. Модуль asynchat.
92. Модуль asyncore.
93. Модуль select.
94. Модуль socket.
95. Модуль ssl.
96. Модуль SocketServer.
97. Модуль ftplib.
98. Пакет http.
99. Модуль smtplib.
100. Пакет urllib.
101. Пакет xmlrpc.
102. Модуль cgi.
103. Модуль cgiib.
104. Поддержка WSGI.
105. Пакет wsgiref.
106. Модуль base64.
107. Модуль binascii.
108. Модуль csv.
109. Пакет email.
110. Модуль hashlib.
111. Модуль hmac.
112. Модуль HTMLParser.
113. Модуль json.
114. Модуль mimetypes.
115. Модуль quopri.
116. Пакет xml.
117. Службы интерпретатора Python.
118. Обработка строк.
119. Модули для доступа к службам операционной системы.
120. Обработка и представление данных в Интернете.
121. Интернационализация.
122. Мультимедийные службы.
123. Модули расширений.
124. Встраивание интерпретатора Python.
125. Модуль ctypes.
126. Дополнительные возможности расширения и встраивания.
127. Jython и IronPython.
128. Классификация паттернов.
129. Описание паттернов.
130. Результаты применения паттернов.
131. Диаграмма классов как средство отображения архитектуры объектно-ориентированной системы.
132. Порождающие паттерны. Порождающий паттерн Singleton. Порож-

дающий паттерн Abstract Factory (абстрактная фабрика).

133. Структурные паттерны. Паттерн Decorator (декоратор, wrapper, обертка). Паттерн Facade (фасад).

134. Паттерны поведения. Паттерн Observer. Паттерн Command (команда).

Типовые практические задания для самоконтроля.

Тема 1. Модули и пакеты в Python.

Задание 1. Установка python-пакетов с помощью pip

Решение.

pip - это система управления пакетами, которая используется для установки и управления программными пакетами, написанными на Python.

Начиная с Python версии 3.4, pip поставляется вместе с интерпретатором python.

Попробуем с помощью pip установить какой-нибудь пакет, например, numpy (<https://pythonworld.ru/numpy>) в консоле (в режиме администратора):

Linux:

```
sudo pip3 install numpy;
```

Windows:

```
pip3 install numpy.
```

Может возникнуть ошибка: "pip3" не является внутренней или внешней командой, исполняемой программой или пакетным файлом.

Тогда придется обращаться напрямую, указав полный путь к программе, например (если Питон установлен в C:\Python36\):
C:\Python36\Tools\Scripts\pip3.exe install numpy

Либо добавлять папку C:\Python36\Tools\Scripts\ в PATH вручную.

Либо сделав эту папку с pip3 текущей:

```
cd "C:\Python36\Tools\Scripts\"
```

```
pip3 install numpy
```

В последних версиях python вызов pip удобно выполнять из консоли:

```
python -m pip <command> [options]
```

Задание 2. Подключение модуля из стандартной библиотеки.

Решение.

Подключить модуль можно с помощью инструкции import. К примеру, подключим модуль os для получения текущей директории:

```
>>>
```

```
>>> import os
```

```
>>> os.getcwd()
```

```
'C:\\Python33'
```

После ключевого слова import указывается название модуля. Одной инструкцией можно подключить несколько модулей, хотя этого не рекомендуется делать, так как это снижает читаемость кода. Импортируем модули time и random.

```
>>>
```

```
>>> import time, random
>>> time.time()
1376047104.056417
>>> random.random()
0.9874550833306869
```

После импортирования модуля его название становится переменной, через которую можно получить доступ к атрибутам модуля. Например, можно обратиться к константе `e`, расположенной в модуле `math`:

```
>>>
>>> import math
>>> math.e
2.718281828459045
```

Если указанный атрибут модуля не будет найден, возбуждятся исключения `AttributeError`. А если не удастся найти модуль для импортирования, то `ImportError`.

```
>>>
>>> import notexist
Traceback (most recent call last):
  File "", line 1, in
    import notexist
ImportError: No module named 'notexist'
>>> import math
>>> math.Ë
Traceback (most recent call last):
  File "", line 1, in
    math.Ë
AttributeError: 'module' object has no attribute 'Ë'
```

Тема 2. Проектирование ИС.

Задание 1. Модуль `fnmatch`.

Модуль `fnmatch` обеспечивает поддержку сопоставления имен файлов с применением шаблонных символов, как это делает командная оболочка в системе UNIX. Этот модуль может использоваться только для сопоставления имен файлов с шаблоном, а получить фактический список файлов, соответствующих шаблону, можно с помощью модуля `glob`. Шаблоны имеют следующий синтаксис:

Для сопоставления имен файлов с шаблонами могут использоваться следующие функции:

```
fnmatch(filename, pattern)
```

Возвращает `True` или `False`, в зависимости от того, соответствует ли имя файла `filename` шаблону `pattern`. Чувствительность к регистру символов зависит от операционной системы (в некоторых системах, таких как Windows, регистр символов не учитывается).

```
fnmatchcase(filename, pattern)
```

Выполняет сопоставление имени файла `filename` с шаблоном `pattern` с уче-

с учетом регистра символов.

```
filter(names, pattern)
```

Применяет функцию `fnmatch()` ко всем именам файлов, перечисленных в последовательности `names`, и возвращает список всех имен, соответствующих шаблону `pattern`.

Примеры

```
fnmatch('foo.gif', '*.gif') # Вернет
True
fnmatch('part37.html', 'part3[0-5].html') # Вернет
False
# Пример поиска файлов в дереве каталогов
# с помощью функции os.walk(), модуля fnmatch и генераторов
def findall(topdir, pattern):
    for path, files, dirs in os.walk(topdir):
        for name in files:
            if fnmatch.fnmatch(name, pattern):
                yield os.path.join(path, name)
# Отыскать все файлы с расширением .py
for pyfile in findall(".", "*.py"):
    print pyfile
```

Тема 3. Паттерны проектирования.

Задание 1. Порождающий паттерн Singleton

Решение.

Назначение паттерна Singleton. Часто в системе могут существовать сущности только в единственном экземпляре, например, система ведения системного журнала сообщений или драйвер дисплея. В таких случаях необходимо уметь создавать единственный экземпляр некоторого типа, предоставлять к нему доступ извне и запрещать создание нескольких экземпляров того же типа.

Описание паттерна Singleton. Архитектура паттерна Singleton основана на идее использования глобальной переменной, имеющей следующие важные свойства:

1. Такая переменная доступна всегда. Время жизни глобальной переменной - от запуска программы до ее завершения.
2. Предоставляет глобальный доступ, то есть, такая переменная может быть доступна из любой части программы.

Для решения этой проблемы паттерн Singleton возлагает контроль над созданием единственного объекта на сам класс. Доступ к этому объекту осуществляется через статическую функцию-член класса, которая возвращает указатель или ссылку на него. Этот объект будет создан только при первом обращении к методу, а все последующие вызовы просто возвращают его адрес.

UML-диаграмма классов паттерна Singleton

Singleton
<u>-instance: Singleton</u>
<u>-Singleton()</u> <u>+getInstance(): Singleton</u>

Пример:

```
# coding: utf-8
"""
```

Одиночка (Singleton) - паттерн, порождающий объекты. Гарантирует, что у класса есть только один экземпляр, и предоставляет к нему глобальную точку доступа.

С помощью паттерна одиночка могут быть реализованы многие паттерны (абстрактная фабрика, строитель, прототип).

```
"""
class SingletonMeta(type):
    def __init__(cls, *args, **kwargs):
        cls._instance = None
        # глобальная точка доступа `Singleton.get_instance()`
        cls.get_instance = classmethod(lambda c:
c._instance)
    def __call__(cls, *args, **kwargs):
        super(SingletonMeta, cls).__init__(*args,
**kwargs)

    def __call__(cls, *args, **kwargs):
        if not cls._instance:
            cls._instance = super(SingletonMeta,
cls).__call__(*args, **kwargs)
        return cls._instance

class Singleton(object):
    __metaclass__ = SingletonMeta

    def __init__(self, name):
        self._name = name

    def get_name(self):
        return self._name

obj1 = Singleton('MyInstance 1')
print obj1.get_name() # MyInstance 1

obj2 = Singleton('MyInstance 2')
print obj2.get_name() # MyInstance 1
```

```
print obj1 is obj2 is Singleton.get_instance() #  
True
```

Текущий контроль знаний.

Текущий срез №1

Вопросы для подготовки к контрольному срезу №1:

1. Подключение модуля из стандартной библиотеки.
2. Использование псевдонимов в модуле.
3. Инструкция from в модуле.
4. Местонахождение модулей в Python.
5. Получение списка всех модулей Python установленных на компью-
тере.
6. Создание своего модуля в Python.
7. Функция dir().
8. Архитектура программы на Python.
9. Пакеты модулей в Python.
10. Установка NumPy.
11. Наиболее важные атрибуты NumPy.
12. Создание массивов посредством NumPy.
13. Печать массивов посредством NumPy.
14. NumPy, базовые операции над массивами.
15. Список математических функций пакета NumPy.
16. Индексы, срезы, итерации NumPy.
17. Манипуляции с формой в NumPy.
18. Объединение массивов посредством NumPy.
19. Разбиение массива посредством NumPy.
20. Копии и представления посредством NumPy.
21. Глубокая копия посредством NumPy.
22. NumPy случайные числа.
23. Списки в массивы посредством NumPy.
24. Модуль numpy.random.
25. Создание массивов со случайными элементами посредством NumPy.
26. Выбор и перемешивание посредством NumPy.
27. Инициализация генератора случайных чисел в NumPy.
28. NumPy linalg – операции линейной алгебры.
29. Возведение в степень в NumPy.
30. Разложения в NumPy.
31. Системы уравнений в NumPy.
32. Matplotlib.
33. Модуль math.
34. Модуль fractions.
35. Модуль cmath.
36. Модуль struct -упаковка данных в бинарный файл.
37. Файлы CSV. Работа со словарями.

38. Модуль `shelve`. Обновление данных. Удаление данных.
39. Модуль `locale`. Создание и удаление папки. Переименование файла. Удаление файла. Существование файла. Работа с операционной системой.
40. Модуль `os.path`. Класс `date`. Класс `time`. Класс `datetime`.
41. Преобразование из строки в дату.
42. Операции с датами.
43. Форматирование дат и времени. Сложение и вычитание дат и времени.
44. Свойства `timedelta`.
45. Сравнение дат.
46. Модуль `logging`.
47. Создание GUI на Python с помощью библиотеки Tkinter.
48. Создание главного окна в Tkinter.
49. Создание виджет в Tkinter. Установка свойств виджет.
50. Определение событий и их обработчиков в Tkinter.
51. Размещение виджет посредством Tkinter.
52. Отображение главного окна в Tkinter
53. Виджеты (графические объекты) и их свойства: Кнопки; Метки; Однострочное текстовое поле; Многострочное текстовое поле; Радиокнопки (переключатели); Флажки; Списки.
54. Виджеты (графические объекты) и их свойства: Frame (рамка); Scale (шкала); Scrollbar (полоса прокрутки); Toplevel (окно верхнего уровня); Метод `bind` модуля Tkinter.
55. Программирование событий в Tkinter: Типы событий; Способ записи; События, производимые мышью; События, производимые с помощью клавиатуры.
56. Переменные Tkinter.
57. Объект Меню (Menu) в GUI: Что такое меню; Создание меню в Tkinter; Привязка функций к меню.
58. Диалоговые окна в Tkinter.
59. Геометрические примитивы графического элемента Canvas (холст) модуля Tkinter.
60. Canvas (холст): методы, идентификаторы и теги.
61. Особенности работы с виджетом Text модуля Tkinter.

Контрольная работа №1. Тема: «Модули и пакеты в Python».

Задание 1. Получите списка всех модулей Python, установленных на компьютере.

Задание 2. Создайте свой модуль в Python.

Задание 3. Организуйте пакеты модулей в Python.

Задание 4. Создайте три больших массива с использованием модуля `numpy`.

Задание 5. Осуществите дискретное преобразование Фурье с использованием модуля `numpy`.

Оценка по выполнению контрольной работы производится в соответствии с таблицей.

Оценка	Оценка/ Процент	Описание критериев оценки
Отлично	A (90-100%)	Все задачи решены правильно. Решение представлено развернуто и полно. Получены полные ответы на теоретические вопросы.
Хорошо	B (82-89%)	Более 85 % задач решены правильно. Получены полные ответы на теоретические вопросы.
	C (75-81%)	От 75 % до 84 % задач решено правильно. Получены ответы на 75% теоретических вопросов
Удовлетворительно	D (67-74%)	От 65 % до 74 % задач решено правильно. Получены ответы на 70% теоретических вопросов.
	E (60-67%)	От 60 % до 64 % задач решено правильно. Не получены ответы на все теоретические вопросы.

Текущий срез №2

Вопросы для подготовки к контрольному срезу №2:

1. Модуль bz2.
2. Модуль filecmp.
3. Модуль fnmatch.
4. Модуль glob.
5. Модуль gzip.
6. Модуль shutil.
7. Модуль tarfile.
8. Модуль tempfile.
9. Модуль zipfile.
10. Модуль zlib.
11. Модуль commands.
12. Модули ConfigParser и configparser.
13. Модуль errno.
14. Модуль fcntl.
15. Модуль io.
16. Модуль logging.
17. Модуль mmap.
18. Модуль msvcrt.
19. Модуль optparse.
20. Модуль signal.
21. Модуль subprocess.
22. Модуль time.
23. Модуль winreg.
24. Параллельное программирование и Python.
25. Модуль multiprocessing.
26. Модуль threading.
27. Модуль queue (Queue).

28. Сопрограммы и микропотoki.
29. Основы разработки сетевых приложений.
30. Модуль asynchat.
31. Модуль asyncore.
32. Модуль select.
33. Модуль socket.
34. Модуль ssl.
35. Модуль SocketServer.
36. Модуль ftplib.
37. Пакет http.
38. Модуль smtplib.
39. Пакет urllib.
40. Пакет xmlrpc.
41. Модуль cgi.
42. Модуль cgiib.
43. Поддержка WSGI.
44. Пакет wsgiref.
45. Модуль base64.
46. Модуль binascii.
47. Модуль csv.
48. Пакет email.
49. Модуль hashlib.
50. Модуль hmac.
51. Модуль HTMLParser.
52. Модуль json.
53. Модуль mimetypes.
54. Модуль quopri.
55. Пакет xml.
56. Службы интерпретатора Python.
57. Обработка строк.
58. Модули для доступа к службам операционной системы.
59. Обработка и представление данных в Интернете.
60. Интернационализация.
61. Мультимедийные службы.
62. Модули расширений.
63. Встраивание интерпретатора Python.
64. Модуль ctypes.
65. Дополнительные возможности расширения и встраивания.
66. Jython и IronPython.

Контрольная работа №2. Тема: «Проектирование ИС».

Задание 1. Получите список всех имен файлов в каталоге C:. Используйте модуль glob.

Задание 2. Осуществите чтение и запись в архивные файлы с расширением zip, используя модуль gzip.

Задание 3. Выполните операции высокого уровня над файлами, как копирование, удаление и переименование с использованием модуля `shutil`.

Оценка по выполнению контрольной работы производится в соответствии с таблицей.

Оценка	Оценка/ Процент	Описание критериев оценки
Отлично	A (90-100%)	Все задачи решены правильно. Решение представлено развернуто и полно. Получены полные ответы на теоретические вопросы.
Хорошо	B (82-89%)	Более 85 % задач решены правильно. Получены полные ответы на теоретические вопросы.
	C (75-81%)	От 75 % до 84 % задач решено правильно. Получены ответы на 75% теоретических вопросов
Удовлетворительно	D (67-74%)	От 65 % до 74 % задач решено правильно. Получены ответы на 70% теоретических вопросов.
	E (60-67%)	От 60 % до 64 % задач решено правильно. Не получены ответы на все теоретические вопросы.

Текущий срез №3

Вопросы для подготовки к контрольному срезу №3:

1. Классификация паттернов
2. Описание паттернов
3. Результаты применения паттернов
4. Диаграмма классов как средство отображения архитектуры объектно-ориентированной системы
5. Порождающие паттерны. Порождающий паттерн Singleton. Порождающий паттерн Abstract Factory (абстрактная фабрика).
6. Структурные паттерны. Паттерн Decorator (декоратор, wrapper, обертка). Паттерн Facade (фасад)
7. Паттерны поведения. Паттерн Observer. Паттерн Command (команда).

Контрольная работа №3. Тема: «Паттерны проектирования».

Вариант 1.

Задание 1. Необходимо обеспечить взаимодействие несовместимых интерфейсов или как создать единый устойчивый интерфейс для нескольких компонентов с разными интерфейсами

Задание 2. Возложить дополнительные обязанности (прозрачные для клиентов) на отдельный объект, а не на класс в целом.

Задание 3. Необходимо управлять доступом к объекту, так чтобы создавать громоздкие объекты "по требованию".

Вариант 2.

Задание 1. В системе должна аккумулироваться, рассчитываться и т. п. необходимая информация.

Задание 2. Как обрабатывать группу или композицию структур объектов одновременно?

Задание 3. Требуется отделить абстракцию от реализации так, чтобы и то, и другое можно было изменять независимо. При использовании наследования реализация жестко привязывается к абстракции, что затрудняет независимую модификацию.

Вариант 3.

Задание 1. Обеспечить низкую связанность при создании экземпляра класса и связывании его с другим классом.

Задание 2. Необходимо обеспечить поддержку множества мелких объектов.

Задание 3. Как спроектировать систему так, чтобы изменение одних ее элементов не влияло на другие?

Оценка по выполнению контрольной работы производится в соответствии с таблицей.

Оценка	Оценка/ Процент	Описание критериев оценки
Отлично	A (90-100%)	Все задачи решены правильно. Решение представлено развернуто и полно. Получены полные ответы на теоретические вопросы.
Хорошо	B (82-89%)	Более 85 % задач решены правильно. Получены полные ответы на теоретические вопросы.
	C (75-81%)	От 75 % до 84 % задач решено правильно. Получены ответы на 75% теоретических вопросов
Удовлетворительно	D (67-74%)	От 65 % до 74 % задач решено правильно. Получены ответы на 70% теоретических вопросов.
	E (60-67%)	От 60 % до 64 % задач решено правильно. Не получены ответы на все теоретические вопросы.

Вопросы для подготовки к зачету:

1. Модули в Python.
2. Архитектура программы на Python.
3. Модуль NumPy.
4. Модуль Matplotlib.
5. Модуль math.
6. Модуль fractions.
7. Модуль cmath.
8. Модуль struct.
9. Файлы CSV.
10. Модуль shelve.
11. Модуль locale.
12. Модуль os.path.
13. Модуль logging.
14. Создание GUI на Python с помощью библиотеки Tkinter.
15. Работа с файлами и каталогами (Модуль bz2. Модуль filecmp. Модуль fnmatch.).
16. Работа с файлами и каталогами (Модуль glob. Модуль gzip. Модуль shutil. Модуль tarfile. Модуль tempfile. Модуль zipfile. Модуль zlib).

17. Службы операционной системы (Модуль `commands`. Модули `ConfigParser` и `configparser`. Модуль `datetime`. Модуль `errno`).

18. Службы операционной системы (Модуль `fcntl`. Модуль `io`. Модуль `logging`. Модуль `mmap`. Модуль `msvcrt`. Модуль `optparse`. Модуль `os`. Модуль `os.path`. Модуль `signal`. Модуль `subprocess`. Модуль `time`. Модуль `winreg`).

19. Поток и многозадачность (Параллельное программирование и Python. Модуль `multiprocessing`. Модуль `threading`. Модуль `queue` (Queue). Сопрограммы и микропоток).

20. Работа с сетью и сокеты (Основы разработки сетевых приложений. Модуль `asynchat`. Модуль `asyncore`. Модуль `select`).

21. Работа с сетью и сокеты (Модуль `socket`. Модуль `ssl`. Модуль `SocketServer`).

22. Разработка интернет-приложений (Модуль `ftplib`. Пакет `http`. Модуль `smtplib`).

23. Разработка интернет-приложений (Пакет `urllib`. Пакет `xmlrpc`).

24. Веб-программирование (Модуль `cgi`. Модуль `cgitb`. Поддержка WSGI. Пакет `wsgiref`).

25. Обработка и представление данных в Интернете (Модуль `base64`. Модуль `binascii`. Модуль `csv`. Пакет `email`).

26. Обработка и представление данных в Интернете (Модуль `hashlib`. Модуль `hmac`. Модуль `HTMLParser`. Модуль `json`. Модуль `mimetypes`. Модуль `quopri`. Пакет `xml`).

27. Различные библиотечные модули (Службы интерпретатора Python. Обработка строк. Модули для доступа к службам операционной системы).

28. Различные библиотечные модули (Сети. Обработка и представление данных в Интернете. Интернационализация. Мультимедийные службы).

29. Расширение и встраивание интерпретатора Python (Модули расширений. Встраивание интерпретатора Python. Модуль `ctypes`).

30. Расширение и встраивание интерпретатора Python (Дополнительные возможности расширения и встраивания. `Jython` и `IronPython`).

31. Структурные паттерны проектирования классов/объектов (Адаптер (Adapter). Декоратор (Decorator) или Оболочка (Wrapper)).

32. Структурные паттерны проектирования классов/объектов (Заместитель (Proxy) или Суррогат (Surrogate). Компоновщик (Composite)).

33. Структурные паттерны проектирования классов/объектов (Мост (Bridge). Приспособленец (Flyweight). Фасад (Facade)).

34. Паттерны проектирования поведения классов/объектов.

35. Интерпретатор (Interpreter).

36. Итератор (Iterator). Команда (Command).

37. Наблюдатель (Observer). Посетитель (Visitor).

38. Посредник (Mediator). Состояние (State). Стратегия (Strategy).

39. Хранитель (Memento). Цепочка обязанностей (Chain of Responsibility).

40. Шаблонный метод (Template Method).

41. Порождающие паттерны проектирования (Абстрактная фабрика (Abstract Factory, Factory). Одиночка (Singleton)).

42. Порождающие паттерны проектирования (Прототип (Prototype). Строитель (Builder). Фабричный метод (Factory Method)).

43. Паттерны интеграции корпоративных информационных систем (Взаимодействие "точка - точка". Взаимодействие "звезда" (интегрирующая среда). Смешанный способ взаимодействия).

б) критерии оценивания компетенций (результатов)

- «зачтено» с рейтинговой оценкой «отлично» выставляется студенту, если сумма баллов за два ответа составляет 90% и более;
- «зачтено» с рейтинговой оценкой «хорошо» выставляется студенту, если сумма баллов за два ответа от 75% до 89%;
- «зачтено» с рейтинговой оценкой «удовлетворительно» выставляется студенту, если сумма баллов за два ответа от 60% до 74%;
- «незачтено» выставляется студенту, если сумма баллов за два ответа менее 60%.

в) описание шкалы оценивания

Итоговым результатом считается оценка, полученная студентом по результатам работы в семестре (выставляется на основании результатов контрольных работ и работы на семинарских занятиях), проставленная преподавателем в зачетной ведомости. Студент может получить интегральную оценку от 70% до 100%.

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

Виды работы	Индикаторы компетенций, проверяемые в процессе выполнения данного вида работы	Доля вида работы в итоговой оценке
Контрольная работа №1 – промежуточная аттестация № 1	Тема 1-3. Самостоятельное изучение отдельных разделов курса, повторение лекционного материала и материала учебников, подготовка к семинару и дискуссии (ИПК-9.1, ИПК-9.2, ИПК-9.3, ИПК-9.4, ИПК-12.1, ИПК-12.2, ИПК-12.3, ИПК-12.4, ИПК-12.5, ИПК-12.6)	от 0% до 25%
Контрольная работа №2 – промежуточная аттестация № 2		от 0% до 25%
Контрольная работа №3 – промежуточная аттестация № 3		от 0% до 25%
Работа на семинарах	Ответы на вопросы преподавателя по теме семинара, выполнение домашних заданий, основанных на лекционном материале.	от 0% до 25%
Итог (зачтено)	Итоговым результатом по курсу считается оценка, полученная студентом на зачете. Ответ студента оценивается в % с учетом шкалы соответствия рейтинговых оценок пятибалльным и европейским оценкам.	от 60% до 100%
Незачтено	Письменное (устное) тестирование по всему пройденному материалу для студентов, чей текущий рейтинг оценивается менее 60%.	менее 60%

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

а) Основная литература:

1. Язык программирования Python: практикум / Р.А. Жуков. – М.: ИН-ФРА-М, 2019. – 216 с. – URL: <https://znanium.com/catalog/product/1000002>.
2. Шелудько, В.М. Основы программирования на языке высокого уровня Python : учебное пособие / В.М. Шелудько ; Министерство науки и высшего образования РФ, Южный федеральный университет, Инженерно-технологическая академия. – Ростов-на-Дону ; Таганрог : Южный федеральный университет, 2017. – 147 с. : ил. – Режим доступа: по подписке. – URL: <https://biblioclub.ru/index.php?page=book&id=500056>
3. Шелудько, В.М. Язык программирования высокого уровня Python: функции, структуры данных, дополнительные модули / В.М. Шелудько ; Министерство науки и высшего образования РФ, Южный федеральный университет, Институт компьютерных технологий и информационной безопасности. – Ростов-на-Дону ; Таганрог : Южный федеральный университет, 2017. – 108 с. : ил. – Режим доступа: по подписке. – URL: <https://biblioclub.ru/index.php?page=book&id=500060>

б) Дополнительная литература:

1. Говорова, С.В. Web-технологии: учебное пособие (курс лекций) : [16+] / С.В. Говорова ; Северо-Кавказский федеральный университет. – Ставрополь : Северо-Кавказский Федеральный университет (СКФУ), 2019. – Ч. 1. – 149 с. : ил. – Режим доступа: по подписке. – URL: <https://biblioclub.ru/index.php?page=book&id=596209>
2. Тузовский, А. Ф. Проектирование и разработка web-приложений : учебное пособие для вузов / А. Ф. Тузовский. — Москва : Издательство Юрайт, 2020. — 218 с. — (Высшее образование). — ISBN 978-5-534-00515-8. — Текст : электронный // ЭБС Юрайт [сайт]. — URL: <https://urait.ru/bcode/451207>
3. Титов, В.А. Разработка WEB-сайта средствами языка HTML : учебное пособие / В.А. Титов, Г.И. Пещеров ; Институт мировых цивилизаций. – Москва : Институт мировых цивилизаций, 2018. – 184 с. : ил., табл. – Режим доступа: по подписке. – URL: <https://biblioclub.ru/index.php?page=book&id=598475>

в) Литература для факультативного чтения:

1. Разработка веб-приложений с использованием Flask на языке Python / М. Гринберг; пер. с англ. А.Н. Киселева. – М.: ДМК Пресс, 2014. – 272 с. – URL: <https://znanium.com/catalog/product/1031120>.
2. Python: Пособие / Р.А. Сузи. – СПб: БХВ-Петербург, 2015. – 759 с. – URL: <https://znanium.com/catalog/product/939857>.
3. Python 3 и PyQt 5. Разработка приложений / Н.А. Прохоренок. – СПб: БХВ-Петербург, 2016. – 833 с. – URL: <https://znanium.com/catalog/product/944629>.

4. Django. Практика создания Web-сайтов на Python / В.А. Дронов. – СПб: БХВ-Петербург, 2016. – 528 с. – URL: <https://znanium.com/catalog/product/941019>.

5. Самое необходимое / Н.А. Прохоренко, В.А. Дронов. – СПб: БХВ-Петербург, 2016. – 464 с. – URL: <https://znanium.com/catalog/product/944129>.

6. Python и машинное обучение: крайне необходимое пособие по новейшей предсказательной аналитике, обязательное для более глубокого понимания методологии машинного обучения / С. Рашка; пер. с англ. А.В. Логунова. – М.: ДМК Пресс, 2017. – 418 с. – URL: <https://znanium.com/catalog/product/1027758>.

7. Python и анализ данных / Уэс Маккинли; пер. с англ. А.А. Слинкина. – М.: ДМК Пресс, 2015. – 482 с. – URL: <https://znanium.com/catalog/product/1027796>.

8. Скрапинг веб-сайтов с помощью Python / Р. Митчелл ; пер. с англ. А. В. Груздева. – М.: ДМК Пресс, 2016. – 280 с. – URL: <https://znanium.com/catalog/product/1027754>.

г) Интернет-ресурс, базы данных:

1. Система федеральных образовательных порталов. Информационно-коммуникационные технологии в образовании. <http://www.ict.edu.ru/lib/>.

2. Интернет университет информационных технологий. <http://www.intuit.ru/>.

3. Система федеральных образовательных порталов. Информационно-коммуникационные технологии в образовании. <http://www.ict.edu.ru/lib/>.

4. Российская национальная библиотека (РНБ). www.hbl-russia.ru.

5. Российская государственная библиотека (РГБ). <http://www.rsl.ru>.

6. ЭБС «Университетская библиотека онлайн» <http://www.biblioclub.ru>.

7. ЭБС «Znanium.com» <http://znanium.com>.

8. ЭБС «Юрайт» <https://biblio-online.ru>.

9. Сайт разработчиков на Питоне - <http://diveinto.python.ru/toc.html>.

10. Самоучитель Python - <http://pythonworld.ru/samouchitel-python>.

11. <https://www.python.org/> – сайт, содержащий необходимые дистрибутивы и полную информацию для языка программирования Python. Интерпретатор для Python можно использовать как программируемый высокоуровневый калькулятор.

12. <http://sympy.org/> – сайт, посвященный свободно распространяемому пакету SymPy, представляющему собой библиотеку Python символьных вычислений.

13. <http://github.com/sympy/sympy> – сайт, посвященный свободно распространяемому пакету SymPy, представляющему собой библиотеку Python символьных вычислений. Сайт свободных новинок и постоянного пользовательского обновления данного пакета (реализованный на принципах Вики).

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

Вид учебных занятий	Организация деятельности студента
Лекция	Написание конспекта лекций: кратко, схематично, последовательно фиксировать основные положения, выводы, формулировки, обобщения; пометить важные мысли, выделять ключевые слова, термины. Проверка терминов, понятий с помощью энциклопедий, словарей, справочников с выписыванием толкований в тетрадь. Обозначить вопросы, термины, материал, который вызывает трудности, пометить и попытаться найти ответ в рекомендуемой литературе. Если самостоятельно не удастся разобраться в материале, необходимо сформулировать вопрос и задать преподавателю на консультации, на практическом занятии. Уделить внимание следующим понятиям (<i>перечисление понятий</i>) и др.
Практические занятия	Проработка рабочей программы, уделяя особое внимание целям и задачам структуре и содержанию дисциплины. Конспектирование источников. Работа с конспектом лекций, подготовка ответов к контрольным вопросам, просмотр рекомендуемой литературы, работа с текстом (<i>указать текст из источника и др.</i>). Прослушивание аудио- и видеозаписей по заданной теме, решение расчетно-графических заданий, решение задач по алгоритму и др.
Контрольная работа / индивидуальные задания	Знакомство с основной и дополнительной литературой, включая справочные издания, зарубежные источники, конспект основных положений, терминов, сведений, требующих для запоминания и являющихся основополагающими в этой теме. Составление аннотаций к прочитанным литературным источникам и др.
Подготовка к зачету	При подготовке к зачету необходимо ориентироваться на конспекты лекций, рекомендуемую литературу и др.

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

Программное обеспечение: Windows, Антивирусная защита ESET NOD 32, Microsoft Office, Microsoft Visual Studio, Microsoft Project, IBM SPSS, Python, PGP, 7z, Aris Express, Open Server, Sublime Text, Notepad++, OpenOffice, LibreOffice, R, GIMP.

Аудитория оборудована техническими средствами для проведения занятий в формате онлайн. Наличие веб-камеры и скайп-гарнитуры.

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

Лаборатория цифрового моделирования.
Оборудование:

Компьютеры - 15 шт. (13 шт. - для студентов),
Системный блок - 15 шт. (13 шт. - для студентов),
Комплект аудио колонок для воспроизведения аудио файла - 1шт.,
Экран - 1 шт;
Мультимедиа-проектор - 1 шт.
Специализированная мебель:
Доска - 1шт.,
Стол преподавателя - 1 шт.,
Стол лаборанта - 1 шт.
Стол студенческий двухместный - 6 шт.,
Стол студенческий одноместный - 1 шт.,
Стол двухместный - 8 шт.,
Стулья - 18 шт.

11. Иные сведения и материалы

12. Лист регистрации внесенных изменений