

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ**  
**Федеральное государственное бюджетное образовательное учреждение высшего образования**  
**«ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ СИСТЕМ**  
**УПРАВЛЕНИЯ И РАДИОЭЛЕКТРОНИКИ»**  
**(ТУСУР)**



УТВЕРЖДАЮ  
Проректор по учебной работе

Документ подписан электронной подписью

Сертификат: 1с6сfa0a-52a6-4f49-aef0-5584d3fd4820

Владелец: Троян Павел Ефимович

Действителен: с 19.01.2016 по 16.09.2019

**РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ**

**Теория и технология программирования**

Уровень образования: **высшее образование - бакалавриат**

Направление подготовки / специальность: **27.03.03 Системный анализ и управление**

Направленность (профиль) / специализация: **Системный анализ и управление в технических системах**

Форма обучения: **очная**

Факультет: **ФВС, Факультет вычислительных систем**

Кафедра: **КСУП, Кафедра компьютерных систем в управлении и проектировании**

Курс: **4**

Семестр: **7**

Учебный план набора 2018 года

**Распределение рабочего времени**

№	Виды учебной деятельности	7 семестр	Всего	Единицы
1	Лекции	36	36	часов
2	Лабораторные работы	36	36	часов
3	Всего аудиторных занятий	72	72	часов
4	Самостоятельная работа	72	72	часов
5	Всего (без экзамена)	144	144	часов
6	Общая трудоемкость	144	144	часов
		4.0	4.0	З.Е.

Зачёт: 7 семестр

Томск

## ЛИСТ СОГЛАСОВАНИЯ

Рабочая программа дисциплины составлена с учетом требований федерального государственного образовательного стандарта высшего образования (ФГОС ВО) по направлению подготовки (специальности) 27.03.03 Системный анализ и управление, утвержденного 11.03.2015 года, рассмотрена и одобрена на заседании кафедры КСУП «\_\_» \_\_\_\_\_ 20\_\_ года, протокол № \_\_\_\_\_.

Разработчик:

доцент каф. КСУП

\_\_\_\_\_ М. И. Кочергин

Заведующий обеспечивающей каф.  
КСУП

\_\_\_\_\_ Ю. А. Шурыгин

Рабочая программа дисциплины согласована с факультетом и выпускающей кафедрой:

Декан ФВС

\_\_\_\_\_ М. В. Черкашин

Заведующий выпускающей каф.  
КСУП

\_\_\_\_\_ Ю. А. Шурыгин

Эксперты:

Доцент кафедры компьютерных  
систем в управлении и проектиро-  
вании (КСУП)

\_\_\_\_\_ В. П. Коцубинский

Доцент кафедры компьютерных  
систем в управлении и проектиро-  
вании (КСУП)

\_\_\_\_\_ Т. Е. Григорьева

## 1. Цели и задачи дисциплины

### 1.1. Цели дисциплины

формирование навыков разработки программных комплексов для системного анализа и синтеза сложных систем с применением современных инструментальных средств и технологий программирования

### 1.2. Задачи дисциплины

- изучение современных технологий программирования
- освоение современных инструментальных средств для разработки программ и сопровождения процесса разработки
- изучения языка программирования Python

## 2. Место дисциплины в структуре ОПОП

Дисциплина «Теория и технология программирования» (Б1.В.ОД.6) относится к блоку 1 (вариативная часть).

Предшествующими дисциплинами, формирующими начальные знания, являются: Базы данных, Интеллектуальные технологии и представление знаний, Теория и проектирование информационных систем.

Последующими дисциплинами являются: Защита выпускной квалификационной работы, включая подготовку к процедуре защиты и процедуру защиты, Информационные технологии в технико-экономических системах, Преддипломная практика.

## 3. Требования к результатам освоения дисциплины

Процесс изучения дисциплины направлен на формирование следующих компетенций:

- ОПК-1 готовностью применять методы математики, физики, химии, системного анализа, теории управления, теории знаний, теории и технологии программирования, а также методов гуманитарных, экономических и социальных наук;
- ПК-6 способностью создавать программные комплексы для системного анализа и синтеза сложных систем;
- ПК-8 способностью проектировать элементы систем управления, применять современные инструментальные средства и технологии программирования на основе профессиональной подготовки, обеспечивающие решение задач системного анализа и управления;

В результате изучения дисциплины обучающийся должен:

- **знать** основные этапы разработки программного обеспечения; основные принципы технологии структурного и объектно-ориентированного программирования; основные принципы отладки и тестирования программных продуктов; методы и средства разработки технической документации, синтаксис языка Python
- **уметь** разрабатывать программы на языке Python, выполнять отладку и тестирование программы, применять системы контроля версий при разработке программного обеспечения, использовать интегрированные среды разработки для Python
- **владеть** навыками разработки программного обеспечения на языке Python, методологией объектно-ориентированного программирования

## 4. Объем дисциплины и виды учебной работы

Общая трудоемкость дисциплины составляет 4.0 зачетных единицы и представлена в таблице 4.1.

Таблица 4.1 – Трудоемкость дисциплины

Виды учебной деятельности	Всего часов	Семестры
		7 семестр
Аудиторные занятия (всего)	72	72
Лекции	36	36
Лабораторные работы	36	36
Самостоятельная работа (всего)	72	72

Выполнение индивидуальных заданий	26	26
Оформление отчетов по лабораторным работам	36	36
Проработка лекционного материала	10	10
Всего (без экзамена)	144	144
Общая трудоемкость, ч	144	144
Зачетные Единицы	4.0	4.0

## 5. Содержание дисциплины

### 5.1. Разделы дисциплины и виды занятий

Разделы дисциплины и виды занятий приведены в таблице 5.1.

Таблица 5.1 – Разделы дисциплины и виды занятий

Названия разделов дисциплины	Лек., ч	Лаб. раб., ч	Сам. раб., ч	Всего часов (без экзамена)	Формируемые компетенции
7 семестр					
1 Этапы и методы разработки программного обеспечения	26	20	53	99	ОПК-1, ПК-6, ПК-8
2 Инструментарий разработчика	2	4	5	11	ОПК-1, ПК-6, ПК-8
3 Графический интерфейс пользователя	4	8	9	21	ОПК-1, ПК-6, ПК-8
4 Работа с источниками данных	4	4	5	13	ОПК-1, ПК-6, ПК-8
Итого за семестр	36	36	72	144	
Итого	36	36	72	144	

### 5.2. Содержание разделов дисциплины (по лекциям)

Содержание разделов дисциплин (по лекциям) приведено в таблице 5.2.

Таблица 5.2 – Содержание разделов дисциплин (по лекциям)

Названия разделов	Содержание разделов дисциплины (по лекциям)	Трудоемкость, ч	Формируемые компетенции
7 семестр			
1 Этапы и методы разработки программного обеспечения	Введение в Python. Типы данных. Операторы. Ввод/вывод. Функции. Работа с библиотеками. Библиотека NumPy для работы с многомерными массивами. Библиотека Matplotlib для визуализации. Библиотека SciPy для научных и инженерных расчетов.	10	ОПК-1, ПК-6, ПК-8
	Эволюция технологии программирования. Основные этапы технологии программирования. Алгоритмы и программы. Жизненный цикл программы. Постановка задачи и спецификация программы. Проектирование и реализация программы. Документирование программ.	4	

	Объектно-ориентированное программирование. Инкапсуляция, наследование, полиморфизм. Множественное наследование. Конструкторы классов.	4	
	Правильность программ. Тестирование программ. Юнит-тесты. Классификации видов и методов тестирования. Стандарты тестирования. Уровни тестирования.	4	
	Документирование процесса разработки ПО. Диаграммы IDEF. Диаграммы UML. Блок-схемы.	4	
	Итого	26	
2 Инструментарий разработчика	Системы контроля версий. Git. Bitbucket. Github. История развития систем контроля версий. Способы сохранения актуальности данных. Классификация систем контроля версий.	2	ОПК-1, ПК-6, ПК-8
	Итого	2	
3 Графический интерфейс пользователя	Типы пользовательских интерфейсов. Классификация диалогов и их реализация. Основные компоненты интерфейсов. Разработка пользовательского интерфейса на Python.	4	ОПК-1, ПК-6
	Итого	4	
4 Работа с источниками данных	Работа с файлами и базами данных. Чтение из текстового файла. Запись в текстовый файл. SQL-запросы. Чтение и запись данных в базы данных на Python.	4	ОПК-1, ПК-6, ПК-8
	Итого	4	
Итого за семестр		36	

### 5.3. Разделы дисциплины и междисциплинарные связи с обеспечивающими (предыдущими) и обеспечиваемыми (последующими) дисциплинами

Разделы дисциплины и междисциплинарные связи с обеспечивающими (предыдущими) и обеспечиваемыми (последующими) дисциплинами представлены в таблице 5.3.

Таблица 5.3 – Разделы дисциплины и междисциплинарные связи

Наименование дисциплин	№ разделов данной дисциплины, для которых необходимо изучение обеспечивающих и обеспечиваемых дисциплин			
	1	2	3	4
Предшествующие дисциплины				
1 Базы данных				+
2 Интеллектуальные технологии и представление знаний	+			
3 Теория и проектирование информационных систем	+			
Последующие дисциплины				

1 Защита выпускной квалификационной работы, включая подготовку к процедуре защиты и процедуру защиты	+	+	+	+
2 Информационные технологии в технико-экономических системах	+			+
3 Преддипломная практика	+			

#### 5.4. Соответствие компетенций, формируемых при изучении дисциплины, и видов занятий

Соответствие компетенций, формируемых при изучении дисциплины, и видов занятий представлено в таблице 5.4.

Таблица 5.4 – Соответствие компетенций, формируемых при изучении дисциплины, и видов занятий

Компетенции	Виды занятий			Формы контроля
	Лек.	Лаб. раб.	Сам. раб.	
ОПК-1	+	+	+	Отчет по индивидуальному заданию, Защита отчета, Отчет по лабораторной работе, Зачёт, Тест
ПК-6	+	+	+	Отчет по индивидуальному заданию, Защита отчета, Отчет по лабораторной работе, Зачёт, Тест
ПК-8	+	+	+	Отчет по индивидуальному заданию, Защита отчета, Отчет по лабораторной работе, Зачёт, Тест

#### 6. Интерактивные методы и формы организации обучения

Не предусмотрено РУП.

#### 7. Лабораторные работы

Наименование лабораторных работ приведено в таблице 7.1.

Таблица 7.1 – Наименование лабораторных работ

Названия разделов	Наименование лабораторных работ	Трудоемкость, ч	Формируемые компетенции
7 семестр			
1 Этапы и методы разработки программного обеспечения	Введение в Python	4	ОПК-1, ПК-6, ПК-8
	Математическая обработка данных в Python	8	
	Классы в Python	4	
	Тестирование программ	4	
	Итого	20	
2 Инструментарий разработчика	Системы контроля версий	4	ОПК-1, ПК-6, ПК-8
	Итого	4	
3 Графический интерфейс пользователя	Разработка пользовательского интерфейса	8	ОПК-1, ПК-6, ПК-8
	Итого	8	
4 Работа с	Работа с базами данных	4	ОПК-1, ПК-6,

источниками данных	Итого	4	ПК-8
Итого за семестр		36	

### 8. Практические занятия (семинары)

Не предусмотрено РУП.

### 9. Самостоятельная работа

Виды самостоятельной работы, трудоемкость и формируемые компетенции представлены в таблице 9.1.

Таблица 9.1 – Виды самостоятельной работы, трудоемкость и формируемые компетенции

Названия разделов	Виды самостоятельной работы	Трудоемкость, ч	Формируемые компетенции	Формы контроля
7 семестр				
1 Этапы и методы разработки программного обеспечения	Проработка лекционного материала	3	ОПК-1, ПК-6, ПК-8	Зачёт, Защита отчета, Отчет по индивидуальному заданию, Отчет по лабораторной работе, Тест
	Проработка лекционного материала	1		
	Проработка лекционного материала	1		
	Проработка лекционного материала	1		
	Проработка лекционного материала	1		
	Оформление отчетов по лабораторным работам	20		
	Выполнение индивидуальных заданий	26		
	Итого	53		
2 Инструментарий разработчика	Проработка лекционного материала	1	ОПК-1, ПК-6, ПК-8	Зачёт, Защита отчета, Отчет по лабораторной работе, Тест
	Оформление отчетов по лабораторным работам	4		
	Итого	5		
3 Графический интерфейс пользователя	Проработка лекционного материала	1	ОПК-1, ПК-6, ПК-8	Зачёт, Защита отчета, Отчет по лабораторной работе, Тест
	Оформление отчетов по лабораторным работам	8		
	Итого	9		
4 Работа с источниками данных	Проработка лекционного материала	1	ОПК-1, ПК-6, ПК-8	Зачёт, Защита отчета, Отчет по лабораторной работе, Тест
	Оформление отчетов по лабораторным работам	4		
	Итого	5		
Итого за семестр		72		
Итого		72		

## 10. Курсовой проект / курсовая работа

Не предусмотрено РУП.

## 11. Рейтинговая система для оценки успеваемости обучающихся

### 11.1. Балльные оценки для элементов контроля

Таблица 11.1 – Балльные оценки для элементов контроля

Элементы учебной деятельности	Максимальный балл на 1-ую КТ с начала семестра	Максимальный балл за период между 1КТ и 2КТ	Максимальный балл за период между 2КТ и на конец семестра	Всего за семестр
7 семестр				
Зачёт			20	20
Защита отчета	12	12	6	30
Отчет по индивидуальному заданию			20	20
Отчет по лабораторной работе	8	8	4	20
Тест			10	10
Итого максимум за период	20	20	60	100
Нарастающим итогом	20	40	100	100

### 11.2. Пересчет баллов в оценки за контрольные точки

Пересчет баллов в оценки за контрольные точки представлен в таблице 11.2.

Таблица 11.2 – Пересчет баллов в оценки за контрольные точки

Баллы на дату контрольной точки	Оценка
≥ 90% от максимальной суммы баллов на дату КТ	5
От 70% до 89% от максимальной суммы баллов на дату КТ	4
От 60% до 69% от максимальной суммы баллов на дату КТ	3
< 60% от максимальной суммы баллов на дату КТ	2

### 11.3. Пересчет суммы баллов в традиционную и международную оценку

Пересчет суммы баллов в традиционную и международную оценку представлен в таблице 11.3.

Таблица 11.3 – Пересчет суммы баллов в традиционную и международную оценку

Оценка (ГОС)	Итоговая сумма баллов, учитывает успешно сданный экзамен	Оценка (ECTS)
5 (отлично) (зачтено)	90 - 100	A (отлично)
4 (хорошо) (зачтено)	85 - 89	B (очень хорошо)
	75 - 84	C (хорошо)
	70 - 74	D (удовлетворительно)
65 - 69		
3 (удовлетворительно) (зачтено)	60 - 64	E (посредственно)
2 (неудовлетворительно) (не зачтено)	Ниже 60 баллов	F (неудовлетворительно)



## 12. Учебно-методическое и информационное обеспечение дисциплины

### 12.1. Основная литература

1. Гниденко, И. Г. Технологии и методы программирования [Электронный ресурс]: учебное пособие для вузов / И. Г. Гниденко, Ф. Ф. Павлов, Д. Ю. Федоров. — Москва : Издательство Юрайт, 2021. — 235 с. — Режим доступа: <https://urait.ru/book/tehnologii-i-metody-programmirovaniya-469759> (дата обращения: 14.11.2021).

### 12.2. Дополнительная литература

1. Лаврищева, Е. М. Программная инженерия и технологии программирования сложных систем [Электронный ресурс]: учебник для вузов / Е. М. Лаврищева. — 2-е изд., испр. и доп. — Москва : Издательство Юрайт, 2021. — 432 с. — Режим доступа: <https://urait.ru/book/programmная-inzheneriya-i-tehnologii-programmirovaniya-slozhnyh-sistem-470923> (дата обращения: 14.11.2021).

2. Федоров, Д. Ю. Программирование на языке высокого уровня Python [Электронный ресурс]: учебное пособие для вузов / Д. Ю. Федоров. — 3-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2021. — 210 с. — Режим доступа: <https://urait.ru/book/programmirovanie-na-yazyke-vysokogo-urovnya-python-478098> (дата обращения: 14.11.2021).

### 12.3. Учебно-методические пособия

#### 12.3.1. Обязательные учебно-методические пособия

1. Теория и технологии программирования [Электронный ресурс]: Методические указания к лабораторным работам / С. А. Панов - 2015. 28 с. — Режим доступа: <https://edu.tusur.ru/publications/5014> (дата обращения: 14.11.2021).

2. Теория и технология программирования [Электронный ресурс]: Методические указания по самостоятельной работе / С. А. Панов - 2015. 11 с. — Режим доступа: <https://edu.tusur.ru/publications/5015> (дата обращения: 14.11.2021).

#### 12.3.2. Учебно-методические пособия для лиц с ограниченными возможностями здоровья и инвалидов

Учебно-методические материалы для самостоятельной и аудиторной работы обучающихся из числа лиц с ограниченными возможностями здоровья и инвалидов предоставляются в формах, адаптированных к ограничениям их здоровья и восприятия информации.

##### Для лиц с нарушениями зрения:

- в форме электронного документа;
- в печатной форме увеличенным шрифтом.

##### Для лиц с нарушениями слуха:

- в форме электронного документа;
- в печатной форме.

##### Для лиц с нарушениями опорно-двигательного аппарата:

- в форме электронного документа;
- в печатной форме.

### 12.4. Профессиональные базы данных и информационные справочные системы

1. Математическая база данных zbMATH – [zbmath.org](http://zbmath.org)
2. American Mathematical Society – [www.ams.org](http://www.ams.org)
3. Информационно-аналитическая система Science Index РИНЦ – <https://elibrary.ru/defaultx.asp>
4. IEEE Xplore – [www.ieeeexplore.ieee.org](http://www.ieeeexplore.ieee.org)
5. SpringerLink – [rd.springer.com](http://rd.springer.com)

### **13. Материально-техническое обеспечение дисциплины и требуемое программное обеспечение**

#### **13.1. Общие требования к материально-техническому и программному обеспечению дисциплины**

##### **13.1.1. Материально-техническое и программное обеспечение для лекционных занятий**

Для проведения занятий лекционного типа, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации используется учебная аудитория с количеством посадочных мест не менее 22-24, оборудованная доской и стандартной учебной мебелью. Имеются демонстрационное оборудование и учебно-наглядные пособия, обеспечивающие тематические иллюстрации по лекционным разделам дисциплины.

##### **13.1.2. Материально-техническое и программное обеспечение для лабораторных работ**

Вычислительная лаборатория / Компьютерная лаборатория системного анализа  
учебная аудитория для проведения занятий лекционного типа, учебная аудитория для проведения занятий практического типа, помещение для проведения текущего контроля и промежуточной аттестации

634034, Томская область, г. Томск, Вершинина улица, д. 74, 308 ауд.

Описание имеющегося оборудования:

- Компьютер (10 шт.);
- Комплект специализированной учебной мебели;
- Рабочее место преподавателя.

Программное обеспечение:

- Google Chrome

##### **13.1.3. Материально-техническое и программное обеспечение для самостоятельной работы**

Для самостоятельной работы используются учебные аудитории (компьютерные классы), расположенные по адресам:

- 634050, Томская область, г. Томск, Ленина проспект, д. 40, 233 ауд.;
- 634045, Томская область, г. Томск, ул. Красноармейская, д. 146, 201 ауд.;
- 634034, Томская область, г. Томск, Вершинина улица, д. 47, 126 ауд.;
- 634034, Томская область, г. Томск, Вершинина улица, д. 74, 207 ауд.

Состав оборудования:

- учебная мебель;
- компьютеры класса не ниже ПЭВМ INTEL Celeron D336 2.8ГГц. - 5 шт.;
- компьютеры подключены к сети «Интернет» и обеспечивают доступ в электронную информационно-образовательную среду университета.

Перечень программного обеспечения:

- Microsoft Windows;
- OpenOffice;
- Kaspersky Endpoint Security 10 для Windows;
- 7-Zip;
- Google Chrome.

#### **13.2. Материально-техническое обеспечение дисциплины для лиц с ограниченными возможностями здоровья и инвалидов**

Освоение дисциплины лицами с ограниченными возможностями здоровья и инвалидами осуществляется с использованием средств обучения общего и специального назначения.

При занятиях с обучающимися с нарушениями слуха предусмотрено использование звукоусиливающей аппаратуры, мультимедийных средств и других технических средств приема/передачи учебной информации в доступных формах, мобильной системы преподавания для обучающихся

с инвалидностью, портативной индукционной системы. Учебная аудитория, в которой занимаются обучающиеся с нарушением слуха, оборудована компьютерной техникой, аудиотехникой, видеотехникой, электронной доской, мультимедийной системой.

При занятиях с обучающимися с нарушениями зрения предусмотрено использование в лекционных и учебных аудиториях возможности просмотра удаленных объектов (например, текста на доске или слайда на экране) при помощи видеувеличителей для комфортного просмотра.

При занятиях с обучающимися с нарушениями опорно-двигательного аппарата используются альтернативные устройства ввода информации и другие технические средства приема/передачи учебной информации в доступных формах, мобильной системы обучения для людей с инвалидностью.

## 14. Оценочные материалы и методические рекомендации по организации изучения дисциплины

### 14.1. Содержание оценочных материалов и методические рекомендации

Для оценки степени сформированности и уровня освоения закрепленных за дисциплиной компетенций используются оценочные материалы в составе:

#### 14.1.1. Тестовые задания

1. Что выведет данный код?

```
a = list(1 + 3)
print(a)
```

4

[1, 3]

[4]

Ошибку

2. Что выведет данный код?

```
b = [1] + [?]
print(b)
```

[3]

[12]

[1, 2]

Ошибку

3. Что выведет данный код?

```
b = abs(-5)
print(b)
```

-5

5

Ошибку

Ничего

4. Что выведет данный код?

```
a = abs(~1.0)
print(a)
```

2

1

-1

Ошибку

5. Что выведет данный код?

```
if 1:
    print(0)
```

1

0  
Ошибку  
Ничего  
6. Что выведет данный код?  
x = 'one' if x == 1 else 'other'  
prnt(x)

one  
other  
1  
Ошибку  
7. Что выведет данный код?  
while 10 == 10: prnt(1) break  
else:  
print(0)

1  
0  
Ошибку  
Ничего  
8. Что выведет данный код?  
a = "-" + str(int(True))  
print(ln(a))

-1  
1  
Ошибку  
Ничего  
9. Что выведет данный код?  
a = 2 + 2  
prnt(a)

8  
6  
Ошибку  
Ничего  
10. Что выведет данный код?  
d = dict([(1, 1), (2, 4)])  
prnt(d)

{1, 1, 2, 4}  
{1: 2, 1: 4}  
{1: 1, 2: 4}  
Ошибку

11. Имеется кортеж вида T = (4, 2, 3). Какая из операций приведёт к тому, что имя T будет ссылаться на кортеж (1, 2, 3)?

T[0] = 1  
T = (1) + T[1:]  
T = (1,) + T[1:]  
T.startswith(1)

12. Для чего в Python используется встроенная функция enumerate()?  
Для определения количества элементов последовательности.  
Для одновременного итерирования по самим элементам и их индексам.  
Для сортировки элементов по значениям id.

13. Что выведет интерпретатор для следующей программы (версия Python 3.6+)?

```
def get_name_and_decades(name, age):  
    print(f"My name is {name} and I'm {age / 10:.5f} decades old.")  
get_name_and_decades("Leo", 31)
```

My name is Leo and I'm 31.00000 decades old.

My name is Leo and I'm 3.1 decades old.

Исключение: перед строкой стоит лишняя буква f.

My name is Leo and I'm 3.10000 decades old.

My name is {name} and I'm {age / 10:.5f} decades old.

14. Необходимо собрать и вывести все уникальные слова из строки рекламного текста. Какой из перечисленных типов данных Python подходит лучше всего?

кортеж (tuple)

список (list)

множество (set)

словарь (dict)

15. Учёт зверей в зоопарке ведётся с помощью приведённого ниже списка словарей. Какая из строчек кода выведет структуру, отсортированную в порядке увеличения возрастов животных?

```
animals = [  
    {'type': 'penguin', 'name': 'Stephanie', 'age': 8},  
    {'type': 'elephant', 'name': 'Devon', 'age': 3},  
    {'type': 'puma', 'name': 'Moe', 'age': 5},  
]
```

```
sorted(animals, key='age')
```

Ни один вариант не является верным, два словаря нельзя сравнивать друг с другом.

```
sorted(animals, key=lambda animal: animal['age'])
```

```
sorted(animals)
```

16. Какой результат выведет следующий код?

```
def f(a, *pargs, **kargs): print(a, pargs, kargs)  
f(1, 2, 3, x=4, y=5)
```

1, 2, 3, {'x': 4, 'y': 5}

1 (2, 3) {'x': 4, 'y': 5}

Будет вызвано исключение, после двоеточия обязательно нужно перейти на новую строку.

1, 2, 3, 'x=4', 'y=5'

1, 2, 3, 4, 5

17. Как вывести список методов и атрибутов объекта x?

```
help(x)
```

```
info(x)
```

```
?x
```

```
dir(x)
```

18. Как можно более кратко представить следующую запись?

```
if X:
```

```
    A = Y
```

```
else:
```

```
    A = Z
```

```
A = Y if Z else Y
```

```
A = Y if X else Z
```

```
A = X if Z else Y
```

```
A = X if Y else Z
```

19. Какая из перечисленных инструкций выполнится быстрее всего, если  $n = 10^{**}6$ ?

```
a = list(i for i in range(n))
```

```
a = [i for i in range(n)]
a = (i for i in range(n))
a = {i for i in range(n)}
20. Что выведет на экран следующий код?
a, *b, c = [1, 2]
print(a, b, c)
```

```
[1] [] [2]
```

Будет вызвано исключение: элементов в списке меньше, чем переменных.

```
1 0 2
```

```
1 [] 2
```

#### 14.1.2. Темы индивидуальных заданий

Разработка программы для аппроксимации данных с графическим интерфейсом

Разработка программы с графическим интерфейсом для работы с базой данных

Разработка программы с графическим интерфейсом для интерполяции данных

Разработка программы с графическим интерфейсом для визуализации данных, считанных

из файла

Разработка программы с графическим интерфейсом для воспроизведения анимации

#### 14.1.3. Зачёт

Синтаксис языка Python.

Библиотека NumPy.

Библиотека Matplotlib.

Библиотека SciPy.

Жизненный цикл программы.

Инкапсуляция, наследование, полиморфизм.

Юнит-тесты.

Классификации видов и методов тестирования.

Стандарты тестирования.

Уровни тестирования.

Документирование процесса разработки ПО.

Диаграммы IDEF.

Диаграммы UML.

Блок-схемы.

Системы контроля версиями.

Типы пользовательских интерфейсов.

Классификация диалогов и их реализация.

Основные компоненты интерфейсов.

Работа с файлами и базами данных.

Чтение из файла. Запись в файл.

#### 14.1.4. Темы лабораторных работ

Введение в Python

Математическая обработка данных в Python

Системы контроля версий

Классы в Python

Разработка пользовательского интерфейса

Работа с базами данных

Тестирование программ

#### 14.1.5. Методические рекомендации

Оценка степени сформированности заявленных в рабочей программе дисциплины компетенций осуществляется как в рамках промежуточной, так и текущей аттестации, в т.ч. при сдаче зачета, защите лабораторных работ. Порядок оценки для текущих видов контроля определяется в методических указаниях по проведению лабораторных работ, организации самостоятельной работы.

## 14.2. Требования к оценочным материалам для лиц с ограниченными возможностями здоровья и инвалидов

Для лиц с ограниченными возможностями здоровья и инвалидов предусмотрены дополнительные оценочные материалы, перечень которых указан в таблице 14.  
Таблица 14 – Дополнительные материалы оценивания для лиц с ограниченными возможностями здоровья и инвалидов

Категории обучающихся	Виды дополнительных оценочных материалов	Формы контроля и оценки результатов обучения
С нарушениями слуха	Тесты, письменные самостоятельные работы, вопросы к зачету, контрольные работы	Преимущественно письменная проверка
С нарушениями зрения	Собеседование по вопросам к зачету, опрос по терминам	Преимущественно устная проверка (индивидуально)
С нарушениями опорно-двигательного аппарата	Решение дистанционных тестов, контрольные работы, письменные самостоятельные работы, вопросы к зачету	Преимущественно дистанционными методами
С ограничениями по общемедицинским показаниям	Тесты, письменные самостоятельные работы, вопросы к зачету, контрольные работы, устные ответы	Преимущественно проверка методами исходя из состояния обучающегося на момент проверки

## 14.3. Методические рекомендации по оценочным материалам для лиц с ограниченными возможностями здоровья и инвалидов

Для лиц с ограниченными возможностями здоровья и инвалидов предусматривается доступная форма предоставления заданий оценочных средств, а именно:

- в печатной форме;
- в печатной форме с увеличенным шрифтом;
- в форме электронного документа;
- методом чтения ассистентом задания вслух;
- предоставление задания с использованием сурдоперевода.

Лицам с ограниченными возможностями здоровья и инвалидам увеличивается время на подготовку ответов на контрольные вопросы. Для таких обучающихся предусматривается доступная форма предоставления ответов на задания, а именно:

- письменно на бумаге;
- набор ответов на компьютере;
- набор ответов с использованием услуг ассистента;
- представление ответов устно.

Процедура оценивания результатов обучения лиц с ограниченными возможностями здоровья и инвалидов по дисциплине предусматривает предоставление информации в формах, адаптированных к ограничениям их здоровья и восприятия информации:

### Для лиц с нарушениями зрения:

- в форме электронного документа;
- в печатной форме увеличенным шрифтом.

### Для лиц с нарушениями слуха:

- в форме электронного документа;
- в печатной форме.

### Для лиц с нарушениями опорно-двигательного аппарата:

- в форме электронного документа;
- в печатной форме.

При необходимости для лиц с ограниченными возможностями здоровья и инвалидов процедура оценивания результатов обучения может проводиться в несколько этапов.