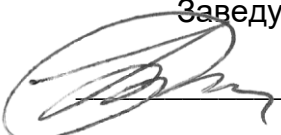


МИНОБРНАУКИ РОССИИ  
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ  
ВЫСШЕГО ОБРАЗОВАНИЯ  
«ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»  
(ФГБОУ ВО «ВГУ»)

УТВЕРЖДАЮ

Заведующий кафедрой  
электроники  
Бобрешов А.М.



31.08.2021 г.

**РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ**  
**Б1.О.15 Программирование**

**1. Код и наименование направления подготовки/специальности:**

*09.03.01 Информатика и вычислительная техника*

**2. Профиль подготовки/специализация:**

*Вычислительные машины, комплексы, системы и сети*

**3. Квалификация (степень) выпускника: бакалавр**

**4. Форма обучения: очная**

**5. Кафедра, отвечающая за реализацию дисциплины: электроники**

**6. Составители программы: Коровченко Игорь Сергеевич, к.ф.-м.н.**

*(ФИО, ученая степень, ученое звание)*

**7. Рекомендована: НМС физического факультета 23.06.2021, № протокола: 6**

**8. Учебный год: 2021/2022**

**Семестр(ы): 2, 3**

**9. Цели и задачи учебной дисциплины:** освоение современных языков программирования, алгоритмов, фреймворков для обработки и анализа данных, а также веб-технологий.

**10. Место учебной дисциплины в структуре ООП:**

Дисциплина является одной из основополагающих для приобретения необходимых специалисту профессиональных навыков. Дисциплина является первой частью курса программирования.

**11. Планируемые результаты обучения по дисциплине/модулю (знания, умения, навыки), соотнесенные с планируемыми результатами освоения образовательной программы (компетенциями выпускников):**

Код	Название компетенции	Код(ы)	Индикатор(ы)	Планируемые результаты обучения
ОПК-8	Способен разрабатывать алгоритмы и программы, пригодные для практического применения	ОПК-8.1	Владеет навыками построения алгоритмов	Использует алгоритмы для решения задач профессиональной деятельности
		ОПК-8.2	Способен выбирать языки программирования с учетом поставленных целей и задач, а также особенностей предметной области	Выбирает язык программирования на основе анализа целей и задач, а также особенностей предметной области
		ОПК-8.3	Применяет на практике операционные системы и оболочки, современные среды разработки программного обеспечения	Учитывает особенности операционных систем и окружения при разработке программ
		ОПК-8.4	Создает и отлаживает код на выбранном языке программирования	Владеет инструментами отладки программ
		ОПК-8.5	Тестирует работоспособность программы	Владеет навыками тестирования программного обеспечения
		ОПК-8.6	Интегрирует программные модули и библиотеки	Владеет системами управления пакетами  Владеет системами сборки программных модулей
ПК-1	Способен выполнять работы и управлять работами по созданию (модификации) и сопровождению ИС, автоматизирующ их задачи организационно го управления и бизнес-процессы	ПК-1.1	Применяет методы и средства проектирования программного обеспечения, методы и средства сборки и интеграции программных модулей и компонент, методы и приемы отладки программного кода	Использует UML для организации программ на основе принципов ООП  Использует блок-схемы и псевдокод для проектирования програмного обеспечения
		ПК-1.2	Применяет языки программирования для создания программного кода	Владеет выбранным языком программирования на базовом уровне
		ПК-1.3	Использует среду программирования и средства системы управления базами данных	Использует интегрированную среду разработчика для работы с данными
		ПК-1.4	Использует возможности имеющейся технической и/или программной архитектуры	Учитывает особенности архитектуры программного обеспечения для разработки программных модулей
		ПК-1.5	Проектирует и создает код на языке программирования в соответствии с техническим заданием	Проектирует программное обеспечение на основе технического задания
		ПК-1.6	Оптимизирует программный код	Проводит оптимизацию и

			с использованием специализированных программных средств и существующих методик разработки программного обеспечения	рефакторинг программ
ПК-2	Способен осуществлять концептуальное, функциональное и логическое проектирование систем среднего и крупного масштаба и сложности	ПК-2.4	Анализирует возможности удовлетворения требований к программному обеспечению	Анализирует код программ с целью исправления ошибок и повышения производительности
		ПК-2.5	Оценивает время и трудоемкость реализации программного обеспечения с учетом требований	Оценивает трудоемкость работы программиста в соответствии с особенностями программного продукта и команды
ПК-4	Способен разрабатывать компоненты системных программных продуктов	ПК-4.1	Применяет на практике методы разработки архитектуры программного обеспечения с учетом особенностей предметной области	Использует различные методы и практики для создания архитектуры программ
		ПК-4.2	Разрабатывает программный код процедур интеграции программных модулей	Использует оптимальные методы для подключения программных модулей
		ПК-4.3	Использует выбранную среду проектирования для разработки процедур интеграции программных модулей	Использует интегрированную среду разработчика для интеграции программных модулей
		ПК-4.4	Применяет методы и средства сборки модулей и компонентов программного обеспечения, разработки процедур для развертывания программного обеспечения, миграции и преобразования данных, создания программных интерфейсов	Использует инструменты сборки программных модулей
		ПК-4.5	Разрабатывает и документирует программные интерфейсы	Описывает программные интерфейсы
		ПК-4.6	Разрабатывает процедуры сборки модулей и компонентов программного обеспечения	Разрабатывает процедуры сборки модулей и компонентов программного обеспечения
ПК-5	Способен разрабатывать требования и проектировать программное обеспечение	ПК-5.1	Использует методы и приемы формализации задач, методы и средства верификации работоспособности программных продуктов	Формализует методы проверки работоспособности программных продуктов
		ПК-5.2	Выполняет процедуры сборки программных модулей и компонент в конечный программный продукт	Осуществляет сборку программ на выбранном языке программирования
		ПК-5.3	Производит настройку параметров программного продукта и осуществляет запуск процедур сборки	Производит конфигурацию программного продукта
		ПК-5.4	Проводит оценку работоспособности программного продукта, документирует выявленные проблемы и способы их устранения	Оценивает ошибки в коде программ, анализирует пути решения, обрабатывает исключения
		ПК-5.5	Обеспечивает соответствие требований заказчиков и разрабатываемых программных	Проводит оценку качества реализации по соответствующим требованиям заказчика

## 12. Объем дисциплины в зачетных единицах/час. — 9 / 324.

Форма промежуточной аттестации экзамен.

## 13. Виды учебной работы

Вид учебной работы	Трудоемкость			
	Всего	По семестрам		
		3	4	
Аудиторные занятия	104	50	54	
в том числе:				
лекции	34	16	18	
практические				
лабораторные	70	34	36	
Самостоятельная работа	148	94	54	
Форма промежуточной аттестации (зачет – 0 час. / экзамен – ___ час.)	62	36	36	
Итого:	324	180	144	

### 13.1. Содержание дисциплины

п/п	Наименование раздела дисциплины	Содержание раздела дисциплины
1. Лекции		
1.1	Введение.	Основные подходы к описанию и оценки алгоритмов.
1.2	Классические алгоритмы	Алгоритмы поиска и сортировки. Численные алгоритмы. Рекурсия. Интегрирование и поиск корней. Связные списки и хэш-таблицы. Деревья. Строковые алгоритмы. Криптографические алгоритмы. Сетевые алгоритмы.
1.3	Веб-технологии	Обзор технологий Интернета. Гипертекстовая разметка HTML. Каскадные стили CSS. Динамические элементы на странице JavaScript.
1.4	Практика использования Python	Анализ данных на Python. Веб-фреймворки на Python.
1.5	Объектно-ориентированное программирование	Основы парадигмы ООП. Наследование, инкапсуляция и полиморфизм.
1.6	Паттерны проектирования	Архитектура приложений на основе парадигмы объектно-ориентированного программирования. MVC, MVVM, VIPER, реактивный подход.
2. Практические занятия		
3. Лабораторные работы		
3.1	Классические алгоритмы	Сортировка сдвигом. Сортировка пузырьком. Сортировка слиянием. Быстрая сортировка. Линейный поиск. Бинарный поиск. Метод Ньютона. Интегрирование методом прямоугольников. Интегрирование методом трапеций. Интегрирование методом адаптивной квадратуры. Псевдографика: построение графика функций. Нахождение наибольшего делителя. Поиск в ширину. Поиск в глубину. Нахождение реакционного расстояния. Регулярные выражения. Шифр Цезаря. Шифр Виженера. Кто сделал это?
3.2	Веб-технологии	Практика HTML. Практика CSS. Практика JavaScript. Свой личный блог.
3.3	Практика использования Python	Умный подсчет расхода воды. Препятствие. Монетки на сдачу. Отцы основатели. Игра Пятнашки. Построение графиков погоды. Бэкенд сайта. Работа со строками. Линейная алгебра.
3.4	Объектно-ориентированное программирование	Использование UML.
3.5	Паттерны проектирования	Создание приложения по подсчету очков в Android. Создание приложения по планированию задач с таймингом.

## 13.2. Темы (разделы) дисциплины и виды занятий

№ п/п	Наименование темы (раздела) дисциплины	Виды занятий (часов)				
		Лекции	Практические	Лабораторные	Самостоятельная работа	Всего
1.	Введение.	2	0	0	12	14
2.	Классические алгоритмы	4	0	22	22	48
3.	Веб-технологии	6	0	6	30	42
4.	Практика использования Python	4	0	6	30	40
5.	Объектно-ориентированное программирование	4	0	2	14	20
6.	Паттерны проектирования	14	0	34	40	88
	Итого:	34	0	70	148	252

## 14. Методические указания для обучающихся по освоению дисциплины

Для успешного освоения дисциплины необходима регулярная и планомерная работа с конспектом лекций и литературой.

После окончания лекции нужно просматривать конспект для определения материала, вызывающего затруднения для понимания. После этого следует обратиться к рекомендуемой в настоящей программе литературе с целью углубленного изучения проблемного вопроса. В общем случае работа лишь с одним литературным источником часто является недостаточной для полного понимания, поэтому необходимо просматривать несколько источников для выбора того, который наиболее полно и доступно освещает изучаемый материал. В случае если проблемы с пониманием остались, необходимо обратиться к преподавателю на ближайшей лекции с заранее сформулированными вопросами.

Для успешного освоения лекционного курса рекомендуется регулярно повторять изученный материал.

Самостоятельная работа студентов способствует глубокому усвоению учебного материала и развитию навыков самообразования. Самостоятельная работа предполагает следующие составляющие:

- работа с текстами: учебниками, справочниками, дополнительной литературой, а также проработка конспектов лекций;
- работа над темами для самостоятельного изучения;
- участие в работе студенческих научных конференций, олимпиад;
- подготовка зачету.

Кроме литературы из основного списка рекомендуется самостоятельно использовать дополнительную. Независимо от вида учебника, работа с ним должна происходить в течение всего семестра. Эффективнее работать с учебником не после, а перед лекцией.

В процессе самостоятельной работы следует занимать активную позицию и пользоваться не только рекомендованной литературой, но и самостоятельно найденными источниками. При изучении дисциплины рекомендуется использовать возможности сети интернет для получения дополнительной информации по рассматриваемой теме.

При использовании дистанционных образовательных технологий и электронного обучения выполнять все указания преподавателей, вовремя подключаться к онлайн-занятиям, ответственно подходить к выполнению заданий для самостоятельной работы.

## 15. Перечень основной и дополнительной литературы, ресурсов интернет, необходимых для освоения дисциплины

а) основная литература:

№ п/п	Источник
1.	Борзунов С. В. Параллельное программирование: задачи и решения : учебное пособие / С.В. Борзунов, С.Д. Кургалин ; Воронеж. гос. ун-т. — Воронеж : Издательский дом ВГУ, 2018. — 112 с.
2.	Завгородний, М. Г. Программирование. Криптографические алгоритмы : учебное пособие /

	М.Г. Завгородний, С.П. Майорова .— Воронеж : Издательский дом ВГУ, 2018 .— Библиогр.: с. 92-93
3.	Блох Дж. Java. Эффективное программирование = Effective Java programming language guide / Джошуа Блох ; пер. с англ. В. Стрельцов ; науч. ред. Р. Усманов ; предисл. Г. Стила .— Москва : Лори, 2017 .— 294 с.
4.	Окулов С. М. Программирование в алгоритмах : [учебные пособия] / С.М. Окулов .— 4-е изд. — Москва : Бином. Лаборатория знаний, 2017 .— 383 с.
5.	Окулов С. М. Динамическое программирование : [учебное пособие] / С.М. Окулов, О.А. Пестов .— Москва : Бином. Лаборатория знаний, 2017 .— 296 с.
6.	Прата С. Язык программирования С++ : [учебник] : лекции и упражнения / Стивен Прата ; [пер. с англ. Ю.И. Корниенко, А.А. Моргунова ; под ред. Ю.Н. Артеменко] .— 6-е изд. — Москва ; Санкт-Петербург ; Киев : Вильямс, 2016 .— 1244 с.
7.	Маклафлин Б. PHP и MySQL. Исчерпывающее руководство / Бретт Маклафлин ; [пер. с англ. О. Сивченко] .— 2-е изд. — Санкт-Петербург [и др.] : Питер, 2017 .— 543 с.

б) дополнительная литература:

№ п/п	Источник
8.	Шилдт Г. С++ : базовый курс / Герберт Шилдт ; [пер. с англ. и ред. Н.М. Ручко] .— 3-е изд. — Москва ; Санкт-Петербург ; Киев : Вильямс, 2015 .— 620 с.
9.	Страуструп Б. Язык программирования С++. Специальное издание = The C++ programming language. Special edition. / Бьерн Страуструп ; пер. с англ. под ред. Н.Н. Мартынова .— Москва : Бином, 2015 .— 1135 с.
10.	Лафоре, Р. Объектно-ориентированное программирование в С++ = Object-Oriented Programming in C++ / Р. Лафоре; [пер. с англ. А. Кузнецова, М. Назарова, В. Шрага ] .— 4-е изд. — Санкт-Петербург : Питер, 2015 .— 923 с.
11.	Лав Р. Linux. Системное программирование = Linux. System Programming / Роберт Лав ; [пер. с англ. О. Сивченко] .— 2-е изд. — Санкт-Петербург [и др.] : Питер, 2015 .— 445 с.
12.	Ускова О. Ф. Информатика и программирование : учебное пособие / О.Ф. Ускова, Н.А. Каплиева ; Воронеж. гос. ун-т .— Воронеж : Издательско-полиграфический центр "Научная книга", 2015 .— 224 с.
13.	Сергиенко М.А. Программирование на языке Java [Электронный ресурс] : учебно-методическое пособие для вузов : [для студ. дневного отд-ния направлений 010300 - Фундаментальная информатика и информационные технологии, 010400 - Прикладная математика и информатика]. Ч. 1 / М.А. Сергиенко ; Воронеж. гос. ун-т .— Электрон. текстовые дан. — Воронеж : Издательский дом ВГУ, 2015 .— Загл. с титула экрана .— Свободный доступ из интранета ВГУ .— Текстовый файл .— Windows 2000 ; Adobe Acrobat Reader .— <URL:http://www.lib.vsu.ru/elib/texts/method/vsu/m15-125.pdf>.
14.	Лафоре Р. Структуры данных и алгоритмы в Java = Data structures @ algorithms in Java / Роберт Лафоре ; [пер. с англ. Е. Матвеева] .— 2-е изд. — Санкт-Петербург [и др.] : Питер, 2014 .— 701 с.
15.	Эванс Б. Java. Новое поколение разработки. Техники Java 7 и многоязычное программирование = The well-Grounded Java developer. Vital techniques of Java 7 and polyglot programming / Бенджамин Эванс, Мартин Вербург ; [пер. с англ. О. Сивченко] .— Санкт-Петербург [и др.] : Питер, 2014 .— 556 с.
16.	Шилдт Г. Java : руководство для начинающих / Герберт Шилдт ; [пер. с англ. и ред. В.В. Вейтмана] .— 4-е изд. — М. [и др.] : Вильямс, 2009 .— 715 с.
17.	Шилдт Г. Искусство программирования на JAVA : пер. с англ. / Герберт Шилдт, Джеймс Холмс .— СПб. [и др.] : БХВ-Петербург, 2005 .— 331 с.
18.	Лутц М. Программирование на Python / Марк Лутц ; Пер. с англ. С.Маккавеева .— 2-е изд. — СПб. : Символ, 2002 .— 1135 с.
19.	Сузи Р. Python / Роман Сузи .— СПб. и др. : БХВ-Петербург, 2002 .— XI, 747 с.
20.	Python. — URL: <a href="https://www.python.org">https://www.python.org</a> (дата обращения 15 июня 2019)
21.	Java. — URL: <a href="https://www.oracle.com/ru/java/">https://www.oracle.com/ru/java/</a> (дата обращения 15 июня 2019)
22.	C++.org. — URL: <a href="http://cplusplus.org">http://cplusplus.org</a> (дата обращения 15 июня 2019)
23.	w3schools.com. — URL: <a href="https://www.w3schools.com">https://www.w3schools.com</a> (дата обращения 15 июня 2019)
24.	Langtangen Hans Petter. A primer on scientific programming with Python / Hans Petter Langtangen .— Dordrecht [etc.] : Springer, 2009 .— XXVII, 693 p.

в) информационные электронно-образовательные ресурсы (официальные ресурсы интернет):

№ п/п	Ресурс
25.	Электронная библиотека Зональной научной библиотеки Воронежского государственного университета : электронно-библиотечная система. – URL : <a href="https://lib.vsu.ru/zgate?Init+elib.xml,simple_elib.xsl+rus">https://lib.vsu.ru/zgate?Init+elib.xml,simple_elib.xsl+rus</a>
26.	Электронно-библиотечная система "БиблиоТех" : электронно-библиотечная система. – URL : <a href="https://lib.vsu.ru/?p=4&amp;t=2d&amp;id=1486">https://lib.vsu.ru/?p=4&amp;t=2d&amp;id=1486</a>
27.	Электронно-библиотечная система "Издательство "Лань" : электронно-библиотечная система. – URL : <a href="https://lib.vsu.ru/?p=4&amp;t=2d&amp;id=1308">https://lib.vsu.ru/?p=4&amp;t=2d&amp;id=1308</a>
28.	Электронно-библиотечная система "Университетская библиотека online" : электронно-библиотечная система. – URL : <a href="https://lib.vsu.ru/?p=4&amp;t=2d&amp;id=1307">https://lib.vsu.ru/?p=4&amp;t=2d&amp;id=1307</a>
29.	Электронно-библиотечная система "Консультант студента" : электронно-библиотечная система. – URL : <a href="https://lib.vsu.ru/?p=4&amp;t=2d&amp;id=1306">https://lib.vsu.ru/?p=4&amp;t=2d&amp;id=1306</a>
30.	Электронно-библиотечная система «ЮРАЙТ» : электронно-библиотечная система. – URL : <a href="https://lib.vsu.ru/?p=4&amp;t=2d&amp;id=1457">https://lib.vsu.ru/?p=4&amp;t=2d&amp;id=1457</a>

## 16. Перечень учебно-методического обеспечения для самостоятельной работы (учебно-методические рекомендации, пособия, задачки, методические указания по выполнению практических (контрольных) работ и др.)

№ п/п	Источник
1.	Груздев Д. В. Программирование С++ (1 курс) [Электронный ресурс] : учебное пособие / Д.В. Груздев ; Воронеж. гос. ун-т .— Электрон. текстовые дан. — Воронеж : Издательский дом ВГУ, 2017 .— Загл. с титула экрана .— Свободный доступ из интранета ВГУ .— Текстовый файл .— Windows 2000; Adobe Acrobat Reader .— <URL: <a href="http://www.lib.vsu.ru/elib/texts/method/vsu/m17-19.pdf">http://www.lib.vsu.ru/elib/texts/method/vsu/m17-19.pdf</a> >.
2.	Груздев Д. В. Практика ЭВМ - JavaScript (3 курс) [Электронный ресурс] : учебное пособие / Д.В. Груздев ; Воронеж. гос. ун-т .— Электрон. текстовые дан. — Воронеж : Издательский дом ВГУ, 2017 .— Загл. с титула экрана .— Свободный доступ из интранета ВГУ .— Текстовый файл .— Windows 2000; Adobe Acrobat Reader .— <URL: <a href="http://www.lib.vsu.ru/elib/texts/method/vsu/m17-20.pdf">http://www.lib.vsu.ru/elib/texts/method/vsu/m17-20.pdf</a> >.
3.	Груздев Д. В. Практика ЭВМ - HTML (3 курс) [Электронный ресурс] : учебное пособие / Д.В. Груздев ; Воронеж. гос. ун-т .— Электрон. текстовые дан. — Воронеж : Издательский дом ВГУ, 2017 .— Загл. с титула экрана .— Свободный доступ из интранета ВГУ .— Текстовый файл .— Windows 2000; Adobe Acrobat Reader .— <URL: <a href="http://www.lib.vsu.ru/elib/texts/method/vsu/m17-21.pdf">http://www.lib.vsu.ru/elib/texts/method/vsu/m17-21.pdf</a> >.
4.	Груздев Д. В. Практика ЭВМ - CSS (3 курс) [Электронный ресурс] : учебное пособие / Д.В. Груздев ; Воронеж. гос. ун-т .— Электрон. текстовые дан. — Воронеж : Издательский дом ВГУ, 2017 .— Загл. с титула экрана .— Свободный доступ из интранета ВГУ .— Текстовый файл .— Windows 2000; Adobe Acrobat Reader .— <URL: <a href="http://www.lib.vsu.ru/elib/texts/method/vsu/m17-22.pdf">http://www.lib.vsu.ru/elib/texts/method/vsu/m17-22.pdf</a> >.

## 17. Информационные технологии, используемые для реализации учебной дисциплины, включая программное обеспечение и информационно-справочные системы (при необходимости)

При реализации дисциплины могут использоваться технологии электронного обучения и дистанционные образовательные технологии на базе образовательного портала "Электронный университет ВГУ" по адресу [edu.vsu.ru](http://edu.vsu.ru), а также другие доступные ресурсы сети Интернет.

## 18. Материально-техническое обеспечение дисциплины:

### Лаб. 401

Компьютер RAMEC МТЛ5-6400/8GB/500GB – 20 шт.

Коммутатор HPJ9981A – 1 шт.

Комплекс для проведения лекций, семинаров и презентаций – 1 шт.

Проектор Optoma W402 – 1шт.

Экран Cactus Wallscreen – 1 шт.

## 19. Фонд оценочных средств:

### 19.1. Перечень компетенций с указанием этапов формирования и планируемых результатов обучения

Код и содержание компетенции (или ее части)	Планируемые результаты обучения (показатели достижения заданного уровня освоения компетенции посредством формирования знаний, умений, навыков)	Этапы формирования компетенции (разделы (темы) дисциплины или модуля и их наименование)	ФОС (средства оценивания)
ОПК-8 Способен разрабатывать алгоритмы и программы, пригодные для практического применения	ОПК-8.1 Владеет навыками построения алгоритмов	1.1. Классические алгоритмы 1.2. Веб-технологии	Устный опрос
	ОПК-8.2 Способен выбирать языки программирования с учетом поставленных целей и задач, а также особенностей предметной области		
	ОПК-8.3 Применяет на практике операционные системы и оболочки, современные среды разработки программного обеспечения	1.1. Классические алгоритмы 1.2. Веб-технологии	Практические задания №№ 1-32
	ОПК-8.4 Создает и отлаживает код на выбранном языке программирования		
	ОПК-8.5 Тестирует работоспособность программы	1.1. Классические алгоритмы 1.2. Веб-технологии	Практические задания №№ 1-32
	ОПК-8.6 Интегрирует программные модули и библиотеки		
ПК-1 Способен выполнять работы и управлять работами по созданию (модификации) и сопровождению ИС, автоматизирующих задачи организационного управления и бизнес-процессы	ПК-1.1 Применяет методы и средства проектирования программного обеспечения, методы и средства сборки и интеграции программных модулей и компонент, методы и приемы отладки программного кода	1.3. Практика использования Python	Устный опрос
	ПК-1.2 Применяет языки программирования для создания программного кода		
	ПК-1.3 Использует среду программирования и средства системы управления базами данных	1.3. Практика использования Python	Практические задания №№ 1-32
	ПК-1.4 Использует возможности имеющейся технической и/или программной архитектуры		
	ПК-1.5 Проектирует и создает код на языке программирования в соответствии с техническим заданием	1.3. Практика использования Python	Практические задания №№ 1-32
	ПК-1.6 Оптимизирует программный код с использованием специализированных программных средств и существующих методик разработки программного обеспечения		
ПК-2 Способен осуществлять концептуальное, функциональное и	ПК-2.4 Анализирует возможности удовлетворения требований к программному обеспечению	1.3. Практика использования Python	Устный опрос
	ПК-2.5 Оценивает время и	1.3. Практика	Практические



логическое проектирование систем среднего и крупного масштаба и сложности	трудоемкость реализации программного обеспечения с учетом требований	использования Python	задания №№ 1-32
ПК-4 Способен разрабатывать компоненты системных программных продуктов	ПК-4.1 Применяет на практике методы разработки архитектуры программного обеспечения с учетом особенностей предметной области	1.2. Веб-технологии 1.4. Объектно-ориентированное программирование	Устный опрос
	ПК-4.2 Разрабатывает программный код процедур интеграции программных модулей	1.5. Паттерны проектирования	
	ПК-4.3 Использует выбранную среду проектирования для разработки процедур интеграции программных модулей	1.2. Веб-технологии 1.4. Объектно-ориентированное программирование	Практические задания №№ 1-34
	ПК-4.4 Применяет методы и средства сборки модулей и компонентов программного обеспечения, разработки процедур для развертывания программного обеспечения, миграции и преобразования данных, создания программных интерфейсов	1.5. Паттерны проектирования	
	ПК-4.5 Разрабатывает и документирует программные интерфейсы	1.2. Веб-технологии 1.4. Объектно-ориентированное программирование	Практические задания №№ 1-34
	ПК-4.6 Разрабатывает процедуры сборки модулей и компонентов программного обеспечения	1.5. Паттерны проектирования	
ПК-5 Способен разрабатывать требования и проектировать программное обеспечение	ПК-5.1 Использует методы и приемы формализации задач, методы и средства верификации работоспособности программных продуктов	1.2. Веб-технологии 1.4. Объектно-ориентированное программирование 1.5. Паттерны проектирования	Практические задания №№ 1-34
	ПК-5.2 Выполняет процедуры сборки программных модулей и компонент в конечный программный продукт		
	ПК-5.3 Производит настройку параметров программного продукта и осуществляет запуск процедур сборки		
	ПК-5.4 Проводит оценку работоспособности программного продукта, документирует выявленные проблемы и способы их устранения		
	ПК-5.5 Обеспечивает соответствие требований заказчиков и разрабатываемых программных продуктов		
Промежуточная аттестация			КИМ

## 19.2 Описание критериев и шкалы оценивания компетенций (результатов обучения) при промежуточной аттестации

Для оценивания результатов обучения на экзамене используются следующие показатели:

- 1) знание учебного материала и владение понятийным аппаратом операционных систем;
- 2) умение связывать теорию с практикой;
- 3) умение иллюстрировать ответ примерами, фактами, данными научных исследований в области разработки операционных систем;

	Уровень	
--	---------	--

Критерии оценивания компетенций	сформированности компетенций	Шкала оценок
Обучающийся в полной мере владеет понятийным аппаратом данной области науки (теоретическими основами дисциплины), способен иллюстрировать ответ примерами, фактами, данными научных исследований, применять теоретические знания для решения практических задач в операционных систем, готов к использованию современных алгоритмов и языков программирования профессионала.	Повышенный уровень	Отлично
Обучающийся владеет понятийным аппаратом функциональной электроники (теоретическими основами дисциплины), готов к использованию современных алгоритмов и языков программирования на уровне продвинутого пользователя	Базовый уровень	Хорошо
Обучающийся владеет частично теоретическими основами дисциплины, фрагментарно способен к использованию современных алгоритмов и языков программирования на уровне обычного пользователя	Пороговый уровень	Удовлетворительно
Обучающийся демонстрирует отрывочные, фрагментарные знания, допускает грубые ошибки.	–	Неудовлетворительно

### **19.3. Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующие этапы формирования компетенций в процессе освоения образовательной программы**

#### 19.3.1 Перечень вопросов к экзамену:

1. Асимптотическая сложность.
2. Алгоритм “Линейный поиск”. Алгоритм “Бинарный поиск”.
3. Алгоритм “Рекурсивный поиск”.
4. Алгоритм “Решето Эратосфена”.
5. Алгоритм “Сортировка выбором (Selection sort)”.
6. Алгоритм “Сортировка вставкой (Insertion sort)”.
7. Алгоритм “Сортировка пузырьком (Bubble sort)”.
8. Алгоритм “Сортировка слиянием (Merge sort)”.
9. Алгоритм “Быстрая сортировка (fast sort)”.
10. Алгоритмы интегрирования: “Формула прямоугольников”.
11. Алгоритмы интегрирования: “Формула трапеций”.
12. Алгоритмы интегрирования: “Адаптивная квадратура”.
13. Алгоритмы интегрирования: “Метод Монте-Карло”.
14. Алгоритмы поиска корней уравнения: “Метод Ньютона-Рафсона”.
15. Алгоритм Евклида.
16. Линейный конгруэнтный генератор и случайные величины.
17. Нахождение простых множителей.
18. Тесты простоты. Малая теорема Ферма.
19. Численные алгоритмы: факториал, числа Фибоначчи, перевод из двоичной в десятичную систему счисления и наоборот.
20. Численные алгоритмы: расчет детерминанта матрицы произвольного порядка.
21. Алгоритм “Ханойская башня”.
22. Определение “стэка” и “кучи”. Стэк вызовов.
23. Связный список. Операции: вставка в начало, вставка в конец, вставка после указанного элемента, удаление.
24. Хэш-таблицы. Хэш-функции.
25. Деревья. Бинарные деревья. Операции: вставка слева, вставка вправо, поиск в глубину, поиск в ширину.
26. Бинарное дерево поиска. Поиск и удаление в бинарном дереве поиска.
27. Веб-приложения: основные технологии клиент-серверных приложений. Основные тэги HTML. Работа с CSS и JavaScript.
28. Строковые алгоритмы: вычисление арифметических выражений. Синтаксические деревья.

25. Строковые алгоритмы: детерминированные конечные автоматы и регулярные выражения.
26. Строковые алгоритмы: недетерминированные конечные автоматы и поиск строк.
27. Строковые алгоритмы: алгоритм "Бойера-Мура", редакционный граф.
28. Алгоритмы шифрования: перестановочные шифры.
29. Алгоритмы шифрования: шифр Цезаря, шифр Виженера, шифры подстановки.
30. Алгоритмы шифрования: блочные шифры.
31. Алгоритмы шифрования: шифрование с открытым ключом, функция Эйлера, обратные величины.
32. Типизация: бестиповая, сильная / слабая, явная / неявная, строгая / нестрогая, динамическая / статическая.
33. Обобщенное программирование.
34. Сетевые алгоритмы: обход в глубину, обход в ширину.
35. Сетевые алгоритмы: основные деревья, минимальные основные деревья.
36. Сетевые алгоритмы: поиск кратчайшего пути с помощью установки меток, поиск кратчайшего пути от корня к узлу.
37. Сетевые алгоритмы: поиск кратчайшего пути с помощью алгоритма Флойда-Уоршелла.

### 19.3.2 Перечень практических заданий

1. Сортировка сдвигом.
2. Сортировка пузырьком.
3. Сортировка слиянием.
4. Быстрая сортировка.
5. Линейный поиск.
6. Бинарный поиск.
7. Метод Ньютона.
8. Интегрирование методом прямоугольников.
9. Интегрирование методом трапеций.
10. Интегрирование методом адаптивной квадратуры.
11. Псевдографика: построение графика функций.
12. Нахождение наибольшего делителя.
13. Поиск в ширину.
14. Поиск в глубину.
15. Нахождение реакционного расстояния.
16. Регулярные выражения.
17. Шифр Цезаря.
18. Шифр Виженера.
19. Кто сделал это?
20. Практика HTML.
21. Практика CSS.
22. Практика JavaScript.
23. Свой личный блог.
24. Умный подсчет расхода воды.
25. Препятствие.
26. Монетки на сдачу.
27. Отцы основатели.
28. Игра Пятнашки.
29. Построение графиков погоды.
30. Бэкенд сайта.
31. Работа со строками.
32. Линейная алгебра.
33. Использование UML.
34. Создание приложения по подсчету очков в Android.
35. Создание приложения по планированию задач с таймингом.

19.3.4 Тестовые задания

19.3.4 Перечень заданий для контрольных работ

19.3.5 Темы курсовых работ

19.3.6 Темы рефератов

#### **19.4. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций**

Оценка знаний, умений и навыков, характеризующая этапы формирования компетенций в рамках изучения дисциплины осуществляется в ходе текущей и промежуточной аттестаций.

Текущая аттестация проводится в соответствии с Положением о текущей аттестации обучающихся по программам высшего образования Воронежского государственного университета. Текущая аттестация проводится в форме тестирования. Критерии оценивания приведены выше.

Промежуточная аттестация проводится в соответствии с Положением о промежуточной аттестации обучающихся по программам высшего образования.

Контрольно-измерительные материалы промежуточной аттестации включают в себя теоретические вопросы, позволяющие оценить уровень полученных знаний и практические задания, позволяющие оценить степень сформированности умений и навыков.

При оценивании используются качественные шкалы оценок. Критерии оценивания приведены выше.