

Документ подписан простой электронной подписью

Информация о владельце:

ФИО: Минцаев Магомед Шавалович

Должность: Ректор

Дата подписания: 2020.03.20

Уникальный программный ключ:

236bcc35c296f119d6aafdc22836b21db52dbc079718866385825f7a4304cc

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

ГРОЗНЕНСКИЙ ГОСУДАРСТВЕННЫЙ НЕФТЯНОЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

имени академика М.Д. Миллионщикова

«УТВЕРЖДАЮ»

Первый проректор

И.Г. Гайрабеков



« 20 » 2020 г.

## РАБОЧАЯ ПРОГРАММА

дисциплины

*«Объектно - ориентированное программирование»*

### Направление подготовки

*09.03.02 Информационные системы и технологии*

### Направленности (профили)

*«Информационные системы и технологии»*

*«Информационные технологии в образовании»*

*«Информационные технологии в дизайне»*

### Квалификация

*бакалавр*

Грозный – 2020

## 1. Цели и задачи дисциплины

Цель преподавания дисциплины «Объектно-ориентированное программирование» состоит в углублении студентами, получающими квалификацию бакалавра, знаний и навыков в области создания приложений, ознакомлении с принципами объектно-ориентированного подхода к проектированию и разработке программ.

Задачами дисциплины являются: изучение теоретических основ современного объектно-ориентированного программирования (ООП) и получение практических навыков применения парадигмы ООП при разработке сложных программ.

## 2. Место дисциплины в структуре образовательной программы

Учебная дисциплина «Объектно-ориентированное программирование» относится к части, формируемой участниками образовательных отношений, ФГОС ВО по направлению подготовки 09.03.02 Информационные системы и технологии (квалификация «бакалавр»).

Для освоения дисциплины «Объектно-ориентированное программирование» студент должен обладать знаниями и умениями, приобретенными в результате освоения предшествующих дисциплин:

- технологии программирования;
- программирование;
- информационные технологии.

Дисциплина «Объектно-ориентированное программирование» является предшествующей и необходимой для изучения следующих дисциплин:

- web-программирование;
- методы и средства проектирования информационных систем и технологий.

## 3. Требования к результатам освоения дисциплины

Процесс изучения дисциплины направлен на формирование следующих компетенций и индикаторов их достижения:

**ОПК-1.** Способность применять естественнонаучные и общетехнические знания, методы математического анализа и моделирования, теоретического и экспериментального исследования в профессиональной деятельности:

– **ИД-1оПК-1-знать:** основы математики, физики, вычислительной техники и программирования;

– **ИД-2оПК-1-уметь:** решать стандартные профессиональные задачи с применением естественнонаучных и общетехнических знаний, методов математического анализа и моделирования;

– **ИД-3оПК-1-иметь навыки:** теоретического и экспериментального исследования объектов профессиональной деятельности.

**ОПК-6.** Способность разрабатывать алгоритмы и программы, пригодные для практического применения в области информационных систем и технологий:

– **ИД-1оПК-6-знать:** основные языки программирования и работы с базами данных, операционные системы и оболочки, современные программные среды разработки информационных систем и технологий;

– **ИД-2оПК-6-уметь:** применять языки программирования и работы с базами данных, современные программные среды разработки информационных систем и

технологий для автоматизации бизнес-процессов, решения прикладных задач различных классов, ведения баз данных и информационных хранилищ;

– **ИД-3опк-6-иметь навыки:** программирования, отладки и тестирования прототипов программно-технических комплексов задач.

**ПК-1.** Способность разрабатывать программное обеспечение (ПО), включая проектирование, отладку, проверку работоспособности и модификацию ПО:

– **ИД-1пк-1-знать:** процедуры интеграции программных модулей;

– **ИД-2пк-1-уметь:** проводить анализ требований к программному обеспечению;

– **ИД-3пк-1-иметь навыки:** разработки технических спецификаций на программные компоненты и их взаимодействие.

**ПК-2.** Способность оценивать качество разрабатываемого программного обеспечения ИС:

– **ИД-1пк-2-знать:** методы тестирования по разработанным тестовым случаям;

– **ИД-2пк-1-уметь:** проводить анализ результатов тестирования;

– **ИД-3пк-1-иметь навыки:** разработки тестовых документов.

#### 4. Объем дисциплины и виды учебной работы

Таблица 1

Вид учебной работы	Всего часов / зач. ед.		Семестры				
			4	5	4	5	
	ОФО	ЗФО	ОФО		ЗФО		
<b>Контактная работа (всего)</b>	<b>99/2,7</b>	<b>32/1</b>	<b>48/1,3</b>	<b>51/1,4</b>	<b>16/0,5</b>	<b>16/0,5</b>	
В том числе:							
Лекции	35/1	12/0,4	16/0,4	17/0,5	6/0,2	6/0,2	
Практические занятия	-	-	-	-	-	-	
Семинары	-	-	-	-	-	-	
Лабораторные работы	70/2	20/0,6	32/0,9	34/0,9	10/0,3	10/0,3	
<b>Самостоятельная работа (всего)</b>	<b>117/3,3</b>	<b>184/5</b>	<b>54/1,5</b>	<b>63/1,8</b>	<b>90/2,5</b>	<b>94/2,5</b>	
В том числе:							
Курсовая работа (проект)	-	-	-	-	-	-	
Расчетно-графические работы	-	-	-	-	-	-	
ИТР	-	-	-	-	-	-	
Рефераты	-	-	-	-	-	-	
Доклады с презентациями	36/1	72/2	18/0,5	18/0,5	36/1	36/1	
<i>И (или) другие виды самостоятельной работы:</i>							
Подготовка к лабораторным работам	36/1	72/2	18/0,5	18/0,5	36/1	36/1	
Подготовка к практическим занятиям	-	-	-	-	-	-	
Подготовка к зачету	18/0,5	18/0,5	18/0,5	-	18/0,5	-	
Подготовка к экзамену	27/0,8	22/0,5	-	27/0,8	-	22/0,5	
<b>Вид отчетности</b>			<b>зач.</b>	<b>экз.</b>	<b>зач.</b>	<b>экз.</b>	
<b>Общая трудоемкость дисциплины</b>	<b>ВСЕГО в часах</b>	<b>216</b>	<b>216</b>	<b>102</b>	<b>114</b>	<b>106</b>	<b>110</b>
	<b>ВСЕГО в зач. единицах</b>	<b>6</b>	<b>6</b>	<b>2,8</b>	<b>3,2</b>	<b>3</b>	<b>3</b>

## 5. Содержание дисциплины

### 5.1. Разделы дисциплины и виды занятий

Таблица 2

№ п/п	Наименование раздела дисциплины по семестрам	Часы лекционных занятий		Часы лабораторных занятий		Часы практических (семинарских) занятий		Всего часов	
		ОФО	ЗФО	ОФО	ЗФО	ОФО	ЗФО	ОФО	ЗФО
<b>4 семестр</b>									
1.	Основы объектно-ориентированного программирования	16	6	32	10	-	-	48	16
<b>5 семестр</b>									
2.	Подходы к разработке сложных программных продуктов	17	6	34	10	-	-	51	16

### 5.2. Лекционные занятия

Таблица 3

№ п/п	Наименование раздела дисциплины	Содержание раздела
1.	Основы объектно-ориентированного программирования	<p>Понятия объекта и класса. Основные идеи и сферы применения объектно-ориентированного подхода.</p> <p>Основные принципы ООП: абстракция, инкапсуляция, наследование и полиморфизм.</p> <p>Правила построения объектной модели: абстрагирование, инкапсуляция, модульность, иерархичность, типизация, параллелизм, сохраняемость.</p> <p>Состояние, поведение и идентичность объекта. Понятие и виды операций над объектами.</p> <p>Способы создания объектов. Отношения между объектами – связь и агрегация.</p> <p>Виды отношений между классами.</p> <p>Способы проведения объектно-ориентированного анализа.</p>
2.	Подходы к разработке сложных программных продуктов	<p>Типизация данных в программировании.</p> <p>Введение в машинное обучение. Методы и подходы в машинном обучении.</p> <p>Синтаксис описания класса. Элементы, входящие в состав класса. Спецификаторы класса.</p> <p>Поля и константы. Методы класса.</p> <p>Конструктор и деструктор класса.</p> <p>Предмет и цель программной инженерии как области знаний. Основные принципы разработки программных продуктов.</p> <p>Уровни требований к программным продуктам.</p> <p>Понятие и основные принципы конфигурационного управления.</p> <p>Виды тестирования программного обеспечения.</p> <p>Верификация и валидация.</p>

### 5.3. Лабораторные занятия

Таблица 4

№ п/п	Наименование раздела дисциплины	Наименование лабораторных работ
1.	Основы объектно-ориентированного программирования	Введение в объектно-ориентированное программирование. Описание предметной области в виде объектной модели. Console Application. Использование принципов ООП. Создание иерархии классов. Создание объектов и свойств.
2.	Подходы к разработке сложных программных продуктов	Form Application. Создание программ с пользовательским интерфейсом. Form Application. Использование принципов ООП. Введение в разработку данных для машинного обучения. Постановка задач классификации в машинном обучении. Методы классификации.

### 5.4. Практические (семинарские) занятия: нет

#### 6. Самостоятельная работа студентов по дисциплине

Способ организации самостоятельной работы: подготовка презентации с устным докладом по заданной тематике; разработка программного кода по заданию преподавателя.

##### 4 семестр

Тематика докладов с презентациями:

1. Объектно-ориентированный язык и мышление. Новая парадигма и способ видения мира.
2. Развитие объектно-ориентированного программирования.
3. Объектные возможности языка C#.
4. Объектные возможности языка Python.
5. Smalltalk – язык объектного программирования.
6. Основы объектно-ориентированного программирования на языке Java.
7. Информация, требуемая для разработки конструктора: структура объекта, условия его создания и использования в программе.
8. Области видимости и время существования переменных. Локальные, статические и глобальные переменные.
9. Интерфейсы. Синтаксис интерфейса. Реализация интерфейса.
10. Виртуальные методы, перегрузка методов и стандартных операторов.
11. Абстрактные классы: назначение, построение, использование.
12. Множественное наследование как способ представления объекта с нескольких точек зрения.
13. Повторное использование кода.
14. Понятие рекурсии; математические рекурсивные функции; примеры рекурсивных процедур.
15. UML: диаграмма классов, диаграмма объектов и диаграмма компонентов.

##### 5 семестр

Тематика докладов с презентациями:

1. Классификация методов проектирования программных продуктов.

2. Место и роль ООП в теории и практике разработки программных систем.
3. Объектно-ориентированное проектирование программных продуктов. Объектно-ориентированный анализ.
4. Средства визуального создания программ и объектно-ориентированное программирование.
5. Объектный подход за пределами программирования.
6. Основные средства анализа и моделирования предметной области в языке UML.
7. Событийно-управляемое, параллельное и распределенное программирование.
8. Компонентно-ориентированный подход. Компоненты и их свойства.
9. Параллельное программирование. Потoki.
10. Сложность разработки ПО. Участники процесса разработки. Проблемы разработки ПО.
11. Методика тестирования объектно-ориентированных программ.
12. Библиотеки подпрограмм: понятие и виды. Схема вызова библиотек. Статическое и динамическое связывание.
13. Организация файловой структуры программы для эффективной разработки, отладки и модификации.
14. Поддержка и сопровождение сложного программного продукта.
15. Организация командной работы над проектом.

### **Примеры заданий**

Провести анализ и декомпозицию предметной области – выделить классы и объекты, их отношения, свойства и методы:

- университет;
- поликлиника;
- аэропорт;
- паспортный стол;
- автомобильный завод;
- гостиница;
- почта и т.д.

### **Учебно-методическое обеспечение для самостоятельной работы студентов:**

1. Тагайцева, С.Г. Предметно-ориентированное программирование [Электронный ресурс]: учебное пособие / С.Г. Тагайцева, Т.В. Юрченко. – Электрон. текстовые данные. – Нижний Новгород: Нижегородский государственный архитектурно-строительный университет, ЭБС АСВ, 2018. – 89 с. – Режим доступа: <http://www.iprbookshop.ru/80821.html> (ЭБС «IPRbooks»).

2. Кулямин, В.В. Технологии программирования. Компонентный подход [Электронный ресурс] / В.В. Кулямин. – 2-е изд. – Электрон. текстовые данные. — М.: Интернет-Университет Информационных Технологий (ИНТУИТ), 2016. – 590 с. – Режим доступа: <http://www.iprbookshop.ru/73733.html> (ЭБС «IPRbooks»).

3. Калентьев, А.А. Новые технологии в программировании [Электронный ресурс]: учебное пособие / А.А. Калентьев, Д.В. Гарайс, А.Е. Горяинов. – Электрон. текстовые данные. – Томск: Томский государственный университет систем управления и радиоэлектроники, Эль Контент, 2014. – 176 с. – Режим доступа: <http://www.iprbookshop.ru/72142.html> (ЭБС «IPRbooks»).

4. Васильев, А.Н. Python на примерах [Электронный ресурс]: практический курс по программированию / А.Н. Васильев. – 2-е изд. – Электрон. текстовые данные. – СПб.: Наука и Техника, 2017. – 432 с. – Режим доступа: <http://www.iprbookshop.ru/73043.html> (ЭБС «IPRbooks»).

## **7. Оценочные средства**

### **Вопросы к рубежной аттестации**

#### **4 семестр**

*К 1-ой рубежной аттестации:*

1. Понятия объекта и класса. Основные идеи и сферы применения объектно-ориентированного подхода.
2. Основные принципы ООП: абстракция, инкапсуляция, наследование и полиморфизм.
3. Правила построения объектной модели: абстрагирование, инкапсуляция, модульность.
4. Правила построения объектной модели: иерархичность, типизация, параллелизм, сохраняемость.

*Ко 2-ой рубежной аттестации:*

1. Состояние, поведение и идентичность объекта. Понятие и виды операций над объектами.
2. Способы создания объектов. Отношения между объектами – связь и агрегация.
3. Виды отношений между классами.
4. Способы проведения объектно-ориентированного анализа.

#### **5 семестр**

*К 1-ой рубежной аттестации:*

1. Типизация данных в программировании.
2. Введение в машинное обучение. Методы и подходы в машинном обучении.
3. Синтаксис описания класса. Элементы, входящие в состав класса. Спецификаторы класса.
4. Поля и константы. Методы класса.
5. Конструктор и деструктор класса.

*Ко 2-ой рубежной аттестации:*

1. Предмет и цель программной инженерии как области знаний. Основные принципы разработки программных продуктов.
2. Уровни требований к программным продуктам.
3. Понятие и основные принципы конфигурационного управления.
4. Виды тестирования программного обеспечения.
5. Верификация и валидация.

#### **4 семестр**

### **Вопросы к зачету**

1. Понятия объекта и класса. Основные идеи и сферы применения объектно-ориентированного подхода.
2. Основные принципы ООП: абстракция, инкапсуляция, наследование и полиморфизм.
3. Правила построения объектной модели: абстрагирование, инкапсуляция,

модульность.

4. Правила построения объектной модели: иерархичность, типизация, параллелизм, сохраняемость.

5. Состояние, поведение и идентичность объекта. Понятие и виды операций над объектами.

6. Способы создания объектов. Отношения между объектами – связь и агрегация.

7. Виды отношений между классами.

8. Способы проведения объектно-ориентированного анализа.

### **5 семестр**

#### **Вопросы к экзамену**

1. Типизация данных в программировании.

2. Введение в машинное обучение. Методы и подходы в машинном обучении.

3. Синтаксис описания класса. Элементы, входящие в состав класса.

Спецификаторы класса.

4. Поля и константы. Методы класса.

5. Конструктор и деструктор класса.

6. Предмет и цель программной инженерии как области знаний. Основные принципы разработки программных продуктов.

7. Уровни требований к программным продуктам.

8. Понятие и основные принципы конфигурационного управления.

9. Виды тестирования программного обеспечения.

10. Верификация и валидация.

Образец билетов рубежной аттестации:

**Грозненский Государственный Нефтяной Технический Университет  
им. акад. М.Д. Миллионщикова**

**Кафедра «Информационные технологии»**

**Дисциплина «Объектно-ориентированное программирование»**

**1-я рубежная аттестация**

**Группа:**

**Семестр: 4**

**Билет №**

1. Основные принципы ООП: абстракция, инкапсуляция, наследование и полиморфизм.

2. Понятия объекта и класса. Основные идеи и сферы применения объектно-ориентированного подхода.

**Преподаватель** \_\_\_\_\_

**Грозненский Государственный Нефтяной Технический Университет  
им. акад. М.Д. Миллионщикова**

**Кафедра «Информационные технологии»**

**Дисциплина «Объектно-ориентированное программирование»**

**2-я рубежная аттестация**

**Группа:**

**Семестр: 4**

**Билет №**

1. Способы создания объектов. Отношения между объектами – связь и агрегация.

2. Способы проведения объектно-ориентированного анализа.

**Преподаватель** \_\_\_\_\_



**Грозненский Государственный Нефтяной Технический Университет**  
**им. акад. М.Д. Миллионщикова**  
**Кафедра «Информационные технологии»**  
**Дисциплина «Объектно-ориентированное программирование»**  
**1-я рубежная аттестация**

**Группа:**

**Семестр: 5**

**Билет №**

1. Синтаксис описания класса. Элементы, входящие в состав класса. Спецификаторы класса.
2. Конструктор и деструктор класса.

**Преподаватель** \_\_\_\_\_

**Грозненский Государственный Нефтяной Технический Университет**  
**им. акад. М.Д. Миллионщикова**  
**Кафедра «Информационные технологии»**  
**Дисциплина «Объектно-ориентированное программирование»**  
**2-я рубежная аттестация**

**Группа:**

**Семестр: 5**

**Билет №**

1. Предмет и цель программной инженерии как области знаний. Основные принципы разработки программных продуктов.
2. Виды тестирования программного обеспечения.

**Преподаватель** \_\_\_\_\_

Образец билета к зачету:

**Грозненский Государственный Нефтяной Технический Университет**  
**им. акад. М.Д. Миллионщикова**  
**Кафедра «Информационные технологии»**  
**Дисциплина «Объектно-ориентированное программирование»**  
**Группа:**

**Семестр: 4**

**Билет №**

1. Правила построения объектной модели: абстрагирование, инкапсуляция, модульность.
2. Способы проведения объектно-ориентированного анализа.

**Подпись преподавателя** \_\_\_\_\_ **Подпись заведующего кафедрой** \_\_\_\_\_

Образец билета к экзамену:

**Грозненский Государственный Нефтяной Технический Университет**  
**им. акад. М.Д. Миллионщикова**  
**Кафедра «Информационные технологии»**  
**Дисциплина «Объектно-ориентированное программирование»**  
**Группа:**

**Семестр: 5**

**Билет №**

1. Типизация данных в программировании.
2. Виды тестирования программного обеспечения.

**Подпись преподавателя** \_\_\_\_\_ **Подпись заведующего кафедрой** \_\_\_\_\_

## Текущий контроль

### Образец типового задания для лабораторных занятий

#### Лабораторная работа на тему «Создание классов»

В языке Python определение класса начинается с ключевого слова `class`, после него следует имя класса и двоеточие.

```
1
2 class Elevator:
3     """ Simple elevator class """
4     # Переменная класса. Сколько людей было перевезено ВСЕМИ лифтами
5     people_lifted = 0
6
7     # Конструктор класса. Вызывается при создании экземпляра класса
8     def __init__(self, name):
9         self.name = name
10        # переменная экземпляра класса. Количество
11        людей перевезенных КОНКРЕТНЫМ лифтом
12        self.people_lifted = 0
13
14        # Метод перевозки людей
15        def lift(self):
16            print ("{} lifted someone".format(self.name))
17            # Увеличиваем количество людей перевезенных ЭТИМ лифтом
18            self.people_lifted += 1
19            # Увеличиваем количество людей перевезенных ВСЕМИ лифтами
20            Elevator.people_lifted += 1
21
22        # Метод печатающий информацию о конкретном лифте
23        def info(self):
24            print (self.name, "lifted", self.people_lifted, "people out of",
25                  Elevator.people_lifted)
```

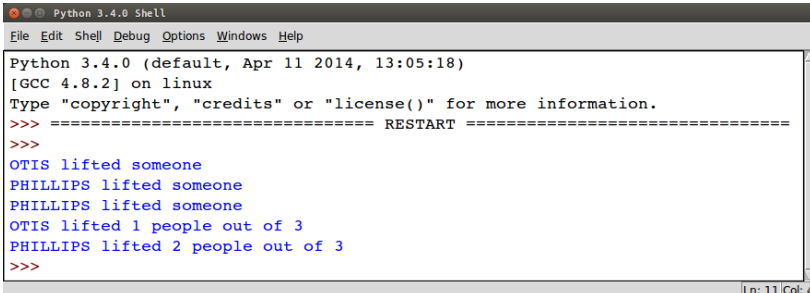
Чтобы создать экземпляр класса, следует любой переменной присвоить значение имени класса, указав в скобках аргументы, которые принимает метод `__init__()`.

```
1 elevator_1 = Elevator("OTIS")
2 elevator_2 = Elevator("PHILLIPS")
```

Чтобы получить доступ к атрибутам и методам класса в Python, следует после объекта поставить точку и написать имя переменной или метода, которые вы хотите использовать:

```
1
2 # Везем человека в лифте под именем OTIS
3 elevator_1.lift()
4 # Везем двоих человек в лифте под именем PHILLIPS
5 elevator_2.lift()
6 elevator_2.lift()
7 # Получаем информацию по лифту под именем OTIS
8 elevator_1.info()
9 # Получаем информацию по лифту под именем PHILLIPS
10 elevator_2.info()
```

#### Результат:



```
Python 3.4.0 Shell
File Edit Shell Debug Options Windows Help
Python 3.4.0 (default, Apr 11 2014, 13:05:18)
[GCC 4.8.2] on linux
Type "copyright", "credits" or "license()" for more information.
>>> ===== RESTART =====
>>>
OTIS lifted someone
PHILLIPS lifted someone
PHILLIPS lifted someone
OTIS lifted 1 people out of 3
PHILLIPS lifted 2 people out of 3
>>>
```

## **8. Учебно-методическое и информационное обеспечение дисциплины**

### *Основная литература:*

1. Уйманова, Н.А. Основы объектно-ориентированного программирования [Электронный ресурс]: практикум / Н.А. Уйманова, М.Г. Таспаева. – Электрон. текстовые данные. – Оренбург: Оренбургский государственный университет, ЭБС АСВ, 2017. – 156 с. – Режим доступа: <http://www.iprbookshop.ru/78808.html> (ЭБС «IPRbooks»).
2. Мейер, Б. Объектно-ориентированное программирование и программная инженерия [Электронный ресурс] / Б. Мейер. – 3-е изд. – Электрон. текстовые данные. – М.: Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Эр Медиа, 2019. – 285 с. – Режим доступа: <http://www.iprbookshop.ru/79706.html> (ЭБС «IPRbooks»).
3. Букунов, С.В. Основы объектно-ориентированного программирования [Электронный ресурс]: учебное пособие / С.В. Букунов, О.В. Букунова. – Электрон. текстовые данные. – СПб.: Санкт-Петербургский государственный архитектурно-строительный университет, ЭБС АСВ, 2017. – 196 с. – Режим доступа: <http://www.iprbookshop.ru/74339.html> (ЭБС «IPRbooks»).
4. Комлев, Н.Ю. Объектно Ориентированное Программирование. Хорошая книга для Хороших Людей [Электронный ресурс] / Н.Ю. Комлев. – Электрон. текстовые данные. – М.: СОЛОН-ПРЕСС, 2014. – 298 с. – Режим доступа: <http://www.iprbookshop.ru/26923.html> (ЭБС «IPRbooks»).

### *Дополнительная литература:*

5. Тагайцева, С.Г. Предметно-ориентированное программирование [Электронный ресурс]: учебное пособие / С.Г. Тагайцева, Т.В. Юрченко. – Электрон. текстовые данные. – Нижний Новгород: Нижегородский государственный архитектурно-строительный университет, ЭБС АСВ, 2018. – 89 с. – Режим доступа: <http://www.iprbookshop.ru/80821.html> (ЭБС «IPRbooks»).
6. Кулямин, В.В. Технологии программирования. Компонентный подход [Электронный ресурс] / В.В. Кулямин. – 2-е изд. – Электрон. текстовые данные. — М.: Интернет-Университет Информационных Технологий (ИНТУИТ), 2016. – 590 с. – Режим доступа: <http://www.iprbookshop.ru/73733.html> (ЭБС «IPRbooks»).
7. Калентьев, А.А. Новые технологии в программировании [Электронный ресурс]: учебное пособие / А.А. Калентьев, Д.В. Гарайс, А.Е. Горяинов. – Электрон. текстовые данные. – Томск: Томский государственный университет систем управления и радиоэлектроники, Эль Контент, 2014. – 176 с. – Режим доступа: <http://www.iprbookshop.ru/72142.html> (ЭБС «IPRbooks»).
8. Васильев, А.Н. Python на примерах [Электронный ресурс]: практический курс по программированию / А.Н. Васильев. – 2-е изд. – Электрон. текстовые данные. – СПб.: Наука и Техника, 2017. – 432 с. – Режим доступа: <http://www.iprbookshop.ru/73043.html> (ЭБС «IPRbooks»).

## **9. Материально-техническое обеспечение дисциплины**

Перечень материально-технических средств учебной аудитории для проведения занятий по дисциплине:

- учебная аудитория, доска;
- стационарные компьютеры;
- мультимедийный проектор;
- настенный экран.

**Составитель:**

Старший преподаватель кафедры  
«Информационные технологии»



/ Мачуева Д.А. /

**СОГЛАСОВАНО:**

Зав. выпускающей кафедрой  
«Информационные технологии»



/ Моисеенко Н.А. /

Директор ДУМР



/ Магомаева М.А. /