

Уральский социально-экономический институт
(филиал) образовательного учреждения профсоюзов
высшего образования «Академия труда и
социальных отношений»

УТВЕРЖДАЮ
Зам. директора по УВР УрСЭИ (филиал)
ОУП ВО «АТиСО»

_____ О.В. Зубкова

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)

Программирование на языке Python

Направление подготовки
09.03.03 ПРИКЛАДНАЯ ИНФОРМАТИКА

Профиль подготовки
Корпоративные информационные системы

Квалификация выпускника
«Бакалавр»

Кафедра: Гуманитарных и естественнонаучных дисциплин

Разработчики программы: к.т.н., доцент Мадудин В.Н.

Оглавление

1.	ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ (МОДУЛЮ), СООТНЕСЕННЫЕ С ПЛАНИРУЕМЫМИ РЕЗУЛЬТАТАМИ ОСВОЕНИЯ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ.....	4
1.1	Планируемые результаты обучения по дисциплине.....	4
1.2	Результаты освоения образовательной программы:	4
2.	МЕСТО ДИСЦИПЛИНЫ (МОДУЛЯ) В СТРУКТУРЕ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ.....	5
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	5
4.	СОДЕРЖАНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ), СТРУКТУРИРОВАННОЕ ПО ТЕМАМ (РАЗДЕЛАМ) С УКАЗАНИЕМ ОТВЕДЕННОГО НА НИХ КОЛИЧЕСТВА АКАДЕМИЧЕСКИХ ИЛИ АСТРОНОМИЧЕСКИХ ЧАСОВ И ВИДОВ УЧЕБНЫХ ЗАНЯТИЙ;	5
4.1	Содержание дисциплины (модуля).....	5
4.2	Разделы/темы дисциплины, их трудоемкость и виды занятий	7
5.	ПЕРЕЧЕНЬ УЧЕБНО-МЕТОДИЧЕСКОГО ОБЕСПЕЧЕНИЯ ДЛЯ САМОСТОЯТЕЛЬНОЙ РАБОТЫ ОБУЧАЮЩИХСЯ ПО ДИСЦИПЛИНЕ (МОДУЛЮ).8	
6.	ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ДЛЯ ПРОВЕДЕНИЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ОБУЧАЮЩИХСЯ ПО ДИСЦИПЛИНЕ (МОДУЛЮ)	11
7.	ОСНОВНАЯ И ДОПОЛНИТЕЛЬНАЯ УЧЕБНАЯ ЛИТЕРАТУРА, НЕОБХОДИМАЯ ДЛЯ ОСВОЕНИЯ ДИСЦИПЛИНЫ (МОДУЛЯ);.....	11
8.	РЕСУРСЫ ИНФОРМАЦИОННО-ТЕЛЕКОММУНИКАЦИОННОЙ СЕТИ «ИНТЕРНЕТ», НЕОБХОДИМЫЕ ДЛЯ ОСВОЕНИЯ ДИСЦИПЛИНЫ (МОДУЛЯ).....	12
9.	МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ (МОДУЛЯ).....	13
10.	ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ, ИСПОЛЬЗУЕМЫЕ ПРИ ОСУЩЕСТВЛЕНИИ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА ПО ДИСЦИПЛИНЕ (МОДУЛЮ), ВКЛЮЧАЯ ПЕРЕЧЕНЬ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ И ИНФОРМАЦИОННЫХ СПРАВОЧНЫХ СИСТЕМ	14
11.	МАТЕРИАЛЬНО-ТЕХНИЧЕСКАЯ БАЗА, НЕОБХОДИМАЯ ДЛЯ ОСУЩЕСТВЛЕНИЯ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА ПО ДИСЦИПЛИНЕ (МОДУЛЮ).....	14
12.	ОБРАЗОВАТЕЛЬНЫЕ ТЕХНОЛОГИИ	15
	Приложение №1 к разделу № 6	16
	Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине (модулю)	16
6.1	ПЕРЕЧЕНЬ КОМПЕТЕНЦИЙ С УКАЗАНИЕМ ЭТАПОВ ИХ ФОРМИРОВАНИЯ В ПРОЦЕССЕ ОСВОЕНИЯ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ.....	16
6.2	ОПИСАНИЕ ПОКАЗАТЕЛЕЙ И КРИТЕРИЕВ ОЦЕНИВАНИЯ КОМПЕТЕНЦИЙ НА РАЗЛИЧНЫХ ЭТАПАХ ИХ ФОРМИРОВАНИЯ, ОПИСАНИЕ ШКАЛ ОЦЕНИВАНИЯ	16
6.3	ТИПОВЫЕ КОНТРОЛЬНЫЕ ЗАДАНИЯ ИЛИ ИНЫЕ МАТЕРИАЛЫ, НЕОБХОДИМЫЕ ДЛЯ ОЦЕНКИ ЗНАНИЙ, УМЕНИЙ, НАВЫКОВ И (ИЛИ) ОПЫТА ДЕЯТЕЛЬНОСТИ, ХАРАКТЕРИЗУЮЩИХ ЭТАПЫ ФОРМИРОВАНИЯ КОМПЕТЕНЦИЙ В ПРОЦЕССЕ ОСВОЕНИЯ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ.....	19
	Типовые контрольные вопросы для подготовки к зачету при проведении промежуточной аттестации по дисциплине	19
	Типовые практические задачи (задания, тесты) билетов для проведения промежуточной аттестации по дисциплине	20

6.4	МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ, ОПРЕДЕЛЯЮЩИЕ ПРОЦЕДУРЫ ОЦЕНИВАНИЯ ЗНАНИЙ, УМЕНИЙ, НАВЫКОВ И (ИЛИ) ОПЫТА ДЕЯТЕЛЬНОСТИ, ХАРАКТЕРИЗУЮЩИХ ЭТАПЫ ФОРМИРОВАНИЯ КОМПЕТЕНЦИЙ.....	25
-----	---	----

1. ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ (МОДУЛЮ), СООТНЕСЕННЫЕ С ПЛАНИРУЕМЫМИ РЕЗУЛЬТАТАМИ ОСВОЕНИЯ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ

Цель изучения дисциплины: формирование у студентов представления о современном языке программирования Python, приобретение навыков программирования в скриптовых языках, приобретение навыков использования библиотек и модулей для ускоренной обработки данных, использование модульного программирования для решения практических задач.

Задачи:

- изучение Python как мультипарадигменного языкового средства, достаточно полно отражающего современные концепции разработки ПО;
- формирование навыков создания приложений на языке Python;
- совершенствование и углубление навыков объектно-ориентированного и функционального программирования;
- знакомство с основами создания приложений для взаимодействия с базами данных.

1.1 Планируемые результаты обучения по дисциплине.

Освоение дисциплины направлено на формирование у студентов следующих компетенций:

ПК-1. Способен проводить работы по интеграции программных модулей и компонент и проверке работоспособности выпусков программных продуктов.

1.2 Результаты освоения образовательной программы:

В результате освоения компетенции **ПК-1** студент должен:

а) знать методы описания алгоритмов, основные абстрактные типы данных и их реализации; основные способы верстки web-страниц, современные языки разметки; современные системы управления контентом сайта (CMS); основы создания интернет-приложений;

б) уметь разрабатывать алгоритмы для конкретных задач, учитывать различия в представлении информационных объектов современными браузерами; проводить работы по интеграции программных модулей с использованием современных языков программирования; разрабатывать вебприложения;

в) владеть навыками использования языка разметки гипертекста с языками программирования; навыками работы с программными комплексами и проверкой работоспособности выпусков программного продукта; способностью проводить интеграцию программных модулей, навыками тестирования разработанной программы.

В результате изучения дисциплины студент должен:

Знать:

- основные понятия языка программирования Python;
- методы описания структур данных на Python;
- классы задач, формулируемых и решаемых на Python.

Уметь:

- разрабатывать программы на языке Python;
- применять изученные методы и структуры данных в соответствии с технологией разработки программ.

Владеть:

- самостоятельного создания простых скриптов;
- разработки программ средней сложности на Python.

2. МЕСТО ДИСЦИПЛИНЫ (МОДУЛЯ) В СТРУКТУРЕ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ

Согласно ФГОС ВО и ОПОП по направлению 09.03.03 Прикладная информатика дисциплина «Программирование на языке Python» является дисциплиной, формируемой участниками образовательных отношений блока Дисциплины (модули) **Б1.В.01**.

Для освоения дисциплины обучающиеся используют знания, умения и компетенции, сформированные в ходе изучения дисциплин «Информатика и программирование», «Базы данных», «Алгоритмы и структуры данных».

Освоение данной дисциплины является основой для последующего изучения дисциплин «Анализ данных и машинное обучение», а также для последующего прохождения учебной и производственной практик и подготовки к итоговой государственной аттестации.

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

Таблица 3.1

Объем дисциплины	Всего часов	
	Для очной формы обучения	Для заочной формы обучения
Общая трудоемкость дисциплины (зачетных един/часов)	4/144	4/144
Контактная работа обучающихся с преподавателем (по видам учебных занятий) (всего)		
Аудиторная работа (всего)	54	16
в том числе:	-	-
Лекции	20	6
Семинары, практические занятия	-	-
Лабораторные работы	34	10
Самостоятельная работа обучающегося (всего)	54	119
Контроль	36	9
Вид промежуточной аттестации (зачет, зачет с оценкой, экзамен)	Экзамен	Экзамен

4. СОДЕРЖАНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ), СТРУКТУРИРОВАННОЕ ПО ТЕМАМ (РАЗДЕЛАМ) С УКАЗАНИЕМ ОТВЕДЕННОГО НА НИХ КОЛИЧЕСТВА АКАДЕМИЧЕСКИХ ИЛИ АСТРОНОМИЧЕСКИХ ЧАСОВ И ВИДОВ УЧЕБНЫХ ЗАНЯТИЙ;

4.1 Содержание дисциплины (модуля)

Тема 1. Установка и настройка среды программирования Python.

Интерпретаторы и компиляторы. Загрузка среды программирования Python с сайта разработчика. Установка Python в ОС Windows. Знакомство с интерфейсом среды программирования Python. Запуск программ, написанных на Python через командную строку

ОС. Стандартные модули Python. Загрузка модулей в среду программирования и выполнение функция модуля.

Тема 2. Типы данных и составные объекты Python.

Ввод и вывод на Python. Форматированный ввод и вывод. Типизация Python. Специальные типы. Числа: целые, с плавающей точкой и комплексное. Строки. Последовательности. Отображения. Арифметические выражения. Составные объекты. Списки. Дополнительные возможности при конструировании списков. Кортежи. Множества. Словари. Файлы. Вспомогательные типы. Стеки. Очереди. Работа с файлами на диске.

Тема 3. Управляющие конструкции ходом выполнения программ на Python.

Полное и неполное ветвление. Реализация многовариантного выбора через ветвления. Параметрический цикл с вариативной и без вариативной части. Цикл с предусловием. Команды прерывания цикла и ее продолжения со следующей итерации (break, continue). Выполнение команд внутри контекста (with). Обработка исключений в программах. Встроенные типы исключений и определение новых исключений.

Тема 4. Пользовательские функции и основы функционального программирования.

Определение пользовательских функций. Передача параметров и возврат результатов. Значения аргументов функции по умолчанию. Произвольный набор аргументов. Именованные аргументы. Правила видимости. Функции, как объекты и замыкания. Декораторы. Итераторы. Генераторы и сопрограммы. Генераторы списков. Выражения-генераторы. Основы декларативного программирования. Оператор Lambda. Атрибуты функций. Выполнение неопределенных функций, командами: eval, exec, compile. Разработка рекурсивной функции на базе процедурной ее формы. Последовательная, параллельная и псевдопараллельная рекурсии. Обработка списков рекурсивными функциями. Работа со списком, как со стеком и очередью.

Тема 5. Модульное программирование. Стандартные и нестандартные модули Python.

Создание и использование модуля. Поиск модулей и компилированные файлы. Стандартные модули: sys, os. Пакеты. Краткая характеристика нестандартных модулей Python.

Тема 6. Математические модули.

Модуль чисел с плавающей точкой Decimal. Модуль рациональных чисел Fractions. Модуль стандартных математических функций Math. Модуль абстрактных базовых классов Numbers. Модуль псевдослучайных чисел Random. Модуль для работы с комплексными числами CMath. Модуль для работы с массивами Array. Модуль сортировки списков Bisect.

Тема 7. Разработка модуля численных методов.

Численные методы решения алгебраических и трансцендентных уравнений. Численные методы решения систем алгебраических уравнений. Численные методы интерполирования. Численные методы интегрирования. Статистическая обработка данных методом наименьших квадратов.

Тема 8. Работа с текстом и строками.

Кодировки символов на Python. Операции над строками. Стандартные модули обработки строк String и Codecs. Модуль приблизительного сравнения двух строк DiffLib. Модуль для работы с кодировкой и регулярными выражениями Re. Модуль преобразования данных Struct. Модуль доступа к базе символов UnicodeData.

Тема 9. Доступ к средствам операционной системы.

Модуль FileCmp для сравнения файловых объектов и каталогов ОС. Модуль Time для работы с системным временем. Модуль Sched для планирования задач ОС. Модуль GetPass для работы с секретной информацией. Модуль GetOpt для обработки данных командной строки. Модуль TempFile для работы с временными файлами. Модули Glob и fnMatch для работы с шаблонами. Модуль Mmap для отображения файлов в память.

4.2 Разделы/темы дисциплины, их трудоемкость и виды занятий

Таблица 4.1

№ п/п	Название раздела, темы	Очная форма обучения					Компетенции	Литература
		Всего	Аудиторные занятия		Самостоятельная работа	Контроль		
			Лекции	Лабораторные занятия				
1	Установка и настройка среды программирования Python.	10	1	3	6		ПК-1	Л1-6
2	Типы данных и составные объекты Python.	10	1	3	6		ПК-1	Л1-6
3	Управляющие конструкции ходом выполнения программ на Python.	13	3	4	6		ПК-1	Л1-6
4	Пользовательские функции и основы функционального программирования.	13	3	4	6		ПК-1	Л1-6
5	Модульное программирование. Стандартные и нестандартные модули Python.	12	2	4	6		ПК-1	Л1-6
6	Математические модули.	12	2	4	6		ПК-1	Л1-6
7	Разработка модуля численных методов.	12	2	4	6		ПК-1	Л1-6
8	Работа с текстом и строками.	13	3	4	6		ПК-1	Л1-6
9	Доступ к средствам операционной системы.	13	3	4	6		ПК-1	Л1-6
	Экзамен	36						
	Всего часов	144	20	34	54	36		
	Зачетные единицы		4					

Вид промежуточной аттестации: Экзамен

Таблица 4.2

№ п/п	Название раздела, темы	Заочная форма обучения					Компетенции	Литература
		Всего	Аудиторные занятия		Самостоятельная работа	Контроль		
			Лекции	Лабораторные занятия				
1	Установка и настройка среды программирования Python.	12	1	1	10		ПК-1	Л1-6
2	Типы данных и составные объекты Python.	14	1	1	12		ПК-1	Л1-6

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)

3	Управляющие конструкции ходом выполнения программ на Python.	15	1	1	13		ПК-1	Л1-6
4	Пользовательские функции и основы функционального программирования.	16	1	1	14		ПК-1	Л1-6
5	Модульное программирование. Стандартные и нестандартные модули Python.	16	1	1	14		ПК-1	Л1-6
6	Математические модули.	16	1	1	14		ПК-1	Л1-6
7	Разработка модуля численных методов.	15		1	14		ПК-1	Л1-6
8	Работа с текстом и строками.	15		1	14		ПК-1	Л1-6
9	Доступ к средствам операционной системы.	16		2	14		ПК-1	Л1-6
Экзамен		9						
Всего часов		144	6	10	119	9		
Зачетные единицы			4					

Вид промежуточной аттестации: Экзамен

5. ПЕРЕЧЕНЬ УЧЕБНО-МЕТОДИЧЕСКОГО ОБЕСПЕЧЕНИЯ ДЛЯ САМОСТОЯТЕЛЬНОЙ РАБОТЫ ОБУЧАЮЩИХСЯ ПО ДИСЦИПЛИНЕ (МОДУЛЮ)

Для закрепления практических навыков в дисциплине предусмотрены практические занятия, которые выполняются студентами самостоятельно и/или под руководством преподавателя.

Для лучшего закрепления практических навыков рекомендуется занятия проводить в небольших группах, в которых студенты имеют возможность повторять демонстрационные действия преподавателя на своих компьютерах. В этом случае лабораторные работы (практикумы) будут выполняться студентами самостоятельно для закрепления изученных материалов.

Перечень лабораторных работ (практикумов) по темам:

Тема 1. Установка и настройка среды программирования Python.

Список литературы по теме приведен в таблице 4.1 и 4.2.

Задания для самостоятельной работы:

1. Загрузка среды программирования Python с сайта разработчика.
2. Установка Python в ОС Windows.
3. Знакомство с интерфейсом среды программирования Python. Запуск программ, написанных на Python через командную строку ОС.
4. Стандартные модули Python. Загрузка модулей в среду программирования и выполнение функция модуля..

Тема 2. Типы данных и составные объекты Python.

Список литературы по теме приведен в таблице 4.1 и 4.2.

Задания для самостоятельной работы:

1. Ввод и вывод на Python. Форматированный ввод и вывод.
2. Типизация Python. Специальные типы. Числа: целые, с плавающей точкой и комплексное.
3. Строки. Последовательности. Отображения.
4. Арифметические выражения.
5. Составные объекты. Списки. Дополнительные возможности при конструировании списков.
6. Кортежи. Множества. Словари. Файлы.
7. Вспомогательные типы. Стеки. Очереди.
8. Работа с файлами на диске.

Тема 3. Управляющие конструкции ходом выполнения программ на Python.

Список литературы по теме приведен в таблице 4.1 и 4.2.

Задания для самостоятельной работы:

1. Полное и неполное ветвление. Реализация многовариантного выбора через ветвления.
2. Параметрический цикл с вариативной и без вариативной части.
3. Цикл с предусловием. Команды прерывания цикла и ее продолжения со следующей итерации (break, continue).
4. Выполнение команд внутри контекста (with). Обработка исключений в программах. Встроенные типы исключений и определение новых исключений.

Тема 4. Пользовательские функции и основы функционального программирования.

Список литературы по теме приведен в таблице 4.1 и 4.2.

Задания для самостоятельной работы:

1. Передача параметров и возврат результатов. Значения аргументов функции по умолчанию. Произвольный набор аргументов. Именованные аргументы. Правила видимости.
2. Функции, как объекты и замыкания.
3. Декораторы. Итераторы.
4. Генераторы и сопрограммы. Генераторы списков. Выражения-генераторы.
5. Основы декларативного программирования. Оператор Lambda. Атрибуты функций. Выполнение неопределенных функций, командами: eval, exec, compile.
6. Разработка рекурсивной функции на базе процедурной ее формы. Последовательная, параллельная и псевдопараллельная рекурсии. Обработка списков рекурсивными функциями. Работа со списком, как со стеком и очередью.

Тема 5. Модульное программирование. Стандартные и нестандартные модули Python.

Список литературы по теме приведен в таблице 4.1 и 4.2.

Задания для самостоятельной работы:

1. Создание и использование модуля.
2. Поиск модулей и компилированные файлы.
3. Стандартные модули: sys, os.
4. Краткая характеристика нестандартных модулей Python.

Тема 6. Математические модули.

Список литературы по теме приведен в таблице 4.1 и 4.2.

Задания для самостоятельной работы:

1. Модуль чисел с плавающей точкой Decimal.
2. Модуль рациональных чисел Fractions.
3. Модуль стандартных математических функций Math.
4. Модуль абстрактных базовых классов Numbers.
5. Модуль псевдослучайных чисел Random.
6. Модуль для работы с комплексными числами CMath.
7. Модуль для работы с массивами Array.
8. Модуль сортировки списков Bisect.

Тема 7. Разработка модуля численных методов.

1. Численные методы решения алгебраических и трансцендентных уравнений.
2. Численные методы решения систем алгебраических уравнений.
3. Численные методы интерполирования.
4. Численные методы интегрирования.

5. Статистическая обработка данных методом наименьших квадратов.

Тема 8. Работа с текстом и строками.

1. Кодировки символов на Python.
2. Операции над строками. Стандартные модули обработки строк String и Codecs.
3. Модуль приблизительного сравнения двух строк DiffLib.
4. Модуль для работы с кодировкой и регулярными выражениями Re.
5. Модуль преобразования данных Struct.
6. Модуль доступа к базе символов UnicodeData.

Тема 9. Доступ к средствам операционной системы.

1. Модуль FileCmp для сравнения файловых объектов и каталогов ОС.
2. Модуль Time для работы с системным временем.
3. Модуль Sched для планирования задач ОС.
4. Модуль GetPass для работы с секретной информацией.
5. Модуль GetOpt для обработки данных командной строки.
6. Модуль TempFile для работы с временными файлами.
7. Модули Glob и fnMatch для работы с шаблонами.
8. Модуль Mmap для отображения файлов в память.

5.1. Примеры задач (практических заданий)

Задание 1.

Есть список $a = [1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89]$.

Выведите все элементы, которые меньше 5.

Задание 2.

Даны списки:

$a = [1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89]$;

$b = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13]$.

Нужно вернуть список, который состоит из элементов, общих для этих двух списков.

Задание 3. Отсортируйте словарь по значению в порядке возрастания и убывания.

Задание 4. Напишите код, который переводит целое число в строку, при том что его можно применить в любой системе счисления.

5.2. Примеры тестовых вопросов

Вопрос 1 Какая библиотека отвечает за время?

- a) Time
- b) localtime
- c) time
- d) clock

Вопрос 2 Что будет результатом этого кода?

```
x = 23
```

```
num = 0 if x > 10 else 11
```

```
print(num)
```

- a) 11
- b) 23
- c) Ошибка

- d) 0
- e) 10

Вопрос 3. Что будет показано в результате?

```
name = "John"  
print('Hi, %s' % name)
```

- a) "Hi, John"
- b) Ошибка
- c) "Hi, "
- d) "Hi, name".

Вопрос 4 Что покажет этот код?

```
for i in range(5):  
    if i % 2 == 0:  
        continue  
    print(i)
```

- a) Числа: 1, 3 и 5
- b) Ошибку из-за неверного вывода
- c) Числа: 1 и 3
- d) Ошибку, так как i не присвоена
- e) Числа: 0, 2 и 4.

Вопрос 5 Где правильно создана переменная? Вопрос про вариант ответа, который не выдаст ошибку при запуске проекта

- a) int num = 2
- b) var num = 2
- c) Нет подходящего варианта
- d) num = float(2)
- e) \$num = 2

6. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ДЛЯ ПРОВЕДЕНИЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ОБУЧАЮЩИХСЯ ПО ДИСЦИПЛИНЕ (МОДУЛЮ)

Приложение №1 к рабочей программе дисциплины

7. ОСНОВНАЯ И ДОПОЛНИТЕЛЬНАЯ УЧЕБНАЯ ЛИТЕРАТУРА, НЕОБХОДИМАЯ ДЛЯ ОСВОЕНИЯ ДИСЦИПЛИНЫ (МОДУЛЯ);

Основная литература

- | |
|---|
| 1. 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: http://biblioclub.ru/index.php?page=book&id=500056 (дата обращения: 31.08.2020). – Библиогр. в кн. – ISBN 978-5-9275-2649-9. – Текст : электронный. |
| 2. 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: <http://biblioclub.ru/index.php?page=book&id=500060> (дата обращения: 31.08.2020). – Библиогр. в кн. – ISBN 978-5-9275-2648-2. – Текст : электронный.

3. Программные системы статистического анализа: обнаружение закономерностей в данных с использованием системы R и языка Python : [16+] / В.М. Волкова, М.А. Семенова, Е.С. Четвертакова, С.С. Вожов ; Новосибирский государственный технический университет. – Новосибирск : Новосибирский государственный технический университет, 2017. – 74 с. : ил., табл. – Режим доступа: по подписке. – URL: <http://biblioclub.ru/index.php?page=book&id=576496> (дата обращения: 31.08.2020). – Библиогр.: с. 48. – ISBN 978-5-7782-3183-2. – Текст : электронный.

4. Баженова, И. Ю. Введение в программирование : учебное пособие / И. Ю. Баженова, В. А. Сухомлин. — 3-е изд. — Москва : Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2020. — 326 с. — ISBN 978-5-4497-0652-2. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: <http://www.iprbookshop.ru/97539.html> (дата обращения: 31.08.2020). — Режим доступа: для авторизир. пользователей

Дополнительная литература

5. Буйначев, С.К. Основы программирования на языке Python : учебное пособие / С.К. Буйначев, Н.Ю. Боклаг ; Уральский федеральный университет имени первого Президента России Б. Н. Ельцина. – Екатеринбург : Издательство Уральского университета, 2014. – 92 с. : табл., ил. – Режим доступа: по подписке. – URL: <http://biblioclub.ru/index.php?page=book&id=275962> (дата обращения: 31.08.2020). – Библиогр. в кн. – ISBN 978-5-7996-1198-9. – Текст : электронный.

6. Хахаев, И.А. Практикум по алгоритмизации и программированию на Python: курс / И.А. Хахаев. – 2-е изд., исправ. – Москва : Национальный Открытый Университет «ИНТУИТ», 2016. – 179 с. : ил. – Режим доступа: по подписке. – URL: <http://biblioclub.ru/index.php?page=book&id=429256> (дата обращения: 31.08.2020). – Библиогр. в кн. – Текст : электронный.

Методические материалы по дисциплине:

Сайт разработчиков на Питоне. Режим доступа: <http://diveinto.python.ru/toc.html>

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

Python Software Foundation. Режим доступа: <http://www.python.org>

8. РЕСУРСЫ ИНФОРМАЦИОННО-ТЕЛЕКОММУНИКАЦИОННОЙ СЕТИ «ИНТЕРНЕТ», НЕОБХОДИМЫЕ ДЛЯ ОСВОЕНИЯ ДИСЦИПЛИНЫ (МОДУЛЯ)

№ п/п	Интернет ресурс (адрес)	Описание ресурса
Профессиональные базы данных		
1.	http://www.intuit.ru/department/pl/python	Интернет-университет информационных технологий (ИНТУИТ)
2.	https://python-ucheba.ru/samouchitel-python	Обучение Python. Сайт-самоучитель для начинающих
3.	https://snakify.org/ru/	Интерактивный учебник Python 3 со 100 упражнениями

9. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ (МОДУЛЯ)

Методические указания по подготовке к экзамену

Наиболее ответственным этапом в обучении студентов является экзаменационная сессия. На ней студенты отчитываются о выполнении учебной программы, об уровне и объеме полученных знаний. Это государственная отчетность студентов за период обучения, за изучение учебной дисциплины, за весь вузовский курс. Поэтому так велика их ответственность за успешную сдачу экзаменационной сессии. На сессии студенты сдают экзамены или зачеты. Зачеты могут проводиться с дифференцированной отметкой или без нее, с записью «зачтено» в зачетной книжке. Экзамен как высшая форма контроля знаний студентов оценивается по пятибалльной системе.

Залогом успешной сдачи всех экзаменов являются систематические, добросовестные занятия студента. Однако это не исключает необходимости специальной работы перед сессией и в период сдачи экзаменов. Специфической задачей студента в период экзаменационной сессии являются повторение, обобщение и систематизация всего материала, который изучен в течение года.

Начинать повторение рекомендуется за месяц-полтора до начала сессии. Прежде чем приступить к нему, необходимо установить, какие учебные дисциплины выносятся на сессию и, если возможно, календарные сроки каждого экзамена или зачета.

Установив выносимые на сессию дисциплины, необходимо обеспечить себя программами. В основу повторения должна быть положена только программа. Не следует повторять ни по билетам, ни по контрольным вопросам. Повторение по билетам нарушает систему знаний и ведет к механическому заучиванию, к «натаскиванию». Повторение по различного рода контрольным вопросам приводит к пропускам и пробелам в знаниях и к недоработке иногда весьма важных разделов программы.

Повторение - процесс индивидуальный; каждый студент повторяет то, что для него трудно, неясно, забыто. Поэтому, прежде чем приступить к повторению, рекомендуется сначала внимательно посмотреть программу курса, установить наиболее трудные, наименее усвоенные разделы и выписать их на отдельном листе.

В процессе повторения анализируются и систематизируются все знания, накопленные при изучении программного материала: данные учебника, записи лекций, конспекты прочитанных книг, заметки, сделанные во время консультаций или семинаров, и др. Ни в коем случае нельзя ограничиваться только одним конспектом, а тем более, чужими записями. Всякого рода записи и конспекты - вещи сугубо индивидуальные, понятные только автору. Готовясь по чужим записям, легко можно впасть в очень грубые заблуждения.

Само повторение рекомендуется вести по темам программы и по главам учебника. Закончив работу над темой (главой), необходимо ответить на вопросы учебника или выполнить задания, а самое лучшее - воспроизвести весь материал.

Консультации, которые проводятся для студентов в период экзаменационной сессии, необходимо использовать для углубления знаний, для восполнения пробелов и для разрешения всех возникших трудностей. Без тщательного самостоятельного продумывания материала беседа с консультантом неизбежно будет носить «общий», поверхностный характер и не принесет нужного результата.

Есть целый ряд принципов («секретов»), которыми следует руководствоваться при подготовке к экзаменам.

Первый - подготовьте свое рабочее место, где все должно способствовать успеху: тишина, расположение учебных пособий, строгий порядок.

Второй - сядьте удобнее за стол, положите перед собой чистые листы бумаги, справа - тетради и учебники. Вспомните все, что знаете по данной теме, и запишите это в виде плана или тезисов на чистых листах бумаги слева. Потом проверьте правильность, полноту и последовательность знаний по тетрадям и учебникам. Выпишите то, что не сумели вспомнить,

на правой стороне листов и там же запишите вопросы, которые следует задать преподавателю на консультации. Не оставляйте ни одного неясного места в своих знаниях.

Третий - работайте по своему плану. Вдвоем рекомендуется готовиться только для взаимопроверки или консультации, когда в этом возникает необходимость.

Четвертый - подготавливая ответ по любой теме, выделите основные мысли в виде тезисов и подберите к ним в качестве доказательства главные факты и цифры. Ваш ответ должен быть кратким, содержательным, концентрированным.

Пятый - помимо повторения теории не забудьте подготовить практическую часть, чтобы свободно и умело показать навыки работы с текстами, картами, различными пособиями, решения задач и т.д.

Шестой - установите четкий ритм работы и режим дня. Разумно чередуйте труд и отдых, питание, нормальный сон и пребывание на свежем воздухе.

Седьмой - толково используйте консультации преподавателя. Приходите на них, продуктивно поработав дома и с заготовленными конкретными вопросами, а не просто послушать, о чем будут спрашивать другие.

Восьмой - бойтесь шпаргалки - она вам не прибавит знаний.

Девятый - не допускайте как излишней самоуверенности, так и недооценки своих способностей и знаний. В основе уверенности лежат твердые знания. Иначе может получиться так, что вам достанется тот единственный вопрос, который вы не повторили.

Десятый - не забывайте связывать свои знания по любому предмету с современностью, с жизнью, с производством, с практикой.

Одиннадцатый - когда на экзамене вы получите свой билет, спокойно сядьте за стол, обдумайте вопрос, набросайте план ответа, подойдите к приборам, картам, подумайте, как теоретически объяснить проделанный опыт. Не волнуйтесь, если что-то забыли.

При подготовке к занятиям необходимо еще раз проверить себя на предмет усвоения основных категорий и ключевых понятий курса.

**10. ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ, ИСПОЛЬЗУЕМЫЕ ПРИ
ОСУЩЕСТВЛЕНИИ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА ПО
ДИСЦИПЛИНЕ (МОДУЛЮ), ВКЛЮЧАЯ ПЕРЕЧЕНЬ ПРОГРАММНОГО
ОБЕСПЕЧЕНИЯ И ИНФОРМАЦИОННЫХ СПРАВОЧНЫХ СИСТЕМ
ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ И ИНФОРМАЦИОННЫЕ СПРАВОЧНЫЕ
СИСТЕМЫ**

№ п/п	Название программы/системы	Описание программы/системы
1.	Microsoft Office 2007, 2019	Пакет Майкрософт Офис.
2.	MS Visual Studio	Интегрированная среда разработки
3.	PyCharm, Spyder	Среда разработки Python
4.	Google Chrome, Mozilla Firefox	Интернет-браузеры.

**11. МАТЕРИАЛЬНО-ТЕХНИЧЕСКАЯ БАЗА, НЕОБХОДИМАЯ ДЛЯ
ОСУЩЕСТВЛЕНИЯ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА ПО
ДИСЦИПЛИНЕ (МОДУЛЮ)**

Лекционные и практические занятия проводятся в аудиториях, оснащенных презентационным оборудованием (компьютер, имеющий выход в Интернет, мультимедийный проектор, экран, акустические системы), доской, рабочими учебными столами и стульями.

При необходимости занятия проводятся в компьютерных классах, оснащенных доской, экраном, рабочими учебными столами и стульями, персональными компьютерами,

объединенными в локальные сети с выходом в Интернет, с установленным лицензионным программным обеспечением, с подключенным к ним периферийным устройством и оборудованием (мультимедийный проектор, акустическая система и пр.).

При проведении практических занятий с использованием индивидуальных учебных заданий студенты должны быть обеспечены калькуляторами.

Для лиц с ОВЗ. В учебных помещениях возможно оборудование специальных учебных мест, предполагающих увеличение размера зоны на одно место с учетом подъезда и разворота кресла-коляски, увеличения ширины прохода между рядами столов. В аудитории хорошее освещение, в соответствии с требованиями СЭС.

В случае обучения слабослышащих обучающихся аудитории по необходимости оборудуются аудиотехникой (микрофонами, динамиками, наушниками или головными телефонами, диктофонами).

12. ОБРАЗОВАТЕЛЬНЫЕ ТЕХНОЛОГИИ

В процессе освоения дисциплины используются следующие образовательные технологии.

Стандартные формы обучения:

- лекции с использованием мультимедийных презентаций;
- практические занятия и лабораторные работы;
- консультации преподавателей;
- самостоятельная работа студентов, в которую входит освоение теоретического материала, подготовка к практическим/лабораторным занятиям, выполнение указанных выше письменных/устных заданий, работа с литературой.

Приложение №1 к разделу № 6

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

6.1 ПЕРЕЧЕНЬ КОМПЕТЕНЦИЙ С УКАЗАНИЕМ ЭТАПОВ ИХ ФОРМИРОВАНИЯ В ПРОЦЕССЕ ОСВОЕНИЯ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ

Этапами формирования компетенций в процессе освоения образовательной программы являются семестры.

№ п/п	Код формируемой компетенции и ее содержание	Этапы (семестры) формирования компетенции в процессе освоения ООП		
		Очная форма обучения	Очно-заочная форма обучения	Заочная форма обучения
1	ПК-1. Способен проводить работы по интеграции программных модулей и компонент и проверку работоспособности выпусков программных продуктов.	6 семестр		3 курс

6.2 ОПИСАНИЕ ПОКАЗАТЕЛЕЙ И КРИТЕРИЕВ ОЦЕНИВАНИЯ КОМПЕТЕНЦИЙ НА РАЗЛИЧНЫХ ЭТАПАХ ИХ ФОРМИРОВАНИЯ, ОПИСАНИЕ ШКАЛ ОЦЕНИВАНИЯ

В зависимости от количества баллов оценивание компетентности студента оценивается по уровням: от 3 до 4 баллов - «минимальный уровень», от 5 до 7 баллов - «базовый уровень», от 8 до 9 баллов - «высокий уровень».

Описание показателей и критериев оценивания компетенций

Показатели оценивания	Критерии оценивания компетенций	Шкала оценивания
Понимание смысла компетенции	<p>Имеет базовые общие знания в рамках диапазона выделенных задач (1 балл)</p> <p>Понимает факты, принципы, процессы, общие понятия в пределах области исследования. В большинстве случаев способен выявить достоверные источники информации, обработать, анализировать информацию. (2 балла)</p> <p>Имеет фактические и теоретические знания в пределах области исследования с пониманием границ применимости (3 балла)</p>	<p>Минимальный уровень</p> <p>Базовый уровень</p> <p>Высокий уровень</p>

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)

Освоение компетенции в рамках изучения дисциплины	Наличие основных умений, требуемых для выполнения простых задач. Способен применять только типичные, наиболее часто встречающиеся приемы по конкретной сформулированной (выделенной) задаче (1 балл)	Минимальный уровень
	Имеет диапазон практических умений, требуемых для решения определенных проблем в области исследования. В большинстве случаев способен выявить достоверные источники информации, обработать, анализировать информацию. (2 балла)	Базовый уровень
	Имеет широкий диапазон практических умений, требуемых для развития творческих решений, абстрагирования проблем. Способен выявлять проблемы и умеет находить способы решения, применяя современные методы и технологии. (3 балла)	Высокий уровень
Способность применять на практике знания, полученные в ходе изучения дисциплины	Способен работать при прямом наблюдении. Способен применять теоретические знания к решению конкретных задач. (1 балл)	Минимальный уровень
	Может взять на себя ответственность за завершение задач в исследовании, приспосабливает свое поведение к обстоятельствам в решении проблем. Затрудняется в решении сложных, неординарных проблем, не выделяет типичных ошибок и возможных сложностей при решении той или иной проблемы (2 балла)	Базовый уровень
	Способен контролировать работу, проводить оценку, совершенствовать действия работы. Умеет выбрать эффективный прием решения задач по возникающим проблемам. (3 балла)	Высокий уровень

Шкала оценки для проведения промежуточной аттестации по дисциплине. Шкала оценки в системе «зачтено – не зачтено»

№ п/п	Оценка за ответ	Характеристика ответа
1	Зачтено	<p>Достаточный объем знаний в рамках изучения дисциплины. В ответе используется научная терминология. Стилистическое и логическое изложение ответа на вопрос правильное.</p> <p>Умеет делать выводы без существенных ошибок.</p> <p>Владеет инструментарием изучаемой дисциплины, умеет его использовать в решении стандартных (типовых) задач.</p> <p>Ориентируется в основных теориях, концепциях и направлениях по изучаемой дисциплине.</p> <p>Активен на практических (лабораторных) занятиях, допустимый уровень культуры исполнения заданий.</p> <p>Количество баллов за освоение компетенций от 3 до 9.</p>
2	Не зачтено	<p>Не достаточно полный объем знаний в рамках изучения дисциплины.</p> <p>В ответе не используется научная терминология.</p> <p>Изложение ответа на вопрос с существенными стилистическими и логическими ошибками.</p> <p>Не умеет делать выводы по результатам изучения дисциплины.</p>

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)

		<p>Слабое владение инструментарием изучаемой дисциплины, не компетентность в решении стандартных (типовых) задач.</p> <p>Не умеет ориентироваться в основных теориях, концепциях и направлениях по изучаемой дисциплине.</p> <p>Пассивность на практических (лабораторных) занятиях, низкий уровень культуры исполнения заданий.</p> <p>Не сформированы компетенции, умения и навыки. Количество баллов за освоение компетенций менее 3.</p> <p>Отказ от ответа или отсутствие ответа.</p>
--	--	--

Шкала оценки письменных ответов по дисциплине

№ п/п	Оценка за ответ	Характеристика ответа
1	Отлично	<p>Материал раскрыт полностью, изложен логично, без существенных ошибок, выводы доказательны и опираются на теоретические знания</p> <p>Количество баллов за освоение компетенций от 8 до 9</p>
2	Хорошо	<p>Основные положения раскрыты, но в изложении имеются незначительные ошибки выводы доказательны, но содержат отдельные неточности</p> <p>Количество баллов за освоение компетенций от 5 до 7</p>
3	Удовлетворительно	<p>Изложение материала не систематизированное, выводы недостаточно доказательны, аргументация слабая.</p> <p>Количество баллов за освоение компетенций от 3 до 4</p>
4	Неудовлетворительно	<p>Не раскрыто основное содержание материала, обнаружено незнание основных положений темы.</p> <p>Не сформированы компетенции, умения и навыки. Количество баллов за освоение компетенций менее 3</p> <p>Ответ на вопрос отсутствует</p>

Шкала оценки выполнения лабораторных и самостоятельных работ

№ п/п	Оценка за выполнение	Характеристика ответа
1	Отлично	<p>Оценку «отлично» заслуживает студент, если индивидуальное учебное задание решено в полном объеме, с соблюдением индивидуального варианта, с требуемой точностью. Решения сопровождаются корректными комментариями и выводами.</p>
2	Хорошо	<p>Оценку «хорошо» заслуживает студент, если индивидуальное учебное задание решено в полном объеме, с соблюдением индивидуального варианта. Решения сопровождаются комментариями и выводами.</p>
3	Удовлетворительно	<p>Оценку «удовлетворительно» заслуживает студент, если индивидуальное учебное задание решено в неполном объеме, но с соблюдением индивидуального варианта. Решения сопровождаются комментариями и выводами.</p>
4	Неудовлетворительно	<p>Оценку «неудовлетворительно» выставляется студенту, если индивидуальное учебное задание не решено, либо решено с ошибками и характеризуется отсутствием выводов.</p>

Для студентов с ограниченными возможностями здоровья предусмотрены следующие оценочные средства:

<i>Категории студентов</i>	<i>Виды оценочных средств</i>	<i>Форма контроля и оценки результатов обучения</i>
С нарушением слуха	Тесты, письменные самостоятельные работы, вопросы к зачету, контрольные работы	Преимущественно письменная проверка
С нарушением зрения	Собеседование по вопросам к зачету, опрос по терминам	Преимущественно устная проверка (индивидуально)
С нарушением опорно-двигательного аппарата	Решение дистанционных тестов, контрольные работы, письменные самостоятельные работы, вопросы к зачету.	Организация взаимодействия с обучающимися посредством электронной почты, письменная проверка

Студентам с ограниченными возможностями здоровья увеличивается время на подготовку ответов на контрольные вопросы. В освоении дисциплины инвалидами и лицами с ограниченными возможностями здоровья большое значение имеет индивидуальная работа. Под индивидуальной работой подразумевается индивидуальная учебная работа (консультации), то есть дополнительное разъяснение учебного материала и углубленное изучение материала с теми обучающимися, которые в этом заинтересованы.

6.3 ТИПОВЫЕ КОНТРОЛЬНЫЕ ЗАДАНИЯ ИЛИ ИНЫЕ МАТЕРИАЛЫ, НЕОБХОДИМЫЕ ДЛЯ ОЦЕНКИ ЗНАНИЙ, УМЕНИЙ, НАВЫКОВ И (ИЛИ) ОПЫТА ДЕЯТЕЛЬНОСТИ, ХАРАКТЕРИЗУЮЩИХ ЭТАПЫ ФОРМИРОВАНИЯ КОМПЕТЕНЦИЙ В ПРОЦЕССЕ ОСВОЕНИЯ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ

Этап формирования компетенций в процессе изучения дисциплины характеризуется следующими типовыми контрольными заданиями.

Типовые контрольные вопросы для подготовки к экзамену при проведении аттестации по дисциплине

1. Типы данных.
2. Переменные.
3. Числовые типы данных.
4. Операции над числовыми типами данных.
5. Строки.
6. Строки unіcod.
7. Вывод данных.
8. Ввод данных.
9. Форматированный ввод/вывод.
10. Списки.
11. Выражения в списках.
12. Оператор del.
13. Использование списков, как стеков.
14. Использование списков, как очередей.
15. Операции сравнения для списков.
16. Диапазоны.
17. Кортежи. Отличие кортежей от словарей
18. Словари.
19. Оператор if. Особенности операторов сравнения.
20. Операторы цикла. Оператор for. Оператор while. Завершение цикла.

21. Продолжение цикла. Оператор pass.
22. Определение функции.
23. Пространство имен функции.
24. Передача параметров. Ключи.
25. Передача в функцию переменного числа аргументов.
26. Элементы функционального программирования.
27. Использование лямбда функций.
28. Функции работы со структурами данных.
29. Функция map(). Примеры применения
30. Функция filter(). Примеры применения
31. Функция reduce(). Примеры применения
32. Документирование функций.
33. Создание модулей.
34. Указание кодировки.
35. Поиск модулей.
36. Компиляция модулей на Python.
37. Стандартные модули Python.
38. Использование функции `__dir()`.
39. Структурирование модулей в пакеты.
40. Импорт модулей и их составляющих из пакета.
41. Ссылки в пакетах.
42. Пакеты и файловая система.
43. Класс File.
44. Открытие файла.
45. Методы класса для File ввода-вывода.
46. Взаимодействие с файловой системой.
47. Модуль path.
48. Объекты и файловый ввод-вывод.
49. Объявление класса
50. Управление атрибутами и методами класса
51. Объявление объектов
52. Множественное наследование
53. Заимствование свойств и методов у родительского класса

Критерии оценки изложены в шкале оценки для проведения промежуточной аттестации по дисциплине в п.6.2.

Типовые практические задачи (задания, тесты) билетов для проведения промежуточной аттестации по дисциплине

ИТОГОВЫЙ ТЕСТ ПО ДИСЦИПЛИНЕ

Вариант 1

Тесты	Контролируемые компетенции
1. Что вернет следующая запись $7//3 + 7//-3$ а) -1 б) 0 в) 1 д) Вернет ошибку	ПК-1
2. Что напечатает следующий код <code>print((1, 2, 3) < (1, 2, 4))</code> а) None	ПК-1

Тесты	Контролируемые компетенции
b) True c) False d) Ошибка	
3. Какое значение получит а? $a = 2,3$ a) 2 b) 3 c) (2,3)	ПК-1
4. Где правильно создана переменная? a) <code>int num = 2</code> b) <code>\$num = 2</code> c) <code>var num = 2</code> d) <code>num = float(2)</code> e) Нет подходящего ответа	ПК-1
5. Что делает следующий код: <code>def a(b, c, d): pass</code> a) Определяет список и инициализирует его b) Определяет функцию, которая ничего не делает c) Определяет функцию, которая передает параметры d) Определяет пустой класс	ПК-1
6. Что выведет следующий код в Python 3.x? <code>print(type(1 / 2))</code> a) <code>class 'int'</code> b) <code>class 'number'</code> c) <code>class 'double'</code> d) <code>class 'tuple'</code> e) <code>class 'float'</code>	ПК-1
7. Какая из функций вернет итерируемый объект? a) <code>len()</code> b) <code>xrange()</code> c) <code>range()</code> d) <code>ord()</code>	ПК-1
8. Каким будет результат $7 / -3$? a) 2 b) -2 c) -3 d) -2.3333333 e) -2.333334	ПК-1
9. Какого типа будет результат следующего выражения: <code>('')</code> a) <code>str(строка)</code> b) <code>tuple(кортеж)</code> c) возникнет синтаксическая ошибка d) <code>unicode строка</code>	ПК-1
10. Укажите все правильные способы использования <code>print</code> : a) <code>print 5</code> b) <code>print "5"</code> c) <code>print(5)</code> d) <code>print(5, "5")</code> e) <code>print() * 5</code>	ПК-1
11. Что такое итерация?	ПК-1

Тесты	Контролируемые компетенции
a) Однократное выполнение какого либо действия. b) Такого понятия не существует. c) Организация обработки данных, при которой действия повторяются многократно. d) Второе название переменной.	
12. Какие операторы итерации существуют? (возможно несколько вариантов ответа) a) for b) from c) def d) while	ПК-1
13. Выберите правильный синтаксис a) for i in list: b) from i in list: c) for i in list d) i in for list:	ПК-1
14. В чем разница между циклами for и while? (возможно несколько вариантов ответа) a) Оператор for выполняет итерацию через функцию коллекции или итерируемого объекта, или генератора. b) Нет разницы. c) Оператор while выполняет итерацию до тех пор, пока условие не будет False. d) while быстрее, чем for.	ПК-1
15. Что делает break? a) Продолжает цикл. b) Останавливает цикл. c) Такого не существует. d) Останавливает выполнение всего кода.	ПК-1

Вариант 2

Тесты	Контролируемые компетенции

Тесты	Контролируемые компетенции
<p>1. Какой результат выполнения этого кода?</p> <pre data-bbox="264 409 727 622"> 3 i = 0 4 while i < 10: 5 i += 1 6 i -= 10 7 print(i) </pre> <p>a) 10 b) 5 c) 0 d) -10</p>	ПК-1
<p>2. Что делает оператор continue?</p> <p>a) Создает бесконечный цикл. b) Останавливает выполнение цикла. c) Начинает следующий проход цикла, минуя оставшееся тело цикла. d) Продолжает выполнять цикл, до оператора break.</p>	ПК-1
<p>3. Для чего else в циклах?</p> <p>a) Не используется в циклах. b) Проверяет, был ли произведен выход из цикла инструкцией break, или же "естественным" образом. c) Проверяет, работает ли цикл. d) Создает условие для цикла.</p>	ПК-1
<p>4. В чем суть наследования?</p> <p>a) Во внедрении других объектов в объект контейнера и их активизации для реализации методов контейнера b) В отношении между суперклассами и подклассами. Первые определяют поведение, а вторые могут наследовать и настраивать его c) Во внедрении других объектов, которым передают запросы на операции объекты контроллеров</p>	ПК-1
<p>5. Во что превращаются суперклассы/подклассы/экземпляры при наследовании?</p> <p>a) В иерархию классов/дерево классов b) В объект контейнера c) Ни во что</p>	ПК-1
<p>6. Какой вид может иметь иерархия классов, образованная с помощью наследования?</p> <p>a) В самом верху класс object, ниже суперкласс, от которого наследуется подкласс и в самом низу экземпляры класса b) В самом верху суперкласс, ниже класс object, от которого наследуется подкласс и в самом низу экземпляры класса c) В самом верху экземпляры класса, ниже подкласс, от которого наследуется суперкласс и в самом низу класс object</p>	ПК-1
<p>7. Как запустить поиск в иерархии классов?</p> <p>a) С помощью выражения объект.атрибут, где объект представляет собой</p>	ПК-1

Тесты	Контролируемые компетенции
<p>объект базового класса или экземпляра</p> <p>b) С помощью выражения объект.атрибут, где объект представляет собой объект экземпляра или класса</p> <p>c) С помощью выражения объект.атрибут, где объект представляет собой объект класса или базового класса</p>	
<p>8. Как происходит поиск в иерархии?</p> <p>a) Сверху вниз</p> <p>b) Снизу вверх</p> <p>c) Этот параметр можно задавать вручную</p>	ПК-1
<p>9. На самом деле иерархия классов явно доступна в специальных атрибутах. Эти атрибуты связывают экземпляры с подклассами, а подклассы с суперклассами. О каких атрибутах идет речь?</p> <p>a) class и bases</p> <p>b) class и bases</p> <p>c) <code>__class__</code> и <code>__bases__</code></p> <p>d) <code>class_</code> и <code>bases_</code></p>	ПК-1
<p>10. Как реализуется настройка? Иными словами, как переопределить подклассу поведение, получаемое от суперкласса?</p> <p>a) Мы должны определить специальный метод, перечислить в нем имена для настройки, и уже тогда настраивать</p> <p>b) Путем переопределения в подклассах атрибутов, наследованных от суперклассов</p> <p>c) Никак, ее невозможно реализовать</p>	ПК-1
<p>11. Выберите верные утверждения про списки:</p> <p>a) списки могут содержать элементы различных типов</p> <p>b) списки неизменяемые</p> <p>c) проверка на вхождение элемента в список происходит за константное время</p> <p>d) проверка на вхождение элемента в список происходит за линейное время</p> <p>e) списки изменяемые</p>	ПК-1
<p>12. К чему приведет обращение к непустому списку по индексу «-1»?</p> <p>a) Вернется последний элемент</p> <p>b) Ошибка IndexError</p> <p>c) Вернется первый элемент</p> <p>d) Ошибка KeyError</p>	ПК-1
<p>13. Выберите верные утверждения про словари:</p> <p>a) словари изменяемые</p> <p>b) поиск ключа в словаре происходит за линейное время</p> <p>c) поиск ключа в словаре происходит за константное время</p> <p>d) словари неизменяемые</p>	ПК-1
<p>14. Можно ли изменять список, находящийся внутри кортежа?</p> <p>a) Нет</p> <p>b) Да</p>	ПК-1
<p>15. В чем отличие стандартного метода списка <code>sort</code> и встроенной функции <code>sorted</code>?</p> <p>a) <code>Sort</code> сортирует исходный список, а <code>sorted</code> возвращает новый</p> <p>b) <code>Sorted</code> сортирует исходный список, а <code>sort</code> возвращает новый</p>	ПК-1

Тесты	Контролируемые компетенции
с) Отличий нет d) Функции sorted не существует	

**6.4 МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ, ОПРЕДЕЛЯЮЩИЕ ПРОЦЕДУРЫ
ОЦЕНИВАНИЯ ЗНАНИЙ, УМЕНИЙ, НАВЫКОВ И (ИЛИ) ОПЫТА
ДЕЯТЕЛЬНОСТИ, ХАРАКТЕРИЗУЮЩИХ ЭТАПЫ ФОРМИРОВАНИЯ
КОМПЕТЕНЦИЙ.**

Формы билетов для проведения промежуточной аттестации по дисциплине

<p>Уральский социально-экономический институт (филиал) Образовательное учреждение профсоюзов высшего образования «Академия труда и социальных отношений» Социально-экономический факультет Кафедра ГЕМД</p> <p>09.03.03 Прикладная информатика Дисциплина: «Программирование на языке Python»</p> <p>Билет к экзамену № 1</p> <p>Вопрос 1. Написать программу выполняющие следующие действия: Создать произвольный список. Добавить новый элемент типа str в конец списка. Добавить новый элемент типа int на место с индексом. Добавить новый элемент типа list в конец списка. Добавить новый элемент типа tuple на место с индексом. Получить элемент по индексу. Удалить элемент. Найти число повторений элемента списка</p> <p>Вопрос 2. Получите первый и последний элемент списка из первого задания.</p> <p>Утверждено на заседании кафедры прикладной информатики и математики, протокол от ____ № ____.</p> <p>Зав. кафедрой _____ <i>И.О. Тимофеева</i></p>
--

Критерии оценки изложены в шкале оценки для проведения промежуточной аттестации по дисциплине (п.6.2.).

Оценка сформированности компетенций

ПК-1. Способен проводить работы по интеграции программных модулей и компонент и проверку работоспособности выпусков программных продуктов.

**Вариант 1
Задание 1.**

Напишите программу, которая считывает три числа и выводит их сумму. Каждое число записано в отдельной строке.

Задание 2.

Напишите программу, которая считывает длины двух катетов в прямоугольном треугольнике и выводит его площадь. Каждое число записано в отдельной строке.

Задание 3.

n студентов делят k яблок поровну, неделившийся остаток остается в корзинке. Сколько яблок достанется каждому студенту? Сколько яблок останется в корзинке? Программа получает на вход числа n и k и должна вывести искомое количество яблок (два числа).

Задание 4.

Дано число n . С начала суток прошло n минут. Определите, сколько часов и минут будут показывать электронные часы в этот момент. Программа должна вывести два числа: количество часов (от 0 до 23) и количество минут (от 0 до 59). Учтите, что число n может быть больше, чем количество минут в сутках.

Задание 5.

Шоколадка имеет вид прямоугольника, разделенного на $n \times m$ долек. Шоколадку можно один раз разломить по прямой на две части. Определите, можно ли таким образом отломить от шоколадки часть, состоящую ровно из k долек. Программа получает на вход три числа: n , m , k и должна вывести YES или NO.

Вариант 2

Задание 1.

Шахматный король ходит по горизонтали, вертикали и диагонали, но только на 1 клетку. Даны две различные клетки шахматной доски, определите, может ли король попасть с первой клетки на вторую одним ходом. Программа получает на вход четыре числа от 1 до 8 каждое, задающие номер столбца и номер строки сначала для первой клетки, потом для второй клетки. Программа должна вывести YES, если из первой клетки ходом короля можно попасть во вторую или NO в противном случае.

Задание 2.

Длина Московской кольцевой автомобильной дороги — 109 километров. Байкер Вася стартует с нулевого километра МКАД и едет со скоростью v километров в час. На какой отметке он остановится через t часов?

Программа получает на вход значение v и t . Если $v > 0$, то Вася движется в положительном направлении по МКАД, если же значение $v < 0$, то в отрицательном.

Программа должна вывести целое число от 0 до 108 — номер отметки, на которой остановится Вася..

Задание 3.

Пирожок в столовой стоит a рублей и b копеек. Определите, сколько рублей и копеек нужно заплатить за n пирожков. Программа получает на вход три числа: a , b , n , и должна вывести два числа: стоимость покупки в рублях и копейках.

Задание 4.

По данному натуральному $n \leq 9$ выведите лесенку из n ступенек, i -я ступенька состоит из чисел от 1 до i без пробелов.

Задание 5.

Август и Беатриса играют в игру. Август загадал натуральное число от 1 до n . Беатриса пытается угадать это число, для этого она называет некоторые множества натуральных чисел. Август отвечает Беатрисе YES, если среди названных ей чисел есть задуманное или NO в противном случае. После нескольких заданных вопросов Беатриса запуталась в том, какие вопросы она задавала и какие ответы получила и просит вас помочь ей определить, какие числа мог задумать Август.

В первой строке задано n - максимальное число, которое мог загадать Август. Далее каждая строка содержит вопрос Беатрисы (множество чисел, разделенных пробелом) и ответ Августа на этот вопрос.

Вы должны вывести через пробел, в порядке возрастания, все числа, которые мог задумать Август.

ЛИСТ СОГЛАСОВАНИЯ

№п/п	Подразделение	Фамилия	Подпись	Дата
1	Кафедра ГЕМД	Тимофеева И.О.		10.06.2020
2	Библиотека	Шпакова Г.В.		10.06.2020
3	УМО	Дерябичева М.О.		10.06.2020