

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ**  
федеральное государственное автономное образовательное учреждение  
высшего образования  
«Балтийский федеральный университет имени Иммануила Канта»  
Высшая школа компьютерных наук и искусственного интеллекта

**РАБОЧАЯ ПРОГРАММА МОДУЛЯ**

**Модуль 3. Фундаментальная информатика**

**Шифр: 02.03.02**

**Направление подготовки: Фундаментальная информатика и информационные  
технологии**

**Профиль: Программная инженерия в искусственном интеллекте**

**Квалификация (степень) выпускника: бакалавр**

Калининград  
2023

## Лист согласования

### Составители:

- Верещагин Михаил Дмитриевич, к.ф.-м.н., доцент
- Мищук Богдан Ростиславович, к.ф.-м.н., доцент
- Тарачков Михаил Владимирович, ассистент,
- Савкин Дмитрий Александрович, доцент.

Рабочая программа утверждена на заседании  
Ученого совета ОНК «Институт высоких технологий»

Протокол № 33 от «27» октября 2023 г.

Председатель Ученого совета ОНК «Институт высоких технологий»

Профессор, д.ф.-м.н.,  
руководитель ОНК «Институт высоких технологий»

А.В. Юров

Директор высшей школы компьютерных наук  
и искусственного интеллекта

М.Д. Верещагин

Руководитель ОПОП ВО

С.С. Головин

## Содержание

1. Название образовательного модуля «Модуль 3. Фундаментальная информатика»
2. Характеристика образовательного модуля
3. Методические указания для обучающихся по освоению модуля
4. Программы дисциплин образовательного модуля
  - 4.1. Программа дисциплины «Основы программирования»
  - 4.2. Программа дисциплины «Объектно-ориентированное программирование»
  - 4.3. Программа дисциплины «Операционные системы»
  - 4.4. Программа дисциплины «Алгоритмы и структуры данных»
  - 4.5. Программа дисциплины «Основы программной инженерии»
  - 4.6. Программа дисциплины «Архитектура компьютеров»
  - 4.7. Программа дисциплины «Язык Python»
  - 4.8. Программа дисциплины «Программирование на языке C#»
  - 4.9. Программа дисциплины «Информационная безопасность»
  - 4.10. Программа дисциплины «Функциональное программирование»
  - 4.11. Программа дисциплины «Тестирование и внедрение ПО»
5. Программа практики
6. Программа итоговой аттестации по модулю

## 1. Название модуля: «Модуль 3. Фундаментальная информатика»

### 2. Характеристика модуля

#### 2.1. Образовательные цели и задачи:

Модуль ставит своей целью создать условия для эффективного формирования и развития универсальных и общепрофессиональных компетенций.

Для достижения поставленной цели необходимо решить следующие задачи:

- Способствовать формированию у обучающихся практических навыков по программированию на современных языках программирования.
- Формировать у обучающихся базовые знания о принципах организации современных ЭВМ, комплексов и систем, овладение студентами основными приемами и методами программного управления средствами вычислительной техники на ассемблерном уровне.
- Формировать базовые знания по вопросам построения компьютерных сетей различной модификации и изучение основных видов операционных систем.
- Способствовать формированию навыков, связанных с методами построения корректных и эффективных алгоритмов и структур данных, их разработкой и использованием в различных сферах.

#### 2.2. Образовательные результаты выпускника

Код компетенции	Результаты освоения образовательной программы (ИДК)	Результаты обучения по дисциплине
<b>УК-1.</b> Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач	<p>УК.1.1. Выбирает источники информации и осуществляет поиск информации для решения поставленных задач</p> <p>УК. 1.2. Демонстрирует умение рассматривать различные точки зрения и выявлять степень доказательности на поставленную задачу</p> <p>УК.1.3. Определяет рациональные идеи для решения поставленных задач</p>	<p><b>знать:</b></p> <ul style="list-style-type: none"><li>• основные математические методы и системы программирования;</li><li>• основные принципы метода системного подхода;</li><li>• базовые структуры данных и алгоритмы их обработки;</li><li>• современные направления создания новых структур данных и алгоритмов их использования;</li></ul> <p><b>уметь:</b></p> <ul style="list-style-type: none"><li>• применять системы программирования для разработки и реализации алгоритмов решения прикладных задач;</li><li>• применять метод системного подхода при выборе соответствующей структуры данных и алгоритмов её</li></ul>

		<p>обработки;</p> <ul style="list-style-type: none"> <li>• анализировать предметную область задачи и предлагать новые структуры данных и алгоритмы их обработки;</li> </ul> <p><b>владеть:</b></p> <ul style="list-style-type: none"> <li>• практическими навыками применения систем программирования и реализации алгоритмов;</li> <li>• применения метода системного подхода при разработке различных алгоритмов.</li> </ul>
<p>ОПК-2. Способен использовать и адаптировать существующие математические методы и системы программирования для разработки и реализации алгоритмов решения прикладных задач.</p>	<p>ОПК-2.1. Имеет представление о существующих базовых математических методах и системах программирования, применяемых для разработки и реализации алгоритмов решения прикладных задач.</p> <p>ОПК-2.2. Применяет математические методы и системы программирования для решения прикладных задач.</p> <p>ОПК-2.3. Разрабатывает и применяет алгоритмические и программные решения прикладных задач.</p>	<p><b>Знать:</b></p> <ul style="list-style-type: none"> <li>• теоретические основы написания программного обеспечения;</li> <li>• базовые понятия и терминологию курса ОС; понятие мультипрограммирования; понятия процесса и ядра; структуру контекста процесса;</li> <li>• фундаментальные компьютерные алгоритмы и структуры данных;</li> <li>• классификацию алгоритмов по степени их сложности и по типам используемых структур данных;</li> <li>• достоинства и недостатки каждого вида структур данных для применения при решении различных задач</li> </ul> <p><b>Уметь:</b></p> <ul style="list-style-type: none"> <li>• писать программный код на одном из языков программирования, составлять алгоритмы для решения задач;</li> <li>• использовать средства и функции операционных систем для управления аппаратными ресурсами;</li> <li>• выполнять оценку эффективности алгоритмов и их применимости;</li> </ul>

		<ul style="list-style-type: none"> <li>• проектировать и реализовывать структуры данных для построения эффективных программных комплексов, используя возможности языков высокого уровня</li> </ul> <p><b>Владеть:</b></p> <ul style="list-style-type: none"> <li>• применения теоретических основ программирования (основные понятия, типы данных, базовые операторы, алгоритмы и структуры данных) для решения задач;</li> <li>• настройки и осуществления работы в многопользовательском режиме; использования языков и систем программирования, инструментальными средствами для решения профессиональных, исследовательских и прикладных задач;</li> <li>• использования существующих структур данных и алгоритмов их обработки при разработке программных модулей и компонент, а также при их верификации.</li> </ul>
<p>ОПК-4. Способен понимать принципы работы современных информационных технологий и использовать их для решения задач профессиональной деятельности.</p>	<p>ОПК-4.1. Имеет представление об основных существующих информационных технологиях, используемых при решении профессиональных задач.</p> <p>ОПК-4.2. Анализирует существующие информационные технологии при решении задач профессиональной деятельности.</p> <p>ОПК-4.3. Применяет существующие информационные технологии при решении задач профессиональной деятельности.</p>	<p><b>Знать:</b></p> <ul style="list-style-type: none"> <li>- основные понятия и алгоритмы по курсу «Объектно-ориентированное программирование»;</li> <li>• основные направления развития вычислительной техники;</li> <li>• понятия идентификатора и дескриптора процесса; понятия приоритета и очереди процессов; понятие событийного программирования; настройки операционных систем для решения различных задач; принципы функционирования и взаимодействия аппаратных и программных средств компьютерной техники; способы настройки ОС Microsoft Windows и Unix для работы в сетях;</li> </ul>

		<p><b>Уметь:</b></p> <ul style="list-style-type: none"> <li>- применять полученные знания в исследовательской работе;</li> <li>- выбирать и использовать программные средства для анализа программного обеспечения;</li> <li>- выявлять причины возникновения дефектов информационных систем;</li> <li>- обеспечивать надежное функционирование информационных систем;</li> <li>• настраивать пользовательский интерфейс и сетевое окружение; устанавливать иерархию процессов; задавать приоритет процессам; использовать системные прерывания; предоставлять доступ к локальным ресурсам и использовать сетевые ресурсы; работать с современным сетевым программным обеспечением: клиентскими программами протокола передачи файлов, клиентскими программами удаленного администрирования, вспомогательными программами сетевых служб;</li> </ul> <p><b>Владеть:</b></p> <ul style="list-style-type: none"> <li>• технологией разработки фиксации и учета дефектов информационных систем и программного обеспечения;</li> <li>• технологией тестирования программного обеспечения;</li> <li>• приемами тестирования информационных систем;</li> <li>• внешним интерфейсом ОС для реализации мультипрограммирования и обеспечения коммуникации процессов; навыками определения узлов в одной подсети; настройки связи в локальных сетях и в сетях Ethernet, Internet, VPN; технологией предоставления доступа к общим ресурсам</li> <li>• работы в среде QtCreator</li> </ul>
--	--	--

<p>ОПК-5. Способен разрабатывать алгоритмы и компьютерные программы, пригодные для практического применения.</p>	<p>ОПК-5.1. Имеет представление о базовых структурах данных и алгоритмах.  ОПК-5.2. Разрабатывает алгоритмы для решения задач профессиональной деятельности.  ОПК-5.3. Создает на основе разработанных алгоритмов программное обеспечение.</p>	<p><b>Знать:</b></p> <ul style="list-style-type: none"> <li>- принципы разработки программного кода</li> <li>- основные концептуальные положения объектно-ориентированного программирования</li> </ul> <p><b>Уметь:</b></p> <ul style="list-style-type: none"> <li>- осуществлять моделирование информационных процессов</li> <li>- разрабатывать программы на алгоритмических языках высокого уровня</li> <li>- производить отладку программного обеспечения</li> </ul> <p><b>Владеть:</b></p> <ul style="list-style-type: none"> <li>- обеспечением жизненного цикла информационных продуктов и услуг</li> </ul>
--	--	--

### 3. Методические указания для обучающихся по освоению модуля

Освоение дисциплин модуля закладывает базу для будущей профессиональной деятельности в сфере ИТ. Оно должно начинаться с внимательного ознакомления с рабочими программами дисциплин, обязательными компонентами которых являются: перечень тем, подлежащих усвоению; задания; списки учебных пособий и рекомендуемой литературы; списки контрольных вопросов, заданий.

При изучении дисциплин модуля необходимо последовательно переходить от дисциплины к дисциплине, от темы к теме, следуя внутренней логике, заложенной в программе дисциплины модуля. Только так можно достичь полного понимания материала, хорошей ориентации в специальной литературе, формирования собственной точки зрения и умений практического характера. Для более глубокого и эффективного освоения дисциплин рекомендуется предварительная подготовка к занятиям.

## 4. Программы дисциплин модуля

### 4.1 Программа дисциплины «Основы программирования»

#### 1. Наименование дисциплины: «Основы программирования».

**Цель** дисциплины: целью освоения дисциплины «Основы программирования» является получение теоретических знаний и практических навыков обучающихся в области программирования.



## 2. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы

Код компетенции	Результаты освоения образовательной программы (ИДК)	Результаты обучения по дисциплине
<p>ОПК-2. Способен использовать и адаптировать существующие математические методы и системы программирования для разработки и реализации алгоритмов решения прикладных задач</p>	<p>ОПК-2.1. Имеет представление о существующих базовых математических методах и системах программирования, применяемых для разработки и реализации алгоритмов решения прикладных задач.</p> <p>ОПК-2.2. Применяет математические методы и системы программирования для решения прикладных задач.</p> <p>ОПК-2.3. Разрабатывает и применяет алгоритмические и программные решения прикладных задач.</p>	<p>- <b>знать</b> основные математические методы и системы программирования;</p> <p>- <b>уметь</b> применять системы программирования для разработки и реализации алгоритмов решения прикладных задач;</p> <p>- <b>владеть</b> практическими навыками применения систем программирования и реализации алгоритмов.</p>
<p>ОПК-5. Способен разрабатывать алгоритмы и компьютерные программы, пригодные для практического применения</p>	<p>ОПК-5.1. Имеет представление о базовых структурах данных и алгоритмах.</p> <p>ОПК-5.2. Разрабатывает алгоритмы для решения задач профессиональной деятельности.</p> <p>ОПК-5.3. Создает на основе разработанных алгоритмов программное обеспечение.</p>	<p>- <b>знать</b> теоретические основы написания программного обеспечения;</p> <p>- <b>уметь</b> писать программный код на одном из языков программирования, составлять алгоритмы для решения задач;</p> <p>- <b>владеть</b> практическими навыками применения теоретических основ программирования (основные понятия, типы данных, базовые операторы, алгоритмы и структуры данных) для решения задач.</p>

### 3. Место дисциплины в структуре образовательной программы

«Основы программирования» представляет собой дисциплину обязательной части (Б1.О.13) направления подготовки бакалавриата 02.03.02 «Фундаментальная информатика и информационные технологии», профиль «Программная инженерия в искусственном интеллекте».

### 4. Виды учебной работы по дисциплине.

Виды учебной работы по дисциплине зафиксированы учебным планом основной профессиональной образовательной программы по указанному направлению и профилю, выражаются в академических часах. Часы контактной работы и самостоятельной работы обучающегося и часы, отводимые на процедуры контроля, могут различаться в учебных планах ОПОП по формам обучения. Объем контактной работы включает часы контактной аудиторной работы (лекции/практические занятия/ лабораторные работы), контактной внеаудиторной работы (контроль самостоятельной работы), часы контактной работы в период аттестации. Контактная работа, в том числе может проводиться посредством электронной информационно-образовательной среды университета с использованием ресурсов сети Интернет и дистанционных технологий

### 5. Содержание дисциплины, структурированное по темам (разделам)

Исходя из рамок, установленных учебным планом по трудоемкости и видам учебной работы по дисциплине, преподаватель самостоятельно выбирает тематику занятий по формам и количеству часов проведения контактной работы: лекции и иные учебные занятия, предусматривающие преимущественную передачу учебной информации преподавателем и (или) занятия семинарского типа (семинары, практические занятия, практикумы, лабораторные работы, коллоквиумы и иные аналогичные занятия), и (или) групповые консультации, и (или) индивидуальную работу обучающихся с преподавателем, в том числе индивидуальные консультации (по курсовым работам/проектам – при наличии курсовой работы/проекта по данной дисциплине в учебном плане). Рекомендуемая тематика занятий максимально полно реализуется в контактной работе с обучающимися очной формы обучения. В случае реализации образовательной программы в заочной / очно-заочной форме трудоемкость дисциплины сохраняется, однако объем учебного материала в значительной части осваивается обучающимися в форме самостоятельной работы. При этом требования к ожидаемым образовательным результатам обучающихся по данной дисциплине не зависят от формы реализации образовательной программы.

№	Наименование раздела	Содержание раздела
1	Основы построение алгоритмов	Блок-схемы. Принципы построения алгоритмов для решения поставленных задач.
2	Введение в язык программирования C++.	Среда разработки, типы данных, базовые конструкции.

3	Системы контроля версий.	Применение системы контроля версий для хранения и передачи создаваемого программного кода.
4	Система автоматизации сборки проекта CMake.	Использование CMake для автоматизации создания проекта.
5	Указатели и динамическая память.	Основы работы с динамическим выделением памяти. Сырые и умные указатели. Идиома RAII.
6	ООП	Абстракция, инкапсуляция, наследование полиморфизм. Создание классов. Использование объектно-ориентированного подхода для написания программного обеспечения. Перегрузка операторов. Шаблоны.
7	Использование библиотек	Основы работы со сторонними библиотеками в собственных проектах.
8	Создание библиотек	Разработка собственных библиотек (статических и динамических). Использование их в проектах на C++ и других языках.
9	Алгоритмы и структуры данных	Псевдокод, теория сложности, базовые алгоритмы поиска, сортировки. Структуры данных.
10	Обработка ошибок	Использование кодов ошибок. Исключения.
11	Математика и программирование	Использование математических моделей в программировании
12	Проектная деятельность	Разработка проектов по результатам обучения

## 6. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине

Рекомендуемая тематика учебных занятий *лекционного* типа (предусматривающих преимущественную передачу учебной информации преподавателями):

№	Наименование раздела	Темы лекций
1	Введение в язык программирования C++.	Лекция 1. Программный код и как его обрабатывает компьютер. Языка программирования. IDE. Компилятор. Типы данных. Арифметические

		<p>операторы. Первая программа.</p> <p>Лекция 2. Условия, операторы сравнения, логические операторы.</p> <p>Лекция 3. Циклы.</p> <p>Лекция 4. Массивы.</p> <p>Лекция 5. Строки.</p> <p>Лекция 6. Функции.</p>
2	Системы контроля версий / Система автоматизации сборки проекта CMake.	<p>Лекция 7. Использование CMake и системы контроля версий.</p> <p>Лекция 8. Ветвление, слияния, pull request в системе контроля версий.</p>
3	Указатели и динамическая память.	<p>Лекция 9. Работа с памятью. Сырые указатели.</p> <p>Лекция 10. Умные указатели. Идиома RAII.</p>
4	ООП	<p>Лекция 11. Создание первого класса. Абстрация, инкапсуляция.</p> <p>Лекция 12. Наследование и полиморфизм.</p> <p>Лекция 13. Перегрузка операторов.</p> <p>Лекция 14. Создание шаблонного класса.</p>
5	Использование библиотек	<p>Лекция 15. Использование библиотеки SFML/OpenCV/ASIO.</p> <p>Лекция 16. Подключение и изучение библиотек сторонних производителей.</p>
6	Создание библиотек	<p>Лекция 17. Создание статической библиотеки.</p> <p>Лекция 18. Создание динамической библиотеки.</p> <p>Лекция 19. Использование собственных библиотек.</p>
7	Алгоритмы и структуры данных	<p>Лекция 20. Понятие алгоритма, блок-схемы, псевдокод. Теория сложности.</p> <p>Лекция 21. Рекурсия.</p> <p>Лекция 22. Алгоритмы сортировки.</p> <p>Лекция 23. Бинарный поиск и бисекция.</p> <p>Лекция 24. Односвязный и двусвязный циклический</p>

		список. Лекция 25. Бинарное дерево поиска.
8	Обработка ошибок	Лекция 26. Коды ошибок. Лекция 27. Исключения.
9	Математика и программирование	Лекция 28. Изучение реализации математических операций и моделей на языке программирования. Лекция 29. Использование математических моделей для анимации процессов.
10	Проектная деятельность	Лекция 30. Создание проектов.

**Рекомендуемая тематика *практических* занятий:**

- Базовые конструкции.
- Условия.
- Циклы.
- Массивы.
- Строки.
- Функции.
- Система контроля версий и CMake.
- Ссылки, указатели, динамическая память.
- Основы работы с библиотекой SFML.
- Основы ООП.
- Улучшение кода при помощи ООП.
- Геймплей.
- Подготовка программного обеспечения к релизу.
- Создание инсталлятора.
- Введение в алгоритмы и структуры данных.
- Рекурсия.
- Алгоритмы сортировки.
- Бинарный поиск и бисекция.
- Односвязный и двусвязный циклический список.
- Бинарное дерево поиска.
- Ветки, слияния, pull request.
- Перегрузка операторов.
- Шаблоны.
- Создание библиотек.
- Тестирование.

**Требования к самостоятельной работе обучающихся**

1. Работа с лекционным материалом, предусматривающая проработку конспекта лекций и учебной литературы, по всем темам из п. 6 настоящей рабочей программы.

2. Выполнение домашнего задания, предусматривающего решение задач, выполнение упражнений, выдаваемых на практических занятиях, по всем темам из п. 6 настоящей рабочей программы.

Руководствуясь положениями статьи 47 и статьи 48 Федерального закона от 29 декабря 2012 г. N 273-ФЗ «Об образовании в Российской Федерации» научно-педагогические работники и иные лица, привлекаемые университетом к реализации данной образовательной программы, пользуются предоставленными академическими правами и свободами в части свободы преподавания, свободы от вмешательства в профессиональную деятельность; свободы выбора и использования педагогически обоснованных форм, средств, методов обучения и воспитания; права на творческую инициативу, разработку и применение авторских программ и методов обучения и воспитания в пределах реализуемой образовательной программы и отдельной дисциплины.

Исходя из рамок, установленных учебным планом по трудоемкости и видам учебной работы по дисциплине, преподаватель самостоятельно выбирает тематику занятий по формам и количеству часов проведения контактной работы: лекции и иные учебные занятия, предусматривающие преимущественную передачу учебной информации преподавателем и (или) занятия семинарского типа (семинары, практические занятия, практикумы, лабораторные работы, коллоквиумы и иные аналогичные занятия), и (или) групповые консультации, и (или) индивидуальную работу обучающихся с преподавателем, в том числе индивидуальные консультации (по курсовым работам/проектам – при наличии курсовой работы/проекта по данной дисциплине в учебном плане).

Рекомендуемая тематика занятий максимально полно реализуется в контактной работе с обучающимися очной формы обучения. В случае реализации образовательной программы в заочной / очно-заочной форме трудоемкость дисциплины сохраняется, однако объем учебного материала в значительной части осваивается обучающимися в форме самостоятельной работы. При этом требования к ожидаемым образовательным результатам обучающихся по данной дисциплине не зависят от формы реализации образовательной программы.

## **7. Методические рекомендации по видам занятий**

Лекционные занятия.

В ходе лекционных занятий обучающимся рекомендуется выполнять следующие действия. Вести конспектирование учебного материала. Обращать внимание на категории, формулировки, раскрывающие содержание тех или иных явлений и процессов, научные выводы и практические рекомендации по их применению. Задавать преподавателю уточняющие вопросы с целью уяснения теоретических положений, разрешения спорных ситуаций.

Желательно оставить в рабочих конспектах поля, на которых во внеаудиторное время можно сделать пометки из рекомендованной литературы, дополняющие материал

прослушанной лекции, а также подчеркивающие особую важность тех или иных теоретических положений.

Практические и семинарские занятия.

На практических и семинарских занятиях в зависимости от темы занятия выполняется поиск информации по решению проблем, практические упражнения, контрольные работы, выработка индивидуальных или групповых решений, итоговое обсуждение с обменом знаниями, участие в круглых столах, разбор конкретных ситуаций, командная работа, представление портфолио и т.п.

Самостоятельная работа.

Самостоятельная работа осуществляется в виде изучения литературы, эмпирических данных по публикациям и конкретных ситуаций из практики, подготовке индивидуальных работ, работа с лекционным материалом, самостоятельное изучение отдельных тем дисциплины; поиск и обзор литературы и электронных источников; чтение и изучение учебника и учебных пособий.

## 8. Фонд оценочных средств

### 8.1. Перечень компетенций с указанием этапов их формирования в процессе освоения образовательной программы в рамках учебной дисциплины

Основными этапами формирования указанных компетенций при изучении обучающимися дисциплины являются последовательное изучение содержательно связанных между собой тем учебных занятий. Изучение каждой темы предполагает овладение обучающимися необходимыми компетенциями. Результат аттестации обучающихся на различных этапах формирования компетенций показывает уровень освоения компетенций.

Контролируемые разделы (темы) дисциплины	Индекс контролируемой компетенции (или её части)	Оценочные средства по этапам формирования компетенций
		текущий контроль по дисциплине
<ul style="list-style-type: none"> <li>• Основы построение алгоритмов</li> </ul>	ОПК-2 ОПК-5	Опрос, решение задач.
<ul style="list-style-type: none"> <li>• Введение в язык программирования C++.</li> </ul>	ОПК-2 ОПК-5	Опрос, решение задач, контрольная работа
<ul style="list-style-type: none"> <li>• Системы контроля версий.</li> </ul>	ОПК-2 ОПК-5	Опрос, решение задач
<ul style="list-style-type: none"> <li>• Система автоматизации сборки проекта CMake.</li> </ul>	ОПК-2 ОПК-5	Опрос, решение задач
<ul style="list-style-type: none"> <li>• Указатели и динамическая память.</li> </ul>	ОПК-2 ОПК-5	Опрос, решение задач
<ul style="list-style-type: none"> <li>• ООП</li> </ul>	ОПК-2 ОПК-5	Опрос, решение задач
<ul style="list-style-type: none"> <li>• Использование библиотек</li> </ul>	ОПК-2 ОПК-5	Опрос, решение задач,
<ul style="list-style-type: none"> <li>• Создание библиотек</li> </ul>	ОПК-2	Опрос, решение задач, контрольная

	ОПК-5	работа
• Алгоритмы и структуры данных	ОПК-2 ОПК-5	Опрос, решение задач
• Обработка ошибок	ОПК-2 ОПК-5	Опрос, решение задач
• Математика и программирование	ОПК-2 ОПК-5	Опрос, решение задач
• Проектная деятельность	ОПК-2 ОПК-5	Опрос, решение задач

## 8.2. Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений и навыков и (или) опыта деятельности процессе текущего контроля

### Примеры вопросов для устного опроса:

#### По Теме 2. Введение в язык программирования C++

- Какие бывают типы данных в C++?
- Что такое IDE?
- Какие бывают этапы компиляции?
- Какие бывают виды циклов?
- Какие существуют способы передачи аргументов в функцию?

#### По Теме 4. Алгоритмы и структуры данных

- Оцените сложность быстрой сортировки.
- Оцените сложность бинарного поиска.
- Теория сложности.
- Оцените сложность операции поиска в односвязном списке.
- Назовите 2 условия существования рекурсии.

### Типовые контрольные задания:

#### Тема: Введение в язык программирования C++

- Напишите программу для определения простоты числа.
- Напишите программу для поиска суммы цифр числа.
- Напишите программу для сортировки массива.
- Найдите наибольший элемент матрицы.

#### Тема: Алгоритмы и структуры данных.

- Напишите реализацию быстрой сортировки.



- Напишите реализацию односвязного списка.
- Определите оптимальную структуру хранения данных, если над ней преобладают операции поиска. Минимальны операции удаления и вставки.

### 8.3. Перечень вопросов и заданий для промежуточной аттестации по дисциплине

#### Вопросы для промежуточного контроля (зачета)

- Императивная парадигма программирования. История развития языков C/C++. Их достоинства и недостатки. Сферы применения.
- Типы данных в C++. Правила наименования переменных. Переполнение и потеря данных.
- Консольный и файловый ввод/вывод. Спецификаторы ввода/вывода.
- Условные операторы. Построение условий при помощи логических операторов.
- Организация циклов.
- Статические массивы. Работа с 1- и 2-мерными массивами. Переполнение массива.
- Указатели и ссылки. Примеры.
- Создание функций. Передача аргументов по значению и по ссылке.
- Кодирование символьной информации в компьютере. Организация чтения текста из файла. Функции библиотеки string.
- Создание собственных библиотек. Пример.
- Понятие сложности алгоритма. Пример расчета сложности для сортировки обменов.
- Волновой алгоритм поиска кратчайшего пути в лабиринте.
- Алгоритм быстрой сортировки.
- Алгоритм бинарного поиска.
- Стек и очередь.
- Алгоритм генерации перестановок.
- Алгоритм генерации подмножеств.
- Нормативные документы в работе программиста.

#### Вопросы для промежуточного контроля (экзамена)

- Оцените сложность быстрой сортировки.
- Оцените сложность сортировки слиянием.
- Оцените сложность сортировки подсчетом.
- Оцените сложность бинарного дерева поиска по основным операциям.
- Напишите псевдокод для односвязного списка.
- Напишите псевдокод для двусвязного циклического списка.
- Напишите псевдокод для бинарного поиска.
- Напишите псевдокод для бисекции.
- Основные типы умных указателей.
- Основные принципы ООП.
- Перегрузка операторов.
- Основные операции системы контроля версий.

- Как используется CMake.
- В чем отличие статической и динамической библиотеки.
- Каким образом подключается сторонняя библиотека к проекту.

#### 8.4. Планируемые уровни сформированности компетенций обучающихся и критерии оценивания

Уровни	Содержательное описание уровня	Основные признаки выделения уровня (этапы формирования компетенции, критерии оценки сформированности)	Пятибалльная шкала (академическая) оценка	Двухбалльная шкала, зачет	БРС, % освоения (рейтинговая оценка)
Повышенный	Творческая деятельность	<i>Включает нижестоящий уровень.</i> Умение самостоятельно принимать решение, решать проблему/задачу теоретического и прикладного характера на основе изученных методов, приемов, технологий	отлично	зачтено	86-100
Базовый	Применение знаний и умений в более широких контекстах учебной и профессиональной деятельности, нежели по образцу с большей степени самостоятельности и инициативы	<i>Включает нижестоящий уровень.</i> Способность собирать, систематизировать, анализировать и грамотно использовать информацию из самостоятельно найденных теоретических источников и иллюстрировать ими теоретические положения или обосновывать практику применения	хорошо		71-85
Удовлетворительный	Репродуктивная	Изложение в пределах задач курса	удовлетвор		55-70

(достаточны й)	деятельность	теоретически практически контролируемого материала	и ительно		
Недостаточн ый	Отсутствие признаков удовлетворительного уровня		неудовлетв орительно	не зачтено	Менее 55

## **9. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины.**

### **Основная литература**

- Кузин, А. В. Программирование на языке Си : учебное пособие / А. В. Кузин, Е. В. Чумакова. — Москва : ФОРУМ : ИНФРА-М, 2019. — 144 с. — (Высшее образование). - ISBN 978-5-00091-066-5. - Текст : электронный. - URL: <https://znanium.com/catalog/product/1007488> (дата обращения: 03.04.2023). – Режим доступа: по подписке.
- Царев, Р. Ю. Программирование на языке Си : учеб. пособие / Р. Ю. Царев. - Красноярск : Сиб. федер. ун-т, 2014. - 108 с. - ISBN 978-5-7638-3006-4. - Текст : электронный. - URL: <https://znanium.com/catalog/product/510946> (дата обращения: 20.11.2023). – Режим доступа: по подписке.
- Михайличенко, Ж. В. Программирование на языке Си : учебно-методическое пособие / Ж. В. Михайличенко, М. А. Кузниченко, В. С. Яне. - 2-е изд., стер. - Москва : ФЛИНТА, 2017. - 103 с. - ISBN 978-5-9765-3434-6. - Текст : электронный. - URL: <https://znanium.com/catalog/product/1583197> (дата обращения: 20.11.2023). – Режим доступа: по подписке.

### **Дополнительная литература**

- Литвиненко, В. А. Программирование на C++ задач на графах: Учебное пособие / Литвиненко В.А. - Таганрог: Южный федеральный университет, 2016. - 83 с.: ISBN 978-5-9275-2311-5. - Текст : электронный. - URL: <https://znanium.com/catalog/product/997083> (дата обращения: 03.04.2023). – Режим доступа: по подписке.

## **10. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины (модуля).**

- НЭБ Национальная электронная библиотека, диссертации и прочие издания
- eLIBRARY.RU Научная электронная библиотека, книги, статьи, тезисы докладов конференций
- Гребенников Электронная библиотека ИД журналы
- ЭБС Консультант студента
- ПРОСПЕКТ ЭБС
- ЭБС ZNANIUM.COM

- ЭБС IBOOKS.RU
- РГБ Информационное обслуживание по МБА
- БЕН РАН
- Электронно-библиотечная система (ЭБС) Кантиана (<https://elib.kantiana.ru/>)

## **11. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине.**

Программное обеспечение обучения включает в себя:

- система электронного образовательного контента БФУ им. И. Канта – [www.lms.kantiana.ru](http://www.lms.kantiana.ru), обеспечивающую разработку и комплексное использование электронных образовательных ресурсов;
- серверное программное обеспечение, необходимое для функционирования сервера и связи с системой электронного обучения через Интернет;
- корпоративная платформа;
- установленное на рабочих местах обучающихся ПО: Microsoft Windows 7, Microsoft Office Standart 2010, антивирусное программное обеспечение Kaspersky Endpoint Security.
- Среда программирования Microsoft Visual Studio (любая версия);
- Qt версии 5.0 и выше

## **12. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине.**

Для проведения занятий лекционного типа, практических и семинарских занятий используются специальные помещения (учебные аудитории), оборудованные техническими средствами обучения – мультимедийной проекционной техникой. Для проведения занятий лекционного типа используются наборы демонстрационного оборудования.

Для проведения лабораторных работ, (практических занятий – при необходимости) используются специальные помещения (учебные аудитории), оснащенные специализированным лабораторным оборудованием: персональными компьютерами с возможностью выхода в интернет и с установленным программным обеспечением, заявленным в п.11.

Для проведения групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации используются специальные помещения (учебные аудитории), оборудованные специализированной мебелью (для обучающихся), меловой / маркерной доской.

Для организации самостоятельной работы обучающимся предоставляются помещения, оснащенные компьютерной техникой с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду университета.

Для обучения инвалидов и лиц с ограниченными возможностями здоровья университетом могут быть представлены специализированные средства обучения, в том числе технические средства коллективного и индивидуального пользования.

## 4.2 Программа дисциплины «Объектно-ориентированное программирование»

### 1. Наименование дисциплины: «Объектно-ориентированное программирование».

Целью освоения дисциплины «Объектно-ориентированное программирование» является изучение концепций и приемов гейм-дизайна применительно к проектированию и разработке компьютерных игр и образовательных мобильных приложений.

### 2. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы

Код компетенции	Результаты освоения образовательной программы (ИДК)	Результаты обучения по дисциплине
ОПК-4 - Способен понимать принципы работы современных информационных технологий и использовать их для решения задач профессиональной деятельности	ОПК-4.1. Имеет представление об основных существующих информационных технологиях, используемых при решении профессиональных задач ОПК-4.2. Анализирует существующие информационные технологии при решении задач профессиональной деятельности ОПК-4.3. Применяет существующие информационные технологии при решении задач профессиональной деятельности	<ul style="list-style-type: none"><li>• <b>знать:</b> основные понятия и алгоритмы по курсу «Объектно-ориентированное программирование»;</li><li>• <b>уметь:</b> применять полученные знания в исследовательской работе;</li><li>• <b>владеть:</b> практическими навыками работы в среде QtCreator</li></ul>
ОПК-5 - Способен разрабатывать алгоритмы и компьютерные программы, пригодные для практического применения	ОПК-5.1. Имеет представление о базовых структурах данных и алгоритмах. ОПК-5.2. Разрабатывает алгоритмы для решения задач профессиональной деятельности. ОПК-5.3. Создает на основе разработанных алгоритмов программное обеспечение	<ul style="list-style-type: none"><li>• <b>знать:</b> основные концептуальные положения объектно-ориентированного программирования;</li><li>• <b>уметь:</b> осуществлять моделирование информационных процессов;</li><li>• <b>владеть:</b> практическими навыками обеспечения жизненного цикла информационных продуктов и услуг</li></ul>

### 3. Место дисциплины в структуре образовательной программы

«Объектно-ориентированное программирование» представляет собой дисциплину обязательной части (Б1.О.13) направления подготовки бакалавриата 02.03.02

«Фундаментальная информатика и информационные технологии», профиль «Программная инженерия в искусственном интеллекте».

#### 4. Виды учебной работы по дисциплине.

Виды учебной работы по дисциплине зафиксированы учебным планом основной профессиональной образовательной программы по указанному направлению и профилю, выражаются в академических часах. Часы контактной работы и самостоятельной работы обучающегося и часы, отводимые на процедуры контроля, могут различаться в учебных планах ОПОП по формам обучения. Объем контактной работы включает часы контактной аудиторной работы (лекции/практические занятия/ лабораторные работы), контактной внеаудиторной работы (контроль самостоятельной работы), часы контактной работы в период аттестации. Контактная работа, в том числе может проводиться посредством электронной информационно-образовательной среды университета с использованием ресурсов сети Интернет и дистанционных технологий

#### 5. Содержание дисциплины, структурированное по темам (разделам)

Исходя из рамок, установленных учебным планом по трудоемкости и видам учебной работы по дисциплине, преподаватель самостоятельно выбирает тематику занятий по формам и количеству часов проведения контактной работы: лекции и иные учебные занятия, предусматривающие преимущественную передачу учебной информации преподавателем и (или) занятия семинарского типа (семинары, практические занятия, практикумы, лабораторные работы, коллоквиумы и иные аналогичные занятия), и (или) групповые консультации, и (или) индивидуальную работу обучающихся с преподавателем, в том числе индивидуальные консультации (по курсовым работам/проектам – при наличии курсовой работы/проекта по данной дисциплине в учебном плане). Рекомендуемая тематика занятий максимально полно реализуется в контактной работе с обучающимися очной формы обучения. В случае реализации образовательной программы в заочной / очно-заочной форме трудоемкость дисциплины сохраняется, однако объем учебного материала в значительной части осваивается обучающимися в форме самостоятельной работы. При этом требования к ожидаемым образовательным результатам обучающихся по данной дисциплине не зависят от формы реализации образовательной программы.

№	Наименование раздела	Содержание раздела
1	Обзор иерархии классов Qt. Философия объектной модели.	Qt - кроссплатформенная среда разработки программ на C++. Программа "Hello world". Модули Qt: QtCore, QtGui, QtNetwork, QtOpenGL, QtSQL, QtSVG, Tamp, QtPhonon. Механизм сигналов и слотов. Организация объектных иерархий. Метаобъектная информация.
2	Интегрированная среда разработки QtCreator.	Создание нового проекта. Структура проекта. Пользовательский интерфейс QtCreator. Редактирование текста. Добавление новых классов. Компиляция и запуск проекта. Работа с qmake. Метаобъектный компилятор МОС. Компилятор

		ресурсов RCC. Создание новых форм. Добавление виджетов. Компоновка графических элементов. Порядок следования табулятора. Сигналы и слоты. Использование форм в проектах. Динамическая загрузка формы.
3	Библиотека контейнеров	Контейнерные классы. Итераторы. Последовательные контейнеры. Ассоциативные контейнеры. Алгоритмы. Строки. Произвольный тип QVariant. Модель общего использования данных.
4	Элементы управления и отображения в графической программе на Qt	Класс QWidget. Стек виджетов. Стандартные виджеты Qt. Рамки. Кнопки. Класс QAbstractButton. Флажки. Переключатели. Группировка кнопок. Надписи. Индикатор прогресса. Электронный индикатор. Класс QAbstractSlider. Ползунок. Полоса прокрутки.
5	Управление автоматическим размещением элементов	Менеджеры компоновки (layout managers). Вертикальный и горизонтальный менеджеры компоновки. Сеточный менеджер компоновки. Менеджер компоновки в виде формы. Разделители QSplitter. Вложенные менеджеры компоновки.
6	Элементы ввода и выбора	Однорядное текстовое поле. Редактор текста. Проверка ввода и классы QIntValidator и QDoubleValidator. Простой список. Иерархические списки. Таблицы. Выпадающие списки. Закладки. Виджет панели инструментов.

#### **6. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине**

<b>№</b>	<b>Наименование раздела</b>	<b>Содержание раздела</b>
1	Обзор иерархии классов Qt. Философия объектной модели.	Лекция 1. Qt - кроссплатформенная среда разработки программ на C++. Программа "Hello world". Модули Qt: QtCore, QtGui, QtNetwork, QtOpenGL, QtSQL, QtSVG, Tmp, QtPhonon. Лекция 2. Механизм сигналов и слотов. Организация объектных иерархий. Метаобъектная информация.
2	Интегрированная среда разработки QtCreator.	Лекция 3. Создание нового проекта. Структура проекта. Пользовательский интерфейс QtCreator. Редактирование текста. Добавление новых классов. Лекция 4. Компиляция и запуск проекта. Работа с qmake. Метаобъектный компилятор МОС. Компилятор ресурсов RCC. Создание новых форм. Лекция 5. Добавление виджетов. Компоновка графических элементов. Порядок следования табулятора. Сигналы и слоты. Использование форм в проектах. Динамическая загрузка формы.
3	Библиотека контейнеров	Лекция 6. Контейнерные классы. Итераторы. Последовательные контейнеры. Ассоциативные контейнеры. Лекция 7. Алгоритмы. Строки. Произвольный тип

		QVariant. Модель общего использования данных.
4	Элементы управления и отображения в графической программе на Qt	Лекция 8. Класс QWidget. Стек виджетов. Стандартные виджеты Qt. Рамки. Кнопки. Класс QAbstractButton. Флажки. Переключатели. Лекция 9. Группировка кнопок. Надписи. Индикатор прогресса. Электронный индикатор. Класс QAbstractSlider. Ползунок. Полоса прокрутки.
5	Управление автоматическим размещением элементов	Лекция 10. Менеджеры компоновки (layout managers). Вертикальный и горизонтальный менеджеры компоновки. Лекция 11. Сеточный менеджер компоновки. Менеджер компоновки в виде формы. Разделители QSplitter. Лекция 12. Вложенные менеджеры компоновки.
6	Элементы ввода и выбора	Лекция 13. Однострочное текстовое поле. Редактор текста. Лекция 14. Проверка ввода и классы QIntValidator и QDoubleValidator. Простой список. Иерархические списки. Лекция 15. Таблицы. Выпадающие списки. Закладки. Лекция 16. Виджет панели инструментов.

**Рекомендуемая тематика учебных занятий лекционного типа** (предусматривающих преимущественную передачу учебной информации преподавателями):

**Рекомендуемая тематика практических занятий:**

№ п/п	Наименование Темы	Содержание темы
1	Обзор иерархии классов Qt. Философия объектной модели.	Лабораторные работы №1-№2
2	Интегрированная среда разработки QtCreator.	Лабораторные работы №3-№4
3	Библиотека контейнеров	Лабораторные работы №5-№6
4	Элементы управления и отображения в графической программе на Qt	Лабораторные работы №7-№8
5	Управление автоматическим размещением элементов	Лабораторные работы №9-№11
6	Элементы ввода и выбора	Лабораторные работы №12-№15

**Требования к самостоятельной работе обучающихся**



1. Работа с лекционным материалом, предусматривающая проработку конспекта лекций и учебной литературы, по всем темам из п. 6 настоящей рабочей программы.

2. Выполнение домашнего задания, предусматривающего решение задач, выполнение упражнений, выдаваемых на практических занятиях, по всем темам из п. 6 настоящей рабочей программы.

Руководствуясь положениями статьи 47 и статьи 48 Федерального закона от 29 декабря 2012 г. N 273-ФЗ «Об образовании в Российской Федерации» научно-педагогические работники и иные лица, привлекаемые университетом к реализации данной образовательной программы, пользуются предоставленными академическими правами и свободами в части свободы преподавания, свободы от вмешательства в профессиональную деятельность; свободы выбора и использования педагогически обоснованных форм, средств, методов обучения и воспитания; права на творческую инициативу, разработку и применение авторских программ и методов обучения и воспитания в пределах реализуемой образовательной программы и отдельной дисциплины.

Исходя из рамок, установленных учебным планом по трудоемкости и видам учебной работы по дисциплине, преподаватель самостоятельно выбирает тематику занятий по формам и количеству часов проведения контактной работы: лекции и иные учебные занятия, предусматривающие преимущественную передачу учебной информации преподавателем и (или) занятия семинарского типа (семинары, практические занятия, практикумы, лабораторные работы, коллоквиумы и иные аналогичные занятия), и (или) групповые консультации, и (или) индивидуальную работу обучающихся с преподавателем, в том числе индивидуальные консультации (по курсовым работам/проектам – при наличии курсовой работы/проекта по данной дисциплине в учебном плане).

Рекомендуемая тематика занятий максимально полно реализуется в контактной работе с обучающимися очной формы обучения. В случае реализации образовательной программы в заочной / очно-заочной форме трудоемкость дисциплины сохраняется, однако объем учебного материала в значительной части осваивается обучающимися в форме самостоятельной работы. При этом требования к ожидаемым образовательным результатам обучающихся по данной дисциплине не зависят от формы реализации образовательной программы.

## **7. Методические рекомендации по видам занятий**

Лекционные занятия.

В ходе лекционных занятий обучающимся рекомендуется выполнять следующие действия. Вести конспектирование учебного материала. Обращать внимание на категории, формулировки, раскрывающие содержание тех или иных явлений и процессов, научные выводы и практические рекомендации по их применению. Задавать преподавателю уточняющие вопросы с целью уяснения теоретических положений, разрешения спорных ситуаций.

Желательно оставить в рабочих конспектах поля, на которых во внеаудиторное время можно сделать пометки из рекомендованной литературы, дополняющие материал

прослушанной лекции, а также подчеркивающие особую важность тех или иных теоретических положений.

Практические и семинарские занятия.

На практических и семинарских занятиях в зависимости от темы занятия выполняется поиск информации по решению проблем, практические упражнения, контрольные работы, выработка индивидуальных или групповых решений, итоговое обсуждение с обменом знаниями, участие в круглых столах, разбор конкретных ситуаций, командная работа, представление портфолио и т.п.

Самостоятельная работа.

Самостоятельная работа осуществляется в виде изучения литературы, эмпирических данных по публикациям и конкретных ситуаций из практики, подготовке индивидуальных работ, работа с лекционным материалом, самостоятельное изучение отдельных тем дисциплины; поиск и обзор литературы и электронных источников; чтение и изучение учебника и учебных пособий.

## 8. Фонд оценочных средств

### 8.1. Перечень компетенций с указанием этапов их формирования в процессе освоения образовательной программы в рамках учебной дисциплины

Основными этапами формирования указанных компетенций при изучении обучающимися дисциплины являются последовательное изучение содержательно связанных между собой тем учебных занятий. Изучение каждой темы предполагает овладение обучающимися необходимыми компетенциями. Результат аттестации обучающихся на различных этапах формирования компетенций показывает уровень освоения компетенций.

Контролируемые разделы (темы) дисциплины	Индекс контролируемой компетенции (или её части)	Оценочные средства по этапам формирования компетенций
		текущий контроль по дисциплине
Обзор иерархии классов Qt. Философия объектной модели.	ОПК-4 ОПК-5	Лабораторные работы Тестирование
Интегрированная среда разработки QtCreator.	ОПК-4 ОПК-5	Лабораторные работы Тестирование
Библиотека контейнеров	ОПК-4 ОПК-5	Лабораторные работы Тестирование
Элементы управления и отображения в графической программе на Qt	ОПК-4 ОПК-5	Лабораторные работы Тестирование
Управление автоматическим размещением элементов	ОПК-4 ОПК-5	Лабораторные работы Тестирование
Элементы ввода и выбора	ОПК-4 ОПК-5	Лабораторные работы Тестирование

### 8.2. Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений и навыков и (или) опыта деятельности в процессе текущего контроля

#### Тема 1.

1. Для чего предназначена система Qt?
2. Перечислите модули Qt.
3. Что такое механизм сигналов и слотов?

#### **Тема 2.**

1. Что такое системы быстрого прототипирования?
2. Перечислите модули Qt.
3. Что такое менеджер компоновки?

#### **Тема 3.**

1. Приведите пример ассоциативного контейнера?
2. Какие операции над строками разрешает Qt?
3. Что такое модель общего использования данных?

#### **Тема 4.**

1. Что такое виджет?
2. Перечислите стандартные виджеты Qt?
3. Что делает метод clicked() в классе QAbstractButton?

#### **Тема 5.**

1. Каково назначение класса QLineEdit?
2. Каково назначение класса QTextEdit?
3. Какой класс позволяет пользователю выбрать один или более элементов из списка?

### **8.3. Перечень вопросов и заданий для промежуточной аттестации по дисциплине**

#### **Вопросы для промежуточного контроля (зачета)**

1. Перегрузка операций.
2. Назначение деструктора класса
3. Наследование классов. Открытое, закрытое и защищенное наследование.
4. Виртуальные функции.
5. Чистые виртуальные функции и абстрактные классы.
6. Написать программу, переводящую десятичное число в шестнадцатеричное
6. Написать программу, переводящую десятичное число в восьмеричное
7. Написать программу, переводящую десятичное число в двоичное
8. Найти первые четыре совершенных числа
9. Подсчитать число слов в текстовом файле
10. Подсчитать длину самой большой последовательности 1 во введенном числе
11. Вывести на экран первых 10 счастливых билетика
12. Написать программу, которая подсчитывает периметр и площадь четырехугольника, вписанного в окружность заданного диаметра
13. Даны произвольные числа  $a$ ,  $b$  и  $c$ . Написать программу, которая выводит 0, если нельзя построить треугольник с такими длинами сторон, иначе напечатать 3,2 или 1 в зависимости от того, равносторонний это треугольник, равнобедренный или какой то другой

14. Напечатать в возрастающем порядке все 3-х значные числа, в десятичной записи которых нет одинаковых цифр
15. Реализовать арифметику больших чисел
16. Создать класс BitString, который реализует битовые операции над последовательностями произвольной длины
17. Создать класс Fraction для работы с дробными десятичными числами

#### 8.4. Планируемые уровни сформированности компетенций обучающихся и критерии оценивания

Уровни	Содержательное описание уровня	Основные признаки выделения уровня (этапы формирования компетенции, критерии оценки сформированности)	Пятибалльная шкала (академическая) оценка	Двухбалльная шкала, зачет	БРС, % освоения (рейтинговая оценка)
Повышенный	Творческая деятельность	<i>Включает нижестоящий уровень.</i> Умение самостоятельно принимать решение, решать проблему/задачу теоретического и прикладного характера на основе изученных методов, приемов, технологий	отлично	зачтено	86-100
Базовый	Применение знаний и умений в более широких контекстах учебной и профессиональной деятельности, нежели по образцу с большей степени самостоятельности и инициативы	<i>Включает нижестоящий уровень.</i> Способность собирать, систематизировать, анализировать и грамотно использовать информацию из самостоятельно найденных теоретических источников и иллюстрировать ими теоретические положения или обосновывать практику применения	хорошо		71-85
Удовлетворительный (достаточный)	Репродуктивная деятельность	Изложение в пределах задач курса теоретически и практически контролируемого материала	удовлетворительно		55-70
Недостаточный	Отсутствие признаков		неудовлетв	не	Менее

ый	удовлетворительного уровня	орительно	зачтено	55
----	----------------------------	-----------	---------	----

## 9. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины.

### Основная литература

- Объектно-ориентированное программирование на C++ : учебник / И. В. Баранова, С. Н. Баранов, И. В. Баженова [и др.]. - Красноярск : Сиб. федер. ун-т, 2019. - 288 с. - ISBN 978-5-7638-4034-6. - Текст : электронный. - URL: <https://znanium.com/catalog/product/1819676> (дата обращения: 19.03.2023). – Режим доступа: по подписке.
- Кузин, А. В. Программирование на языке Си : учебное пособие / А. В. Кузин, Е. В. Чумакова. — Москва : ФОРУМ : ИНФРА-М, 2019. — 144 с. — (Высшее образование). - ISBN 978-5-00091-066-5. - Текст : электронный. - URL: <https://znanium.com/catalog/product/1007488> (дата обращения: 03.04.2023). – Режим доступа: по подписке.
- Литвиненко, В. А. Программирование на C++ задач на графах: Учебное пособие / Литвиненко В.А. - Таганрог:Южный федеральный университет, 2016. - 83 с.: ISBN 978-5-9275-2311-5. - Текст : электронный. - URL: <https://znanium.com/catalog/product/997083> (дата обращения: 03.04.2023). – Режим доступа: по подписке.

## 10. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины (модуля).

- НЭБ Национальная электронная библиотека, диссертации и прочие издания
- eLIBRARY.RU Научная электронная библиотека, книги, статьи, тезисы докладов конференций
- Гребенников Электронная библиотека ИД журналы
- ЭБС Консультант студента
- ПРОСПЕКТ ЭБС
- ЭБС ZNANIUM.COM
- ЭБС IBOOKS.RU
- РГБ Информационное обслуживание по МБА
- БЕН РАН
- Электронно-библиотечная система (ЭБС) Кантиана (<https://elib.kantiana.ru/>)

## 11. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине.

Программное обеспечение обучения включает в себя:

- система электронного образовательного контента БФУ им. И. Канта – [www.lms.kantiana.ru](http://www.lms.kantiana.ru), обеспечивающую разработку и комплексное использование электронных образовательных ресурсов;
- серверное программное обеспечение, необходимое для функционирования сервера и связи с системой электронного обучения через Интернет;
- корпоративная платформа;

- установленное на рабочих местах обучающихся ПО: Microsoft Windows 10, Microsoft Office Standart 2016, антивирусное программное обеспечение Kaspersky Endpoint Security.
- Среда программирования Microsoft Visual Studio (любая версия);
- Qt версии 5.0 и выше

## **12. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине.**

Для проведения занятий лекционного типа, практических и семинарских занятий используются специальные помещения (учебные аудитории), оборудованные техническими средствами обучения – мультимедийной проекционной техникой. Для проведения занятий лекционного типа используются наборы демонстрационного оборудования.

Для проведения лабораторных работ, (практических занятий – при необходимости) используются специальные помещения (учебные аудитории), оснащенные специализированным лабораторным оборудованием: персональными компьютерами с возможностью выхода в интернет и с установленным программным обеспечением, заявленным в п.11.

Для проведения групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации используются специальные помещения (учебные аудитории), оборудованные специализированной мебелью (для обучающихся), меловой / маркерной доской.

Для организации самостоятельной работы обучающимся предоставляются помещения, оснащенные компьютерной техникой с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду университета.

Для обучения инвалидов и лиц с ограниченными возможностями здоровья университетом могут быть представлены специализированные средства обучения, в том числе технические средства коллективного и индивидуального пользования

### **4.3 Программа дисциплины «Операционные системы»**

#### **1.Наименование дисциплины: «Операционные системы».**

**Цель** дисциплины: целью освоения дисциплины «Операционные системы» обучение студентов методам функционирования и разработки операционных систем, а также применения основных алгоритмов, в том числе — параллельных.

#### **2. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы**

Код компетенции	Результаты освоения образовательной программы (ИДК)	Результаты обучения по дисциплине
ОПК-5. Способен устанавливать и	ОПК-5.1. Инсталлирует программное обеспечение	<b>Знать:</b> основные характеристики архитектур и системной

<p>сопровождать программное обеспечение информационных систем и баз данных, в том числе отечественного происхождения, с учетом информационной безопасности</p>	<p>информационных систем и баз данных, в том числе отечественного происхождения, с учетом информационной безопасности</p> <p>ОПК-5.2. Сопровождает программное обеспечение информационных систем и баз данных, в том числе отечественного происхождения, с учетом информационной безопасности</p>	<p>организации вычислительной системы, компьютеров и операционных систем и взаимосвязь их основных компонентов;</p> <p>основные системные задачи и проблемы, решаемые в рамках операционных систем;</p> <p>типовую структуру операционной системы, задачи и основные характеристики функциональных модулей, составляющих операционную систему;</p> <p>основные понятия, алгоритмы и методы организации управления процессами в операционных системах;</p> <p>основные понятия, алгоритмы и методы организации взаимодействия процессов;</p> <p>основные понятия, алгоритмы и методы организации файловых систем;</p> <p>основные понятия, алгоритмы и методы организации планирования в операционных системах;</p> <p>основные понятия, алгоритмы и методы организации управления внешними устройствами;</p> <p>основные понятия, алгоритмы и методы организации управления оперативной памятью.</p> <p><b>Уметь:</b> формировать обоснованную оценку организации и функционирования тех или иных компонентов операционных систем в контексте их системной взаимосвязи с аппаратурой компьютера;</p> <p>использовать современные операционные системы;</p> <p>разрабатывать элементы распределенных компонентов системного программного обеспечения, основанных на использовании библиотек системных вызовов;</p> <p>использовать современные языки разработки системного программного обеспечения (язык Си);</p> <p>создавать алгоритмические</p>
--	---	--

		<p>модели типовых задач, проводить спецификацию задачи, реализовывать программы на алгоритмических языках высокого уровня, оценивать сложность полученных алгоритмов.</p> <p><b>Владеть:</b> опытом решения практических задач, связанных с разработкой программного обеспечения на основе использования библиотек системных вызовов и системных библиотек программ; разработки компонентов программного обеспечения в среде ОС UNIX с использованием языка программирования Си; разработки параллельных программ на основе использования различных средств взаимодействия процессов ОС UNIX (базовые средства взаимодействия процессов ОС UNIX, IPC и др.).</p>
--	--	--

### 3. Место дисциплины в структуре образовательной программы

«Операционные системы» представляет собой дисциплину обязательной части (Б1.О.13) направления подготовки бакалавриата 02.03.02 «Фундаментальная информатика и информационные технологии», профиль «Программная инженерия в искусственном интеллекте».

### 4. Виды учебной работы по дисциплине.

Виды учебной работы по дисциплине зафиксированы учебным планом основной профессиональной образовательной программы по указанному направлению и профилю, выражаются в академических часах. Часы контактной работы и самостоятельной работы обучающегося и часы, отводимые на процедуры контроля, могут различаться в учебных планах ОПОП по формам обучения. Объем контактной работы включает часы контактной аудиторной работы (лекции/практические занятия/ лабораторные работы), контактной внеаудиторной работы (контроль самостоятельной работы), часы контактной работы в период аттестации. Контактная работа, в том числе может проводиться посредством электронной информационно-образовательной среды университета с использованием ресурсов сети Интернет и дистанционных технологий

### 5. Содержание дисциплины, структурированное по темам (разделам)



Исходя из рамок, установленных учебным планом по трудоемкости и видам учебной работы по дисциплине, преподаватель самостоятельно выбирает тематику занятий по формам и количеству часов проведения контактной работы: лекции и иные учебные занятия, предусматривающие преимущественную передачу учебной информации преподавателем и (или) занятия семинарского типа (семинары, практические занятия, практикумы, лабораторные работы, коллоквиумы и иные аналогичные занятия), и (или) групповые консультации, и (или) индивидуальную работу обучающихся с преподавателем, в том числе индивидуальные консультации (по курсовым работам/проектам – при наличии курсовой работы/проекта по данной дисциплине в учебном плане). Рекомендуемая тематика занятий максимально полно реализуется в контактной работе с обучающимися очной формы обучения. В случае реализации образовательной программы в заочной / очно-заочной форме трудоемкость дисциплины сохраняется, однако объем учебного материала в значительной части осваивается обучающимися в форме самостоятельной работы. При этом требования к ожидаемым образовательным результатам обучающихся по данной дисциплине не зависят от формы реализации образовательной программы.

№ п/п	Наименование разделов (тем) дисциплины	Содержание разделов (тем) дисциплин
•	Тема 1. Этапы развития вычислительной техники и программного обеспечения.	Этапы развития вычислительной техники и программного обеспечения.
•	Тема 2. Основы архитектуры вычислительной системы.	Структура вычислительной системы. Ресурсы ВС - физические ресурсы, виртуальные ресурсы. Уровень операционной системы. Структура вычислительной системы. Ресурсы ВС - физические, виртуальные. Уровень систем программирования. Структура вычислительной системы. Ресурсы ВС - физические ресурсы, виртуальные ресурсы. Уровень прикладных системы.
•	Тема 3. Основы архитектуры компьютеров.	Основы архитектуры компьютера. Основные компоненты и характеристики. Структура и функционирование ЦП. Основы архитектуры компьютера. Основные компоненты и характеристики. Оперативное запоминающее устройство. Расслоение памяти. Основы архитектуры компьютера. Основные компоненты и характеристики. Кэширование ОЗУ. Основы архитектуры компьютера. Аппарат прерываний. Последовательность действий в вычислительной системе при обработке прерываний. Основы архитектуры компьютера. Внешние устройства. Организация управления и потоков данных при обмене с внешними устройствами. Основы архитектуры компьютера. Иерархия памяти.

•	Тема 4. Основы архитектуры операционных систем.	Аппаратная поддержка ОС. Мультипрограммный режим. Аппаратная поддержка ОС и систем программирования.. Организация регистровой памяти ЦП (регистровые окна, стек). Аппаратная поддержка ОС. Виртуальная оперативная память. Аппаратная поддержка ОС. Пример организации страничной виртуальной памяти.
•	Тема 5. Управление процессами. Основные концепции.	Управление процессами. Определение процесса, типы. Жизненный цикл, состояния процесса. Свопинг. Модели жизненного цикла процесса. Контекст процесса.
•	Тема 6. Управление процессами. Реализация процессов в ОС UNIX.	Реализация процессов в ОС UNIX. Определение процесса. Контекст, тело процесса. Состояния процесса. Аппарат системных вызовов в ОС UNIX. Реализация процессов в ОС UNIX. Базовые средства управления процессами в ОС UNIX. Загрузка ОС UNIX, формирование нулевого и первого процессов.
•	Тема 7. Управление процессами. Планирование.	Процессы и потоки: основные понятия. Процессоры: компоненты и понятие архитектуры. Процессоры: параллелизм на уровне инструкций. Планировщик процессов и алгоритмы планирования: определение и задачи. Алгоритмы планирования в пакетных системах. Алгоритмы планирования в интерактивных системах. Алгоритмы планирования в системах реального времени. Алгоритмы планирования в распределенных системах.
•	Тема 8. Управление процессами. Взаимодействие процессов.	Взаимодействие процессов. Разделяемые ресурсы. Критические секции. Взаимное исключение. Тупики. Взаимодействие процессов. Некоторые способы реализации взаимного исключения: семафоры Дейкстры, мониторы, обмен сообщениями. Взаимодействие процессов. Классические задачи синхронизации процессов. “Обедающие философы”. Взаимодействие процессов. Классические задачи синхронизации процессов. “Читатели и писатели”.
•	Тема 9. Реализация межпроцессного взаимодействия в ОС UNIX. Базовые средства реализации взаимодействия процессов в ОС UNIX.	Базовые средства взаимодействия процессов в ОС UNIX. Сигналы. Примеры программирования. Базовые средства взаимодействия процессов в ОС UNIX. Неименованные каналы. Примеры программирования . Базовые средства взаимодействия процессов в ОС UNIX. Именованные каналы. Примеры программирования. Базовые средства взаимодействия процессов в ОС UNIX. Взаимодействие процессов по схеме ”подчиненный-главный”. Общая схема трассировки процессов.

•	Тема 10. Реализация межпроцессного взаимодействия в ОС UNIX. IPC – система межпроцессного взаимодействия.	Система межпроцессного взаимодействия ОС UNIX. Именованые разделяемых объектов. Очереди сообщений. Пример. Система межпроцессного взаимодействия ОС UNIX . Именованые разделяемых объектов. Разделяемая память. Пример. Система межпроцессного взаимодействия ОС UNIX . Именованые разделяемых объектов. Массив семафоров. Пример.
•	Тема 11. Реализация межпроцессного взаимодействия в ОС UNIX. Сокеты – унифицированный интерфейс программирования распределенных систем.	Сокеты. Типы сокетов. Коммуникационный домен. Схема работы с сокетами с установлением соединения. Сокеты. Схема работы с сокетами без установления соединения. Общая классификация средств взаимодействия процессов в ОС UNIX
•	Тема 12. Файловые системы. Основные концепции.	Файловые системы. Структурная организация файлов. Атрибуты файлов. Основные правила работы с файлами. Типовые программные интерфейсы работы с файлами. Файловые системы. Модели реализации файловых систем. Понятие индексного дескриптора. Файловые системы. Координация использования пространства внешней памяти. Квотирование пространства ФС. Надежность ФС. Проверка целостности ФС.
•	Тема 13. Файловые системы. Примеры реализаций файловых систем.	Примеры реализаций файловых систем. Организация файловой системы ОС UNIX. Виды файлов. Права доступа. Логическая структура каталогов. Примеры реализаций файловых систем Внутренняя организация ФС. Модель версии UNIX SYSTEM V. Примеры реализаций файловых систем. Внутренняя организация ФС. Принципы организации файловой системы FFS UNIX BSD.
•	Тема 14. Управление памятью. Базовые концепции, задачи и стратегии управления оперативной памятью.	Управление оперативной памятью. Одиночное непрерывное распределение. Распределение разделами. Распределение перемещаемыми разделами. Управление оперативной памятью. Страничное распределение. Управление оперативной памятью. Сегментное распределение. Вычислительная система. Кэширование информационных потоков на уровнях аппаратуры и ОС.
•	Тема 15. Управление внешними устройствами. Общие концепции.	Управление внешними устройствами. Архитектура организации управления внешними устройствами, основные подходы, характеристики. Управление внешними устройствами. Планирование дисковых обменов, основные алгоритмы.

		Управление внешними устройствами. Организация RAID систем, основные решения, характеристики.
•	Тема 16. Управление внешними устройствами. ОС UNIX – работа с внешними устройствами.	Внешние устройства в ОС UNIX. Типы устройств, файлы устройств, драйверы. Внешние устройства в ОС UNIX. Системная организация обмена с файлами. Буферизация обменов с блокоориентированными устройствами.

### **6. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине**

Рекомендуемая тематика учебных занятий в форме контактной работы

**Рекомендуемая тематика учебных занятий лекционного типа**  
(предусматривающих преимущественную передачу учебной информации преподавателями):

№ п/п	Наименование разделов (тем) дисциплины	Содержание разделов (тем) дисциплин
•	Тема 1. Этапы развития вычислительной техники и программного обеспечения.	Этапы развития вычислительной техники и программного обеспечения.
•	Тема 2. Основы архитектуры вычислительной системы.	Структура вычислительной системы. Ресурсы ВС - физические ресурсы, виртуальные ресурсы. Уровень операционной системы. Структура вычислительной системы. Ресурсы ВС - физические, виртуальные. Уровень систем программирования. Структура вычислительной системы. Ресурсы ВС - физические ресурсы, виртуальные ресурсы. Уровень прикладных системы.
•	Тема 3. Основы архитектуры компьютеров.	Основы архитектуры компьютера. Основные компоненты и характеристики. Структура и функционирование ЦП. Основы архитектуры компьютера. Основные компоненты и характеристики. Оперативное запоминающее устройство. Расслоение памяти. Основы архитектуры компьютера. Основные компоненты и характеристики. Кэширование ОЗУ. Основы архитектуры компьютера. Аппарат прерываний. Последовательность действий в вычислительной системе при обработке прерываний. Основы архитектуры компьютера. Внешние устройства. Организация управления и потоков данных при обмене с внешними устройствами. Основы архитектуры компьютера. Иерархия памяти.
•	Тема 4. Основы архитектуры операционных систем.	Аппаратная поддержка ОС. Мультипрограммный режим. Аппаратная поддержка ОС и систем

		<p>программирования.. Организация регистровой памяти ЦП (регистровые окна, стек).  Аппаратная поддержка ОС. Виртуальная оперативная память.  Аппаратная поддержка ОС. Пример организации страничной виртуальной памяти.</p>
•	Тема 5. Управление процессами. Основные концепции.	<p>Управление процессами. Определение процесса, типы. Жизненный цикл, состояния процесса. Свопинг. Модели жизненного цикла процесса. Контекст процесса.</p>
•	Тема 6. Управление процессами. Реализация процессов в ОС UNIX.	<p>Реализация процессов в ОС UNIX. Определение процесса. Контекст, тело процесса. Состояния процесса. Аппарат системных вызовов в ОС UNIX. Реализация процессов в ОС UNIX. Базовые средства управления процессами в ОС UNIX. Загрузка ОС UNIX, формирование нулевого и первого процессов.</p>
•	Тема 7. Управление процессами. Планирование.	<p>Процессы и потоки: основные понятия.  Процессоры: компоненты и понятие архитектуры.  Процессоры: параллелизм на уровне инструкций.  Планировщик процессов и алгоритмы планирования: определение и задачи.  Алгоритмы планирования в пакетных системах.  Алгоритмы планирования в интерактивных системах.  Алгоритмы планирования в системах реального времени.  Алгоритмы планирования в распределенных системах.</p>
•	Тема 8. Управление процессами. Взаимодействие процессов.	<p>Взаимодействие процессов. Разделяемые ресурсы. Критические секции. Взаимное исключение. Тупики.  Взаимодействие процессов. Некоторые способы реализации взаимного исключения: семафоры Дейкстры, мониторы, обмен сообщениями.  Взаимодействие процессов. Классические задачи синхронизации процессов. “Обедающие философы”.  Взаимодействие процессов. Классические задачи синхронизации процессов. “Читатели и писатели”.</p>
•	Тема 9. Реализация межпроцессного взаимодействия в ОС UNIX. Базовые средства реализации взаимодействия процессов в ОС UNIX.	<p>Базовые средства взаимодействия процессов в ОС UNIX. Сигналы. Примеры программирования.  Базовые средства взаимодействия процессов в ОС UNIX. Неименованные каналы. Примеры программирования .  Базовые средства взаимодействия процессов в ОС UNIX. Именованные каналы. Примеры программирования.  Базовые средства взаимодействия процессов в ОС UNIX. Взаимодействие процессов по схеме ”подчиненный-главный”. Общая схема трассировки процессов.</p>
•	Тема 10. Реализация межпроцессного взаимодействия в ОС UNIX. IPC – система	<p>Система межпроцессного взаимодействия ОС UNIX. Именование разделяемых объектов. Очереди сообщений. Пример.</p>

	межпроцессного взаимодействия.	Система межпроцессного взаимодействия ОС UNIX . Именованние разделяемых объектов. Разделяемая память. Пример. Система межпроцессного взаимодействия ОС UNIX . Именованние разделяемых объектов. Массив семафоров. Пример.
•	Тема 11. Реализация межпроцессного взаимодействия в ОС UNIX. Сокеты – унифицированный интерфейс программирования распределенных систем.	Сокеты. Типы сокетов. Коммуникационный домен. Схема работы с сокетами с установлением соединения. Сокеты. Схема работы с сокетами без установления соединения. Общая классификация средств взаимодействия процессов в ОС UNIX
•	Тема 12. Файловые системы. Основные концепции.	Файловые системы. Структурная организация файлов. Атрибуты файлов. Основные правила работы с файлами. Типовые программные интерфейсы работы с файлами. Файловые системы. Модели реализации файловых систем. Понятие индексного дескриптора. Файловые системы. Координация использования пространства внешней памяти. Квотирование пространства ФС. Надежность ФС. Проверка целостности ФС.
•	Тема 13. Файловые системы. Примеры реализаций файловых систем.	Примеры реализаций файловых систем. Организация файловой системы ОС UNIX. Виды файлов. Права доступа. Логическая структура каталогов. Примеры реализаций файловых систем Внутренняя организация ФС. Модель версии UNIX SYSTEM V. Примеры реализаций файловых систем. Внутренняя организация ФС. Принципы организации файловой системы FFS UNIX BSD.
•	Тема 14. Управление памятью. Базовые концепции, задачи и стратегии управления оперативной памятью.	Управление оперативной памятью. Одиночное непрерывное распределение. Распределение разделами. Распределение перемещаемыми разделами. Управление оперативной памятью. Страничное распределение. Управление оперативной памятью. Сегментное распределение. Вычислительная система. Кэширование информационных потоков на уровнях аппаратуры и ОС.
•	Тема 15. Управление внешними устройствами. Общие концепции.	Управление внешними устройствами. Архитектура организации управления внешними устройствами, основные подходы, характеристики. Управление внешними устройствами. Планирование дисковых обменов, основные алгоритмы. Управление внешними устройствами. Организация RAID систем, основные решения, характеристики.
•	Тема 16. Управление внешними	Внешние устройства в ОС UNIX. Типы устройств,

	устройствами. ОС UNIX – работа с внешними устройствами.	файлы устройств, драйверы. Внешние устройства в ОС UNIX. Системная организация обмена с файлами. Буферизация обменов с блокоориентированными устройствами.
--	--	---

### **Рекомендуемая тематика *практических* занятий:**

Тематика практических занятий совпадает с темами лекционных занятий.

На практических занятиях решаются задачи по теме занятия.

### **Требования к самостоятельной работе обучающихся**

1. Работа с лекционным материалом, предусматривающая проработку конспекта лекций и учебной литературы, по всем темам из п. 6 настоящей рабочей программы.

2. Выполнение домашнего задания, предусматривающего решение задач, выполнение упражнений, выдаваемых на лабораторных занятиях, по всем темам из п. 6 настоящей рабочей программы.

Руководствуясь положениями статьи 47 и статьи 48 Федерального закона от 29 декабря 2012 г. N 273-ФЗ «Об образовании в Российской Федерации» научно-педагогические работники и иные лица, привлекаемые университетом к реализации данной образовательной программы, пользуются предоставленными академическими правами и свободами в части свободы преподавания, свободы от вмешательства в профессиональную деятельность; свободы выбора и использования педагогически обоснованных форм, средств, методов обучения и воспитания; права на творческую инициативу, разработку и применение авторских программ и методов обучения и воспитания в пределах реализуемой образовательной программы и отдельной дисциплины.

Исходя из рамок, установленных учебным планом по трудоемкости и видам учебной работы по дисциплине, преподаватель самостоятельно выбирает тематику занятий по формам и количеству часов проведения контактной работы: лекции и иные учебные занятия, предусматривающие преимущественную передачу учебной информации преподавателем и (или) занятия семинарского типа (семинары, практические занятия, практикумы, лабораторные работы, коллоквиумы и иные аналогичные занятия), и (или) групповые консультации, и (или) индивидуальную работу обучающихся с преподавателем, в том числе индивидуальные консультации (по курсовым работам/проектам – при наличии курсовой работы/проекта по данной дисциплине в учебном плане).

Рекомендуемая тематика занятий максимально полно реализуется в контактной работе с обучающимися очной формы обучения. В случае реализации образовательной программы в заочной / очно-заочной форме трудоемкость дисциплины сохраняется, однако объем учебного материала в значительной части осваивается обучающимися в форме самостоятельной работы. При этом требования к ожидаемым образовательным результатам обучающихся по данной дисциплине не зависят от формы реализации образовательной программы.

## 7. Методические рекомендации по видам занятий

Лекционные занятия.

В ходе лекционных занятий обучающимся рекомендуется выполнять следующие действия. Вести конспектирование учебного материала. Обращать внимание на категории, формулировки, раскрывающие содержание тех или иных явлений и процессов, научные выводы и практические рекомендации по их применению. Задавать преподавателю уточняющие вопросы с целью уяснения теоретических положений, разрешения спорных ситуаций.

Желательно оставить в рабочих конспектах поля, на которых во внеаудиторное время можно сделать пометки из рекомендованной литературы, дополняющие материал прослушанной лекции, а также подчеркивающие особую важность тех или иных теоретических положений.

Лабораторные занятия.

На лабораторных занятиях в зависимости от темы занятия выполняется поиск информации по решению проблем, практические упражнения, контрольные работы, выработка индивидуальных или групповых решений, итоговое обсуждение с обменом знаниями, участие в круглых столах, разбор конкретных ситуаций, командная работа, представление портфолио и т.п.

Самостоятельная работа.

Самостоятельная работа осуществляется в виде изучения литературы, эмпирических данных по публикациям и конкретных ситуаций из практики, подготовке индивидуальных работ, работа с лекционным материалом, самостоятельное изучение отдельных тем дисциплины; поиск и обзор литературы и электронных источников; чтение и изучение учебника и учебных пособий.

## 8. Фонд оценочных средств

### 8.1. Перечень компетенций с указанием этапов их формирования в процессе освоения образовательной программы в рамках учебной дисциплины

Основными этапами формирования указанных компетенций при изучении обучающимися дисциплины являются последовательное изучение содержательно связанных между собой тем учебных занятий. Изучение каждой темы предполагает овладение обучающимися необходимыми компетенциями. Результат аттестации обучающихся на различных этапах формирования компетенций показывает уровень освоения компетенций.

Контролируемые разделы (темы) дисциплины	Индекс контролируемой компетенции (или её части)	Оценочные средства по этапам формирования компетенций
		текущий контроль по дисциплине
Тема 1. Этапы развития вычислительной техники и программного обеспечения	ОПК-5	коллоквиум
Тема 2. Основы архитектуры	ОПК-5	коллоквиум



вычислительной системы.		
Тема 3. Основы архитектуры компьютеров.	ОПК-5	коллоквиум
Тема 4. Основы архитектуры операционных систем.	ОПК-5	коллоквиум
Тема 5. Управление процессами. Основные концепции.	ОПК-5	коллоквиум
Тема 6. Управление процессами. Реализация процессов в ОС UNIX	ОПК-5	коллоквиум
Тема 7. Управление процессами. Планирование.	ОПК-5	коллоквиум
Тема 8. Управление процессами. Взаимодействие процессов.	ОПК-5	коллоквиум
Тема 9. Реализация межпроцессного взаимодействия в ОС UNIX. Базовые средства реализации взаимодействия процессов в ОС UNIX	ОПК-5	коллоквиум
Тема 10. Реализация межпроцессного взаимодействия в ОС UNIX. IPC – система межпроцессного взаимодействия.	ОПК-5	коллоквиум
Тема 11. Реализация межпроцессного взаимодействия в ОС UNIX. Сокеты – унифицированный интерфейс программирования распределенных систем	ОПК-5	коллоквиум
Тема 12. Файловые системы. Основные концепции.	ОПК-5	коллоквиум
Тема 13. Файловые системы. Примеры реализаций файловых систем	ОПК-5	коллоквиум
Тема 14. Управление памятью. Базовые концепции, задачи и стратегии управления оперативной памятью.	ОПК-5	коллоквиум
Тема 15. Управление внешними устройствами. Общие концепции	ОПК-5	коллоквиум
Тема 16. Управление внешними устройствами. ОС UNIX – работа с внешними устройствами	ОПК-5	коллоквиум

## 8.2. Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений и навыков и (или) опыта деятельности процессе текущего контроля

### Пример вопросов коллоквиума.

#### Вариант 4. 10 вопросов.

- Определение виртуального ресурса (устройства).

- Минимальные требования к аппаратуре для обеспечения корректного мультипрограммирования.
- Дать определение понятия «аппарат» виртуальной памяти.
- Основное преимущество протокола UDP по сравнению с протоколом TCP?
- Сформулировать общее (не являющееся определением процессов в Unix) определение процесса.
- Указать основное преимущество использование битовых массивов для учета свободных блоков файловой системы.
- Дать определение семафора Дейкстры.
- Для доступа взаимодействующих процессов к разделяемому ресурсу  $R$  используется семафор Дейкстры  $S$ . На входе в критические секции ресурса  $R$  выполняется операция  $Down(S)$ , на выходе  $Up(S)$ .  $S$  имеет начальное значение  $N$ . Какую модель доступа к ресурсу  $R$  демонстрирует этот пример?
- Имеется программная система, реализующая с использованием сокетов модель клиент-сервер. К серверу п Написать программу на Си, выводящую на стандартное устройство вывода текст командной строки, посредством которой данная отключено  $K$  клиентов. Какое количество сокетов создано на сервере в этот момент времен?
- Написать программу на Си, выводящую на стандартное устройство вывода текст командной строки, посредством которой данная программа была запущена.

### 8.3. Перечень вопросов и заданий для промежуточной аттестации по дисциплине

#### Типовые задания для проведения промежуточной (экзамен) аттестации

##### Вопросы к экзамену

1. Этапы развития вычислительной техники и программного обеспечения.
2. Структура вычислительной системы. Ресурсы ВС - физические ресурсы, виртуальные ресурсы. Уровень операционной системы.
3. Структура вычислительной системы. Ресурсы ВС - физические, виртуальные. Уровень систем программирования.
4. Структура вычислительной системы. Ресурсы ВС - физические ресурсы, виртуальные ресурсы. Уровень прикладных системы.
5. Структура вычислительной системы. Понятие виртуальной машины.
6. Основы архитектуры компьютера. Основные компоненты и характеристики. Структура и функционирование ЦП.
7. Основы архитектуры компьютера. Основные компоненты и характеристики. Оперативное запоминающее устройство. Расслоение памяти.
8. Основы архитектуры компьютера. Основные компоненты и характеристики. Кэширование ОЗУ.
9. Основы архитектуры компьютера. Аппарат прерываний. Последовательность действий в вычислительной системе при обработке прерываний.
10. Основы архитектуры компьютера. Внешние устройства. Организация управления и потоков данных при обмене с внешними устройствами.
11. Основы архитектуры компьютера. Иерархия памяти.

12. Аппаратная поддержка ОС. Мультипрограммный режим.
13. Аппаратная поддержка ОС и систем программирования.. Организация регистровой памяти ЦП (регистровые окна, стек).
14. Аппаратная поддержка ОС. Виртуальная оперативная память.
15. Аппаратная поддержка ОС. Пример организации страничной виртуальной памяти.
16. Многомашинные, многопроцессорные ассоциации. Классификация. Примеры.
17. Многомашинные, многопроцессорные ассоциации. Терминальные комплексы. Компьютерные сети.
18. Операционные системы. Основные компоненты и логические функции. Базовые понятия: ядро, процесс, ресурс, системные вызовы. Структурная организация ОС.
19. Операционные системы. Пакетная ОС, ОС разделения времени, ОС реального времени, распределенные и сетевые ОС.
20. Организация сетевого взаимодействия. Эталонная модель ISO/OSI. Протокол, интерфейс. Стек протоколов. Логическое взаимодействие сетевых устройств.
21. Организация сетевого взаимодействия. Семейство протоколов TCP/IP, соответствие модели ISO/OSI. Взаимодействие между уровнями протоколов семейства TCP/IP. IP адресация.
22. Управление процессами. Определение процесса, типы. Жизненный цикл, состояния процесса. Свопинг. Модели жизненного цикла процесса. Контекст процесса.
23. Реализация процессов в ОС UNIX. Определение процесса. Контекст, тело процесса. Состояния процесса. Аппарат системных вызовов в ОС UNIX.
24. Реализация процессов в ОС UNIX. Базовые средства управления процессами в ОС UNIX. Загрузка ОС UNIX, формирование нулевого и первого процессов.
25. Взаимодействие процессов. Разделяемые ресурсы. Критические секции. Взаимное исключение. Тупики.
26. Взаимодействие процессов. Некоторые способы реализации взаимного исключения: семафоры Дейкстры, мониторы, обмен сообщениями.
27. Взаимодействие процессов. Классические задачи синхронизации процессов. “Обедающие философы”.
28. Взаимодействие процессов. Классические задачи синхронизации процессов. “Читатели и писатели”.
29. Базовые средства взаимодействия процессов в ОС UNIX. Сигналы. Примеры программирования.
30. Базовые средства взаимодействия процессов в ОС UNIX. Неименованные каналы. Примеры программирования .
31. Базовые средства взаимодействия процессов в ОС UNIX. Именованные каналы. Примеры программирования.
32. Базовые средства взаимодействия процессов в ОС UNIX. Взаимодействие процессов по схеме ”подчиненный-главный”. Общая схема трассировки процессов.
33. Система межпроцессного взаимодействия ОС UNIX. Именование разделяемых объектов. Очереди сообщений. Пример.
34. Система межпроцессного взаимодействия ОС UNIX . Именование разделяемых объектов. Разделяемая память. Пример.
35. Система межпроцессного взаимодействия ОС UNIX . Именование разделяемых объектов. Массив семафоров. Пример.

36. Сокеты. Типы сокетов. Коммуникационный домен. Схема работы с сокетами с установлением соединения.
37. Сокеты. Схема работы с сокетами без установления соединения.
38. Общая классификация средств взаимодействия процессов в ОС UNIX.
39. Файловые системы. Структурная организация файлов. Атрибуты файлов. Основные правила работы с файлами. Типовые программные интерфейсы работы с файлами.
40. Файловые системы. Модели реализации файловых систем. Понятие индексного дескриптора.
41. Файловые системы. Координация использования пространства внешней памяти. Квотирование пространства ФС. Надежность ФС. Проверка целостности ФС.
42. Примеры реализаций файловых систем. Организация файловой системы ОС UNIX. Виды файлов. Права доступа. Логическая структура каталогов.
43. Примеры реализаций файловых систем. Внутренняя организация ФС. Модель версии UNIX SYSTEM V.
44. Примеры реализаций файловых систем. Внутренняя организация ФС. Принципы организации файловой системы FFS UNIX BSD.
45. Управление внешними устройствами. Архитектура организации управления внешними устройствами, основные подходы, характеристики.
46. Управление внешними устройствами. Планирование дисковых обменов, основные алгоритмы.
47. Управление внешними устройствами. Организация RAID систем, основные решения, характеристики.
48. Внешние устройства в ОС UNIX. Типы устройств, файлы устройств, драйверы.
49. Внешние устройства в ОС UNIX. Системная организация обмена с файлами. Буферизация обменов с блокоориентированными устройствами.
50. Управление оперативной памятью. Одиночное непрерывное распределение. Распределение разделами. Распределение перемещаемыми разделами.
51. Управление оперативной памятью. Страничное распределение.
52. Управление оперативной памятью. Сегментное распределение.
53. Вычислительная система. Кэширование информационных потоков на уровнях аппаратуры и ОС.
54. Язык программирования C. Общая характеристика. Типы, данные, классы памяти. Правила видимости. Структура программы. Препроцессор. Интерфейс с ОС UNIX.

### **Пример экзаменационного билета**

#### **Билет**

1. Структура вычислительной системы. Понятие виртуальной машины.
2. Система межпроцессного взаимодействия ОС UNIX. Именованные разделяемые объекты. Очереди сообщений. Пример.
3. Управление внешними устройствами. Организация RAID систем, основные решения, характеристики.

#### 8.4. Планируемые уровни сформированности компетенций обучающихся и критерии оценивания

Уровни	Содержательное описание уровня	Основные признаки выделения уровня (этапы формирования компетенции, критерии оценки сформированности)	Пятибалльная шкала (академическая) оценка	Двухбалльная шкала, зачет	БРС, % освоения (рейтинговая оценка)
Повышенный	Творческая деятельность	<i>Включает нижестоящий уровень.</i>  Умение самостоятельно принимать решение, решать проблему/задачу теоретического и прикладного характера на основе изученных методов, приемов, технологий	отлично	зачтено	86-100
Базовый	Применение знаний и умений в более широких контекстах учебной и профессиональной деятельности, нежели по образцу с большей степени самостоятельности и инициативы	<i>Включает нижестоящий уровень.</i> Способность собирать, систематизировать, анализировать и грамотно использовать информацию из самостоятельно найденных теоретических источников и иллюстрировать ими теоретические положения или обосновывать практику применения	хорошо		71-85
Удовлетворительный (достаточный)	Репродуктивная деятельность	Изложение в пределах задач курса теоретически и практически контролируемого	удовлетворительно		55-70

		материала			
Недостаточный	Отсутствие признаков удовлетворительного уровня		неудовлетворительно	не зачтено	Менее 55

## **9. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины.**

### **Основная литература**

1. Партыка, Т. Л. Операционные системы, среды и оболочки : учебное пособие / Т.Л. Партыка, И.И. Попов. — 5-е изд., перераб. и доп. — Москва : ФОРУМ : ИНФРА-М, 2021. — 560 с. — (Среднее профессиональное образование). - ISBN 978-5-00091-501-1. - Текст : электронный. - URL: <https://znanium.com/catalog/product/1189335> (дата обращения: 20.11.2023). – Режим доступа: по подписке.
2. Буза, М. К. Архитектура компьютеров: Учебник / Буза М.К. - Мн.:Вышэйшая школа, 2015. - 414 с.: ISBN 978-985-06-2652-3. - Текст : электронный. - URL: <https://znanium.com/catalog/product/1011033> (дата обращения: 20.11.2023). – Режим доступа: по подписке.
3. Урбанович, П. П. Компьютерные сети : учебное пособие / П. П. Урбанович, Д. М. Романенко. - Москва ; Вологда : Инфра-Инженерия, 2022. - 460 с. - ISBN 978-5-9729-0962-9. - Текст : электронный. - URL: <https://znanium.com/catalog/product/1902692> (дата обращения: 20.11.2023). – Режим доступа: по подписке.

### **Дополнительная литература**

- И.В.Машечкин, М.И.Петровский, П.Д.Скулачев, А.Н.Терехин, "Системное программное обеспечение: файловые системы ОС Unix и Windows NT". Москва, Диалог-Москва, 1997г. 47 с.
- Б.Керниган, Д.Ритчи, «Язык программирования Си», Издательский дом «Вильямс», 2 изд. 2006г. 296с.
- А. Робачевский, «Операционная система Unix», 2 изд. ВHV Санкт-Петербург, 2007 г.
- Герасимов С.В., Машечкин И.В., Петровский М.И., Попов И.С., Терехин А.Н.,
- Чернов А.В., «Организация кэширования». М., МГУ, 2011 г. 24 с.
- Герасимов С.В., Машечкин И.В., Петровский М.И., Попов И.С., Терехин А.Н.,
- Чернов А.В., «Инструментальные средства разработки ПО в ОС UNIX». М., МГУ, 2011 г. 66 с.
- Руденко Т.В. Сборник задач и упражнений по языку Си. М., МГУ. 1999 г.
- Попов И.С. Операционные системы: планирование выполнения процессов. М., МГУ. 2015г. 52с.
- Устюгов М.В. Введение в TCP/IP. Под ред. Машечкина И.В., М. МГУ., 1996.
-

## **10. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины (модуля).**

- НЭБ Национальная электронная библиотека, диссертации и прочие издания
- eLIBRARY.RU Научная электронная библиотека, книги, статьи, тезисы докладов конференций
- Гребенников Электронная библиотека ИД журналы
- ЭБС Консультант студента
- ПРОСПЕКТ ЭБС
- ЭБС ZNANIUM.COM
- ЭБС IBOOKS.RU
- РГБ Информационное обслуживание по МБА
- БЕН РАН
- Электронно-библиотечная система (ЭБС) Кантиана (<https://elib.kantiana.ru/>)

## **11. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине.**

Программное обеспечение обучения включает в себя:

- система электронного образовательного контента БФУ им. И. Канта – [www.lms.kantiana.ru](http://www.lms.kantiana.ru), обеспечивающую разработку и комплексное использование электронных образовательных ресурсов;
- серверное программное обеспечение, необходимое для функционирования сервера и связи с системой электронного обучения через Интернет;
- корпоративная платформа;
- установленное на рабочих местах обучающихся ПО: Microsoft Windows 10, Microsoft Office Standart 2010, антивирусное программное обеспечение Kaspersky Endpoint Security.
- специализированное ПО: NetEmul, VirtualBox.

## **12. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине.**

Для проведения занятий лекционного типа, практических и семинарских занятий используются специальные помещения (учебные аудитории), оборудованные техническими средствами обучения – мультимедийной проекционной техникой. Для проведения занятий лекционного типа используются наборы демонстрационного оборудования.

Для проведения лабораторных работ, (практических занятий – при необходимости) используются специальные помещения (учебные аудитории), оснащенные специализированным лабораторным оборудованием: персональными компьютерами с возможностью выхода в интернет и с установленным программным обеспечением, заявленным в п.11.

Для проведения групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации используются специальные помещения (учебные аудитории),

оборудованные специализированной мебелью (для обучающихся), меловой / маркерной доской.

Для организации самостоятельной работы обучающимся предоставляются помещения, оснащенные компьютерной техникой с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду университета.

Для обучения инвалидов и лиц с ограниченными возможностями здоровья университетом могут быть представлены специализированные средства обучения, в том числе технические средства коллективного и индивидуального пользования.

#### 4.4. Программа дисциплины «Алгоритмы и структуры данных»

##### 1. Наименование дисциплины: «Алгоритмы и структуры данных».

**Цель дисциплины:** формирование у обучающихся компетенций, связанных с применением существующих и разработкой новых структур хранения данных, а также с применением эффективных алгоритмов по работе с этими структурами.

##### 2. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы

Код компетенции	Результаты освоения образовательной программы (ИДК)	Результаты обучения по дисциплине
УК-1 Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач	<p>УК.1.1. Выбирает источники информации и осуществляет поиск информации для решения поставленных задач</p> <p>УК. 1.2. Демонстрирует умение рассматривать различные точки зрения и выявлять степень доказательности на поставленную задачу</p> <p>УК.1.3. Определяет рациональные идеи для решения поставленных задач</p>	<p><b>-знать:</b></p> <ul style="list-style-type: none"> <li>- основные принципы метода системного подхода;</li> <li>- базовые структуры данных и алгоритмы их обработки;</li> <li>- современные направления создания новых структур данных и алгоритмов их использования;</li> </ul> <p><b>-уметь:</b></p> <ul style="list-style-type: none"> <li>- применять метод системного подхода при выборе соответствующей структуры данных и алгоритмов её обработки;</li> <li>- анализировать предметную область задачи и предлагать новые структуры данных и алгоритмы их обработки;</li> </ul> <p><b>-владеть:</b> практическими навыками применения метода системного подхода при разработке различных алгоритмов.</p>
ОПК-2. Способен использовать и адаптировать	ОПК-2.1. Имеет представление о существующих базовых	<p><b>-знать:</b></p> <ul style="list-style-type: none"> <li>- фундаментальные компьютерные алгоритмы и структуры данных;</li> <li>- классификацию алгоритмов по</li> </ul>



<p>существующие математические методы и системы программирования для разработки и реализации алгоритмов решения прикладных задач.</p>	<p>математических методах и системах программирования, применяемых для разработки и реализации алгоритмов решения прикладных задач.</p> <p>ОПК-2.2. Применяет математические методы и системы программирования для решения прикладных задач.</p> <p>ОПК-2.3. Разрабатывает и применяет алгоритмические и программные решения прикладных задач.</p>	<p>степени их сложности и по типам используемых структур данных;</p> <ul style="list-style-type: none"> <li>- достоинства и недостатки каждого вида структур данных для применения при решении различных задач</li> </ul> <p><b>-уметь:</b></p> <ul style="list-style-type: none"> <li>- выполнять оценку эффективности алгоритмов и их применимости;</li> <li>- проектировать и реализовывать структуры данных для построения эффективных программных комплексов, используя возможности языков высокого уровня</li> </ul> <p><b>- владеть:</b> практическими навыками использования существующих структур данных и алгоритмов их обработки при разработке программных модулей и компонент, а также при их верификации</p>
---	--	--

### 3. Место дисциплины в структуре образовательной программы

«Алгоритмы и структуры данных» представляет собой дисциплину обязательной части (Б1.О.13) направления подготовки бакалавриата 02.03.02 «Фундаментальная информатика и информационные технологии», профиль «Программная инженерия в искусственном интеллекте».

### 4. Виды учебной работы по дисциплине.

Виды учебной работы по дисциплине зафиксированы учебным планом основной профессиональной образовательной программы по указанному направлению и профилю, выражаются в академических часах. Часы контактной работы и самостоятельной работы обучающегося и часы, отводимые на процедуры контроля, могут различаться в учебных планах ОПОП по формам обучения. Объем контактной работы включает часы контактной аудиторной работы (лекции/практические занятия/ лабораторные работы), контактной внеаудиторной работы (контроль самостоятельной работы), часы контактной работы в период аттестации. Контактная работа, в том числе может проводиться посредством электронной информационно-образовательной среды университета с использованием ресурсов сети Интернет и дистанционных технологий

### 5. Содержание дисциплины, структурированное по темам (разделам)

Исходя из рамок, установленных учебным планом по трудоемкости и видам учебной работы по дисциплине, преподаватель самостоятельно выбирает тематику занятий по формам и количеству часов проведения контактной работы: лекции и иные

учебные занятия, предусматривающие преимущественную передачу учебной информации преподавателем и (или) занятия семинарского типа (семинары, практические занятия, практикумы, лабораторные работы, коллоквиумы и иные аналогичные занятия), и (или) групповые консультации, и (или) индивидуальную работу обучающихся с преподавателем, в том числе индивидуальные консультации (по курсовым работам/проектам – при наличии курсовой работы/проекта по данной дисциплине в учебном плане). Рекомендуемая тематика занятий максимально полно реализуется в контактной работе с обучающимися очной формы обучения. В случае реализации образовательной программы в заочной / очно-заочной форме трудоемкость дисциплины сохраняется, однако объем учебного материала в значительной части осваивается обучающимися в форме самостоятельной работы. При этом требования к ожидаемым образовательным результатам обучающихся по данной дисциплине не зависят от формы реализации образовательной программы.

№	Наименование раздела	Содержание раздела
1	Введение	<p>Предмет дисциплины и ее задачи. Связь с другими дисциплинами учебного плана направления и специальности.</p> <p>Вход и выход алгоритма. Пред- и постусловие. Математическая индукция. Инвариант. Эквивалентность алгоритмов. Представление алгоритмов в командах компьютера и на языке высокого уровня.</p> <p>Временная и емкостная сложности алгоритма. Полиномиальные и экспоненциальные виды функций сложности. Исследование временной сложности в наихудшем и в среднем. Экспериментальное исследование сложности алгоритмов.</p>
2	Структуры данных	<p>Стеки, очереди, списки, их моделирование с помощью массивов. Представления множеств в виде массивов и списков. Графы и различные способы их представления. Таблицы. Выбор наиболее эффективных структур данных.</p>
3	Сортировки	<p>Простейшие алгоритмы сортировки. Алгоритм Шелла. Быстрая сортировка Хоара, оценка его сложности в среднем. Пирамидальная сортировка. Сортировка слиянием. Цифровая сортировка. Цифровая сортировка строк.</p> <p>Особенности задачи сортировки информации на файлах. Сбалансированное слияние. Многофазная сортировка, ее анализ. Особенности практической реализации.</p>
4	Порядковые статистики	<p>Задача определения k-го элемента. Алгоритм, основанный на быстрой сортировке, его сложность в среднем. Алгоритм, эффективный в наихудшем случае.</p>
5	Хеш-таблицы	<p>Задача хеширования. Хеш-функция. Хеш-таблица с областью переполнения, поиск, удаление элементов. Хеш-таблица с открытой адресацией,</p>

		эффективность поиска в среднем. Применение хеш-таблиц в файлах.
6	Бинарные деревья. Основные понятия	Понятие бинарного дерева. Представление бинарного дерева. Свойства. Обход бинарного дерева рекурсивный.
7	Бинарные деревья поиска	Понятие бинарного дерева поиска. Базовый интерфейс двоичного дерева поиска.
8	Красно-черные деревья	Сбалансированное дерево поиска. Понятие красно-черного дерева. Структура узла дерева. Основные операции. Операции восстановления структуры красно-чётного дерева. Применение.
9	АВЛ-деревья	Сбалансированные по высоте деревья. Понятие AVL-дерева. Операция балансировки вершины. Виды вращений. Операции над AVL-деревьями. Эффективность. Применение.
10	2-3 деревья и другие	Понятие 2-3 дерева. Поиск в 2-3 деревьях. Добавление нового узла 2-3 дерева. Преобразование 2-3 дерева. B-дерево k-го порядка. B+ дерево. AA-деревья. Сравнения разных видов деревьев.
11	Кучи	Понятие кучи. Двоичная куча. Реализация очереди с приоритетом. Примеры применения двоичной кучи. Биноминальные кучи. Фибоначчиевы кучи.
12	Вычислительная геометрия	Элементы вычислительной геометрии. Базовые процедуры. Поиск пересекающихся отрезков. Задача о ближайших точках. Задача об наиболее удалённых точках. Проверка выпуклости многоугольника. Построение звёздчатого многоугольника. Нахождение выпуклой оболочки множества точек. Триангуляция многоугольника.
13	Графы. Основные понятия	Начальные понятия теории графов. Представление графа. Операции над графами. Понятие подграфа.
14	Методы поиска в глубину и в ширину в графах	Поиск в глубину (Depth-first search, DFS). Поиск в глубину в неориентированном графе. Глубинный остовный лес. Поиск в глубину в ориентированном графе. Решение задачи топологической сортировки методом поиска в глубину. Поиск компонент связности в графе. Метод поиска в ширину (BFS, Breadth-first search). Нахождение кратчайшего пути в лабиринте. Минимальное остовное дерево. Понятие остовного дерева. Способы построения остовных деревьев. Алгоритм Борувки. Алгоритм Крускала. Алгоритм Прима.
15	Методы поиска кратчайшего расстояния	Алгоритм Дейкстры нахождения кратчайшего пути. Алгоритм Беллмана-Форда. Алгоритм Флойда нахождения кратчайших путей между парами вершин. Эйлеровы графы. Алгоритмы поиска выхода из лабиринта.
16	Поиск решения в комбинаторных задачах	Перебор вариантов. Бэктрекинг, общий алгоритм. Оптимизационные задачи. Метод ветвей и границ для решения оптимизационных задач. Задача коммивояжера. Оценки трудоемкости.

		Приближенные решения задачи коммивояжера. Приближенное решение задачи коммивояжера с помощью минимального остова.
17	Комбинаторные задачи на графах	Минимальная раскраска графа, переборный алгоритм. Приближенные алгоритмы раскраски графа, основанные на понятии соцветных вершин. Раскраска методом ветвей и границ. Гамильтонов цикл. Поиск клик в графе. Узельное покрытие.
18	Динамическое программирование	Понятие динамического программирования. Признаки возможности применения динамического программирования. Постановка задачи динамического программирования. Этапы разработки алгоритма динамического программирования. Примеры решения задач методом динамического программирования.
19	Поиск цепочек символов	Основные понятия. Наивный (прямой) поиск подстроки. Алгоритм Рабина-Карпа. Алгоритм Бойера—Мура. Алгоритм Бойера-Мура со сдвигом по стоп-символам. Алгоритм Кнута-Морриса-Пратта. Сравнение алгоритмов.
20	Жадные алгоритмы	Понятие жадного алгоритма. Задача о заявках. Задача о рюкзаке. Размен монет. Кодирование информации. Кодирование по Хаффмену.
21	NP-полнота задачи выполнимости	Классы P и NP задач. Теорема Кука о задаче выполнимости булевых формул. NP-полнота задачи выполнимости. Задача 3-выполнимости. Раскраска графа. Клики. Узельное покрытие. Гамильтоновы циклы. Задача коммивояжера.
22	Связь задач по сложности	NP-трудные задачи. Класс языков P-SPACE. Связь ДМТ и НМТ по емкостной сложности. Связь классов языков P-SPACE, P, NP-полных и NP-трудных.

## 6. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине

Рекомендуемая тематика учебных занятий *лекционного* типа (предусматривающих преимущественную передачу учебной информации преподавателями):

№	Наименование раздела	Тема лекции
1	Введение	Лекция 1. Предмет дисциплины и ее задачи. Связь с другими дисциплинами учебного плана направления и специальности.
2	Структуры данных	Лекция 2. Стеки, очереди, списки, их моделирование с помощью массивов. Представления множеств в виде массивов и списков. Лекция 3. Графы и различные способы их представления. Таблицы. Выбор наиболее эффективных структур данных.
3	Сортировки	Лекция 4. Простейшие алгоритмы сортировки. Алгоритм Шелла. Быстрая сортировка Хоара,

		оценка его сложности в среднем. Лекция 5. Пирамидальная сортировка. Сортировка слиянием. Цифровая сортировка. Цифровая сортировка строк. Лекция 6. Особенности задачи сортировки информации на файлах. Сбалансированное слияние. Многофазная сортировка, ее анализ. Особенности практической реализации.
4	Порядковые статистики	Лекция 8. Задача определения k-го элемента. Алгоритм, основанный на быстрой сортировке, его сложность в среднем. Алгоритм, эффективный в наихудшем случае.
5	Хеш-таблицы	Лекция 9. Задача хеширования. Хеш-функция. Хеш-таблица с областью переполнения, поиск, удаление элементов. Хеш-таблица с открытой адресацией, эффективность поиска в среднем. Применение хеш-таблиц в файлах.
6	Бинарные деревья. Основные понятия	Лекция 10. Понятие бинарного дерева. Представление бинарного дерева. Свойства. Обход бинарного дерева рекурсивный.
7	Бинарные деревья поиска	Лекция 11. Понятие бинарного дерева поиска. Базовый интерфейс двоичного дерева поиска.
8	Красно-черные деревья	Лекция 12. Понятие красно-черного дерева. Структура узла дерева. Основные операции.
9	АВЛ-деревья	Лекция 13. Понятие АВЛ-дерево
10	2-3 деревья и другие	Лекция 14. Понятие 2-3 дерева. В-дерево k-го порядка. В+ дерево. АА-деревья. Сравнения разных видов деревьев.
11	Кучи	Лекция 15. Понятие кучи. Реализация очереди с приоритетом. Примеры применения двоичной кучи.
12	Вычислительная геометрия	Лекция 16. Элементы вычислительной геометрии. Базовые процедуры. Лекция 17. Проверка выпуклости многоугольника. Построение звёзчатого многоугольника. Нахождение выпуклой оболочки множества точек.
13	Графы. Основные понятия	Лекция 18. Начальные понятия теории графов. Представление графа. Операции над графами. Понятие подграфа.
14	Методы поиска в глубину и в ширину в графах	Лекция 19. Поиск в глубину (Depth-first search, DFS). Поиск в глубину в неориентированном графе. Лекция 20. Метод поиска в ширину (BFS, Breadth-first search). Нахождение кратчайшего пути в лабиринте.
15	Методы поиска кратчайшего расстояния	Лекция 21. Алгоритм Дейкстры нахождения кратчайшего пути. Алгоритм Беллмана-Форда. Лекция 22. Алгоритм Флойда нахождения кратчайших путей между парами вершин. Эйлеровы графы.
16	Поиск решения в комбинаторных задачах	Лекция 23. Перебор вариантов. Бэктрекинг, общий алгоритм. Оптимизационные задачи. Лекция 24. Задача коммивояжера.
17	Комбинаторные задачи	Лекция 25. Минимальная раскраска графа,

	на графах	переборный алгоритм. Лекция 26. Гамильтонов цикл. Поиск клик в графе.
18	Динамическое программирование	Лекция 27. Понятие динамического программирования. Признаки возможности применения динамического программирования. Лекция 28. Постановка задачи динамического программирования. Этапы разработки алгоритма динамического программирования. Примеры решения задач методом динамического программирования.
19	Поиск цепочек символов	Лекция 29. Основные понятия. Наивный (прямой) поиск подстроки. Алгоритм Рабина-Карпа. Алгоритм Бойера—Мура. Лекция 30. Алгоритм Бойера-Мура со сдвигом по стоп-символам. Алгоритм Кнута-Морриса-Пратта. Сравнение алгоритмов.
20	Жадные алгоритмы	Лекция 31. Понятие жадного алгоритма. Задача о заявках. Задача о рюкзаке. Лекция 32. Размен монет. Кодирование информации. Кодирование по Хаффмену.
21	NP-полнота задачи выполнимости	Лекция 33. Классы P и NP задач. Теорема Кука о задаче выполнимости булевых формул. NP-полнота задачи выполнимости. Лекция 34. Задача 3-выполнимости. Раскраска графа. Клики. Узельное покрытие. Гамильтоновы циклы. Задача коммивояжера.
22	Связь задач по сложности	Лекция 35. NP-трудные задачи. Класс языков P-SPACE. Связь ДМТ и НМТ по емкостной сложности. Связь классов языков P-SPACE, P, NP-полных и NP-трудных.

**Рекомендуемая тематика практических занятий:**

№ п/п	Наименование Темы	Содержание темы
1	Тема 2. Структуры данных	Задача о скобках. Задача об арифметическом выражении. Задача о простых множителях.
2	Тема 3. Сортировки	Метод «пузырька» $O(n^2)$ . Сортировка вставками $O(n^2)$ . Сортировка посредством выбора $O(n^2)$ . Сортировка Шелла. Корневая сортировка. Пирамидальная сортировка. Сортировка слиянием. Быстрая сортировка. Внешняя многофазная сортировка
3	Тема 5. Хеш-таблицы	Хеш-таблицы с наложением. Хеш-таблицы со списками.
4	Тема 6. Бинарные деревья. Основные понятия	Рекурсивные обходы (прямой, центрированный, концевой). Не рекурсивный прямой обход.
5	Тема 7. Бинарные деревья поиска	Операции над БДП: поиск, добавление, удаление. Сортировка бинарным деревом поиска.
6	Тема 8. Красно-черные деревья	Операции над красно-черными деревьями: поиск, добавление, удаление
7	Тема 12.	Записать алгоритмы нахождения точек пересечения двух

	Вычислительная геометрия	прямых, прямой и отрезка, двух отрезков, прямой и окружности, отрезка и окружности, двух окружностей.
8	Тема 14. Методы поиска в глубину и в ширину в графах	Найти в заданном графе кратчайшие пути из заданной вершины до всех остальных вершин с помощью поиска в ширину Найти в заданном графе количество и состав компонент связности с помощью поиска в ширину. Найти в заданном графе количество и состав компонент связности с помощью поиска в глубину Найти в заданном орграфе количество и состав сильно связанных компонент с помощью поиска в глубину. Реализовать алгоритм Крускала нахождения минимального покрывающего дерева. Реализовать алгоритм Прима нахождения минимального покрывающего дерева.
9	Тема 15. Методы поиска кратчайшего расстояния	Реализовать алгоритм Дейкстры поиска кратчайших путей из одной вершины, используя в качестве приоритетной очереди обычный массив Реализовать алгоритм Беллмана-Форда поиска кратчайших путей из одной вершины Реализовать алгоритм нахождения эйлерова цикла в неориентированном графе, заданном матрицей смежности.
10	Тема 17. Комбинаторные задачи на графах	Решить задачу о раскраске графа.
11	Тема 18. Динамическое программирование	Решить дискретную задачу о рюкзаке.
12	Тема 19. Поиск цепочек символов	Реализовать алгоритм поиска по образцу с помощью конечного автомата Реализовать алгоритм Кнута-Морриса-Пратта для поиска по образцу Реализовать алгоритм Бойера-Мура для поиска по образцу Реализовать алгоритм Рабина для поиска по образцу
13	Тема 21. NP-полнота задачи выполнимости	Решить задачу о раскладке по ящикам Решить задачу о суммах подмножеств, используя жадный алгоритм.

На практических занятиях решаются задачи по теме занятия.

### **Требования к самостоятельной работе обучающихся**

1. Работа с лекционным материалом, предусматривающая проработку конспекта лекций и учебной литературы, по всем темам из п. 6 настоящей рабочей программы.
2. Выполнение домашнего задания, предусматривающего решение задач, выполнение упражнений, выдаваемых на практических занятиях, по всем темам из п. 6 настоящей рабочей программы.

Руководствуясь положениями статьи 47 и статьи 48 Федерального закона от 29 декабря 2012 г. N 273-ФЗ «Об образовании в Российской Федерации» научно-

педагогические работники и иные лица, привлекаемые университетом к реализации данной образовательной программы, пользуются предоставленными академическими правами и свободами в части свободы преподавания, свободы от вмешательства в профессиональную деятельность; свободы выбора и использования педагогически обоснованных форм, средств, методов обучения и воспитания; права на творческую инициативу, разработку и применение авторских программ и методов обучения и воспитания в пределах реализуемой образовательной программы и отдельной дисциплины.

Исходя из рамок, установленных учебным планом по трудоемкости и видам учебной работы по дисциплине, преподаватель самостоятельно выбирает тематику занятий по формам и количеству часов проведения контактной работы: лекции и иные учебные занятия, предусматривающие преимущественную передачу учебной информации преподавателем и (или) занятия семинарского типа (семинары, практические занятия, практикумы, лабораторные работы, коллоквиумы и иные аналогичные занятия), и (или) групповые консультации, и (или) индивидуальную работу обучающихся с преподавателем, в том числе индивидуальные консультации (по курсовым работам/проектам – при наличии курсовой работы/проекта по данной дисциплине в учебном плане).

Рекомендуемая тематика занятий максимально полно реализуется в контактной работе с обучающимися очной формы обучения. В случае реализации образовательной программы в заочной / очно-заочной форме трудоемкость дисциплины сохраняется, однако объем учебного материала в значительной части осваивается обучающимися в форме самостоятельной работы. При этом требования к ожидаемым образовательным результатам обучающихся по данной дисциплине не зависят от формы реализации образовательной программы.

## **7. Методические рекомендации по видам занятий**

Лекционные занятия.

В ходе лекционных занятий обучающимся рекомендуется выполнять следующие действия. Вести конспектирование учебного материала. Обращать внимание на категории, формулировки, раскрывающие содержание тех или иных явлений и процессов, научные выводы и практические рекомендации по их применению. Задавать преподавателю уточняющие вопросы с целью уяснения теоретических положений, разрешения спорных ситуаций.

Желательно оставить в рабочих конспектах поля, на которых во внеаудиторное время можно сделать пометки из рекомендованной литературы, дополняющие материал прослушанной лекции, а также подчеркивающие особую важность тех или иных теоретических положений.

Практические и семинарские занятия.

На практических и семинарских занятиях в зависимости от темы занятия выполняется поиск информации по решению проблем, практические упражнения, контрольные работы, выработка индивидуальных или групповых решений, итоговое обсуждение с обменом знаниями, участие в круглых столах, разбор конкретных ситуаций, командная работа, представление портфолио и т.п.

Самостоятельная работа.



Самостоятельная работа осуществляется в виде изучения литературы, эмпирических данных по публикациям и конкретных ситуаций из практики, подготовке индивидуальных работ, работа с лекционным материалом, самостоятельное изучение отдельных тем дисциплины; поиск и обзор литературы и электронных источников; чтение и изучение учебника и учебных пособий.

## 8. Фонд оценочных средств

### 8.1. Перечень компетенций с указанием этапов их формирования в процессе освоения образовательной программы в рамках учебной дисциплины

Основными этапами формирования указанных компетенций при изучении обучающимися дисциплины являются последовательное изучение содержательно связанных между собой тем учебных занятий. Изучение каждой темы предполагает овладение обучающимися необходимыми компетенциями. Результат аттестации обучающихся на различных этапах формирования компетенций показывает уровень освоения компетенций.

Контролируемые разделы (темы) дисциплины	Индекс контролируемой компетенции (или её части)	Оценочные средства по этапам формирования компетенций
		текущий контроль по дисциплине
Тема 1. Введение	УК-1, ОПК-2	Лабораторная работа
Тема 2. Структуры данных	УК-1, ОПК-2	Лабораторная работа
Тема 3. Сортировки	УК-1, ОПК-2	Лабораторная работа
Тема 4. Порядковые статистики	УК-1, ОПК-2	Тест
Тема 5. Хеш-таблицы	УК-1, ОПК-2	Тест
Тема 6. Бинарные деревья. Основные понятия	УК-1, ОПК-2	Тест
Тема 7. Бинарные деревья поиска	УК-1, ОПК-2	Тест
Тема 8. Красно-черные деревья	УК-1, ОПК-2	Лабораторная работа
Тема 9. AVL-деревья	УК-1, ОПК-2	Лабораторная работа
Тема 10. 2-3 деревья и другие	УК-1, ОПК-2	Лабораторная работа
Тема 11. Кучи	УК-1, ОПК-2	Тест
Тема 12. Вычислительная геометрия	УК-1, ОПК-2	Лабораторная работа
Тема 13. Графы. Основные понятия	УК-1, ОПК-2	Тест
Тема 14. Методы поиска в глубину и в ширину в графах	УК-1, ОПК-2	Лабораторная работа
Тема 15. Методы поиска кратчайшего расстояния	УК-1, ОПК-2	Лабораторная работа
Тема 16. Поиск решения в комбинаторных задачах	УК-1, ОПК-2	Тест

Тема 17. Комбинаторные задачи на графах	УК-1, ОПК-2	Лабораторная работа
Тема 18. Динамическое программирование	УК-1, ОПК-2	Лабораторная работа
Тема 19. Поиск цепочек символов	УК-1, ОПК-2	Лабораторная работа
Тема 20. Жадные алгоритмы	УК-1, ОПК-2	Лабораторная работа
Тема 21. NP-полнота задачи выполнимости	УК-1, ОПК-2	Лабораторная работа
Тема 22. Связь задач по сложности	УК-1, ОПК-2	Тест

## 8.2. Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений и навыков и (или) опыта деятельности процессе текущего контроля

### Типовой тест к теме «AVL-деревья»

1. Отметить правильный (ые) ответ (ы): «Основное свойство бинарного дерева поиска»	<p>А) Если <math>x</math> — узел бинарного дерева поиска, а узел <math>y</math> находится в левом поддереве <math>x</math>, то <math>key[y] \geq key[x]</math>. Если узел <math>y</math> находится в правом поддереве <math>x</math>, то <math>key[x] \geq key[y]</math>.</p> <p>Б) Если <math>x</math> — узел бинарного дерева поиска, а узел <math>y</math> находится в левом поддереве <math>x</math>, то <math>key[y] \geq key[x]</math>. Если узел <math>y</math> находится в правом поддереве <math>x</math>, то <math>key[x] \leq key[y]</math>.</p> <p><b>В) Если <math>x</math> — узел бинарного дерева поиска, а узел <math>y</math> находится в левом поддереве <math>x</math>, то <math>key[y] \leq key[x]</math>. Если узел <math>y</math> находится в правом поддереве <math>x</math>, то <math>key[x] \leq key[y]</math>.</b></p> <p>Г) Если <math>x</math> — узел бинарного дерева поиска, а узел <math>y</math> находится в левом поддереве <math>x</math>, то <math>key[y] &gt; key[x]</math>. Если узел <math>y</math> находится в правом поддереве <math>x</math>, то <math>key[x] &lt; key[y]</math>.</p>
2. Отметить правильный (ые) ответ (ы): «Выберите операции над AVL-деревьями»	<p><b>А) Добавление вершины</b></p> <p>Б) Перекрашивание вершины</p> <p><b>В) Удаление вершины</b></p> <p><b>Г) Поиск вершины</b></p> <p>Д) Разбиение на два дерева</p> <p><b>Е) Слияние двух деревьев</b></p>
3. Продолжить правильно утверждение: «AVL-дерево – это»	<p>А) сбалансированное двоичное дерево, в котором поддерживается следующее свойство: для каждой его вершины высота её двух поддеревьев различается не более чем на 1.</p> <p><b>Б) сбалансированное двоичное дерево поиска, в котором поддерживается следующее свойство: для каждой его вершины высота её двух поддеревьев различается не более, чем на 1.</b></p> <p>В) сбалансированное двоичное дерево поиска, в котором поддерживается следующее свойство: для каждой его вершины высота её двух поддеревьев различается более, чем на 1.</p>
4. Отметить правильный (ые) ответ (ы): «Для AVL-деревьев определены»	<p><b>А) малое левое вращение</b></p> <p><b>Б) большое правое вращение</b></p>

следующие виды вращений:»	<b>В) малое правое вращение</b> Г) перестановка вершин <b>Д) большое левое вращение</b>
5. Продолжить правильно утверждение: «Балансировкой вершины	<b>А) Балансировкой вершины называется операция, которая в случае разницы высот левого и правого поддеревьев <math> h(L) - h(R)  &lt; 2</math>, изменяет связи предок-потомок в поддереве данной вершины так, чтобы восстановилось свойство дерева <math> h(L) - h(R)  \leq 1</math>, иначе ничего не меняет.</b> Б) Балансировкой вершины называется операция, которая в случае разницы высот левого и правого поддеревьев $ h(L) - h(R)  < 3$ , изменяет связи предок-потомок в поддереве данной вершины так, чтобы восстановилось свойство дерева $ h(L) - h(R)  \leq 2$ , иначе ничего не меняет. В) Балансировкой вершины называется операция, которая в случае разницы высот левого и правого поддеревьев $ h(L) - h(R)  > 2$ , изменяет связи предок-потомок в поддереве данной вершины так, чтобы восстановилось свойство дерева $ h(L) - h(R)  \leq 1$ , иначе ничего не меняет.

### Типовые тесты КСР

1. Выберите правильное продолжение определения: «Конечный граф $G = (V, E)$ состоит из конечного множества вершин $V = \{v_1, v_2, \dots\}$ и ...	А) неполного множества ребер $E\{e_1, e_2, \dots\}$ Б) счётного множества ребер $E\{e_1, e_2, \dots\}$ В) бесконечного множества ребер $E\{e_1, e_2, \dots\}$ <b>Г) конечного множества ребер <math>E\{e_1, e_2, \dots\}</math></b>
2. Выберите правильное продолжение определения: «Граф называется ориентированным, если пара вершин $(v, w)$ , соответствующая каждому ребру, ...»	А) не задана Б) неупорядочена <b>В) упорядочена</b> Г) отсортирована
3. Выберите правильное продолжение определения: «Кликкой графа G называется..	<b>А) подмножество его вершин, такое, что между каждой парой вершин этого подмножества существует ребро, и это подмножество не принадлежит никакому большему подмножеству с тем же свойством</b> Б) любое подмножество его вершин В) максимальный полный обход графа G <b>Г) максимальный полный подграф графа G</b>
4. На рисунке изображены операции <div style="text-align: center;"> <p style="text-align: center;"><math>G</math>                  <math>G'</math>                  <math>G''</math></p> </div>	<b>А) стягивание ребра</b> <b>Б) подразбиение ребра</b> В) удаление ребра Г) удаление вершины
5. Выберите правильное продолжение определения: «Маршрут в графе – это...»	<b>А) последовательность вершин <math>x_1, x_2, \dots, x_n</math>, такая, что для каждого <math>i = 1, 2, \dots, n - 1</math> вершины <math>x_i</math> и <math>x_{i+1}</math> соединены ребром.</b> Б) последовательность вершин $x_1, x_2, \dots, x_n$ , такая, что для каждого $i = 1, 2, \dots, n - 1$ вершины $x_i$ и $x_{i+1}$ соединены несколькими ребрами

	В) последовательность вершин $x_1, x_2 \dots x_n$ , идущих друг за другом
6 Выберите правильное продолжение определения: «Путь – это...»	А) маршрут, в котором ребра могут повторяться <b>Б) маршрут, в котором все ребра различны</b> В) маршрут, в котором возможно повторное прохождение по одному и тому же ребру
7. Отметить правильный (ые) ответ (ы): «К достаточным условиям существования гамильтоновых графов относится...»	А) условие, что всякий полный граф является гамильтоновым <b>Б) условие, что если для любой пары вершин <math>v_u</math> и <math>v_v</math> графа <math>G</math> с <math>m</math> вершинами справедливо неравенство <math>\rho(v_u) + \rho(v_v) \geq m</math>,</b> В) условие, что всякий оргграф является гамильтоновым Г) условие, что если в простом графе с $n \geq 3$ вершинами $\rho(v) \geq n/2$ для любой вершины $v$
8. Ниже представлен программный код const MAX_N = 10; var graph: array [1..MAX_N,1..MAX_N] of boolean; visited: array [1..MAX_N] of boolean; procedure dfs (v: integer) ; var i: integer; begin visited[v] := true; for i := 1 to MAX_N do if graph[v, i] and not visited[i] then dfs(i) ; end;	А) поиска в глубину Б) поиска в ширину В) сортировки вставками Г) сортировки обмeнами
9. Отметить правильный (ые) ответ (ы): «Представленная итерационная формула является частью алгоритма...» $A_{ij}^k = \min(A_{ij}^{k-1}, A_{ik}^{k-1}, A_{kj}^{k-1})$	А) алгоритма Дейкстры Б) алгоритма Беллмана-Форда <b>В) алгоритма Флойда</b> Г) формулы Зейделя
10. Отметить правильный (ые) ответ (ы): «Для нахождения выхода из лабиринта необходимо руководствоваться...»	А) пятью правилами <b>Б) тремя правилами</b> В) четырьмя правилами Г) здравым смыслом
10. Отметить правильный (ые) ответ (ы): «Какие этапы относятся к этапам динамического программирования»	А). Описание структуры оптимального решения. <b>Б) Рекурсивное определение значения, соответствующего оптимальному решению.</b> <b>В) Вычисление значения, соответствующего оптимальному решению, с помощью метода восходящего анализа.</b> Г) Составление оптимального решения на основе информации, полученной на предыдущих этапах. Д) Внесение изменений в структуру оптимального решения
11. Отметить правильный (ые) ответ (ы): «К жадным алгоритмам можно отнести»	А) приближённую задачу о коммивояжёре Б) приближённую задачу о раскладке по рюкзакам <b>В) приближённую задачу о раскраске графа</b> Г) задачу о назначении

12. Правильно продолжить утверждение: «В алгоритме Бойера-Мура применяются массивы»	А) прыжков и перемещений Б) скачков и сдвигов <b>В) прыжков и сдвигов</b>
--	---

### Примерные темы практических групповых заданий

- **Задача. Кампус**

Имя входного файла:	building.in
Имя выходного файла:	building.out
Ограничение по времени:	1 секунда
Ограничение по памяти:	256 мегабайт

Новое здание кампуса Университета Байтбурга имеет  $n$  этажей, пронумерованных снизу вверх от 1 до  $n$ . Комнаты студентов расположены в нескольких подъездах.

В каждом подъезде на этажах, номер которых кратен числу  $k$ , расположено по  $x$  комнат, а на остальных этажах расположено по  $y$  комнат.

Комнаты внутри каждого подъезда пронумерованы последовательными натуральными числами. Номера комнат на первом этаже имеют наименьшие значения в этом подъезде, затем следуют номера комнат на втором этаже, и так далее. Комнаты в первом подъезде пронумерованы, начиная с 1, в каждом следующем подъезде нумерация комнат начинается с числа, следующего после максимального номера комнаты в предыдущем подъезде.

На рис. 1 показаны номера комнат в здании с  $n = 7$  этажами, 3 подъездами, и параметрами  $k = 3$ ,  $x = 2$ ,  $y = 3$ .

	Подъезд 1	Подъезд 2	Подъезд 3
7 этаж	17, 18, 19	36, 37, 38	55, 56, 57
6 этаж	15, 16	34, 35	53, 54
5 этаж	12, 13, 14	31, 32, 33	50, 51, 52
4 этаж	9, 10, 11	28, 29, 30	47, 48, 49
3 этаж	7, 8	26, 27	45, 46
2 этаж	4, 5, 6	23, 24, 25	42, 43, 44
1 этаж	1, 2, 3	20, 21, 22	39, 40, 41

Рис. 1. Пример нумерации комнат в здании

Для организации расселения студентов администрация кампуса должна по номеру комнаты оперативно определять этаж, на котором она находится.

Требуется написать программу, которая по заданным числам  $n$ ,  $k$ ,  $x$  и  $y$ , а также по номерам комнат, определяет для каждой комнаты, на каком этаже она находится.

#### Формат входного файла

Первая строка входного файла содержит натуральные числа  $n$ ,  $k$ ,  $x$  и  $y$  ( $1 \leq n \leq 10^9$ ,  $1 \leq k \leq n$ ,  $1 \leq x, y \leq 10^9$ ). Соседние числа разделены ровно одним пробелом.

Вторая строка входного файла содержит натуральное число  $q$  — количество номеров комнат, для которых требуется определить этаж ( $1 \leq q \leq 1000$ ).

Третья строка содержит  $q$  целых чисел  $a_1, a_2, \dots, a_q$  — номера комнат ( $1 \leq a_i \leq 10^{18}$ ). Можно считать, что в здании так много подъездов, что все комнаты с заданными номерами существуют.

### Формат выходного файла

Требуется вывести  $q$  чисел, по одному на строке. Для каждого номера комнаты во входном файле требуется вывести номер этажа, на котором она находится.

#### Пример входных и выходных файлов

building.in	building.out
7 3 2 3	1
4	7
1 19 20 50	1
	5

- **Задача. «Шахматный матч»**

**Входной файл:** chess.in

**Выходной файл:** chess.out

**Ограничение времени:** 1 секунда на тест

**Ограничение памяти:** 128 М байт

Марк и Максим играют между собой шахматный матч. Вероятность того, что в одной партии победит Марк, равна  $a/(a+b+c)$ . Вероятность того, что в одной партии победит Максим, равна  $b/(a+b+c)$ . Соответственно вероятность ничьей равна  $c/(a+b+c)$ . Мальчики договорились, что матч будет состоять не более, чем из  $N$  партий. Но если кто-то из них вырвется вперед на  $K$  очков, то матч сразу заканчивается. Ваша задача – найти ожидаемую продолжительность шахматного матча.

**Вход**

Во входном файле записаны пять целых чисел –  $a, b, c, N, K$  ( $1 \leq a, b, c \leq 10^6, 3 \leq N \leq 10, 1 \leq K < N$ ).

**Выход**

Запишите в выходной файл ожидаемое количество партий, которые будут сыграны в матче, с четырьмя дробными цифрами.

Примеры входа и выхода

chess.in	chess.out
1 2 1 5 5	5.0000
1 2 1 5 4	4.9336
1 2 1 5 2	3.6133
1 2 1 5 1	1.3320

**Пояснение**

Победитель партии получает 1 очко, проигравший – 0 очков, если партия заканчивается вничью, то оба игрока получают по  $\frac{1}{2}$  очка.

- **Задача. "Волшебник"**

**Ограничение времени:** 1 секунда на тест

**Ограничение памяти:** 256 М байт

Волшебник имеет  $N$  магических предметов ( $1 \leq N \leq 30$ ), каждый из которых характеризуется своей ценностью  $v_i$  ( $0 < v_i \leq 10000$ ). Он может произнести  $M$  заклинаний ( $1 \leq M \leq 10$ ), изменяющих ценность имеющихся предметов. Каждое заклинание может быть произнесено не более одного раза. Произнесенное заклинание действует на все

имеющиеся предметы. Заклинания делятся на 2 типа. После сотворения заклинания первого типа с номером  $j$  стоимость предмета  $i$  изменяется в  $D_{ij}$  раз (если  $1 < D_{ij} \leq 100$ , абсолютная величина стоимости увеличивается, при  $0 \leq D_{ij} < 1$  уменьшается, при  $D_{ij} = 1$  остается неизменной). Заклинание второго типа с номером  $j$  изменяет стоимость предмета  $i$  на  $R_{ij}$  (если  $R_{ij} > 0$ , стоимость увеличивается, при  $R_{ij} < 0$  - уменьшается, при  $R_{ij} = 0$  остается неизменной). Волшебник должен с помощью известных ему заклинаний добиться того, чтобы суммарная ценность имеющихся предметов была максимальной.

#### Вход

Текстовый файл **WIZARD.IN** содержит  $M + 2$  строки. Первая строка содержит значения  $N$  и  $M$ . Следующая строка содержит значения  $v_i$  ( $i = 1, \dots, N$ ). Наконец, каждая из последних  $M$  строк соответствует одному заклинанию. Для заклинания первого типа эта строка содержит символ  $*$  и значения  $D_{ij}$  ( $i = 1, \dots, N$ ). Для заклинания второго типа она содержит символ  $+$  и значения  $R_{ij}$  ( $i = 1, \dots, N$ ). Данные в строках входного файла разделяются одним или несколькими пробелами.

#### Выход

Выходные данные помещаются в текстовый файл **WIZARD.OUT** и содержат две строки. Первая строка содержит получившуюся суммарную стоимость предметов (с точностью до 0.001), вторая -  $M$  разделенных одним пробелом чисел  $t_j$  ( $j = 1, \dots, M$ ), где  $t_j = k$ , если заклинание  $j$  было произнесено  $k$ -м по счету, и  $t_j = 0$ , если заклинание не было произнесено.

Примеры входа и выхода

WIZARD.IN	WIZARD.OUT
4 2	29.000
2 2 2 2	1 2
* 3 2 1 2	
* 0.5 1 1 5	

### 8.3. Перечень вопросов и заданий для промежуточной аттестации по дисциплине

#### Вопросы для промежуточного контроля (экзамена)

- Алгоритмы сортировки сложности  $n^2$ : сортировка обменами, сортировка выбором.
- Алгоритмы сортировки сложности  $n^2$ : сортировка "пузырьком", сортировка бинарными вставками.
- Стек. Задача о скобках. Задача об арифметическом выражении.
- Простая очередь в линейном и закольцованном массиве. Задача о простых множителях.
- Односвязный и двусвязный список.
- Сортировка слиянием. Рекурсивный и не рекурсивный алгоритмы.
- Сортировка QUICKSORT. Случайный выбор среднего элемента, медиана трех, метод Синглтона.
- Сортировка Хоара. Сортировка списка.
- Порядковые статистики.
- Двоичная куча. Сортировка HEAPSORT.
- Хеш-таблицы с наложением.
- Хеш-таблицы со списками.

- Произвольное дерево в формате "предок-левый потомок-правый сосед".
- Бинарное дерево. Линейные бесскобочные записи. Линейные скобочные записи.
- Бинарное дерево. Рекурсивные обходы. Обход сверху - вниз. Не рекурсивный полный обход.
- Бинарное дерево поиска. Операции над БДП: поиск, добавление, удаление. Сортировка бинарным деревом поиска.
- Построение AVL-дерева, добавление вершин.
- Построение AVL-дерева, добавление и удаление вершин.
- Построение AVL-дерева, поиск минимального и максимального элемента.
- Построение прошитого AVL-дерева, поиск соседнего элемента.
- Построение 2-3-дерева, добавление вершин.
- Построение 2-3-дерева, добавление и удаление вершин.

### **Вопросы для промежуточного контроля (зачета с оценкой)**

- Геометрия на плоскости. Точка, прямая, луч, отрезок, окружность.
- Геометрия на плоскости. Ориентированная площадь треугольника. Простой многоугольник. Выпуклый многоугольник.
- Выпуклая оболочка. Метод "обертывания". Метод Грэхема.
- Основные понятия теории графов.
- Поиск в ширину. Проверка графа на двудольность.
- Поиск в глубину. Сильно связные компоненты.
- Топологическая сортировка поиском в глубину. Топологическая сортировка последовательным удалением вершин.
- Минимальное покрывающее дерево. Алгоритм Крускала. Алгоритм Прима.
- Остовное дерево наименьшей стоимости, алгоритм Прима.
- Остовное дерево наименьшей стоимости, алгоритм Крускала.
- Остовное дерево наименьшей стоимости, алгоритм Крускала с алгоритмом быстрого объединения множеств (сжатие путей).
- Кратчайшие пути из одной вершины. Алгоритм Дейкстры. Алгоритм Беллмана - Форда.
- Кратчайшие пути для всех пар вершин. Алгоритм Флойда-Уоршола.
- Жадные алгоритмы. Задача о размене. Задача о назначениях.
- Динамическое программирование. Метод "снизу-вверх". Метод "сверху-вниз с динамической таблицей". Задача о НОП. Задача о произведении матриц. Задача об оптимальной триангуляции.
- Комбинаторные алгоритмы. Рекурсивные и не рекурсивные генераторы подмножеств, выборки, перестановки.
- Простейший алгоритм распознавания подцепочки.
- Алгоритм распознавания подцепочки, вычисляющий функцию отказов.
- Алгоритм Бауэра-Мура распознавания подцепочки.
- Поиск в лабиринте. Рекурсивный вариант.
- Поиск в лабиринте. Нерекурсивный вариант.
- Бэктрекинг для какой-либо головоломки. Рекурсивный вариант.



#### 8.4. Планируемые уровни сформированности компетенций обучающихся и критерии оценивания

Уровни	Содержательное описание уровня	Основные признаки выделения уровня (этапы формирования компетенции, критерии оценки сформированности)	Пятибалльная шкала (академическая) оценка	Двухбалльная шкала, зачет	БРС, % освоения (рейтинговая оценка)
Повышенный	Творческая деятельность	<i>Включает нижестоящий уровень.</i> Умение самостоятельно принимать решение, решать проблему/задачу теоретического и прикладного характера на основе изученных методов, приемов, технологий	отлично	зачтено	86-100
Базовый	Применение знаний и умений в более широких контекстах учебной и профессиональной деятельности, нежели по образцу с большей степени самостоятельности и инициативы	<i>Включает нижестоящий уровень.</i> Способность собирать, систематизировать, анализировать и грамотно использовать информацию из самостоятельно найденных теоретических источников и иллюстрировать ими теоретические положения или обосновывать практику применения	хорошо		71-85
Удовлетворительный (достаточный)	Репродуктивная деятельность	Изложение в пределах задач курса теоретически и практически контролируемого материала	удовлетворительно		55-70
Недостаточный	Отсутствие признаков удовлетворительного уровня		неудовлетворительно	не зачтено	Менее 55

#### 9. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины.

##### Основная литература

- Белов, В. В. Алгоритмы и структуры данных : учебник / В. В. Белов, В. И. Чистякова. - Москва : КУРС : ИНФРА-М, 2020. - 240 с. - (Бакалавриат). - ISBN 978-5-906818-25-6. - Текст : электронный. - URL: <https://znanium.com/catalog/product/1057212> (дата обращения: 30.03.2023). – Режим доступа: по подписке.
- Царев, Р. Ю. Алгоритмы и структуры данных (CDIO): Учебник / Царев Р.Ю., Прокопенко А.В. - Краснояр.:СФУ, 2016. - 204 с.: ISBN 978-5-7638-3388-1. - Текст : электронный. - URL: <https://znanium.com/catalog/product/967108> (дата обращения: 20.11.2023). – Режим доступа: по подписке.
- Вирт, Н. Алгоритмы и структуры данных : учебное пособие / Н. Вирт ; пер. с англ. Ф. В. Ткачева. - 3-е изд. - Москва : ДМК Пресс, 2023. - 274 с. - ISBN 978-5-89818-313-4. - Текст : электронный. - URL: <https://znanium.com/catalog/product/2102600> (дата обращения: 20.11.2023). – Режим доступа: по подписке.

### Дополнительная литература

- Царев, Р. Ю. Алгоритмы и структуры данных (CDIO): Учебник / Царев Р.Ю., Прокопенко А.В. - Краснояр.:СФУ, 2016. - 204 с.: ISBN 978-5-7638-3388-1. - Текст : электронный. - URL: <https://znanium.com/catalog/product/967108> (дата обращения: 30.03.2023). – Режим доступа: по подписке.
- Колдаев, В. Д. Структуры и алгоритмы обработки данных [Электронный ресурс]: учеб. пособие для вузов/ В. Д. Колдаев. - Москва: РИОР; Москва: ИНФРА-М, 2014. - 1 эл. опт. диск (CD-ROM), 294 с.: ил., табл.. - (Высшее образование - бакалавриат). - Библиогр.: с. 285. - Лицензия до 23.06.2020 г.. - ISBN 978-5-369-01264-2. - ISBN 978-5-16-009012-2: 15100.00 р. Имеются экземпляры в отделах /There are copies in departments: всего /all 2: ЭБС Кантиана(1), ч.з.N1(1) Свободны / free: ЭБС Кантиана(1), ч.з.N1(1)
- Гданский, Н. И. Основы теории и алгоритмы на графах : учебное пособие / Н. И. Гданский. — Москва : ИНФРА-М, 2020. — 206 с. — (Высшее образование: Бакалавриат). - ISBN 978-5-16-014386-6. - Текст : электронный. - URL: <https://znanium.com/catalog/product/978686> (дата обращения: 30.03.2023). – Режим доступа: по подписке.

### 10. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины (модуля).

- НЭБ Национальная электронная библиотека, диссертации и прочие издания
- eLIBRARY.RU Научная электронная библиотека, книги, статьи, тезисы докладов конференций
- Гребенников Электронная библиотека ИД журналы
- ЭБС Консультант студента
- ПРОСПЕКТ ЭБС
- ЭБС ZNANIUM.COM
- ЭБС IBOOKS.RU
- РГБ Информационное обслуживание по МБА
- БЕН РАН
- Электронно-библиотечная система (ЭБС) Кантиана (<https://elib.kantiana.ru/>)

## **11. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине.**

Программное обеспечение обучения включает в себя:

- система электронного образовательного контента БФУ им. И. Канта – [www.lms.kantiana.ru](http://www.lms.kantiana.ru), обеспечивающую разработку и комплексное использование электронных образовательных ресурсов;
- серверное программное обеспечение, необходимое для функционирования сервера и связи с системой электронного обучения через Интернет;
- корпоративная платформа;
- установленное на рабочих местах обучающихся ПО: Microsoft Windows 7, Microsoft Office Standart 2010, антивирусное программное обеспечение Kaspersky Endpoint Security.
- Visual Studio 2010 и выше;
- Компиляторы языков C, C++, C#, Java, Python и другие

## **12. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине.**

Для проведения занятий лекционного типа, практических и семинарских занятий используются специальные помещения (учебные аудитории), оборудованные техническими средствами обучения – мультимедийной проекционной техникой. Для проведения занятий лекционного типа используются наборы демонстрационного оборудования.

Для проведения лабораторных работ, (практических занятий – при необходимости) используются специальные помещения (учебные аудитории), оснащенные специализированным лабораторным оборудованием: персональными компьютерами с возможностью выхода в интернет и с установленным программным обеспечением, заявленным в п.11.

Для проведения групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации используются специальные помещения (учебные аудитории), оборудованные специализированной мебелью (для обучающихся), меловой / маркерной доской.

Для организации самостоятельной работы обучающимся предоставляются помещения, оснащенные компьютерной техникой с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду университета.

Для обучения инвалидов и лиц с ограниченными возможностями здоровья университетом могут быть представлены специализированные средства обучения, в том числе технические средства коллективного и индивидуального пользования.

### **4.5. Программа дисциплины «Основы программной инженерии»**

#### **1.Наименование дисциплины: «Основы программной инженерии».**

**Цель** дисциплины: формирование у студентов теоретической и практической основы для овладения современными инженерными принципами создания надежного, качественного программного обеспечения, удовлетворяющего требованиям международных стандартов, а также понимания необходимости применения данных принципов для обеспечения высокой корпоративной культуры коллективов специалистов, обеспечивающих жизненный цикл коммерческих программных продуктов.

## 2. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы

Код компетенции	Результаты освоения образовательной программы (ИДК)	Результаты обучения по дисциплине
<p>ОПК-6. Способен понимать принципы работы современных информационных технологий и использовать их для решения задач профессиональной деятельности</p>	<p>ОПК-6.1. Знает принципы работы современных информационных технологий ОПК-6.2. Использует современные информационные технологии для решения задач профессиональной деятельности</p>	<p><b>Знать:</b></p> <ol style="list-style-type: none"> <li>1. основные понятия и принципы программной инженерии;</li> <li>2. основные модели жизненного цикла программного обеспечения (ПО);</li> <li>3. основные виды деятельности при разработке, развитии и сопровождении ПО;</li> <li>4. задачи и техники управления проектами по разработке и сопровождению ПО;</li> <li>5. основные методы выделения, анализа и описания требований к ПО;</li> <li>6. основные методы проектирования, анализа и описания архитектуры ПО;</li> <li>7. основные методы контроля качества ПО;</li> <li>8. основы компонентных технологий разработки ПО.</li> </ol> <p><b>Уметь:</b></p> <ol style="list-style-type: none"> <li>1. определять, описывать и анализировать требования к ПО;</li> <li>2. описывать и анализировать архитектуру ПО;</li> <li>3. выделять и применять образцы проектирования ПО;</li> <li>4. создавать и анализировать тестовые наборы для ПО;</li> <li>5. анализировать удобство использования ПО.</li> </ol> <p><b>Владеть:</b></p> <ol style="list-style-type: none"> <li>1. навыками применения</li> </ol>

		современных техник разработки и анализа ПО.
--	--	---

### 3. Место дисциплины в структуре образовательной программы

«Основы программной инженерии» представляет собой дисциплину обязательной части (Б1.О.13) направления подготовки бакалавриата 02.03.02 «Фундаментальная информатика и информационные технологии», профиль «Программная инженерия в искусственном интеллекте».

### 4. Виды учебной работы по дисциплине.

Виды учебной работы по дисциплине зафиксированы учебным планом основной профессиональной образовательной программы по указанному направлению и профилю, выражаются в академических часах. Часы контактной работы и самостоятельной работы обучающегося и часы, отводимые на процедуры контроля, могут различаться в учебных планах ОПОП по формам обучения. Объем контактной работы включает часы контактной аудиторной работы (лекции/практические занятия/ лабораторные работы), контактной внеаудиторной работы (контроль самостоятельной работы), часы контактной работы в период аттестации. Контактная работа, в том числе может проводиться посредством электронной информационно-образовательной среды университета с использованием ресурсов сети Интернет и дистанционных технологий

### 5. Содержание дисциплины, структурированное по темам (разделам)

Исходя из рамок, установленных учебным планом по трудоемкости и видам учебной работы по дисциплине, преподаватель самостоятельно выбирает тематику занятий по формам и количеству часов проведения контактной работы: лекции и иные учебные занятия, предусматривающие преимущественную передачу учебной информации преподавателем и (или) занятия семинарского типа (семинары, практические занятия, практикумы, лабораторные работы, коллоквиумы и иные аналогичные занятия), и (или) групповые консультации, и (или) индивидуальную работу обучающихся с преподавателем, в том числе индивидуальные консультации (по курсовым работам/проектам – при наличии курсовой работы/проекта по данной дисциплине в учебном плане). Рекомендуемая тематика занятий максимально полно реализуется в контактной работе с обучающимися очной формы обучения. В случае реализации образовательной программы в заочной / очно-заочной форме трудоемкость дисциплины сохраняется, однако объем учебного материала в значительной части осваивается обучающимися в форме самостоятельной работы. При этом требования к ожидаемым образовательным результатам обучающихся по данной дисциплине не зависят от формы реализации образовательной программы.

№ п/п	Наименование разделов (тем) дисциплины	Содержание разделов (тем) дисциплин
•	Предмет и основные принципы программной инженерии	Предмет и основные принципы программной инженерии.

		Жизненный цикл программного обеспечения и процессы его разработки Управление разработкой программного обеспечения
•	Анализ требований к программному обеспечению	Анализ требований к программному обеспечению Качество программного обеспечения и методы его контроля
•	Архитектура программного обеспечения и ее проектирование	Архитектура программного обеспечения и ее проектирование Удобство использования программного обеспечения Компонентные технологии разработки распределенных программных систем

### **6. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине**

**Рекомендуемая тематика учебных занятий лекционного типа** (предусматривающих преимущественную передачу учебной информации преподавателями):

№ п/п	Наименование разделов (тем) дисциплины	Тема лекции
•	Предмет и основные принципы программной инженерии	Лекция 1 «Предмет и основные принципы программной инженерии»
•	Анализ требований к программному обеспечению	Лекция 2 «Анализ требований к программному обеспечению»
•	Архитектура программного обеспечения и ее проектирование	Лекция 3 «Архитектура программного обеспечения и ее проектирование»

#### **Рекомендуемая тематика практических занятий:**

1. Предмет и основные принципы программной инженерии
2. Анализ требований к программному обеспечению
3. Архитектура программного обеспечения и ее проектирование

#### **Требования к самостоятельной работе обучающихся**

1. Работа с лекционным материалом, предусматривающая проработку конспекта лекций и учебной литературы, по всем темам из п. 6 настоящей рабочей программы.
2. Выполнение домашнего задания, предусматривающего решение задач, выполнение упражнений, выдаваемых на практических занятиях, по всем темам из п. 6 настоящей рабочей программы.

Руководствуясь положениями статьи 47 и статьи 48 Федерального закона от 29 декабря 2012 г. N 273-ФЗ «Об образовании в Российской Федерации» научно-педагогические работники и иные лица, привлекаемые университетом к реализации данной образовательной программы, пользуются предоставленными академическими

правами и свободами в части свободы преподавания, свободы от вмешательства в профессиональную деятельность; свободы выбора и использования педагогически обоснованных форм, средств, методов обучения и воспитания; права на творческую инициативу, разработку и применение авторских программ и методов обучения и воспитания в пределах реализуемой образовательной программы и отдельной дисциплины.

Исходя из рамок, установленных учебным планом по трудоемкости и видам учебной работы по дисциплине, преподаватель самостоятельно выбирает тематику занятий по формам и количеству часов проведения контактной работы: лекции и иные учебные занятия, предусматривающие преимущественную передачу учебной информации преподавателем и (или) занятия семинарского типа (семинары, практические занятия, практикумы, лабораторные работы, коллоквиумы и иные аналогичные занятия), и (или) групповые консультации, и (или) индивидуальную работу обучающихся с преподавателем, в том числе индивидуальные консультации (по курсовым работам/проектам – при наличии курсовой работы/проекта по данной дисциплине в учебном плане).

Рекомендуемая тематика занятий максимально полно реализуется в контактной работе с обучающимися очной формы обучения. В случае реализации образовательной программы в заочной / очно-заочной форме трудоемкость дисциплины сохраняется, однако объем учебного материала в значительной части осваивается обучающимися в форме самостоятельной работы. При этом требования к ожидаемым образовательным результатам обучающихся по данной дисциплине не зависят от формы реализации образовательной программы.

## **7. Методические рекомендации по видам занятий**

Лекционные занятия.

В ходе лекционных занятий обучающимся рекомендуется выполнять следующие действия. Вести конспектирование учебного материала. Обращать внимание на категории, формулировки, раскрывающие содержание тех или иных явлений и процессов, научные выводы и практические рекомендации по их применению. Задавать преподавателю уточняющие вопросы с целью уяснения теоретических положений, разрешения спорных ситуаций.

Желательно оставить в рабочих конспектах поля, на которых во внеаудиторное время можно сделать пометки из рекомендованной литературы, дополняющие материал прослушанной лекции, а также подчеркивающие особую важность тех или иных теоретических положений.

Практические и семинарские занятия.

На практических и семинарских занятиях в зависимости от темы занятия выполняется поиск информации по решению проблем, практические упражнения, контрольные работы, выработка индивидуальных или групповых решений, итоговое обсуждение с обменом знаниями, участие в круглых столах, разбор конкретных ситуаций, командная работа, представление портфолио и т.п.

Самостоятельная работа.

Самостоятельная работа осуществляется в виде изучения литературы, эмпирических данных по публикациям и конкретным ситуациям из практики, подготовке

индивидуальных работ, работа с лекционным материалом, самостоятельное изучение отдельных тем дисциплины; поиск и обзор литературы и электронных источников; чтение и изучение учебника и учебных пособий.

## 8. Фонд оценочных средств

### 8.1. Перечень компетенций с указанием этапов их формирования в процессе освоения образовательной программы в рамках учебной дисциплины

Основными этапами формирования указанных компетенций при изучении обучающимися дисциплины являются последовательное изучение содержательно связанных между собой тем учебных занятий. Изучение каждой темы предполагает овладение обучающимися необходимыми компетенциями. Результат аттестации обучающихся на различных этапах формирования компетенций показывает уровень освоения компетенций.

Контролируемые разделы (темы) дисциплины	Индекс контролируемой компетенции (или её части)	Оценочные средства по этапам формирования компетенций
		текущий контроль по дисциплине
Тема 1. Введение в тему атак на модели машинного обучения	ПК-3	Опрос
Тема 2. Схемы атак	ПК-3	Опрос
Тема 3. Атаки на системы искусственного интеллекта	ПК-3	Опрос

### 8.2. Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений и навыков и (или) опыта деятельности процессе текущего контроля

Контрольная работа № 1		
1. Заполните следующую таблицу, указав в соответствующих столбцах номера перечисленных ниже характеристик более свойственных сложным программным системам, простым программам, а также тех, которые не связаны со сложностью ПО.		
Более свойственно сложным программам	Более свойственно простым программам	Не связано с тем, сложная программа или нет
<ul style="list-style-type: none"> <li>• Исходный код, написанный на языке С</li> <li>• Необходимость разработки архитектуры до перехода к кодированию</li> <li>• Много людей вовлечено в создание системы</li> <li>• Отсутствие поддержки пользователей</li> <li>• Высокая стоимость разработки</li> </ul>		
2. Обозначим различные виды деятельности при разработке ПО следующим образом.		
<ul style="list-style-type: none"> <li>• Анализ требований</li> <li>• Проектирование</li> <li>• Кодирование</li> </ul>		



- Тестирование
- Развертывание
- Эксплуатация

Поставьте галочки в первом столбце приведенной ниже таблицы рядом с последовательностями выполнения этих видов деятельности, которые могут возникать при использовании итеративной модели жизненного цикла ПО.

	ACBCBABCDEF
	ABABABCDEF
	ABCDEF CDEF
	BACDBCDEF
	ABABCABCDEF
	ABCBADEF
	ABCABABCDEF
	ABCDEABDF

3. Заполните следующую таблицу, поместив

**в первый столбец** номера перечисленных ниже техник, используемых при разработке ПО в соответствии с Rational Unified Process (RUP),

**во второй столбец** — номера техник, используемых в Extreme Programming (XP),

**в третий столбец** — номера техник, используемых в обоих видах процессов,

**в четвертый столбец** — номера техник, не используемых ни в одном из этих двух процессов

Техники, используемые в RUP	Техники, используемые в XP	Техники, используемые и в RUP, и в XP	Техники, не используемые ни в RUP, ни в XP

- Использование наиболее простых решений
- Использование компонентной архитектуры
- Фиксация требований в виде вариантов использования
- Совместные инспекции кода
- Коллективное владение кодом
- Нацеленность на создание продукта, работоспособного в реальном окружении
- Использование моделей как основного средства коммуникации

4. Отметьте те из указанных элементов, которые входят в технику «программирование парами».

	Разработка кода выполняется парой программистов, один из которых пишет код, другой в это время отдыхает.
	Разработка кода выполняется парой программистов, один из которых пишет код, другой в это время дает ему советы и думает над улучшением кода.
	Пара программистов в течение всего проекта работает вместе.
	Объединение программистов в пары меняется от задачи к задаче.
	Разработка кода выполняется парой программистов, один из которых отвечает за все, сделанное ими, и руководит работой другого, а тот пишет код.
	Разработка кода выполняется парой программистов попеременно, то одним, то другим, они меняются местами каждые 5-10 минут.

5. Занесите номера приведенных ниже описаний проблем предметной области в первый столбец таблицы, функций ПО — во второй столбец, требований к ПО — в третий.

Проблемы предметной области	Функции ПО	Требования к ПО

- ПО анализа геологических данных должно повысить точность определения положения рудных тел и снизить ошибки определения их состава.
- Web-сайт компании должен выдерживать нагрузки до 50000 запросов в час и должен быть недоступен не более 10-ти минут в год.
- Система контроля городского движения должна существенно снизить количество аварий.
- Пользователь-аналитик должен уже в первый день работы с системой понимать, как с ее помощью можно получить все виды годовых, квартальных и месячных отчетов о финансовой деятельности компании.
- ПО контроля процесса производства должно автоматически отсеивать бракованные детали.
- Пользователь должен иметь возможность установить размер шрифта всех выделенных символов равным любому целому числу пунктов, от 5 до 72.

6.К каким из характеристик качества ПО по ISO 9126 относятся описанные ошибки? Перечислите в первом столбце каждой строки с описанием ошибки номера соответствующих характеристик качества из предыдущего задания.

	Внесение первого же изменения, которое было затребовано пользователями, привело к кардинальной перестройке архитектуры системы.
	Код системы не содержит комментариев, плохо отформатирован и труден для восприятия.
	Для выполнения наиболее часто выполняемой пользователями операции системы — получения аналитического отчета за некоторый временной отрезок — требуется нажать не менее 6-ти кнопок на 4-х диалогах.
	Для установки системы необходимо разархивировать архив, содержащий установочный пакет, в директорию, находящуюся в корне диска C:, создать в реестре системы 5 новых записей и занести туда конфигурационные данные системы, создать .dat файл в установочной директории и записать туда в определенном формате конфигурацию драйвера видеокарты.
	Система размером около 50000 строк состоит из одного класса, который имеет только один public метод, все остальные 863 метода в этом классе — private.

### 8.3. Перечень вопросов и заданий для промежуточной аттестации по дисциплине

#### Вопросы для промежуточного контроля (зачет)

##### Вопросы к зачету

1. Понятие программной системы.
2. Предмет и основные принципы программной инженерии. Свойства правильно спроектированных интерфейсов: адекватность, полнота, простота, минимальность.
3. Жизненный цикл ПО. Виды деятельности по разработке и сопровождению ПО. Основные модели жизненного цикла ПО: водопадная, итеративная, спиральная.
4. Стандарты на технологические процессы разработки и сопровождения ПО. Стандарты ISO 12207 и CMMI.
5. Примеры процессов разработки ПО. Унифицированный процесс разработки (RUP). Экстремальное программирование (XP).

6. Требования к ПО. Уровни абстракции требований: потребности, функции и детальные требования. Характеристики требований по стандарту IEEE 830.
7. Характеристики и атрибуты качества ПО по стандарту ISO 9126. Ошибки в ПО.
8. Методы контроля качества ПО. Тестирование ПО. Виды тестирования и характеристики тестов.
9. Архитектура ПО. Основные архитектурные стили: конвейер, интерактивные системы, вызов-возврат, хранилища данных, интерпретация. Анализ архитектуры с помощью метода SAAM.
10. Графические нотации, используемые при разработке ПО. Диаграммы потоков данных, диаграммы сущностей и связей, основы языка UML.
11. Образцы проектирования ПО. Образцы анализа. Идиомы. Образцы организации работ.
12. Удобство использования ПО. Принципы организации удобного интерфейса и типичные проблемы удобства ПО.
13. Понятие распределенной программной системы. Основные характеристики распределенных систем.
14. Технологии разработки Web-приложений на основе J2EE и .NET.
15. Планирование проектов по разработке ПО. Диаграммы задач и зависимостей. Выделение критического пути на графе задач.

#### 8.4 Планируемые уровни сформированности компетенций обучающихся и критерии оценивания

Уровни	Содержательное описание уровня	Основные признаки выделения уровня (этапы формирования компетенции, критерии оценки сформированности)	Пятибалльная шкала (академическая) оценка	Двухбалльная шкала, зачет	БРС, % освоения (рейтинговая оценка)
Повышенный	Творческая деятельность	<i>Включает нижестоящий уровень.</i>  Умение самостоятельно принимать решение, решать проблему/задачу теоретического и прикладного характера на основе изученных методов, приемов, технологий	отлично	зачтено	86-100
Базовый	Применение знаний и умений в	<i>Включает нижестоящий уровень.</i> Способность собирать,	хорошо		71-85

	более широких контекстах учебной и профессиональной деятельности, нежели по образцу с большей степени самостоятельности и инициативы	систематизировать, анализировать и грамотно использовать информацию из самостоятельно найденных теоретических источников и иллюстрировать ими теоретические положения или обосновывать практику применения			
Удовлетворительный (достаточный)	Репродуктивная деятельность	Изложение в пределах задач курса теоретически и практически контролируемого материала	удовлетворительно		55-70
Недостаточный	Отсутствие признаков удовлетворительного уровня		неудовлетворительно	не зачтено	Менее 55

## **9. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины.**

### **Основная литература**

- Конструирование программного обеспечения : учебное пособие / под ред. Л.Г. Гагариной. — Москва : ИНФРА-М, 2024. — 319 с. — (Высшее образование). — DOI 10.12737/1893880. - ISBN 978-5-16-017861-5. - Текст : электронный. - URL: <https://znanium.com/catalog/product/1893880> (дата обращения: 20.11.2023). – Режим доступа: по подписке.
- Проскуряков, А. В. Качество и тестирование программного обеспечения. Метрология программного обеспечения : учебное пособие / А. В. Проскуряков ; Южный федеральный университет. - Ростов-на-Дону ; Таганрог : Издательство Южного федерального университета, 2022. - 197 с. - ISBN 978-5-9275-4044-0. - Текст : электронный. - URL: <https://znanium.com/catalog/product/2057599> (дата обращения: 20.11.2023). – Режим доступа: по подписке.
- Белладжио, Д. Разработка программного обеспечения: управление изменениями : практическое руководство / Д. Белладжио, Т. Миллиган ; пер. с англ. Н. А. Мухина. — 2-е изд. - Москва : ДМК Пресс, 2023. - 385 с. - ISBN 978-5-89818-614-2. - Текст : электронный. - URL: <https://znanium.com/catalog/product/2108492> (дата обращения: 20.11.2023). – Режим доступа: по подписке..

- Буч, Г. Язык UML. Руководство пользователя : практическое руководство / Г. Буч, Дж. Рамбо, И. Якобсон ; пер. с англ. Н. Мухина. — 3-е изд. - Москва : ДМК Пресс, 2022. - 495 с. - ISBN 978-5-89818-247-2. - Текст : электронный. - URL: <https://znanium.com/catalog/product/2110008> (дата обращения: 20.11.2023). – Режим доступа: по подписке. А. Якобсон, Г. Буч, Дж. Рамбо. Унифицированный процесс разработки программного обеспечения. СПб.: Питер, 2002.

#### **Дополнительная литература**

- 1. SoftwareEngineeringBodyofKnowledge, 2005.
- Э. Дж. Брауде. Технология разработки программного обеспечения. СПб.: Питер, 2004.
- У. Ройс. Управление проектами по созданию программного обеспечения. М.: Лори, 2002.
- Д. Леффингуэлл, Д. Уидриг. Принципы работы с требованиями к программному обеспечению. Унифицированный подход. М.: Вильямс, 2002.
- В. В. Липаев. Методы обеспечения качества крупномасштабных программных средств. М., Синтег, 2003.
- М. Fowler. AnalysisPatterns: ReusableObjectModels. Addison-Wesley, 1997.
- М. Фаулер и др. Архитектура корпоративных программных приложений. М.: Вильямс, 2004.
- Б. Майер. Объектно-ориентированное программирование. Концепции разработки. М.: Русская редакция, 2004.
- Дж. Рамбо, А. Якобсон, Г. Буч. UML: Специальный справочник. СПб.: Питер, 2002.
- Г. Буч. Объектно-ориентированный анализ и проектирование с примерами приложений на C++. Второе издание. М.: Бином, СПб.: Невский диалект, 2000.
- Б. У. Перри. Jавасервлеты и JSP: сборник рецептов. М.: Кудиц-Образ, 2005.
- Х. Дейтел, П. Дейтел, С. Сантри. Технологии программирования на Java 2. Книга 3: Корпоративные системы, сервлеты, JSP, Web-сервисы. М.: Бином, 2003.
- А. Купер. Психбольница в руках пациентов. СПб.: Символ-Плюс, 2004

#### **10. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины (модуля).**

- НЭБ Национальная электронная библиотека, диссертации и прочие издания
- eLIBRARY.RU Научная электронная библиотека, книги, статьи, тезисы докладов конференций
- Гребенников Электронная библиотека ИД журналы
- ЭБС Консультант студента
- ПРОСПЕКТ ЭБС
- ЭБС ZNANIUM.COM
- ЭБС IBOOKS.RU
- РГБ Информационное обслуживание по МБА
- БЕН РАН

- Электронно-библиотечная система (ЭБС) Кантиана (<https://elib.kantiana.ru/>)

## **11. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине.**

Программное обеспечение обучения включает в себя:

- система электронного образовательного контента БФУ им. И. Канта – [www.lms.kantiana.ru](http://www.lms.kantiana.ru), обеспечивающую разработку и комплексное использование электронных образовательных ресурсов;
- серверное программное обеспечение, необходимое для функционирования сервера и связи с системой электронного обучения через Интернет;
- корпоративная платформа;
- установленное на рабочих местах обучающихся ПО: Microsoft Windows 7, Microsoft Office Standart 2010, антивирусное программное обеспечение Kaspersky Endpoint Security.
- Среда программирования Microsoft Visual Studio (любая версия);
- Qt версии 5.0 и выше

## **12. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине.**

Для проведения занятий лекционного типа, практических и семинарских занятий используются специальные помещения (учебные аудитории), оборудованные техническими средствами обучения – мультимедийной проекционной техникой. Для проведения занятий лекционного типа используются наборы демонстрационного оборудования.

Для проведения лабораторных работ, (практических занятий – при необходимости) используются специальные помещения (учебные аудитории), оснащенные специализированным лабораторным оборудованием: персональными компьютерами с возможностью выхода в интернет и с установленным программным обеспечением, заявленным в п.11.

Для проведения групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации используются специальные помещения (учебные аудитории), оборудованные специализированной мебелью (для обучающихся), меловой / маркерной доской.

Для организации самостоятельной работы обучающимся предоставляются помещения, оснащенные компьютерной техникой с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду университета.

Для обучения инвалидов и лиц с ограниченными возможностями здоровья университетом могут быть представлены специализированные средства обучения, в том числе технические средства коллективного и индивидуального пользования.

### **4.6 Программа дисциплины «Архитектура компьютеров»**

#### **1.Наименование дисциплины: «Архитектура компьютеров».**

**Цель дисциплины:** Формирование связного представления об организации современных вычислительных систем.–Аппаратура, системные программы, прикладные программы .

**2. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы**

Код компетенции	Результаты освоения образовательной программы (ИДК)	Результаты обучения по дисциплине
<p>ОПК-2. Способен применять компьютерные/суперкомпьютерные методы, современное программное обеспечение, в том числе отечественного происхождения, для решения задач профессиональной деятельности</p>	<p>ОПК-2.1. Выбирает компьютерные/суперкомпьютерные методы для решения задач профессиональной деятельности;  ОПК-2.2. Использует современное программное обеспечение, в том числе отечественного происхождения, для решения задач профессиональной деятельности</p>	<p><b>Знать:</b></p> <ul style="list-style-type: none"> <li>• принципы организации и функционирования компьютера;</li> <li>• основные концепции различных архитектур компьютеров;</li> <li>• понятия машинно-зависимых языков и основные конструкции языка Ассемблера;</li> <li>• основы функционирования систем программирования;</li> <li>• особенности архитектуры современных компьютеров и аппаратные способы повышения их быстродействия.</li> </ul> <p><b>Уметь:</b></p> <ul style="list-style-type: none"> <li>• находить и анализировать научно-техническую информацию по изучаемому предмету, в том числе в электронном виде;</li> <li>• разрабатывать программы на машинно-зависимом языке (Ассемблере);</li> <li>• отображать конструкции языков высокого уровня на язык машины;</li> <li>• организовывать связь программ, написанных на языке высокого уровня (Free Pascal) и на машинно-зависимом языке (Ассемблер).</li> </ul> <p><b>Владеть:</b></p> <ul style="list-style-type: none"> <li>• навыками анализа и</li> </ul>

		<p>понимания особенностей архитектур ЭВМ;</p> <ul style="list-style-type: none"> <li>• навыками использования систем программирования;</li> </ul> <p>навыками разработки и реализации программ на Ассемблере.</p>
--	--	---

### 3. Место дисциплины в структуре образовательной программы

«Архитектура компьютеров» представляет собой дисциплину обязательной части (Б1.О.13) направления подготовки бакалавриата 02.03.02 «Фундаментальная информатика и информационные технологии», профиль «Программная инженерия в искусственном интеллекте».

### 4. Виды учебной работы по дисциплине.

Виды учебной работы по дисциплине зафиксированы учебным планом основной профессиональной образовательной программы по указанному направлению и профилю, выражаются в академических часах. Часы контактной работы и самостоятельной работы обучающегося и часы, отводимые на процедуры контроля, могут различаться в учебных планах ОПОП по формам обучения. Объем контактной работы включает часы контактной аудиторной работы (лекции/практические занятия/ лабораторные работы), контактной внеаудиторной работы (контроль самостоятельной работы), часы контактной работы в период аттестации. Контактная работа, в том числе может проводиться посредством электронной информационно-образовательной среды университета с использованием ресурсов сети Интернет и дистанционных технологий

### 5. Содержание дисциплины, структурированное по темам (разделам)

Исходя из рамок, установленных учебным планом по трудоемкости и видам учебной работы по дисциплине, преподаватель самостоятельно выбирает тематику занятий по формам и количеству часов проведения контактной работы: лекции и иные учебные занятия, предусматривающие преимущественную передачу учебной информации преподавателем и (или) занятия семинарского типа (семинары, практические занятия, практикумы, лабораторные работы, коллоквиумы и иные аналогичные занятия), и (или) групповые консультации, и (или) индивидуальную работу обучающихся с преподавателем, в том числе индивидуальные консультации (по курсовым работам/проектам – при наличии курсовой работы/проекта по данной дисциплине в учебном плане). Рекомендуемая тематика занятий максимально полно реализуется в контактной работе с обучающимися очной формы обучения. В случае реализации образовательной программы в заочной / очно-заочной форме трудоемкость дисциплины сохраняется, однако объем учебного материала в значительной части осваивается обучающимися в форме самостоятельной работы. При этом требования к ожидаемым



образовательным результатам обучающихся по данной дисциплине не зависят от формы реализации образовательной программы.

№ п/п	Наименование разделов (тем) дисциплины	Содержание разделов (тем) дисциплин
•	<p>Архитектура ЭВМ. Схема ЭВМ. Способы представления целых и вещественных чисел. Обзор различных архитектур: трёхадресная, одноадресная, стековая, с регистрами, с модификацией адресов. Принципы Дж. фон Неймана.</p>	<p>Понятие архитектуры ЭВМ. Требования быстродействия, надежности и ограниченной стоимости при построении ЭВМ. Схема устройства ЭВМ: ЦП, ОП, внешние устройства. Назначение устройств. ЦП. Понятия машинной операции, машинной команды, системы команд процессора. ОП. Понятия ячейки ОП, адреса ячейки, объема ОП, машинного слова. Виды внешних устройств: внешняя память, устройства ввода-вывода. Отличия внешней памяти от ОП. Шина. Модели архитектуры ЭВМ с одной шиной, с несколькими шинами. Каналы ввода-вывода. Представление чисел в ЭВМ: числа без знака, со знаком. Сложение и вычитание знаковых/беззнаковых чисел. Арифметические флаги, определение значения флагов. Представление вещественных чисел с плавающей точкой. Нормализованные числа, диапазон представимости, отсутствие ассоциативности умножения. Алгоритмы выполнения сложения и умножения. Вещественные числа в ПК. Принципы Джона фон Неймана.</p>
•	<p>Язык Ассемблер. Основные конструкции языка: лексемы, предложения, выражения. Арифметические команды, переходы, логические команды и сдвиги. Работа со стеком. Процедуры. Макросредства. Многомодульные программы.</p>	<p>Архитектура ПК: схема ЦП, регистры ЦП. ОЗУ: байт, слово, двойное слово, четверённое слово. Машинные команды: переменный формат, виды операндов. Представление данных: числа, символы. Типы предложений языка MASM. Лексемы: идентификаторы, числа, строки. Директивы определения данных. Директивы EQU и =. Константные и адресные выражения. Команды языка MASM. Запись операндов. Команды пересылок. Оператор PTR. Арифметические команды. Знаковое и беззнаковое расширения чисел. Команды перехода. Действие команд перехода. Условные переходы. Проверка выполнимости условий перехода по значению флагов. Команды LOOP, JCXZ. Структура программы. Вспомогательные команды ввода-вывода. Массивы. Описание, доступ к элементу. Операторы TYPE, LENGTH, SIZE, OFFSET. Правила записи адресных выражений. Структуры. Описание типа, значение имени поля.</p>

		<p>Описание переменной-структуры. Доступ к полю структуры. Работа с массивами структур.</p> <p>Битовые команды: логические команды, сдвиги. Связь между арифметическими операциями и битовыми командами. Работа с упакованными данными.</p> <p>Записи. Описание типа, значение имени поля. Операторы WIDTH, MASK. Описание переменной, работа с полем записи.</p> <p>Стек. Описание сегмента стека. Загрузка регистров SS и ESP. Команды PUSH и POP. Работа со стеком через регистр EBP.</p> <p>Процедуры. Описание процедуры. Близкие и дальние процедуры. Команды CALL и RET. Передача параметров в регистрах. Передача параметров в стеке. Соглашения о передаче параметров; стандартные входные и выходные действия процедур.</p> <p>Строковые команды. Операнды: источник и получатель. Флаг DF, команды CLD и STD. Строковые примитивы. Префиксы повторения. Макросредства. Условное ассемблирование. Блоки повторения, макросы общего вида: распознавание формальных параметров в теле макроса, списки фактических параметров, виды фактических параметров. Директива LOCAL.</p> <p>Многомодульные программы. Описание модуля. Внешние и общие имена. Включение ассемблерной процедуры в программу на языке Free Pascal.</p>
<ul style="list-style-type: none"> <li>•</li> </ul>	<p>Элементы систем программирования. Принцип работы двухпроходного ассемблера. Схема работы компоновщика объектных модулей. Статический загрузчик и схема его работы. Понятие о динамической загрузке модулей.</p>	<p>Процесс ассемблирования. Таблицы ассемблера. Два прохода ассемблера.</p> <p>Структура объектного модуля. Работа компоновщика: объединение сегментов, редактирование внешних связей.</p> <p>Структура загрузочного модуля. Работа загрузчика. Понятие динамического связывания. Динамическое связывание в системе Multics. Явное и неявное связывание в Windows.</p>
<ul style="list-style-type: none"> <li>•</li> </ul>	<p>Особенности архитектур современных компьютеров. Многопрограммный режим работы ЭВМ. Система прерываний. Конвейер. Расслоение оперативной памяти; кеш-память.</p>	<p>Многопрограммный режим работы ЭВМ. Аппаратная поддержка многопрограммного режима.</p> <p>Понятие прерывания. Виды прерываний: внутренние и внешние, маскируемые прерывания. Аппаратная и программная реакция на прерывание.</p> <p>Конвейер: основная идея, суперскалярная архитектура. Обработка переходов: отсрочка ветвления, динамическое и статическое предсказание ветвления, спекулятивное выполнение. Связь команд по данным: RAW, WAR и WAW.</p>

	Расслоение ОП. Кэш-память: основная идея, работа с кэш-памятью.
--	---

## 6. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине

Рекомендуемая тематика учебных занятий в форме контактной работы

**Рекомендуемая тематика учебных занятий лекционного типа**  
(предусматривающих преимущественную передачу учебной информации преподавателями):

№ п/п	Наименование разделов (тем) дисциплины	Содержание разделов (тем) дисциплин
•	Архитектура ЭВМ. Схема ЭВМ. Способы представления целых и вещественных чисел. Обзор различных архитектур: трёхадресная, одноадресная, стековая, с регистрами, с модификацией адресов. Принципы Дж. фон Неймана.	<p>Понятие архитектуры ЭВМ. Требования быстродействия, надежности и ограниченной стоимости при построении ЭВМ.</p> <p>Схема устройства ЭВМ: ЦП, ОП, внешние устройства. Назначение устройств.</p> <p>ЦП. Понятия машинной операции, машинной команды, системы команд процессора.</p> <p>ОП. Понятия ячейки ОП, адреса ячейки, объема ОП, машинного слова.</p> <p>Виды внешних устройств: внешняя память, устройства ввода-вывода. Отличия внешней памяти от ОП. Шина. Модели архитектуры ЭВМ с одной шиной, с несколькими шинами. Каналы ввода-вывода.</p> <p>Представление чисел в ЭВМ: числа без знака, со знаком. Сложение и вычитание знаковых/беззнаковых чисел. Арифметические флаги, определение значения флагов.</p> <p>Представление вещественных чисел с плавающей точкой. Нормализованные числа, диапазон представимости, отсутствие ассоциативности умножения. Алгоритмы выполнения сложения и умножения. Вещественные числа в ПК.</p> <p>Принципы Джона фон Неймана.</p>
•	Язык Ассемблер. Основные конструкции языка: лексемы, предложения, выражения. Арифметические команды, переходы, логические команды и сдвиги. Работа со стекком. Процедуры. Макросредства. Многомодульные программы.	<p>Архитектура ПК: схема ЦП, регистры ЦП. ОЗУ: байт, слово, двойное слово, четверённое слово.</p> <p>Машинные команды: переменный формат, виды операндов. Представление данных: числа, символы.</p> <p>Типы предложений языка MASM. Лексемы: идентификаторы, числа, строки. Директивы определения данных. Директивы EQU и =.</p> <p>Константные и адресные выражения.</p> <p>Команды языка MASM. Запись операндов.</p> <p>Команды пересылок. Оператор PTR.</p> <p>Арифметические команды. Знаковое и беззнаковое</p>

		<p>расширения чисел.          Команды перехода. Действие команд перехода.          Условные переходы. Проверка выполнимости условий перехода по значению флагов. Команды LOOP, JCXZ.          Структура программы. Вспомогательные команды ввода-вывода.          Массивы. Описание, доступ к элементу. Операторы TYPE, LENGTH, SIZE, OFFSET. Правила записи адресных выражений.          Структуры. Описание типа, значение имени поля. Описание переменной-структуры. Доступ к полю структуры. Работа с массивами структур.          Битовые команды: логические команды, сдвиги. Связь между арифметическими операциями и битовыми командами. Работа с упакованными данными.          Записи. Описание типа, значение имени поля. Операторы WIDTH, MASK. Описание переменной, работа с полем записи.          Стек. Описание сегмента стека. Загрузка регистров SS и ESP. Команды PUSH и POP. Работа со стеком через регистр EBP.          Процедуры. Описание процедуры. Близкие и дальние процедуры. Команды CALL и RET. Передача параметров в регистрах. Передача параметров в стеке. Соглашения о передаче параметров; стандартные входные и выходные действия процедур.          Строковые команды. Операнды: источник и получатель. Флаг DF, команды CLD и STD.          Строковые примитивы. Префиксы повторения.          Макросредства. Условное ассемблирование. Блоки повторения, макросы общего вида: распознавание формальных параметров в теле макроса, списки фактических параметров, виды фактических параметров. Директива LOCAL.          Многомодульные программы. Описание модуля. Внешние и общие имена. Включение ассемблерной процедуры в программу на языке Free Pascal.</p>
<ul style="list-style-type: none"> <li>•</li> </ul>	<p>Элементы систем программирования. Принцип работы двухпроходного ассемблера. Схема работы компоновщика объектных модулей. Статический загрузчик и схема его работы. Понятие о динамической загрузке модулей.</p>	<p>Процесс ассемблирования. Таблицы ассемблера. Два прохода ассемблера.          Структура объектного модуля. Работа компоновщика: объединение сегментов, редактирование внешних связей.          Структура загрузочного модуля. Работа загрузчика. Понятие динамического связывания. Динамическое связывание в системе Multics. Явное и неявное связывание в Windows.</p>
<ul style="list-style-type: none"> <li>•</li> </ul>	<p>Особенности архитектур современных компьютеров.</p>	<p>Мультипрограммный режим работы ЭВМ. Аппаратная поддержка мультипрограммного</p>

	Мультипрограммный режим работы ЭВМ. Система прерываний. Конвейер. Расслоение оперативной памяти; кэш-память.	режима. Понятие прерывания. Виды прерываний: внутренние и внешние, маскируемые прерывания. Аппаратная и программная реакция на прерывание. Конвейер: основная идея, суперскалярная архитектура. Обработка переходов: отсрочка ветвления, динамическое и статическое предсказание ветвления, спекулятивное выполнение. Связь команд по данным: RAW, WAR и WAW. Расслоение ОП. Кэш-память: основная идея, работа с кэш-памятью.
--	--	--

### **Рекомендуемая тематика *практических* занятий:**

Тематика практических занятий совпадает с темами лекционных занятий.  
На практических занятиях решаются задачи по теме занятия.

### **Требования к самостоятельной работе обучающихся**

1. Работа с лекционным материалом, предусматривающая проработку конспекта лекций и учебной литературы, по всем темам из п. 6 настоящей рабочей программы.
2. Выполнение домашнего задания, предусматривающего решение задач, выполнение упражнений, выдаваемых на практических занятиях, по всем темам из п. 6 настоящей рабочей программы.

Руководствуясь положениями статьи 47 и статьи 48 Федерального закона от 29 декабря 2012 г. N 273-ФЗ «Об образовании в Российской Федерации» научно-педагогические работники и иные лица, привлекаемые университетом к реализации данной образовательной программы, пользуются предоставленными академическими правами и свободами в части свободы преподавания, свободы от вмешательства в профессиональную деятельность; свободы выбора и использования педагогически обоснованных форм, средств, методов обучения и воспитания; права на творческую инициативу, разработку и применение авторских программ и методов обучения и воспитания в пределах реализуемой образовательной программы и отдельной дисциплины.

Исходя из рамок, установленных учебным планом по трудоемкости и видам учебной работы по дисциплине, преподаватель самостоятельно выбирает тематику занятий по формам и количеству часов проведения контактной работы: лекции и иные учебные занятия, предусматривающие преимущественную передачу учебной информации преподавателем и (или) занятия семинарского типа (семинары, практические занятия, практикумы, лабораторные работы, коллоквиумы и иные аналогичные занятия), и (или) групповые консультации, и (или) индивидуальную работу обучающихся с преподавателем, в том числе индивидуальные консультации (по курсовым работам/проектам – при наличии курсовой работы/проекта по данной дисциплине в учебном плане).

Рекомендуемая тематика занятий максимально полно реализуется в контактной работе с обучающимися очной формы обучения. В случае реализации образовательной программы в заочной / очно-заочной форме трудоемкость дисциплины сохраняется, однако объем учебного материала в значительной части осваивается обучающимися в форме самостоятельной работы. При этом требования к ожидаемым образовательным результатам обучающихся по данной дисциплине не зависят от формы реализации образовательной программы.

## **7. Методические рекомендации по видам занятий**

Лекционные занятия.

В ходе лекционных занятий обучающимся рекомендуется выполнять следующие действия. Вести конспектирование учебного материала. Обращать внимание на категории, формулировки, раскрывающие содержание тех или иных явлений и процессов, научные выводы и практические рекомендации по их применению. Задавать преподавателю уточняющие вопросы с целью уяснения теоретических положений, разрешения спорных ситуаций.

Желательно оставить в рабочих конспектах поля, на которых во внеаудиторное время можно сделать пометки из рекомендованной литературы, дополняющие материал прослушанной лекции, а также подчеркивающие особую важность тех или иных теоретических положений.

Практические и семинарские занятия.

На практических и семинарских занятиях в зависимости от темы занятия выполняется поиск информации по решению проблем, практические упражнения, контрольные работы, выработка индивидуальных или групповых решений, итоговое обсуждение с обменом знаниями, участие в круглых столах, разбор конкретных ситуаций, командная работа, представление портфолио и т.п.

Самостоятельная работа.

Самостоятельная работа осуществляется в виде изучения литературы, эмпирических данных по публикациям и конкретных ситуаций из практики, подготовке индивидуальных работ, работа с лекционным материалом, самостоятельное изучение отдельных тем дисциплины; поиск и обзор литературы и электронных источников; чтение и изучение учебника и учебных пособий.

## **8. Фонд оценочных средств**

### **8.1. Перечень компетенций с указанием этапов их формирования в процессе освоения образовательной программы в рамках учебной дисциплины**

Основными этапами формирования указанных компетенций при изучении обучающимися дисциплины являются последовательное изучение содержательно связанных между собой тем учебных занятий. Изучение каждой темы предполагает овладение обучающимися необходимыми компетенциями. Результат аттестации обучающихся на различных этапах формирования компетенций показывает уровень освоения компетенций.

Контролируемые разделы (темы) дисциплины	Индекс контролируемой компетенции (или её части)	Оценочные средства по этапам формирования компетенций
		текущий контроль по дисциплине
Архитектура ЭВМ. Схема ЭВМ. Способы представления целых и вещественных чисел. Обзор различных архитектур: трёхадресная, одноадресная, стековая, с регистрами, с модификацией адресов. Принципы Дж. фон Неймана.	ОПК-2	выполнение заданий на практических (семинарских) занятиях
Язык Ассемблер. Основные конструкции языка: лексемы, предложения, выражения. Арифметические команды, переходы, логические команды и сдвиги. Работа со стеком. Процедуры. Макросредства. Многомодульные программы.	ОПК-2	выполнение заданий на практических (семинарских) занятиях
Элементы систем программирования. Принцип работы двухпроходного ассемблера. Схема работы компоновщика объектных модулей. Статический загрузчик и схема его работы. Понятие о динамической загрузке модулей.	ОПК-2	выполнение заданий на практических (семинарских) занятиях
Особенности архитектур современных компьютеров. Мультипрограммный режим работы ЭВМ. Система прерываний. Конвейер. Расслоение оперативной памяти; кэш-память.	ОПК-2	выполнение заданий на практических (семинарских) занятиях

## 8.2. Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений и навыков и (или) опыта деятельности процессе текущего контроля

В качестве оценочных средств текущего контроля успеваемости предусмотрены:

выполнение заданий на практических (семинарских) занятиях

Примеры заданий для семинарских занятий

1. Написать программу для учебной машины УМ-3. Эта программа должна сначала вводить целочисленный массив X длины 100, затем печатать число S, равное количеству

одновременно отрицательных и кратных трём элементов массива X. При записи кодов операций использовать мнемонические обозначения.

2. Выписать вид внутреннего машинного представления целой переменной X (в двоичном или шестнадцатеричном виде):

```
X    dw    -1023
```

3. Написать полную программу на Ассемблере, которая вводит (по **inint**) целое знаковое число X в формате **dd** и выводит (по **outword**) число значащих *чётных* цифр (т.е. '0', '2', '4', '6', '8') в десятичной записи значения числа X. Цифра является значащей, если её удаление меняет величину числа.

4. Написать полную программу на Ассемблере, которая вводит текст до точки и выводит (по **outword**) сумму *нечётных цифр*, расположенных в этом тексте после первой "\*". Считать, что таких цифр не более MaxLongword.

5. Пусть на Паскале дано описание типа массива:

```
const n=5000; type MAS=array[1..n,1..n] of char;
```

Написать на Ассемблере *процедуру* со стандартными соглашениями о связях, которая получает в качестве параметров адрес массива типа MAS и длину массива N. Процедура должна все символы-*цифры* на главной диагонали этой матрицы заменить на символы "+".

Привести пример вызова этой процедуры.

6. Написать макрос с заголовком

```
First_1 macro X
```

параметр которого может быть только форматов m8, m16 или m32. Макрос переставляет в начало все 1 (биты со значением единица)

во внутреннем машинном представлении X, например, для X = 10101010b необходимо получить X = 11110000b. Макроопределение должно настраиваться на тип параметра..

7. Написать на Ассемблере неголовной модуль, содержащий описание процедуры без параметров с именем Del\_3. Эта процедура должна уменьшать в три раза значение знаковой переменной размером в слово (**dw**) с именем Perem3, описанной в каком-то другом модуле.

### 8.3. Перечень вопросов и заданий для промежуточной аттестации по дисциплине

#### Вопросы для промежуточного контроля (экзамен)

- Понятие архитектуры ЭВМ. Требования быстродействия, надежности и ограниченной стоимости при построении ЭВМ.
- Схема устройства ЭВМ: ЦП, ОП, внешние устройства. Назначение устройств.
- ЦП. Понятия машинной операции, машинной команды, системы команд процессора.
- ОП. Понятия ячейки ОП, адреса ячейки, объема ОП, машинного слова.
- Виды внешних устройств: внешняя память, устройства ввода-вывода. Отличия внешней памяти от ОП. Шина. Модели архитектуры ЭВМ с одной шиной, с несколькими шинами. Каналы ввода-вывода.
- Представление чисел в ЭВМ: числа без знака, со знаком. Сложение и вычитание знаковых/беззнаковых чисел. Арифметические флаги, определение значения флагов.



- Представление вещественных чисел с плавающей точкой. Нормализованные числа, диапазон представимости, отсутствие ассоциативности умножения. Алгоритмы выполнения сложения и умножения. Вещественные числа в ПК.
- Принципы Джона фон Неймана.
- Трехадресная учебная машина. Устройство ЦП. Такт работы процессора. Выполнение арифметических команд и команд перехода.
- Двухадресная учебная машина. Преимущества по сравнению с УМ-3. Исключение третьего адреса из команд. Выполнение команд условного перехода.
- Одноадресная учебная машина. Преимущества по сравнению с УМ-3 и УМ-2. Выполнение арифметических команд.
- Учебная машина с переменным форматом команд. Достоинства, недостатки.
- Стековая учебная машина. Программирование формул в УМ-С.
- Учебная машина с регистрами. Структура ЦП. Длинные и короткие машинные команды. Преимущества использования регистров.
- Учебная машина с модификацией адресов. Понятие самомодифицирующейся программы. Модификация адресов. Преимущества.
- Автоматизация программирования в машинных кодах.
- Архитектура ПК: схема ЦП, регистры ЦП. ОЗУ: байт, слово, двойное слово, учетверённое слово. Машинные команды: переменный формат, виды операндов. Представление данных: числа, символы.
- Типы предложений языка MASM. Лексемы: идентификаторы, числа, строки. Директивы определения данных. Директивы EQU и =. Константные и адресные выражения.
- Команды языка MASM. Запись операндов. Команды пересылок. Оператор PTR.
- Арифметические команды. Знаковое и беззнаковое расширения чисел.
- Команды перехода. Действие команд перехода. Условные переходы. Проверка выполнимости условий перехода по значению флагов. Команды LOOP, JCXZ.
- Структура программы. Вспомогательные команды ввода-вывода.
- Массивы. Описание, доступ к элементу. Операторы TYPE, LENGTH, SIZE, OFFSET. Правила записи адресных выражений.
- Структуры. Описание типа, значение имени поля. Описание переменной-структуры. Доступ к полю структуры. Работа с массивами структур.
- Битовые команды: логические команды, сдвиги. Связь между арифметическими операциями и битовыми командами. Работа с упакованными данными.
- Записи. Описание типа, значение имени поля. Операторы WIDTH, MASK. Описание переменной, работа с полем записи.
- Стек. Описание сегмента стека. Загрузка регистров SS и ESP. Команды PUSH и POP. Работа со стеком через регистр EBP.
- Процедуры. Описание процедуры. Близкие и дальние процедуры. Команды CALL и RET. Передача параметров в регистрах. Передача параметров в стеке. Соглашения о передаче параметров; стандартные входные и выходные действия процедур.
- Строковые команды. Операнды: источник и получатель. Флаг DF, команды CLD и STD. Строковые примитивы. Префиксы повторения.

- Макросредства. Условное ассемблирование. Блоки повторения, макросы общего вида: распознавание формальных параметров в теле макроса, списки фактических параметров, виды фактических параметров. Директива LOCAL.
- Многомодульные программы. Описание модуля. Внешние и общие имена. Включение ассемблерной процедуры в программу на языке Free Pascal.
- Процесс ассемблирования. Таблицы ассемблера. Два прохода ассемблера.
- Структура объектного модуля. Работа компоновщика: объединение сегментов, редактирование внешних связей.
- Структура загрузочного модуля. Работа загрузчика.
- Понятие динамического связывания. Динамическое связывание в системе Multics. Явное и неявное связывание в Windows.
- Мультипрограммный режим работы ЭВМ. Аппаратная поддержка мультипрограммного режима.
- Понятие прерывания. Виды прерываний: внутренние и внешние, маскируемые прерывания. Аппаратная и программная реакция на прерывание.
- Конвейер: основная идея, суперскалярная архитектура. Обработка переходов: отсрочка ветвления, динамическое и статическое предсказание ветвления, спекулятивное выполнение. Связь команд по данным: RAW, WAR и WAW.
- Расслоение ОП. Кэш-память: основная идея, работа с кэш-памятью.

Пример варианта письменного экзамена.

1. Что такое такт работы процессора в машине Фон-Неймана? Перечислить основные этапы такта работы.

**Ответ:**

Такт –

Этапы –

2. Что будет напечатано в результате выполнения последовательности команд

<pre> mov  AX, -254 shl  AH, 1 imul AH shl  AL, 1 outi AX </pre>	<b>Ответ:</b>
--	---------------

3. Написать полную программу на Ассемблере, которая вводит (по макрокомандам **inint**) два числа со знаком формата **dd** и печатает частное от деления меньшего числа на большее. При равенстве чисел вывести "Числа равны", вместо деления на ноль вывести "Деление на ноль". (Привести ответ на обратной стороне листа.)

4. Вычеркнуть синтаксически неверные команды

mul AL	jmp EAX	cmp CF,1	mov EBX,'A'	add 5000[ESI],0	mov EAX[EBX],5
5. Выписать справа фрагмент на Ассемблере (не более 4-х команд), который реализует операцию над флагом переноса CF OF := not OF Можно использовать регистр AL.					Ответ:

6. Пусть есть описания на языке Free Pascal

```
type MAS=array[1..k] of char;
function LastDigit(var z:MAS; n:longword):char;
```

Описать на Ассемблере функцию LastDigit, которая возвращает последнюю цифру (символ из диапазона '0'..'9') этого массива или значение 0FFh, если в массиве нет символов-цифр. Функция должна удовлетворять стандартным соглашениям о связях. Привести пример вызова этой функции, сделав на Ассемблере необходимые описания. (Дать ответ на обратной стороне листа.)

<p>7. Написать макроопределение с заголовком M11 macro v где v – имя переменной. Если параметр v имеет формат m8 или m16, то надо реализовать присваивание v:= 11-v иначе должно получаться пустое макрорасширение. Макрорасширение должно содержать не более двух команд. Диагностику о возможных ошибках в переданных параметрах не выводить.</p>	<p>Ответ:</p>
---	---------------

<p>8. Указать значения регистра CL (в виде знакового и беззнакового десятичных чисел) и флагов CF, OF, SF, ZF после выполнения команд mov CL,-70 add CL,183</p>	<p>Ответ: CL = (зн.), CL= (беззн.) CF = OF = SF = ZF =</p>
---	--

**8.4. Планируемые уровни сформированности компетенций обучающихся и критерии оценивания**

Уровни	Содержательное описание уровня	Основные признаки выделения уровня (этапы формирования компетенции, критерии оценки сформированности)	Пятибалльная шкала (академическая) оценка	Двухбалльная шкала, зачет	БРС, % освоения (рейтинговая оценка)
Повышенной	Творческая деятельность	<p><i>Включает нижестоящий уровень.</i></p> <p>Умение самостоятельно принимать решение, решать проблему/задачу теоретического и прикладного характера на основе изученных методов, приемов, технологий</p>	отлично	зачтено	86-100

Базовый	Применение знаний и умений в более широких контекстах учебной и профессиональной деятельности, нежели по образцу с большей степени самостоятельности и инициативы	<i>Включает нижестоящий уровень.</i> Способность собирать, систематизировать, анализировать и грамотно использовать информацию из самостоятельно найденных теоретических источников и иллюстрировать ими теоретические положения или обосновывать практику применения	хорошо		71-85
Удовлетворительный (достаточный)	Репродуктивная деятельность	Изложение в пределах задач курса теоретически и практически контролируемого материала	удовлетворительно		55-70
Недостаточный	Отсутствие признаков удовлетворительного уровня		неудовлетворительно	не зачтено	Менее 55

## 9. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины.

### Основная литература

1. Лисицин, Д. В. Программирование на языке ассемблера : учебное пособие / Д. В. Лисицин. - Новосибирск : Изд-во НГТУ, 2018. - 100 с. - ISBN 978-5-7782-3679-0. - Текст : электронный. - URL: <https://znanium.com/catalog/product/1866916> (дата обращения: 20.11.2023). – Режим доступа: по подписке.
2. Назаров, С. В. Операционные среды, системы и оболочки. Основы структурной и функциональной организации : Учеб. пособие / С. В. Назаров. - Москва : КУДИЦ-ПРЕСС, 2007. - 504 с.: ил. - ISBN 978-5-91136-036-8. - Текст : электронный. - URL: <https://znanium.com/catalog/product/369379> (дата обращения: 20.11.2023). – Режим доступа: по подписке.
3. Степина, В. В. Архитектура ЭВМ и вычислительные системы : учебник / В.В. Степина. — Москва : КУРС : ИНФРА-М, 2023. — 384 с. — (Среднее профессиональное образование). - ISBN 978-5-906923-07-3. - Текст : электронный. - URL: <https://znanium.com/catalog/product/1916205> (дата обращения: 20.11.2023). – Режим доступа: по подписке.

### **Дополнительная литература**

1. Баула В. Г., Бордаченкова Е. А. Задачи письменного экзамена по курсу "Архитектура ЭВМ и язык ассемблера" М., МАКС Пресс, 2019.
2. Королев Л.Н. Архитектура ЭВМ. М., Научный мир, 2005.
3. Хамахер К., Вранешич З., Заки С. Организация ЭВМ. СПб., Питер, 2003.
4. Корнеев В.В., Киселев А.В. Современные микропроцессоры. СПб., БХВ-Петербург, 2003.
5. Ю-Чжень Лю, Гибсон Г. Микропроцессоры семейства 8086/8088. М., Радио и связь, 1987.
6. Юров В.И. Assembler. Специальный справочник. СПб., Питер, 2005.

### **10. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины (модуля).**

- НЭБ Национальная электронная библиотека, диссертации и прочие издания
- eLIBRARY.RU Научная электронная библиотека, книги, статьи, тезисы докладов конференций
- Гребенников Электронная библиотека ИД журналы
- ЭБС Консультант студента
- ПРОСПЕКТ ЭБС
- ЭБС ZNANIUM.COM
- ЭБС IBOOKS.RU
- РГБ Информационное обслуживание по МБА
- БЕН РАН
- Электронно-библиотечная система (ЭБС) Кантиана (<https://elib.kantiana.ru/>)

### **11. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине.**

Программное обеспечение обучения включает в себя:

- система электронного образовательного контента БФУ им. И. Канта – [www.lms.kantiana.ru](http://www.lms.kantiana.ru), обеспечивающую разработку и комплексное использование электронных образовательных ресурсов;
- серверное программное обеспечение, необходимое для функционирования сервера и связи с системой электронного обучения через Интернет;
- корпоративная платформа;
- установленное на рабочих местах обучающихся ПО: Microsoft Windows 7, Microsoft Office Standart 2010, антивирусное программное обеспечение Kaspersky Endpoint Security.
- Среда программирования Microsoft Visual Studio (любая версия);
- Qt версии 5.0 и выше

## 12. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине.

Для проведения занятий лекционного типа, практических и семинарских занятий используются специальные помещения (учебные аудитории), оборудованные техническими средствами обучения – мультимедийной проекционной техникой. Для проведения занятий лекционного типа используются наборы демонстрационного оборудования.

Для проведения лабораторных работ, (практических занятий – при необходимости) используются специальные помещения (учебные аудитории), оснащенные специализированным лабораторным оборудованием: персональными компьютерами с возможностью выхода в интернет и с установленным программным обеспечением, заявленным в п.11.

Для проведения групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации используются специальные помещения (учебные аудитории), оборудованные специализированной мебелью (для обучающихся), меловой / маркерной доской.

Для организации самостоятельной работы обучающимся предоставляются помещения, оснащенные компьютерной техникой с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду университета.

Для обучения инвалидов и лиц с ограниченными возможностями здоровья университетом могут быть представлены специализированные средства обучения, в том числе технические средства коллективного и индивидуального пользования.

### 4.7 Программа дисциплины «Язык Python»

#### 1.Наименование дисциплины: «Язык Python».

**Цель** дисциплины: целью освоения дисциплины «Язык Python» освоение методов разработки современных программных и информационных решений на языке программирования Python.

#### 2. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы

Код компетенции	Результаты освоения образовательной программы (ИДК)	Результаты обучения по дисциплине
ПК-2. Способен разрабатывать и тестировать программные компоненты решения задач в системах искусственного интеллекта	ПК-2.1. Настраивает программное обеспечение и участвует в разработке программных компонентов систем искусственного интеллекта ПК-2.2. Разрабатывает приложения систем искусственного интеллекта	<ul style="list-style-type: none"><li>• <b>Знать</b> основные принципы разработки программ с применение изучаемых языков.</li><li>• <b>Уметь</b> создавать современные программные и информационные решения.</li><li>• <b>Владеть</b> практическими навыками</li></ul>

		программирования на основе изучаемых языков
--	--	--

### 3. Место дисциплины в структуре образовательной программы

«Язык Python» представляет собой дисциплину обязательной части (Б1.О.13) направления подготовки бакалавриата 02.03.02 «Фундаментальная информатика и информационные технологии», профиль «Программная инженерия в искусственном интеллекте».

### 4. Виды учебной работы по дисциплине.

Виды учебной работы по дисциплине зафиксированы учебным планом основной профессиональной образовательной программы по указанному направлению и профилю, выражаются в академических часах. Часы контактной работы и самостоятельной работы обучающегося и часы, отводимые на процедуры контроля, могут различаться в учебных планах ОПОП по формам обучения. Объем контактной работы включает часы контактной аудиторной работы (лекции/практические занятия/ лабораторные работы), контактной внеаудиторной работы (контроль самостоятельной работы), часы контактной работы в период аттестации. Контактная работа, в том числе может проводиться посредством электронной информационно-образовательной среды университета с использованием ресурсов сети Интернет и дистанционных технологий

### 5. Содержание дисциплины, структурированное по темам (разделам)

Исходя из рамок, установленных учебным планом по трудоемкости и видам учебной работы по дисциплине, преподаватель самостоятельно выбирает тематику занятий по формам и количеству часов проведения контактной работы: лекции и иные учебные занятия, предусматривающие преимущественную передачу учебной информации преподавателем и (или) занятия семинарского типа (семинары, практические занятия, практикумы, лабораторные работы, коллоквиумы и иные аналогичные занятия), и (или) групповые консультации, и (или) индивидуальную работу обучающихся с преподавателем, в том числе индивидуальные консультации (по курсовым работам/проектам – при наличии курсовой работы/проекта по данной дисциплине в учебном плане). Рекомендуемая тематика занятий максимально полно реализуется в контактной работе с обучающимися очной формы обучения. В случае реализации образовательной программы в заочной / очно-заочной форме трудоемкость дисциплины сохраняется, однако объем учебного материала в значительной части осваивается обучающимися в форме самостоятельной работы. При этом требования к ожидаемым образовательным результатам обучающихся по данной дисциплине не зависят от формы реализации образовательной программы.

№	Наименование раздела	Содержание раздела
1	Язык Python. Базовые типы данных.	Особенности языка Python. IDE. Интерактивный и пакетный режим работы языка Python. Переменные. Int, float, str, list. Коллективные типы данных. List, Tuple, Set, Dict. Стек и очередь. List и Set comprehension. Вложение

		структур данных Арифметические операции. Ввод и вывод.
2	Функции. Lamda-выпажения. Модули.	Определение функции. Передача параметров и возврат значений. Локальные, нелокальные и глобальные переменные. Рекурсия. Функция как переменная и функции высших порядков. Замыкания. Docstring. Lambda-выражения. Стандартные библиотеки. Подключение модулей. Создание своих модулей. Иерархическая структуризация модулей.
3	Классы, ООП.	Объектно ориентированное программирование. Классы. Инстансы. Переопределение операторов. Наследование.
4	Стандартные библиотеки языка Python.	Стандартные библиотеки языка Python. os, Glob,sys, re, math, random, statistics, urllib, datetime, timeit, doctest, unittest, template, zipfile,array
5	Реализация GUI в языке Python.	Базовые представления о GUI. Обзор основных библиотек для работы с GUI. TKinter
6	Библиотеки Python для работы с данными, математикой и ИИ	Библиотеки Numpy, SciPy, Matplotlib, SymPy, Pandas, SkLearn. Назначение, принципы работы и варианты использования

## 6. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине

**Рекомендуемая тематика учебных занятий лекционного типа** (предусматривающих преимущественную передачу учебной информации преподавателями):

№	Наименование раздела	Темы лекций
1	Язык Python. Базовые типы данных.	Лекция 1 . Особенности языка Python. IDE. Интерактивный и пакетный режим работы языка Python. Лекция 2 . Переменные. Int, float, str, list. Коллективные типы данных. List, Tuple, Set, Dict. Лекция 3 . Стек и очередь. List и Set comprehension. Вложение структур данных Арифметические операции. Ввод и вывод.
2	Функции. Lamda-выпажения. Модули.	Лекция 4 . Определение функции. Передача параметров и возврат значений. Локальные, нелокальные и глобальные переменные. Рекурсия. Функция как переменная и функции высших порядков. Лекция 5 . Замыкания. Docstring. Lambda-выражения. Лекция 6 . Стандартные библиотеки. Подключение модулей. Создание своих модулей. Иерархическая структуризация модулей.
3	Классы, ООП.	Лекция 7-8. Объектно ориентированное программирование. Классы. Инстансы. Переопределение



		операторов. Наследование.
4	Стандартные библиотеки языка Python.	Лекция 9. Стандартные библиотеки языка Python.
5	Реализация GUI в языке Python.	Лекция 10. Базовые представления о GUI. Обзор основных библиотек для работы с GUI. Лекция 11-14. TKinter
6	Библиотеки Python для работы с данными, математикой и ИИ	Лекция 15-20 . Библиотеки Numpy, SciPy, Matplotlib, SymPy, Pandas, SkLearn. Назначение, принципы работы и варианты использования

### Рекомендуемая тематика лабораторных занятий:

№ п/п	Наименование темы	Содержание темы
1	Язык Python. Базовые типы данных.	Написание программы демонстрирующей работу со сложными структурами данных
2	Функции. Lamda-выпажения. Модули.	Написание программы демонстрирующей работу с функциями и/или модулями
3	Классы, ООП.	Написание программы демонстрирующей работу с классами
4	Стандартные библиотеки языка Python.	Написание программы демонстрирующей работу с файловой системой и работу с исключениями
5	Реализация GUI в языке Python.	Написание программы демонстрирующей работу с GUI на основе TKinter
6	Библиотеки Python для работы с данными, математикой и ИИ	Решение задач по обработке данных с использованием специализированных библиотек.  Визуализация задач по обработке данных с использованием специализированных библиотек

### Требования к самостоятельной работе обучающихся

1. Работа с лекционным материалом, предусматривающая проработку конспекта лекций и учебной литературы, по всем темам из п. 6 настоящей рабочей программы.
2. Выполнение домашнего задания, предусматривающего решение задач, выполнение упражнений, выдаваемых на лабораторных занятиях, по всем темам из п. 6 настоящей рабочей программы.

Руководствуясь положениями статьи 47 и статьи 48 Федерального закона от 29 декабря 2012 г. N 273-ФЗ «Об образовании в Российской Федерации» научно-педагогические работники и иные лица, привлекаемые университетом к реализации данной образовательной программы, пользуются предоставленными академическими правами и свободами в части свободы преподавания, свободы от вмешательства в профессиональную деятельность; свободы выбора и использования педагогически обоснованных форм, средств, методов обучения и воспитания; права на творческую инициативу, разработку и применение авторских программ и методов обучения и воспитания в пределах реализуемой образовательной программы и отдельной дисциплины.

Исходя из рамок, установленных учебным планом по трудоемкости и видам учебной работы по дисциплине, преподаватель самостоятельно выбирает тематику занятий по формам и количеству часов проведения контактной работы: лекции и иные учебные занятия, предусматривающие преимущественную передачу учебной информации преподавателем и (или) занятия семинарского типа (семинары, практические занятия, практикумы, лабораторные работы, коллоквиумы и иные аналогичные занятия), и (или) групповые консультации, и (или) индивидуальную работу обучающихся с преподавателем, в том числе индивидуальные консультации (по курсовым работам/проектам – при наличии курсовой работы/проекта по данной дисциплине в учебном плане).

Рекомендуемая тематика занятий максимально полно реализуется в контактной работе с обучающимися очной формы обучения. В случае реализации образовательной программы в заочной / очно-заочной форме трудоемкость дисциплины сохраняется, однако объем учебного материала в значительной части осваивается обучающимися в форме самостоятельной работы. При этом требования к ожидаемым образовательным результатам обучающихся по данной дисциплине не зависят от формы реализации образовательной программы.

## **7. Методические рекомендации по видам занятий**

Лекционные занятия.

В ходе лекционных занятий обучающимся рекомендуется выполнять следующие действия. Вести конспектирование учебного материала. Обращать внимание на категории, формулировки, раскрывающие содержание тех или иных явлений и процессов, научные выводы и практические рекомендации по их применению. Задавать преподавателю уточняющие вопросы с целью уяснения теоретических положений, разрешения спорных ситуаций.

Желательно оставить в рабочих конспектах поля, на которых во внеаудиторное время можно сделать пометки из рекомендованной литературы, дополняющие материал прослушанной лекции, а также подчеркивающие особую важность тех или иных теоретических положений.

Лабораторные занятия.

На лабораторных занятиях в зависимости от темы занятия выполняется поиск информации по решению проблем, практические упражнения, контрольные работы, выработка индивидуальных или групповых решений, итоговое обсуждение с обменом

знаниями, участие в круглых столах, разбор конкретных ситуаций, командная работа, представление портфолио и т.п.

Самостоятельная работа.

Самостоятельная работа осуществляется в виде изучения литературы, эмпирических данных по публикациям и конкретных ситуаций из практики, подготовке индивидуальных работ, работа с лекционным материалом, самостоятельное изучение отдельных тем дисциплины; поиск и обзор литературы и электронных источников; чтение и изучение учебника и учебных пособий.

## 8. Фонд оценочных средств

### 8.1. Перечень компетенций с указанием этапов их формирования в процессе освоения образовательной программы в рамках учебной дисциплины

Основными этапами формирования указанных компетенций при изучении обучающимися дисциплины являются последовательное изучение содержательно связанных между собой тем учебных занятий. Изучение каждой темы предполагает овладение обучающимися необходимыми компетенциями. Результат аттестации обучающихся на различных этапах формирования компетенций показывает уровень освоения компетенций.

Контролируемые разделы (темы) дисциплины	Индекс контролируемой компетенции (или её части)	Оценочные средства по этапам формирования компетенций
		текущий контроль по дисциплине
Язык Python. Базовые типы данных.	ПК-2	Опрос, выполнение лабораторных работ.
Функции. Lambda-выпажения. Модули.	ПК-2	Опрос, выполнение лабораторных работ.
Классы, ООП.	ПК-2	Опрос, выполнение лабораторных работ.
Стандартные библиотеки языка Python.	ПК-2	Опрос, выполнение лабораторных работ.
Реализация GUI в языке Python.	ПК-2	Опрос, выполнение лабораторных работ.
Библиотеки Python для работы с данными, математикой и ИИ	ПК-2	Опрос, выполнение лабораторных работ.

### 8.2. Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений и навыков и (или) опыта деятельности в процессе текущего контроля

**Примеры вопросов для устного опроса:**

- Язык Python. Особенности реализации
- Базовые типы данных языка Python. Отличия в реализации.
- Условия и циклы
- Функции. Lamda-выражения, условия применения.
- Структуры данных
- Классы, ООП.
- Исключения и их обработка
- Стандартные библиотеки языка Python. Отличия от пользовательских библиотек.

### **Типовая лабораторная работа:**

#### **Лабораторная работа №1**

**Написание программы демонстрирующей работу с функциями.**

**Цель работы:** освоить основные навыки программирования с использованием функций Python.

#### **Задания:**

Написать программу используя функции и необходимые технологии, в рамках двух из предложенных задач.

### **8.3. Перечень вопросов и заданий для промежуточной аттестации по дисциплине**

#### **Вопросы для промежуточного контроля (экзамена):**

- Язык Python
- Базовые типы данных языка Python
- Условия и циклы
- Функции. Lamda-выражения
- Структуры данных
- Модули
- Классы, ООП.
- Исключения и их обработка
- Стандартные библиотеки языка Python
- Библиотеки для работы с математикой
- Реализация GUI в языке Python
- Работа с графическими файлами
  - Работа с компьютерными сетями

#### 8.4. Планируемые уровни сформированности компетенций обучающихся и критерии оценивания

Уровни	Содержательное описание уровня	Основные признаки выделения уровня (этапы формирования компетенции, критерии оценки сформированности)	Пятибалльная шкала (академическая) оценка	Двухбалльная шкала, зачет	БРС, % освоения (рейтинговая оценка)
Повышенный	Творческая деятельность	<i>Включает нижестоящий уровень.</i> Умение самостоятельно принимать решение, решать проблему/задачу теоретического и прикладного характера на основе изученных методов, приемов, технологий	отлично	зачтено	86-100
Базовый	Применение знаний и умений в более широких контекстах учебной и профессиональной деятельности, нежели по образцу с большей степени самостоятельности и инициативы	<i>Включает нижестоящий уровень.</i> Способность собирать, систематизировать, анализировать и грамотно использовать информацию из самостоятельно найденных теоретических источников и иллюстрировать ими теоретические положения или обосновывать практику применения	хорошо		71-85
Удовлетворительный (достаточный)	Репродуктивная деятельность	Изложение в пределах задач курса теоретически и практически контролируемого материала	удовлетворительно		55-70
Недостаточный	Отсутствие признаков удовлетворительного уровня		неудовлетворительно	не зачтено	Менее 55

#### 9. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины.

## Основная литература

1. Шелудько, В. М. Основы программирования на языке высокого уровня Python : учебное пособие / В. М. Шелудько ; Южный федеральный университет. - Ростов-на-Дону ; Таганрог : Издательство Южного федерального университета, 2017. - 146 с. - ISBN 978-5-9275-2649-9. - Текст : электронный. - URL: <https://znanium.com/catalog/product/1021662> (дата обращения: 03.04.2023). – Режим доступа: по подписке.
2. Рамальо, Л. Python. К вершинам мастерства / Лучано Рамальо ; пер. с англ. А.А. Слинкина. - Москва : ДМК Пресс, 2016. - 768 с. - ISBN 978-5-97060-384-0. - Текст : электронный. - URL: <https://znanium.com/catalog/product/1028052> (дата обращения: 20.11.2023). – Режим доступа: по подписке.
3. Саммерфильд, М. Python на практике. Создание качественных программ с использованием параллелизма, библиотек и паттернов : практическое пособие / М. Саммерфильд ; пер. с англ. А. А. Слинкина. - 2-е изд. - Москва : ДМК Пресс, 2023. - 340 с. - ISBN 978-5-89818-322-6. - Текст : электронный. - URL: <https://znanium.com/catalog/product/2102609> (дата обращения: 20.11.2023). – Режим доступа: по подписке.

## Дополнительная литература

- Гуриков, С. Р. Основы алгоритмизации и программирования на Python : учебное пособие / С.Р. Гуриков. — Москва : ИНФРА-М, 2022. — 343 с. — (Высшее образование: Бакалавриат). - ISBN 978-5-16-017142-5. - Текст : электронный. - URL: <https://znanium.com/catalog/product/1356003> (дата обращения: 23.03.2023). – Режим доступа: по подписке.

## 10. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины (модуля).

- НЭБ Национальная электронная библиотека, диссертации и прочие издания
- eLIBRARY.RU Научная электронная библиотека, книги, статьи, тезисы докладов конференций
- Гребенников Электронная библиотека ИД журналы
- ЭБС Консультант студента
- ПРОСПЕКТ ЭБС
- ЭБС ZNANIUM.COM
- ЭБС IBOOKS.RU
- РГБ Информационное обслуживание по МБА
- БЕН РАН
- Электронно-библиотечная система (ЭБС) Кантиана (<https://elib.kantiana.ru/>)

## **11. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине.**

Программное обеспечение обучения включает в себя:

- система электронного образовательного контента БФУ им. И. Канта – [www.lms.kantiana.ru](http://www.lms.kantiana.ru), обеспечивающую разработку и комплексное использование электронных образовательных ресурсов;
- серверное программное обеспечение, необходимое для функционирования сервера и связи с системой электронного обучения через Интернет;
- корпоративная платформа;
- установленное на рабочих местах обучающихся ПО: Microsoft Windows 10, Microsoft Office Standart 2010, антивирусное программное обеспечение Kaspersky Endpoint Security.
- GNU C++;
- Oracle Java;
- Deductor.
- среда разработки (JVE), компилятор (JVK) и виртуальная машина для исполнения кода (JVM) фирмы Oracle.
- Python 2.7.15 (Anaconda2 5.2.0 64-bit)
- Python 3.6.5 (Anaconda3 5.2.0 64-bit)

## **12. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине.**

Для проведения занятий лекционного типа, практических и семинарских занятий используются специальные помещения (учебные аудитории), оборудованные техническими средствами обучения – мультимедийной проекционной техникой. Для проведения занятий лекционного типа используются наборы демонстрационного оборудования.

Для проведения лабораторных работ, (практических занятий – при необходимости) используются специальные помещения (учебные аудитории), оснащенные специализированным лабораторным оборудованием: персональными компьютерами с возможностью выхода в интернет и с установленным программным обеспечением, заявленным в п.11.

Для проведения групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации используются специальные помещения (учебные аудитории), оборудованные специализированной мебелью (для обучающихся), меловой / маркерной доской.

Для организации самостоятельной работы обучающимся предоставляются помещения, оснащенные компьютерной техникой с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду университета.

Для обучения инвалидов и лиц с ограниченными возможностями здоровья университетом могут быть представлены специализированные средства обучения, в том числе технические средства коллективного и индивидуального пользования.

#### 4.8. Программа дисциплины «Программирование на языке C#»

##### 1. Наименование дисциплины: «Программирование на языке C#».

**Цель** дисциплины: целью освоения дисциплины «Программирование на языке C#» освоение методов разработки современных программных и информационных решений на языке C#.

##### 2. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы

Код компетенции	Результаты освоения образовательной программы (ИДК)	Результаты обучения по дисциплине
ОПК-4. Способен участвовать в разработке технической документации программных продуктов и комплексов с использованием стандартов, норм и правил, а также в управлении проектами создания информационных систем на стадиях жизненного цикла	ОПК-4.1. Участвует в разработке технической документации программных продуктов и комплексов с использованием стандартов, норм и правил ОПК-4.2. управляет проектами создания информационных систем на стадиях жизненного цикла	<b>Знать:</b> основы программирования для решения задач, базовый синтаксис и общая структура языка программирования C#. принципы построения приложений на языке C# с графическим пользовательским интерфейсом <b>Уметь:</b> работать со средой визуального программирования MS Visual Studio, и языком программирования высокого уровня C#. создавать простые, но хорошо документированные и структурированные приложения на языке C# с графическим пользовательским интерфейсом, <b>Владеть:</b> разработкой консольных приложений на языке C# для реализации алгоритмов обработки данных разработкой приложений с графическим интерфейсом на языке C#

##### 3. Место дисциплины в структуре образовательной программы

«Программирование на языке C#» представляет собой дисциплину обязательной части (Б1.О.13) направления подготовки бакалавриата 02.03.02 «Фундаментальная информатика



и информационные технологии», профиль «Программная инженерия в искусственном интеллекте».

#### 4. Виды учебной работы по дисциплине.

Виды учебной работы по дисциплине зафиксированы учебным планом основной профессиональной образовательной программы по указанному направлению и профилю, выражаются в академических часах. Часы контактной работы и самостоятельной работы обучающегося и часы, отводимые на процедуры контроля, могут различаться в учебных планах ОПОП по формам обучения. Объем контактной работы включает часы контактной аудиторной работы (лекции/практические занятия/ лабораторные работы), контактной внеаудиторной работы (контроль самостоятельной работы), часы контактной работы в период аттестации. Контактная работа, в том числе может проводиться посредством электронной информационно-образовательной среды университета с использованием ресурсов сети Интернет и дистанционных технологий

#### 5. Содержание дисциплины, структурированное по темам (разделам)

Исходя из рамок, установленных учебным планом по трудоемкости и видам учебной работы по дисциплине, преподаватель самостоятельно выбирает тематику занятий по формам и количеству часов проведения контактной работы: лекции и иные учебные занятия, предусматривающие преимущественную передачу учебной информации преподавателем и (или) занятия семинарского типа (семинары, практические занятия, практикумы, лабораторные работы, коллоквиумы и иные аналогичные занятия), и (или) групповые консультации, и (или) индивидуальную работу обучающихся с преподавателем, в том числе индивидуальные консультации (по курсовым работам/проектам – при наличии курсовой работы/проекта по данной дисциплине в учебном плане). Рекомендуемая тематика занятий максимально полно реализуется в контактной работе с обучающимися очной формы обучения. В случае реализации образовательной программы в заочной / очно-заочной форме трудоемкость дисциплины сохраняется, однако объем учебного материала в значительной части осваивается обучающимися в форме самостоятельной работы. При этом требования к ожидаемым образовательным результатам обучающихся по данной дисциплине не зависят от формы реализации образовательной программы.

№ п/п	Наименование разделов (тем) дисциплины	Содержание разделов (тем) дисциплин
•	Интегрированная среда C#	Основы программирования в среде визуального интерфейса. C# — система быстрой разработки приложений. Интегрированная среда C#. Общий вид окна среды. Главное меню. Структура меню C#. Инспектор объектов, страницы свойств и событий. Различные способы задания свойств (значение, список, диалог). Окно формы. Окно редактора кода. Палитра компонентов. Быстрые кнопки. Управление конфигурациями окон среды. Структура кода модуля.
•	Компилятор C#	Платформа .NET. Основы C# (история и

		особенности). Написание, компиляция и запуск первой программы на C#.
•	Проекты C#	Виды проектов в C#. Организация проекта в C#, основные файлы проектов. Основные проектные операции. Создание и сохранение проекта. Менеджер проекта. Оформление завершеного проекта.
•	Язык C#	Синтаксис языка. Компилятор. Файлы проекта C#. Области видимости и время жизни. Константы, переменные, типизированные константы. Переменные. Процедуры и функции. Операции. Операторы. Условные операторы выбора if. Условный оператор множественного выбора case. Операторы цикла for, repeat, while. Прерывание цикла: оператор break, процедуры continue, exit и abort. Исключения. Типы данных в языке C++. Процедуры и функции C++. Строка описания формата и функция Format. Математические функции. Процедуры и функции преобразования дат и времени. Процедуры и функции файлового ввода/вывода и управления файлами.
•	Визуальная среда разработки	Основы программирования в среде визуального интерфейса. C# — система быстрой разработки приложений. Размещение компонентов на форме. Работа с группой компонентов. Инструментальные средства поддержки разработки кода. Применение Code Insight—Знатока Кода. Исследователь кода Code Explorer. Просмотр иерархии классов, модулей и глобальных символов — Object Browser. Получение информации о классах, свойствах, методах из окна Редактора Кода. Навигация в коде, закладки и дополнительные окна редактирования. Отладка приложений. Настройка инструментальной панели. Настройка палитры компонентов. Настройка Редактора Кода. Настройка Code Insight. Настройка Code Explorer и Object Browser. Настройка отладчика.
•	Компоненты C#	Организация библиотеки компонентов. Страницы библиотеки компонентов. Программный доступ к свойствам и методам объектов. Настройка палитры компонентов. Создание и запись в библиотеку шаблонов компонентов и групп компонентов. Компоненты ввода и отображения текстовой, цифровой и иерархической информации. Компоненты выбора из списков. Таблица строк—компонент StringGrid. Ввод и отображение чисел, дат и времени. Секционированное отображение текстов. Компоненты отображения графической информации. Компонент Shape. Построение графиков и диаграмм. Мультимедиа и анимация. Универсальный проигрыватель Media Player. Воспроизведение видеоклипов. Кнопки,

		индикаторы, управляющие элементы. Компоненты — меню. Главное меню — компонент MainMenu. Контекстное всплывающее меню - компонент PopupMenu. Панели и компоненты внешнего оформления. Инструментальные панели. Перестраиваемые панели. Системные диалоги. Диалоги открытия и сохранения файлов. Диалог выбора шрифта. Диалог выбора цвета. Диалоги печати и установки принтера. Диалоги поиска и замены текста. Компоненты организации управления приложением.
•	Разработка приложений для Windows	Общие рекомендации по разработке графического интерфейса. Многооконные приложения. Меню. Методика проектирования меню и инструментальной панели. компоновка. Печать из приложения текстов и изображений. Проектирование окон с изменяемыми размерами. Масштабирование компонентов. Обработка событий клавиатуры и мыши. Перетаскивание объектов. Буксировка компонентов в окне приложения. Управление формами. Модальные формы. Графика и мультимедиа Построение графических изображений. Установка и настройка приложения: работа с системным реестром. Автономные приложения и пакеты.
•	Исключения и файлы	Возможность сохранять данные для последующего использования с помощью файлов данных. Чтение и запись во внешние файлы данных. Исключения C# и способы их обработки.

## 6. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине

Рекомендуемая тематика учебных занятий в форме контактной работы

Рекомендуемая тематика учебных занятий лекционного типа (предусматривающих преимущественную передачу учебной информации преподавателями):

№ п/п	Наименование разделов (тем) дисциплины	Содержание разделов (тем) дисциплин
•	Интегрированная среда C#	Основы программирования в среде визуального интерфейса. C# — система быстрой разработки приложений. Интегрированная среда C#. Общий вид окна среды. Главное меню. Структура меню C#. Инспектор объектов, страницы свойств и событий. Различные способы задания свойств (значение, список, диалог). Окно формы. Окно редактора кода. Палитра компонентов. Быстрые кнопки. Управление конфигурациями окон среды. Структура кода модуля.

•	Компилятор C#	Платформа .NET. Основы C# (история и особенности). Написание, компиляция и запуск первой программы на C#.
•	Проекты C#	Виды проектов в C#. Организация проекта в C#, основные файлы проектов. Основные про-ектные операции. Создание и сохранение проекта. Менеджер проекта. Оформление завершеного проекта.
•	Язык C#	Синтаксис языка. Компилятор. Файлы проекта C#. Области видимости и время жизни. Кон-станты, переменные, типизированные константы. Переменные. Процедуры и функции. Операции. Операторы. Условные операторы вы-бора if. Условный оператор множественного выбора case . Операторы цикла for, repeat, while. Прерывание цикла: оператор break, процедуры continue, exit и abort. Исключения. Типы данных в языке C++. Процедуры и функции C++. Строка описания формата и функция Format. Математические функции. Процедуры и функции преобразования дат и времени. Процедуры и функции файлового ввода/вывода и управления файлами.
•	Визуальная среда разработки	Основы программирования в среде визуального интерфейса. C# — система быстрой разработки приложений. Размещение компонентов на форме. Работа с группой компонентов. Инструментальные средства поддержки разработки кода. Применение Code Insight—Знатока Кода. Исследова-тель кода Code Explorer. Просмотр иерархии классов, модулей и глобальных символов — Object Browser. Получение информации о классах, свойствах, методах из окна Редактора Кода. Навигация в коде, закладки и дополнительные окна редактирования. Отладка приложений. Настройка инструментальной панели. Настройка палитры компонентов. Настройка Редактора Кода. Настройка Code Insight. Настройка Code Explorer и Object Browser. Настройка отладчика.
•	Компоненты C#	Организация библиотеки компонентов. Страницы библиотеки компонентов. Программный доступ к свойствам и методам объектов. Настройка палитры компонентов. Создание и за-пись в библиотеку шаблонов компонентов и групп компонентов. Компоненты ввода и отображения текстовой, цифровой и иерархической информации. Компоненты выбора из списков. Таблица строк—компонент StringGrid. Ввод и отображение чисел, дат и времени. Секционированное отображение текстов. Компоненты отображения графической информации. Компонент Shape. Построение графиков и диаграмм. Мультимедиа и анимация. Универсальный проигрыватель Media Player

		<p>Воспроизведение видеоклипов. Кнопки, индикаторы, управляющие элементы. Компоненты — меню. Главное меню — компонент MainMenu. Контекстное всплывающее меню - компонент PopupMenu. Панели и компоненты внешнего оформления. Инструментальные панели. Перестраиваемые панели. Системные диалоги. Диалоги открытия и сохранения файлов. Диалог выбора шрифта. Диалог выбора цвета. Диалоги печати и установки принтера. Диалоги поиска и замены текста. Компоненты организации управления приложением.</p>
•	Разработка приложений для Windows	<p>Общие рекомендации по разработке графического интерфейса. Многооконные приложения. Меню. Методика проектирования меню и инструментальной панели. компоновка. Печать из приложения текстов и изображений. Проектирование окон с изменяемыми размерами. Масштабирование компонентов. Обработка событий клавиатуры и мыши. Перетаскивание объектов. Буксировка компонентов в окне приложения. Управление формами. Модальные формы. Графика и мультимедиа Построение графических изображений. Установка и настройка приложения: работа с системным реестром. Автономные приложения и пакеты.</p>
•	Исключения и файлы	<p>Возможность сохранять данные для последующего использования с помощью файлов данных. Чтение и запись во внешние файлы данных. Исключения C# и способы их обработки.</p>

### **Рекомендуемая тематика практических занятий:**

Тематика практических занятий совпадает с темами лекционных занятий.

На практических занятиях решаются задачи по теме занятия.

### **Требования к самостоятельной работе обучающихся**

1. Работа с лекционным материалом, предусматривающая проработку конспекта лекций и учебной литературы, по всем темам из п. 6 настоящей рабочей программы.
2. Выполнение домашнего задания, предусматривающего решение задач, выполнение упражнений, выдаваемых на лабораторных занятиях, по всем темам из п. 6 настоящей рабочей программы.

Руководствуясь положениями статьи 47 и статьи 48 Федерального закона от 29 декабря 2012 г. N 273-ФЗ «Об образовании в Российской Федерации» научно-педагогические работники и иные лица, привлекаемые университетом к реализации данной образовательной программы, пользуются предоставленными академическими правами и свободами в части свободы преподавания, свободы от вмешательства в

профессиональную деятельность; свободы выбора и использования педагогически обоснованных форм, средств, методов обучения и воспитания; права на творческую инициативу, разработку и применение авторских программ и методов обучения и воспитания в пределах реализуемой образовательной программы и отдельной дисциплины.

Исходя из рамок, установленных учебным планом по трудоемкости и видам учебной работы по дисциплине, преподаватель самостоятельно выбирает тематику занятий по формам и количеству часов проведения контактной работы: лекции и иные учебные занятия, предусматривающие преимущественную передачу учебной информации преподавателем и (или) занятия семинарского типа (семинары, практические занятия, практикумы, лабораторные работы, коллоквиумы и иные аналогичные занятия), и (или) групповые консультации, и (или) индивидуальную работу обучающихся с преподавателем, в том числе индивидуальные консультации (по курсовым работам/проектам – при наличии курсовой работы/проекта по данной дисциплине в учебном плане).

Рекомендуемая тематика занятий максимально полно реализуется в контактной работе с обучающимися очной формы обучения. В случае реализации образовательной программы в заочной / очно-заочной форме трудоемкость дисциплины сохраняется, однако объем учебного материала в значительной части осваивается обучающимися в форме самостоятельной работы. При этом требования к ожидаемым образовательным результатам обучающихся по данной дисциплине не зависят от формы реализации образовательной программы.

## **7. Методические рекомендации по видам занятий**

Лекционные занятия.

В ходе лекционных занятий обучающимся рекомендуется выполнять следующие действия. Вести конспектирование учебного материала. Обращать внимание на категории, формулировки, раскрывающие содержание тех или иных явлений и процессов, научные выводы и практические рекомендации по их применению. Задавать преподавателю уточняющие вопросы с целью уяснения теоретических положений, разрешения спорных ситуаций.

Желательно оставить в рабочих конспектах поля, на которых во внеаудиторное время можно сделать пометки из рекомендованной литературы, дополняющие материал прослушанной лекции, а также подчеркивающие особую важность тех или иных теоретических положений.

Лабораторные занятия.

На лабораторных занятиях в зависимости от темы занятия выполняется поиск информации по решению проблем, практические упражнения, контрольные работы, выработка индивидуальных или групповых решений, итоговое обсуждение с обменом знаниями, участие в круглых столах, разбор конкретных ситуаций, командная работа, представление портфолио и т.п.

Самостоятельная работа.

Самостоятельная работа осуществляется в виде изучения литературы, эмпирических данных по публикациям и конкретным ситуациям из практики, подготовке индивидуальных работ, работа с лекционным материалом, самостоятельное изучение

отдельных тем дисциплины; поиск и обзор литературы и электронных источников; чтение и изучение учебника и учебных пособий.

## 8. Фонд оценочных средств

### 8.1. Перечень компетенций с указанием этапов их формирования в процессе освоения образовательной программы в рамках учебной дисциплины

Основными этапами формирования указанных компетенций при изучении обучающимися дисциплины являются последовательное изучение содержательно связанных между собой тем учебных занятий. Изучение каждой темы предполагает овладение обучающимися необходимыми компетенциями. Результат аттестации обучающихся на различных этапах формирования компетенций показывает уровень освоения компетенций.

Контролируемые разделы (темы) дисциплины	Индекс контролируемой компетенции (или её части)	Оценочные средства по этапам формирования компетенций
		текущий контроль по дисциплине
Интегрированная среда C#	ОПК-4	Опрос
Компилятор C#	ОПК-4	Опрос
Проекты C#	ОПК-4	Опрос
Язык C#	ОПК-4	Опрос
Визуальная среда разработки	ОПК-4	Опрос
Компоненты C#	ОПК-4	Опрос
Разработка приложений для Windows	ОПК-4	Опрос
Исключения и файлы	ОПК-4	Опрос

### 8.2. Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений и навыков и (или) опыта деятельности в процессе текущего контроля

#### Примерные тестовые задания:

- Что понимается под простым типом?
  - тип данных, значения которого не содержат компонент, например, целое или символ, в отличие от массива или структуры;
  - тип данных, значения которого содержат компоненты, например массивы, структуры;
  - тип данных, значения которого являются массивами;

- тип данных, значения которого являются целыми числами;
- Что понимается под простым типом?
  - тип данных, значения которого не содержат компонент, например, целое или символ, в отличие от массива или структуры;
  - тип данных, значения которого содержат компоненты, например массивы, структуры;
  - тип данных, значения которого являются массивами;
  - тип данных, значения которого являются целыми числами;
- Сколько вызовов системных функций всегда будет в программе, независимо от ее реализации?
  - всегда 2;
  - по крайней мере один;
  - не менее двух;
- Сколько вызовов системных функций всегда будет в программе, независимо от ее реализации?
  - всегда 2;
  - по крайней мере один;
  - не менее двух;
- Что должно стать результатом анализа создаваемой системы?
  - описание архитектуры;
  - прототипы системы;
  - выработка общих тактических приемов;
  - описание назначения системы;
- Что должно стать результатом анализа создаваемой системы?
  - описание архитектуры;
  - прототипы системы;
  - выработка общих тактических приемов;
  - описание назначения системы;
- Каким типом будете пользоваться для хранения и обработки данных о количестве членов вашей семьи?
  - short;
  - float;
  - long;
- Каким типом будете пользоваться для хранения и обработки данных о количестве членов вашей семьи?
  - short;
  - float;
  - long;



- Что понимается под простым типом?
  - тип данных, значения которого не содержат компонент, например, целое или символ, в отличии от массива или структуры;
  - тип данных, значения которого содержат компоненты, например массивы, структуры;
  - тип данных, значения которого являются массивами;
  - тип данных, значения которого являются целыми числами;
  
- Что понимается под простым типом?
  - тип данных, значения которого не содержат компонент, например, целое или символ, в отличии от массива или структуры;
  - тип данных, значения которого содержат компоненты, например массивы, структуры;
  - тип данных, значения которого являются массивами;
  - тип данных, значения которого являются целыми числами;
  
- Что понимается под модульным программированием?
  - организация программы в виде совокупности модулей со строгим соблюдением правил их взаимодействия;
  - организация программы в виде совокупности модулей со строгим соблюдением правил их взаимодействия, описания интерфейса и описания реализации;
  - организация программы в виде совокупности модулей со строгим соблюдением правил их взаимодействия, простейшим модулем является функция;
  - организация программы в виде совокупности строк с соблюдением правил их взаимодействия;
  
- Что понимается под модульным программированием?
  - организация программы в виде совокупности модулей со строгим соблюдением правил их взаимодействия;
  - организация программы в виде совокупности модулей со строгим соблюдением правил их взаимодействия, описания интерфейса и описания реализации;
  - организация программы в виде совокупности модулей со строгим соблюдением правил их взаимодействия, простейшим модулем является функция;
  - организация программы в виде совокупности строк с соблюдением правил их взаимодействия;
  
- Чем должно завершаться определение функции, возвращающей значение?
  - определение функции, возвращающей значения, обязано завершаться оператором `return`;
  - определение функции, возвращающей значения, обязано завершаться оператором `break`;
  - определение функции, возвращающей значения должно иметь пустое тело;
  - определение функции, возвращающей значения должно иметь оператор перехода;
  
- Чем должно завершаться определение функции, возвращающей значение?

- определение функции, возвращающей значения, обязано завершаться оператором return;
  - определение функции, возвращающей значения, обязано завершаться оператором break;
  - определение функции, возвращающей значения должно иметь пустое тело;
  - определение функции, возвращающей значения должно иметь оператор перехода;
- Функции, которые вызывают сами себя, называются
    - динамическими;
    - рекурсивными;
    - априорными;
- Функции, которые вызывают сами себя, называются
    - динамическими;
    - рекурсивными;
    - априорными;
- Параметр-ссылка является
    - именем;
    - адресом;
    - значением;
- Параметр-ссылка является
    - именем;
    - адресом;
    - значением;
- Если перед именем формального параметра в заголовке функции указан только тип, такой параметр называют
    - параметром-аргументом;
    - параметром-значением;
    - параметром-спецификатором;
- Если перед именем формального параметра в заголовке функции указан только тип, такой параметр называют
    - параметром-аргументом;
    - параметром-значением;
    - параметром-спецификатором;
- Оператор return содержит
    - значение функции;
    - параметры стека;
    - массив переменных;
- Оператор return содержит
    - значение функции;

- параметры стека;
  - массив переменных;
- К чему может привести общедоступность глобальных переменных?
    - к переполнению стека;
    - к несогласованному их изменению разными функциями;
    - к невозможности контекстного вывода значений (динамических и статических);
- К чему может привести общедоступность глобальных переменных?
    - к переполнению стека;
    - к несогласованному их изменению разными функциями;
    - к невозможности контекстного вывода значений (динамических и статических);
- Чем не может являться операция: << ?
    - операция вывода;
    - сдвиг влево;
    - меньше или равно;
- Чем не может являться операция: << ?
    - операция вывода;
    - сдвиг влево;
    - меньше или равно;
- Какие значения могут принимать переменные, объявленные следующим образом: int a,b;?
    - целые;
    - вещественные;
    - значимые;
- Какие значения могут принимать переменные, объявленные следующим образом: int a,b;?
    - целые;
    - вещественные;
    - значимые;
- Правильность вызова системных функций производится компилятором с помощью
    - данных заголовочных файлов;
    - параметров первичной инициализации системы;
    - шаблонов потоковых данных;
- Правильность вызова системных функций производится компилятором с помощью
    - данных заголовочных файлов;
    - параметров первичной инициализации системы;
    - шаблонов потоковых данных;
- Каждая строка программы, содержащая какое-либо объявление или выполняемое действие, оканчивается
    - точкой;

- системным комментарием;
- точкой с запятой;
- Каждая строка программы, содержащая какое-либо объявление или выполняемое действие, оканчивается
  - точкой;
  - системным комментарием;
  - точкой с запятой;

## Примеры комплектов задач

- Создайте консольное приложение, спрашивающее у пользователя имя и отвечающая в ответ, что приложение под авторством [ФИО автора] приветствует [Имя пользователя].
- Создайте консольное приложение, запрашивающее у пользователя 2 числа и арифметический знак операции между ними и выводящее результат этой операции.
- Создайте консольное приложение, запрашивающее у пользователя размер одномерного целочисленного массива, генерирующее одномерный массив случайным образом и выводящее элементы этого массива большие среднего арифметического.
- Создайте консольное приложение для чтения файлов в формате csv (формат задается самостоятельно).
- Создайте графическое приложение с четырьмя кнопками, на каждой кнопке должно быть указано одно из названий элементов из панели элементов, по нажатию на кнопку приложение должно выдать сообщение с кратким описанием названного элемента.
- Создайте приложение, в котором будет строка меню внизу окна, 3 кнопки справа в серой колонке и в оставшемся пространстве 4 прямоугольника по 2 в каждой строке, Строка статуса
- Создать приложение-окно с градиентной заливкой и с вертикальным и горизонтальным скроллбарами, в зависимости от состояний скроллбаров менять градиент
- Создать приложение-окно с градиентной заливкой и, в зависимости от положения курсора мыши в окне менять градиент

## 8.3. Перечень вопросов и заданий для промежуточной аттестации по дисциплине

### Вопросы для промежуточного контроля (экзамен)

- Язык программирования C#. Общая характеристика языка: место в классификации языков, синтаксис и семантика языка, основные объекты языка, операторы, структура программы.
- Понятие среды программирования. Компоненты среды. Компиляция и компоновка программы.
- Синтаксические и логические ошибки. Тестирование и отладка программы.

- Данные в языке C#: константы и переменные. Скалярные типы данных. Модификаторы типов.
- Данные числовых типов в языке C#: объявление, характеристика, допустимые операции, приведение типов. Пример использования.
- Операции языка C#. Приоритет операций. Оператор и операция присваивания в языке C#. Множественное присваивание. Выражения.
- Функции форматного ввода и вывода. Параметры. Управляющая строка. Спецификаторы формата. Управляющие символы.
- Алгоритмическая конструкция ветвления: полная и неполная форма, блок-схемы. Условная операция. Условный оператор в языке C#: структура оператора, полная и неполная формы, использование сложных условий. Пример на языке C#.
- Алгоритмическая конструкция выбора: понятие, блок-схема. Оператор выбора в языке C#: структура оператора. Пример программы на C#.
- Циклические алгоритмы: понятие, виды (перечислить). Алгоритмическая конструкция цикла с предусловием (понятие, использование, блок-схема). Оператор цикла `for`: структура оператора, пример использования.
- Циклические алгоритмы: понятие, виды (перечислить). Алгоритмическая конструкция цикла с предусловием (понятие, использование, блок-схема). Оператор цикла с предусловием `while` в языке C#: структура оператора, допустимые и недопустимые условия, пример использования.
- Циклические алгоритмы: понятие, виды (перечислить). Алгоритмическая конструкция цикла с постусловием (понятие, использование, блок-схема). Оператор цикла с постусловием в языке C#: структура оператора, допустимые и недопустимые условия, пример использования.
- Типовые циклические алгоритмы: максимум/минимум, сумма/произведение, количество.
- Линейный массив: понятие массива, объявление, инициализация массива, индексация элементов. Формирование и вывод массива.
- Указатели. Понятие указателя, объявление, инициализация. Операции, применимые к указателям. Указатель на указатель.
- Использование указателей при работе с массивами. Получение адресов и значений элементов. Последовательный перебор элементов. Примеры.
- Функции в языке C#: понятие, объявление, прототипы функций. Вызов функции. Типы возвращаемых значений.

- Функции в языке C#: параметры формальные и фактические, механизм передачи параметров. Передача параметров «по значению» и «по ссылке». Пример использования.
- Локальные и глобальные переменные. Области действия и области видимости. Экранирование переменных.
- Библиотечные функции. Заголовочные файлы. Подключение библиотек. Функции математической библиотеки.
- Массив символов и строка в языке C#. Ввод и вывод строк. Простейшие алгоритмы сканирования и обработки строки. Пример.
- Строка в языке C#: библиотечные функции для обработки строк. Примеры использования.
- Основные структурные части ИСР C#.
- Основные проектные операции C#.
- Назначение основных элементов знатока кода C#.
- Постройте иерархию базовых классов VCL?
- Перечислите и опишите компоненты отображения информации.
- Опишите особенности работы с канвой. Укажите компоненты, обладающие канвой.
- Перечислите основные методы канвы.
- Перечислите компоненты VCL, реализующие работу системных диалогов различного назначения. Укажите их основные свойства и методы.
- Укажите основные принципы работы с приложением, окном, страницей.
- Перечислите основные события окна и полотна.
- Язык C#. Назначение класса исключений.
- Язык C#. Классификация стандартных функций.
- Приложение стандарта WPF
- Установка и настройка приложения по стандарту WPF.

#### 8.4. Планируемые уровни сформированности компетенций обучающихся и критерии оценивания

Уровни	Содержательное описание уровня	Основные признаки выделения уровня (этапы формирования компетенции, критерии оценки)	Пятибалльная шкала (академическая) оценка	Двухбалльная шкала, зачет	БРС, % освоения (рейтинговая)
--------	--------------------------------	--	---	---------------------------	-------------------------------

		сформированности)			оценка)
Повышенны й	Творческая деятельность	<i>Включает нижестоящий уровень.</i> Умение самостоятельно принимать решение, решать проблему/задачу теоретического и прикладного характера на основе изученных методов, приемов, технологий	отлично	зачтено	86-100
Базовый	Применение знаний и умений в более широких контекстах учебной и профессиона льной деятельности, нежели по образцу с большой степени самостоятель ности и инициативы	<i>Включает нижестоящий уровень.</i> Способность собирать, систематизировать, анализировать и грамотно использовать информацию из самостоятельно найденных теоретических источников и иллюстрировать ими теоретические положения или обосновывать практику применения	хорошо		71-85
Удовлетвори тельный (достаточно й)	Репродуктивн ая деятельность	Изложение в пределах задач курса теоретически и практически контролируемого материала	удовлетвор ительно		55-70
Недостаточн ый	Отсутствие удовлетворительного уровня	признаков	неудовлетв орительно	не зачтено	Менее 55

## 9. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины.

### Основная литература

- Кокоса, К. Управление памятью в .NET для профессионалов : практическое руководство / К. Кокоса. - Москва : ДМК Пресс, 2020. - 800 с. - ISBN 978-5-97060-800-5. - Текст : электронный. - URL: <https://znanium.com/catalog/product/1210679> (дата обращения: 20.11.2023). – Режим доступа: по подписке.
- Агафонов, Е. Д. Прикладное программирование : учебное пособие / Е. Д. Агафонов, Г. В. Ващенко. - Красноярск : СФУ, 2015. - 112 с. - ISBN 978-5-7638-

3165-8. - Текст : электронный. - URL: <https://znanium.com/catalog/product/550046> (дата обращения: 20.11.2023). – Режим доступа: по подписке.

- Бельков, С. А. Прикладное программирование с использованием языка С-Шарп : учебно-методическое пособие / С. А. Бельков. - Екатеринбург : Изд-во Уральского ун-та, 2017. - 120 с. - ISBN 978-5-7996-2035-6. - Текст : электронный. - URL: <https://znanium.com/catalog/product/1936343> (дата обращения: 20.11.2023). – Режим доступа: по подписке.

### **Дополнительная литература**

- Городняя, Л. В. Парадигма программирования : учебное пособие для вузов / Л. В. Городняя. — 2-е изд., стер. — Санкт-Петербург : Лань, 2021. — 232 с
- Свердлов, С. З. Языки программирования и методы трансляции : учебное пособие для вузов / С. З. Свердлов. — 2-е изд., стер. — Санкт-Петербург : Лань, 2021. — 564 с.

### **10. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины (модуля).**

- НЭБ Национальная электронная библиотека, диссертации и прочие издания
- eLIBRARY.RU Научная электронная библиотека, книги, статьи, тезисы докладов конференций
- Гребенников Электронная библиотека ИД журналы
- ЭБС Консультант студента
- ПРОСПЕКТ ЭБС
- ЭБС ZNANIUM.COM
- ЭБС IBOOKS.RU
- РГБ Информационное обслуживание по МБА
- БЕН РАН
- Электронно-библиотечная система (ЭБС) Кантиана (<https://elib.kantiana.ru/>)

### **11. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине.**

Программное обеспечение обучения включает в себя:

- система электронного образовательного контента БФУ им. И. Канта – [www.lms.kantiana.ru](http://www.lms.kantiana.ru), обеспечивающую разработку и комплексное использование электронных образовательных ресурсов;
- серверное программное обеспечение, необходимое для функционирования сервера и связи с системой электронного обучения через Интернет;
- корпоративная платформа
- установленное на рабочих местах обучающихся ПО: Microsoft Windows 10, Microsoft Office Standart 2010, антивирусное программное обеспечение Kaspersky Endpoint Security.
- GNU C++;
- Oracle Java;
- Deductor.



- среда разработки (JVE), компилятор (JVK) и виртуальная машина для исполнения кода (JVM) фирмы Oracle.
- Python 2.7.15 (Anaconda2 5.2.0 64-bit)
- Python 3.6.5 (Anaconda3 5.2.0 64-bit)

## **12. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине.**

Для проведения занятий лекционного типа, практических и семинарских занятий используются специальные помещения (учебные аудитории), оборудованные техническими средствами обучения – мультимедийной проекционной техникой. Для проведения занятий лекционного типа используются наборы демонстрационного оборудования.

Для проведения лабораторных работ, (практических занятий – при необходимости) используются специальные помещения (учебные аудитории), оснащенные специализированным лабораторным оборудованием: персональными компьютерами с возможностью выхода в интернет и с установленным программным обеспечением, заявленным в п.11.

Для проведения групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации используются специальные помещения (учебные аудитории), оборудованные специализированной мебелью (для обучающихся), меловой / маркерной доской.

Для организации самостоятельной работы обучающимся предоставляются помещения, оснащенные компьютерной техникой с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду университета.

Для обучения инвалидов и лиц с ограниченными возможностями здоровья университетом могут быть представлены специализированные средства обучения, в том числе технические средства коллективного и индивидуального пользования.

### **4.9. Программа дисциплины «Информационная безопасность»**

#### **1.Наименование дисциплины: «Информационная безопасность».**

**Цель дисциплины:** показать студентам важность задач обеспечения информационной безопасности и изучить основные методы защиты информации. Рассматриваются основные типы угроз (нарушение конфиденциальности, целостности и доступности) и основные функции систем защиты.

#### **2. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы**

Код компетенции	Результаты освоения образовательной программы (ИДК)	Результаты обучения по дисциплине
ОПК-6. Способен понимать	ОПК-6.1. Знает принципы работы современных	<b>Знать:</b> угрозы безопасности информации

<p>принципы работы современных информационных технологий и использовать их для решения задач профессиональной деятельности</p>	<p>информационных технологий ОПК-6.2. Использует современные информационные технологии для решения задач профессиональной деятельности</p>	<p>автоматизированных системах и вычислительных сетях; основные принципы, методы и технологии идентификации и аутентификации; основные политики систем управления доступом, их свойства и критерии безопасности; технологии и основные методы управления доступом к информации применяемые на практике; технологии регистрации и учета; технологии обеспечения и проверки целостности информации; технологии защиты информации и программного обеспечения от вредоносных программ; основы администрирования вычислительных систем; принципы и методы противодействия несанкционированному информационному воздействию на вычислительные системы; технологии межсетевого экранирования и обнаружения вторжений; принципы организации информационных систем в соответствии с требованиями по защите информации.</p> <p><b>Уметь:</b> анализировать и оценивать угрозы информационной безопасности автоматизированных систем;</p> <p>осуществлять выбор функциональной структуры системы обеспечения безопасности информации, обрабатываемой в автоматизированных системах; обосновывать выбор технологий для обеспечения безопасности информации, обрабатываемой в автоматизированных системах; осуществлять меры противодействия нарушениям сетевой безопасности с использованием различных программных и аппаратных средств защиты.</p>
--	--	---

		<p><b>Владеть:</b> навыками использования методов и средств выявления угроз безопасности информации в автоматизированных системах; применения методов противодействия угрозам ИБ; разработки моделей угроз безопасности информации в автоматизированных системах; установки и настройки программных и программно-аппаратных средств защиты информации от несанкционированного доступа в операционных системах общего пользования; использования методик анализа сетевого трафика для исследования сети и обнаружения признаков сетевых атак; анализа функционирования сетевых средств безопасности.</p>
--	--	---

### 3. Место дисциплины в структуре образовательной программы

«Информационная безопасность» представляет собой дисциплину обязательной части (Б1.О.13) направления подготовки бакалавриата 02.03.02 «Фундаментальная информатика и информационные технологии», профиль «Программная инженерия в искусственном интеллекте».

### 4. Виды учебной работы по дисциплине.

Виды учебной работы по дисциплине зафиксированы учебным планом основной профессиональной образовательной программы по указанному направлению и профилю, выражаются в академических часах. Часы контактной работы и самостоятельной работы обучающегося и часы, отводимые на процедуры контроля, могут различаться в учебных планах ОПОП по формам обучения. Объем контактной работы включает часы контактной аудиторной работы (лекции/практические занятия/ лабораторные работы), контактной внеаудиторной работы (контроль самостоятельной работы), часы контактной работы в период аттестации. Контактная работа, в том числе может проводиться посредством электронной информационно-образовательной среды университета с использованием ресурсов сети Интернет и дистанционных технологий

### 5. Содержание дисциплины, структурированное по темам (разделам)

Исходя из рамок, установленных учебным планом по трудоемкости и видам учебной работы по дисциплине, преподаватель самостоятельно выбирает тематику занятий по формам и количеству часов проведения контактной работы: лекции и иные

учебные занятия, предусматривающие преимущественную передачу учебной информации преподавателем и (или) занятия семинарского типа (семинары, практические занятия, практикумы, лабораторные работы, коллоквиумы и иные аналогичные занятия), и (или) групповые консультации, и (или) индивидуальную работу обучающихся с преподавателем, в том числе индивидуальные консультации (по курсовым работам/проектам – при наличии курсовой работы/проекта по данной дисциплине в учебном плане). Рекомендуемая тематика занятий максимально полно реализуется в контактной работе с обучающимися очной формы обучения. В случае реализации образовательной программы в заочной / очно-заочной форме трудоемкость дисциплины сохраняется, однако объем учебного материала в значительной части осваивается обучающимися в форме самостоятельной работы. При этом требования к ожидаемым образовательным результатам обучающихся по данной дисциплине не зависят от формы реализации образовательной программы.

№ п/п	Наименование разделов (тем) дисциплины	Содержание разделов (тем) дисциплин
•	Тема 1. Задачи и методы обеспечения информационной безопасности	<p>Термины и определения. Классификация угроз несанкционированного доступа к информации в АС. Общая характеристика источников угроз несанкционированного доступа в АС. Общая характеристика уязвимостей АС и вычислительных сетей. Угрозы программно-математических воздействий. Компьютерные вирусы и “троянские кони”. Модели нарушителя. Основные функции систем защиты информации.</p> <p>Процедура проверки подлинности субъектов и объектов, параметры парольной идентификации, особенности аутентификации в вычислительных сетях: задачи аутентификации, авторизации и акаунтинга (AAA).</p> <p>Модель системы защиты с полным перекрытием, субъектно-объектная модель системы защиты, понятие изолированной системы, особенности моделирования механизмов безопасности операционных систем и баз данных, основные виды моделей и политик управления доступом — ограниченность моделей и проблемы изменения прав доступа.</p> <p>Методы аутентификации и разграничения доступа в операционных системах Windows и Linux.</p>
•	Тема 2. Теоретические основы информационной безопасности операционных систем и баз данных.	<p>Строгие протоколы аутентификации. Протокол Нидхема-Шредера для симметричной и асимметричной криптографии. Протоколы на основе ключевых хеш-функций. Использование цифровой подписи.</p> <p>Матрица доступа, пятимерное пространства безопасности Хартсона, модели HRU и Take-Grant, основные результаты, их достоинства и недостатки, основные направления развития.</p> <p>MLS модель «военной безопасности», модель</p>

		<p>Белла-ЛаПадуды, решетки безопасности Деннинг. Модель Биба.</p> <p>Тематические классификаторы и решетки мультирубрик.</p> <p>Использование функциональной структуры организации для управления доступом, индивидуально групповая модель управления доступом.</p>
<ul style="list-style-type: none"> <li>•</li> </ul>	<p>Тема 3. Информационная безопасность вычислительных сетей.</p>	<p>Субъекты и объекты компьютерных атак в сетях, виды сетевых атак; методы защиты вычислительных сетей: задачи аутентификации, авторизации и акаунтинга (AAA), сервера безопасности (RADIUS, Kerberos). Задачи фильтрации сетевого трафика. Межсетевые экраны. Фильтрация пакетов. Анализ приложений. Анализ состояний. Прокси сервер. DLP системы. Понятие DMZ.</p> <p>Управление доступом в распределенных системах. Методы оптимизации и методы теории игр при моделировании систем защиты. Теоретико-игровые модели сетевых атак. Модели «доверия» в социальных сетях.</p> <p>Реальность угроз. Типы атак. Структура типовой атаки. Сканирование. Атаки на разных уровнях протокола TCP/IP (ARP-спуффинг, атаки на маршрутизатор, атаки на DNS, атаки HTTP). Методы обнаружения вторжений.</p> <p>Построение VPN, протоколы SSL,SSH,TLS,IPSec.</p> <p>Сети с открытым доступом к каналам связи. Аутентификация, Авторизация – повышенные требования для WiFi, GSM, LTE сетей. Контроль доступа. Основные уязвимости и риски.</p>
<ul style="list-style-type: none"> <li>•</li> </ul>	<p>Тема 4. Методическое и организационное обеспечение информационной безопасности.</p>	<p>Критериальные пространства безопасности. Задача оценки эффективности защиты информации. Понятие риска безопасности, вероятностная модель Клементса. Идентификация рисков, основания для управления рисками для обеспечения непрерывности. Измерение эффективности систем защиты в качественных и количественных шкалах. Экономические модели оценки эффективности. Классификации и упорядоченные классы требований безопасности. Стандарты безопасности.</p> <p>Субъективность оценки эффективности, понятие доверия в безопасности, методы доверия, требования доверия, управление доверием, обеспечение уровня доверия к среде. Принципиальные ограничения моделей эффективности в условиях критических объектов безопасности и угроз инсайдера.</p> <p>Эволюция подходов и моделей управления</p>

		<p>безопасностью. Процессный характер управления, этапы и факторы управления. Система управления, иерархия политик безопасности. Технологии и инструменты аудита безопасности. Мониторинг безопасности, идентификация событий безопасности, нормализация, корреляция и классификация событий безопасности.</p> <p>Управление фильтрацией прикладного уровня, мониторинг прикладного потока через контур сегмента вычислительной среды, угрозы ошибок фильтрации, задача оптимального фильтра. Технологии управление правами для различных моделей доступа, проблема администратора, расщепление полномочий. Технологии управление безопасностью в виртуальных средах: сертификация среды обработки, доверенный супервизор, функциональная и ресурсная инкапсуляция. Идеология «Общих критериев», сеть высокоуровневых сущностей, диалектика зависимости целей, предположений, угроз и политик для среды и объекта защиты, стойкость функций безопасности.</p>
•	Тема 5. Проблемные вопросы обеспечения информационной безопасности автоматизированных систем и вычислительных сетей.	<p>Виртуальные вычисления в центрах обработки данных, «облачные вычисления».</p> <p>Понятие, виды (по памяти, по времени, статистические), обнаружение и методы противодействия; утечки информации в статистических БД; теоретико-вероятностная модель «невыводимости» и «невлияния».</p> <p>Понятие анонимных сетей. Примеры анонимных сетей. TOR. I2P. Уязвимости. Обнаружение.</p> <p>Безопасность SDN. Разделение потока данных и управляющего потока. Возможные виды атак. Скрытые каналы.</p>

## 6. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине

**Рекомендуемая тематика учебных занятий лекционного типа**  
(предусматривающих преимущественную передачу учебной информации преподавателями):

№ п/п	Наименование разделов (тем) дисциплины	Содержание разделов (тем) дисциплин
•	Тема 1. Задачи и методы обеспечения информационной безопасности	Термины и определения. Классификация угроз несанкционированного доступа к информации в АС. Общая характеристика источников угроз несанкционированного доступа в АС. Общая характеристика уязвимостей АС и вычислительных сетей. Угрозы программно-математических воздействий. Компьютерные вирусы и «троянские

		<p>кони”. Модели нарушителя. Основные функции систем защиты информации.</p> <p>Процедура проверки подлинности субъектов и объектов, параметры парольной идентификации, особенности аутентификации в вычислительных сетях: задачи аутентификации, авторизации и акаунтинга (AAA).</p> <p>Модель системы защиты с полным перекрытием, субъектно-объектная модель системы защиты, понятие изолированной системы, особенности моделирования механизмов безопасности операционных систем и баз данных, основные виды моделей и политик управления доступом — ограниченность моделей и проблемы изменения прав доступа.</p> <p>Методы аутентификации и разграничения доступа в операционных системах Windows и Linux.</p>
<ul style="list-style-type: none"> <li>•</li> </ul>	<p>Тема 2. Теоретические основы информационной безопасности операционных систем и баз данных.</p>	<p>Строгие протоколы аутентификации. Протокол Нидхема-Шредера для симметричной и асимметричной криптографии. Протоколы на основе ключевых хеш-функций. Использование цифровой подписи.</p> <p>Матрица доступа, пятимерное пространства безопасности Хартсона, модели HRU и Take-Grant, основные результаты, их достоинства и недостатки, основные направления развития.</p> <p>MLS модель «военной безопасности», модель Белла-ЛаПадулы, решетки безопасности Деннинг. Модель Биба.</p> <p>Тематические классификаторы и решетки мультирубрик.</p> <p>Использование функциональной структуры организации для управления доступом, индивидуально групповая модель управления доступом.</p>
<ul style="list-style-type: none"> <li>•</li> </ul>	<p>Тема 3. Информационная безопасность вычислительных сетей.</p>	<p>Субъекты и объекты компьютерных атак в сетях, виды сетевых атак; методы защиты вычислительных сетей: задачи аутентификации, авторизации и акаунтинга (AAA), сервера безопасности (RADIUS, Kerberos). Задачи фильтрации сетевого трафика. Межсетевые экраны. Фильтрация пакетов. Анализ приложений. Анализ состояний. Прокси сервер. DLP системы. Понятие DMZ.</p> <p>Управление доступом в распределенных системах. Методы оптимизации и методы теории игр при моделировании систем защиты. Теоретико-игровые модели сетевых атак. Модели «доверия» в социальных сетях.</p> <p>Реальность угроз. Типы атак. Структура типовой атаки. Сканирование. Атаки на разных уровнях</p>

		<p>протокола TCP/IP (ARP-спуфинг, атаки на маршрутизатор, атаки на DNS, атаки HTTP). Методы обнаружения вторжений. Построение VPN, протоколы SSL,SSH,TLS,IPSec. Сети с открытым доступом к каналам связи. Аутентификация, Авторизация – повышенные требования для WiFi, GSM, LTE сетей. Контроль доступа. Основные уязвимости и риски.</p>
•	Тема 4. Методическое и организационное обеспечение информационной безопасности.	<p>Критериальные пространства безопасности. Задача оценки эффективности защиты информации. Понятие риска безопасности, вероятностная модель Клементса. Идентификация рисков, основания для управления рисками для обеспечения непрерывности. Измерение эффективности систем защиты в качественных и количественных шкалах. Экономические модели оценки эффективности. Классификации и упорядоченные классы требований безопасности. Стандарты безопасности.</p> <p>Субъективность оценки эффективности, понятие доверия в безопасности, методы доверия, требования доверия, управление доверием, обеспечение уровня доверия к среде. Принципиальные ограничения моделей эффективности в условиях критических объектов безопасности и угроз инсайдера.</p> <p>Эволюция подходов и моделей управления безопасностью. Процессный характер управления, этапы и факторы управления. Система управления, иерархия политик безопасности. Технологии и инструменты аудита безопасности. Мониторинг безопасности, идентификация событий безопасности, нормализация, корреляция и классификация событий безопасности.</p> <p>Управление фильтрацией прикладного уровня, мониторинг прикладного потока через контур сегмента вычислительной среды, угрозы ошибок фильтрации, задача оптимального фильтра. Технологии управление правами для различных моделей доступа, проблема администратора, расщепление полномочий. Технологии управление безопасностью в виртуальных средах: сертификация среды обработки, доверенный супервизор, функциональная и ресурсная инкапсуляция. Идеология «Общих критериев», сеть высокоуровневых сущностей, диалектика зависимости целей, предположений, угроз и политик для среды и объекта защиты, стойкость функций безопасности.</p>
•	Тема 5. Проблемные вопросы обеспечения информационной безопасности автоматизированных	<p>Виртуальные вычисления в центрах обработки данных, «облачные вычисления».</p> <p>Понятие, виды (по памяти, по времени,</p>



	систем и вычислительных сетей.	статистические), обнаружение и методы противодействия; утечки информации в статистических БД; теоретико-вероятностная модель «невыводимости» и «невлияния». Понятие анонимных сетей. Примеры анонимных сетей. TOR. I2P. Уязвимости. Обнаружение. Безопасность SDN. Разделение потока данных и управляющего потока. Возможные виды атак. Скрытые каналы.
--	--------------------------------	---

### **Рекомендуемая тематика *практических* занятий:**

Тематика практических занятий совпадает с темами лекционных занятий.

На практических занятиях решаются задачи по теме занятия.

### **Требования к самостоятельной работе обучающихся**

1. Работа с лекционным материалом, предусматривающая проработку конспекта лекций и учебной литературы, по всем темам из п. 6 настоящей рабочей программы.

2. Выполнение домашнего задания, предусматривающего решение задач, выполнение упражнений, выдаваемых на лабораторных занятиях, по всем темам из п. 6 настоящей рабочей программы.

Руководствуясь положениями статьи 47 и статьи 48 Федерального закона от 29 декабря 2012 г. N 273-ФЗ «Об образовании в Российской Федерации» научно-педагогические работники и иные лица, привлекаемые университетом к реализации данной образовательной программы, пользуются предоставленными академическими правами и свободами в части свободы преподавания, свободы от вмешательства в профессиональную деятельность; свободы выбора и использования педагогически обоснованных форм, средств, методов обучения и воспитания; права на творческую инициативу, разработку и применение авторских программ и методов обучения и воспитания в пределах реализуемой образовательной программы и отдельной дисциплины.

Исходя из рамок, установленных учебным планом по трудоемкости и видам учебной работы по дисциплине, преподаватель самостоятельно выбирает тематику занятий по формам и количеству часов проведения контактной работы: лекции и иные учебные занятия, предусматривающие преимущественную передачу учебной информации преподавателем и (или) занятия семинарского типа (семинары, практические занятия, практикумы, лабораторные работы, коллоквиумы и иные аналогичные занятия), и (или) групповые консультации, и (или) индивидуальную работу обучающихся с преподавателем, в том числе индивидуальные консультации (по курсовым работам/проектам – при наличии курсовой работы/проекта по данной дисциплине в учебном плане).

Рекомендуемая тематика занятий максимально полно реализуется в контактной работе с обучающимися очной формы обучения. В случае реализации образовательной

программы в заочной / очно-заочной форме трудоемкость дисциплины сохраняется, однако объем учебного материала в значительной части осваивается обучающимися в форме самостоятельной работы. При этом требования к ожидаемым образовательным результатам обучающихся по данной дисциплине не зависят от формы реализации образовательной программы.

## **7. Методические рекомендации по видам занятий**

Лекционные занятия.

В ходе лекционных занятий обучающимся рекомендуется выполнять следующие действия. Вести конспектирование учебного материала. Обращать внимание на категории, формулировки, раскрывающие содержание тех или иных явлений и процессов, научные выводы и практические рекомендации по их применению. Задавать преподавателю уточняющие вопросы с целью уяснения теоретических положений, разрешения спорных ситуаций.

Желательно оставить в рабочих конспектах поля, на которых во внеаудиторное время можно сделать пометки из рекомендованной литературы, дополняющие материал прослушанной лекции, а также подчеркивающие особую важность тех или иных теоретических положений.

Лабораторные занятия.

На лабораторных занятиях в зависимости от темы занятия выполняется поиск информации по решению проблем, практические упражнения, контрольные работы, выработка индивидуальных или групповых решений, итоговое обсуждение с обменом знаниями, участие в круглых столах, разбор конкретных ситуаций, командная работа, представление портфолио и т.п.

Самостоятельная работа.

Самостоятельная работа осуществляется в виде изучения литературы, эмпирических данных по публикациям и конкретных ситуаций из практики, подготовке индивидуальных работ, работа с лекционным материалом, самостоятельное изучение отдельных тем дисциплины; поиск и обзор литературы и электронных источников; чтение и изучение учебника и учебных пособий.

## **8. Фонд оценочных средств**

### **8.1. Перечень компетенций с указанием этапов их формирования в процессе освоения образовательной программы в рамках учебной дисциплины**

Основными этапами формирования указанных компетенций при изучении обучающимися дисциплины являются последовательное изучение содержательно связанных между собой тем учебных занятий. Изучение каждой темы предполагает овладение обучающимися необходимыми компетенциями. Результат аттестации обучающихся на различных этапах формирования компетенций показывает уровень освоения компетенций.

Контролируемые разделы (темы) дисциплины	Индекс контролируемой	Оценочные средства по этапам формирования
--	-----------------------	---

	компетенции (или её части)	компетенций
		текущий контроль по дисциплине
Тема 1. Задачи и методы обеспечения информационной безопасности	ОПК-6	контрольная работа
Тема 2. Теоретические основы информационной безопасности операционных систем и баз данных.	ОПК-6	контрольная работа
Тема 3. Информационная безопасность вычислительных сетей.	ОПК-6	контрольная работа
Тема 4. Методическое и организационное обеспечение информационной безопасности.	ОПК-6	контрольная работа
Тема 5. Проблемные вопросы обеспечения информационной безопасности автоматизированных систем и вычислительных сетей.	ОПК-6	контрольная работа

## **8.2. Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений и навыков и (или) опыта деятельности процессе текущего контроля**

В качестве оценочных средств текущего контроля успеваемости предусмотрены:

контрольная работа

### **Вариант контрольной работы №1**

- Дать определение следующим терминам:
  - Безопасность информации
  - Конфиденциальность
  - Целостность
  - Доступность
  - Угроза
  - Атака
  - Объект
  - Субъект
  - Пользователь
  - Граф доступов
  - Политика безопасности

- Описать принцип атаки «Троянским конем» в дискреционной политике безопасности.
- Вычислить величину потока информации от  $X$  к  $Z$  для следующего оператора:  
if  $((X==1) \& (Y==1))$  the  $Z=1$ ;  
здесь  $X, Y, Z$  – биты, равномерно распределенные на множестве  $\{0,1\}$ .

### Вариант контрольной работы №2

**Задача 1.** Для команды машины Тьюринга выписать команды модели ХРУ, которые ее представляют.

**Задача 2.** Постройте граф создания для систем МТМД.

command a1(x:α, y:β, z:β)

«создать» субъект x с типом α

end;

command a2(x:α, y:γ, z:β, s:γ, o:δ)

«создать» объект y с типом γ;

«создать» субъект s с типом δ;

end;

command a3(x:ε, y:δ, z:β, s:γ, o:δ)

«создать» субъект o с типом δ;

«создать» объект x с типом ε;

end;

**Задача 3.** Модель Take-Grant. Описание и основные теоремы.

## 8.3. Перечень вопросов и заданий для промежуточной аттестации по дисциплине

### Вопросы для промежуточного контроля (экзамен)

1. Основные уязвимости информационных систем. Проблема защиты компьютерных систем. Основные понятия и определения компьютерной безопасности.
2. Угрозы безопасности. Модель нарушителя информационной безопасности.
3. Понятие защищенности (безопасности) компьютерной информации. Конфиденциальность, целостность и доступность информации.
4. Субъектно-объектная модель компьютерной системы. Понятие потока, доступа и правил разграничения доступа. Основные типы политик разграничения доступа.
5. Монитор безопасности КС и гарантирование выполнения политики безопасности.
6. Изолированная программная среда.
7. Дискреционные модели безопасности компьютерных систем. Пяти-мерное пространство Хартсона
8. Модели безопасности на основе матрицы доступа. Способы организации матрицы доступа и управления доступом в компьютерных системах

9. Дискреционные модели распространения прав доступа. Модель и теоремы безопасности Харрисона-Руззо-Ульмана.
10. Модель TAKE-GRANT.
11. Расширенная модель TAKE-GRANT.
12. Основы политики мандатного доступа. Решетка безопасности.
13. Модель Белла-ЛаПадулы и основная теорема безопасности
14. Основные расширения модели Белла-ЛаПадулы.
15. Скрытые каналы утечки информации и теоретико-информационные модели безопасности (модель невлияния и модель невыводимости).
16. Модели ролевого доступа. Иерархические системы ролей. Принципы наделения ролей полномочиями.
17. Модели обеспечения целостности. Дискреционная модель Кларка-Вильсона.
18. Модели обеспечения целостности. Мандатная модель Кена Биба.
19. Объединение мандатных моделей Белла-ЛаПадулы и Кена Биба.

#### 8.4. Планируемые уровни сформированности компетенций обучающихся и критерии оценивания

Уровни	Содержательное описание уровня	Основные признаки выделения уровня (этапы формирования компетенции, критерии оценки сформированности)	Пятибалльная шкала (академическая) оценка	Двухбалльная шкала, зачет	БРС, % освоения (рейтинговая оценка)
Повышенный	Творческая деятельность	<i>Включает нижестоящий уровень.</i> Умение самостоятельно принимать решение, решать проблему/задачу теоретического и прикладного характера на основе изученных методов, приемов, технологий	отлично	зачтено	86-100
Базовый	Применение знаний и умений в более широких контекстах учебной и профессиональной деятельности, нежели по образцу с большей степени самостоятель	<i>Включает нижестоящий уровень.</i> Способность собирать, систематизировать, анализировать и грамотно использовать информацию из самостоятельно найденных теоретических источников и иллюстрировать ими теоретические положения или обосновывать практику	хорошо		71-85

	ности и инициативы	применения			
Удовлетворительный (достаточный)	Репродуктивная деятельность	Изложение в пределах задач курса теоретически и практически контролируемого материала	удовлетворительно		55-70
Недостаточный	Отсутствие признаков удовлетворительного уровня		неудовлетворительно	не зачтено	Менее 55

## 9. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины.

### Основная литература

1. Бабаш, А. В. Криптографические методы защиты информации. Том 1 : учебно-методическое пособие / А. В. Бабаш. — 2-е изд., перераб. и доп. — Москва : РИОР : ИНФРА-М, 2021. — 413 с. — (Высшее образование: Бакалавриат). - ISBN 978-5-369-01267-3. - Текст : электронный. - URL: <https://znanium.com/catalog/product/1215714> (дата обращения: 20.11.2023). – Режим доступа: по подписке.
2. Мельников, Д.А. Информационная безопасность открытых систем : учебник / Д.А. Мельников. — 3-е изд., стер. — Москва : ФЛИНТА, 2019. - 444 с. - ISBN 978-5-9765-1613-7. - Текст : электронный. - URL: <https://znanium.com/catalog/product/1042499> (дата обращения: 20.11.2023). – Режим доступа: по подписке.
- 3) Шаньгин, В. Ф. Защита информации в компьютерных системах и сетях : учебное пособие / В. Ф. Шаньгин. - 2-е изд. - Москва : ДМК Пресс, 2023. - 594 с. - ISBN 978-5-89818-506-0. - Текст : электронный. - URL: <https://znanium.com/catalog/product/2107178> (дата обращения: 20.11.2023). – Режим доступа: по подписке.
- 4) Гришина, Н. В. Основы управления информационной безопасностью : учебно-методическое пособие / Н.В. Гришина. — Москва : ИНФРА-М, 2021. — 99 с. — (Высшее образование: Бакалавриат). - ISBN 978-5-16-110048-6. - Текст : электронный. - URL: <https://znanium.com/catalog/product/1859951> (дата обращения: 20.11.2023). – Режим доступа: по подписке.

### Дополнительная литература

- 1) Галатенко В.А. Стандарты информационной безопасности [Текст] : Курс лекций: Учеб. пособие / В. А. Галатенко ; Под ред. В.Б. Бетелина. - 2-е изд. - М. : Интернет-Университет Информационных технологий, 2012. - 264 с. - 2000 экз. - ISBN 978-5-9556-0053-6 : 262-51; 262-50.
- 2) Девянин П.Н. Модели безопасности компьютерных систем. Управление доступом и информационными потоками [Электронный ресурс] : Учеб. пособие /

П. Н. Девянин. - М. : Горячая линия-Телеком, 2012. - 320 с. - Доступ к электронной версии книги открыт на сайте <http://e.lanbook.com/>. - ISBN 978-5-9912-0147-6.

#### **10. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины (модуля).**

- НЭБ Национальная электронная библиотека, диссертации и прочие издания
- eLIBRARY.RU Научная электронная библиотека, книги, статьи, тезисы докладов конференций
- Гребенников Электронная библиотека ИД журналы
- ЭБС Консультант студента
- ПРОСПЕКТ ЭБС
- ЭБС ZNANIUM.COM
- ЭБС IBOOKS.RU
- РГБ Информационное обслуживание по МБА
- БЕН РАН
- Электронно-библиотечная система (ЭБС) Кантиана (<https://elib.kantiana.ru/>)

#### **11. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине.**

Программное обеспечение обучения включает в себя:

- система электронного образовательного контента БФУ им. И. Канта – [www.lms.kantiana.ru](http://www.lms.kantiana.ru), обеспечивающую разработку и комплексное использование электронных образовательных ресурсов;
- серверное программное обеспечение, необходимое для функционирования сервера и связи с системой электронного обучения через Интернет;
- корпоративная платформа
- установленное на рабочих местах обучающихся ПО: Microsoft Windows 10, Microsoft Office Standart 2010, антивирусное программное обеспечение Kaspersky Endpoint Security.
- GNU C++;
- Oracle Java;
- Deductor.
- среда разработки (JVE), компилятор (JVK) и виртуальная машина для исполнения кода (JVM) фирмы Oracle.
- Python 2.7.15 (Anaconda2 5.2.0 64-bit)
- Python 3.6.5 (Anaconda3 5.2.0 64-bit)

#### **12. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине.**

Для проведения занятий лекционного типа, практических и семинарских занятий используются специальные помещения (учебные аудитории), оборудованные техническими

средствами обучения – мультимедийной проекционной техникой. Для проведения занятий лекционного типа используются наборы демонстрационного оборудования.

Для проведения лабораторных работ, (практических занятий – при необходимости) используются специальные помещения (учебные аудитории), оснащенные специализированным лабораторным оборудованием: персональными компьютерами с возможностью выхода в интернет и с установленным программным обеспечением, заявленным в п.11.

Для проведения групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации используются специальные помещения (учебные аудитории), оборудованные специализированной мебелью (для обучающихся), меловой / маркерной доской.

Для организации самостоятельной работы обучающимся предоставляются помещения, оснащенные компьютерной техникой с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду университета.

Для обучения инвалидов и лиц с ограниченными возможностями здоровья университетом могут быть представлены специализированные средства обучения, в том числе технические средства коллективного и индивидуального пользования.

#### 4.10. Программа дисциплины «Функциональное программирование»

##### 1. Наименование дисциплины: «Функциональное программирование».

**Цель дисциплины:** знакомство слушателей с парадигмой функционального программирования на примере языка Хаскель. В рамках курса рассматриваются как базовые средства языка, относящиеся к «чистому» функциональному программированию, так и дополнительные его возможности: мутируемые структуры данных, макросы, потоки, средства объектно-ориентированного программирования. Также даётся короткий обзор математических основ функционального программирования.

##### 2. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы

Код компетенции	Результаты освоения образовательной программы (ИДК)	Результаты обучения по дисциплине
ОПК-4. Способен участвовать в разработке технической документации программных продуктов и комплексов с использованием стандартов, норм и правил, а также в управлении	ОПК-4.1. Участвует в разработке технической документации программных продуктов и комплексов с использованием стандартов, норм и правил ОПК-4.2. управляет проектами создания информационных систем на стадиях жизненного цикла	<b>Знать:</b> 1. особенности функционального программирования, 2. основные понятия лямбда-исчисления, 3. синтаксис и семантику базовых средств языка Хаскель и средств функционального ядра языка Лисп; <b>Уметь:</b> 1. программировать



<p>проектами создания информационных систем на стадиях жизненного цикла</p>		<p>алгоритмы обработки базовых структур данных на базе средств функциональной парадигмы программирования;  <b>Владеть:</b>  1. навыками составления функциональных программ на языках Лисп и Хаскель,  2. навыками анализа функциональных программ с целью повышения эффективности вычислений;</p>
---	--	--

### 3. Место дисциплины в структуре образовательной программы

«Функциональное программирование» представляет собой дисциплину обязательной части (Б1.О.13) направления подготовки бакалавриата 02.03.02 «Фундаментальная информатика и информационные технологии», профиль «Программная инженерия в искусственном интеллекте».

### 4. Виды учебной работы по дисциплине.

Виды учебной работы по дисциплине зафиксированы учебным планом основной профессиональной образовательной программы по указанному направлению и профилю, выражаются в академических часах. Часы контактной работы и самостоятельной работы обучающегося и часы, отводимые на процедуры контроля, могут различаться в учебных планах ОПОП по формам обучения. Объем контактной работы включает часы контактной аудиторной работы (лекции/практические занятия/ лабораторные работы), контактной внеаудиторной работы (контроль самостоятельной работы), часы контактной работы в период аттестации. Контактная работа, в том числе может проводиться посредством электронной информационно-образовательной среды университета с использованием ресурсов сети Интернет и дистанционных технологий

### 5. Содержание дисциплины, структурированное по темам (разделам)

Исходя из рамок, установленных учебным планом по трудоемкости и видам учебной работы по дисциплине, преподаватель самостоятельно выбирает тематику занятий по формам и количеству часов проведения контактной работы: лекции и иные учебные занятия, предусматривающие преимущественную передачу учебной информации преподавателем и (или) занятия семинарского типа (семинары, практические занятия, практикумы, лабораторные работы, коллоквиумы и иные аналогичные занятия), и (или) групповые консультации, и (или) индивидуальную работу обучающихся с преподавателем, в том числе индивидуальные консультации (по курсовым работам/проектам – при наличии курсовой работы/проекта по данной дисциплине в учебном плане). Рекомендуемая тематика занятий максимально полно реализуется в

контактной работе с обучающимися очной формы обучения. В случае реализации образовательной программы в заочной / очно-заочной форме трудоемкость дисциплины сохраняется, однако объем учебного материала в значительной части осваивается обучающимися в форме самостоятельной работы. При этом требования к ожидаемым образовательным результатам обучающихся по данной дисциплине не зависят от формы реализации образовательной программы.

№ п/п	Наименование разделов (тем) дисциплины	Содержание разделов (тем) дисциплин
•	Тема 1. Введение: особенности функционального программирования.	Основные парадигмы (модели) вычислений Функциональная парадигма и функциональные языки. Свойства функциональных языков: модульность, чистота (отсутствие побочных эффектов), прозрачность вычислений. Роль переменной в функциональном программировании. Достоинства и недостатки функциональной парадигмы.
•	Тема 2. Основные понятия лямбда-исчисления.	Синтаксис функциональных выражений в лямбда-исчислении. Операции абстракции и аппликации. Связанные и свободные переменные выражения. Понятие абстрактного синтаксиса. Правила вывода (редукции) в чистом лямбда-исчислении. Нормальный и аппликативный порядок редукции, их связь с энергичным и ленивым вычислением в функциональных языках. Реализация рекурсивных функций в лямбда-исчислении.
•	Тема 3. Функциональные средства языка Лисп.	Основные структуры данных языка Лисп: атомы и списки. Синтаксис функциональных выражений. Обычные и особые функции. Встроенные функции базового набора, средства определения новых функций, анонимные функции. Представление атомов и списков в памяти компьютера, понятие S-выражения. Виды рекурсивной обработки списков. Понятие функционала (функции высшего порядка). Основные встроенные функционалы. Проблема функционального аргумента, способы ее решения. Замыкание функционального аргумента. Функционалы с функциональным значением. Лисп: динамическое связывание и типизация.
•	Тема 4. Доказательство свойств рекурсивных функций.	Простая и обобщенная математическая индукция. Вполне- и обоснованно-упорядоченные множества. Способы введения порядка на множествах списочных выражений. Структурная индукция. Примеры доказательства правильности рекурсивных функций обработки списков.
•	Тема 5. Базовые средства языка Хаскель.	Особенности языка Хаскель: встроенное каррирование функций, ленивые вычисления, полиморфизм, статическая типизация и автоматический вывод типов. Особенности синтаксиса выражений. Встроенные типы данных и

	<p>операций: числовые, булевский, символьный, кортежи, списки. Охранные выражения. Конструкции LET и WHERE. Встроенные функционалы: отображения, фильтрация, левая и правая свертка. Анонимные функции. Полиморфные функции и типы. Классы типов. Синонимы типов. Конструкторы данных и конструкторы типов. Рекурсивные типы: списки, деревья. Списочные выражения, бесконечные списки.</p>
--	---

## 6. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине

Рекомендуемая тематика учебных занятий в форме контактной работы

**Рекомендуемая тематика учебных занятий лекционного типа**  
(предусматривающих преимущественную передачу учебной информации преподавателями):

№ п/п	Наименование разделов (тем) дисциплины	Содержание разделов (тем) дисциплин
•	Тема 1. Введение: особенности функционального программирования.	Основные парадигмы (модели) вычислений Функциональная парадигма и функциональные языки. Свойства функциональных языков: модульность, чистота (отсутствие побочных эффектов), прозрачность вычислений. Роль переменной в функциональном программировании. Достоинства и недостатки функциональной парадигмы.
•	Тема 2. Основные понятия лямбда-исчисления.	Синтаксис функциональных выражений в лямбда-исчислении. Операции абстракции и аппликации. Связанные и свободные переменные выражения. Понятие абстрактного синтаксиса. Правила вывода (редукции) в чистом лямбда-исчислении. Нормальный и аппликативный порядок редукции, их связь с энергичным и ленивым вычислением в функциональных языках. Реализация рекурсивных функций в лямбда-исчислении.
•	Тема 3. Функциональные средства языка Лисп.	Основные структуры данных языка Лисп: атомы и списки. Синтаксис функциональных выражений. Обычные и особые функции. Встроенные функции базового набора, средства определения новых функций, анонимные функции. Представление атомов и списков в памяти компьютера, понятие S-выражения. Виды рекурсивной обработки списков. Понятие функционала (функции высшего порядка). Основные встроенные функционалы. Проблема функционального аргумента, способы ее решения.

		Замыкание функционального аргумента. Функционалы с функциональным значением. Лисп: динамическое связывание и типизация.
•	Тема 4. Доказательство свойств рекурсивных функций.	Простая и обобщенная математическая индукция. Вполне- и обоснованно-упорядоченные множества. Способы введения порядка на множествах списочных выражений. Структурная индукция. Примеры доказательства правильности рекурсивных функций обработки списков.
•	Тема 5. Базовые средства языка Хаскель.	Особенности языка Хаскель: встроенное каррирование функций, ленивые вычисления, полиморфизм, статическая типизация и автоматический вывод типов. Особенности синтаксиса выражений. Встроенные типы данных и операций: числовые, булевский, символьный, кортежи, списки. Охранные выражения. Конструкции LET и WHERE. Встроенные функционалы: отображения, фильтрация, левая и правая свертка. Анонимные функции. Полиморфные функции и типы. Классы типов. Синонимы типов. Конструкторы данных и конструкторы типов. Рекурсивные типы: списки, деревья. Списочные выражения, бесконечные списки.

#### **Рекомендуемая тематика *практических* занятий:**

Тематика практических занятий совпадает с темами лекционных занятий.

На практических занятиях решаются задачи по теме занятия.

#### **Требования к самостоятельной работе обучающихся**

1. Работа с лекционным материалом, предусматривающая проработку конспекта лекций и учебной литературы, по всем темам из п. 6 настоящей рабочей программы.

2. Выполнение домашнего задания, предусматривающего решение задач, выполнение упражнений, выдаваемых на лабораторных занятиях, по всем темам из п. 6 настоящей рабочей программы.

Руководствуясь положениями статьи 47 и статьи 48 Федерального закона от 29 декабря 2012 г. N 273-ФЗ «Об образовании в Российской Федерации» научно-педагогические работники и иные лица, привлекаемые университетом к реализации данной образовательной программы, пользуются предоставленными академическими правами и свободами в части свободы преподавания, свободы от вмешательства в профессиональную деятельность; свободы выбора и использования педагогически обоснованных форм, средств, методов обучения и воспитания; права на творческую инициативу, разработку и применение авторских программ и методов обучения и воспитания в пределах реализуемой образовательной программы и отдельной дисциплины.

Исходя из рамок, установленных учебным планом по трудоемкости и видам учебной работы по дисциплине, преподаватель самостоятельно выбирает тематику занятий по формам и количеству часов проведения контактной работы: лекции и иные учебные занятия, предусматривающие преимущественную передачу учебной информации преподавателем и (или) занятия семинарского типа (семинары, практические занятия, практикумы, лабораторные работы, коллоквиумы и иные аналогичные занятия), и (или) групповые консультации, и (или) индивидуальную работу обучающихся с преподавателем, в том числе индивидуальные консультации (по курсовым работам/проектам – при наличии курсовой работы/проекта по данной дисциплине в учебном плане).

Рекомендуемая тематика занятий максимально полно реализуется в контактной работе с обучающимися очной формы обучения. В случае реализации образовательной программы в заочной / очно-заочной форме трудоемкость дисциплины сохраняется, однако объем учебного материала в значительной части осваивается обучающимися в форме самостоятельной работы. При этом требования к ожидаемым образовательным результатам обучающихся по данной дисциплине не зависят от формы реализации образовательной программы.

## **7. Методические рекомендации по видам занятий**

Лекционные занятия.

В ходе лекционных занятий обучающимся рекомендуется выполнять следующие действия. Вести конспектирование учебного материала. Обращать внимание на категории, формулировки, раскрывающие содержание тех или иных явлений и процессов, научные выводы и практические рекомендации по их применению. Задавать преподавателю уточняющие вопросы с целью уяснения теоретических положений, разрешения спорных ситуаций.

Желательно оставить в рабочих конспектах поля, на которых во внеаудиторное время можно сделать пометки из рекомендованной литературы, дополняющие материал прослушанной лекции, а также подчеркивающие особую важность тех или иных теоретических положений.

Лабораторные занятия.

На лабораторных занятиях в зависимости от темы занятия выполняется поиск информации по решению проблем, практические упражнения, контрольные работы, выработка индивидуальных или групповых решений, итоговое обсуждение с обменом знаниями, участие в круглых столах, разбор конкретных ситуаций, командная работа, представление портфолио и т.п.

Самостоятельная работа.

Самостоятельная работа осуществляется в виде изучения литературы, эмпирических данных по публикациям и конкретным ситуациям из практики, подготовке индивидуальных работ, работа с лекционным материалом, самостоятельное изучение отдельных тем дисциплины; поиск и обзор литературы и электронных источников; чтение и изучение учебника и учебных пособий.

## **8. Фонд оценочных средств**

## 8.1. Перечень компетенций с указанием этапов их формирования в процессе освоения образовательной программы в рамках учебной дисциплины

Основными этапами формирования указанных компетенций при изучении обучающимися дисциплины являются последовательное изучение содержательно связанных между собой тем учебных занятий. Изучение каждой темы предполагает овладение обучающимися необходимыми компетенциями. Результат аттестации обучающихся на различных этапах формирования компетенций показывает уровень освоения компетенций.

Контролируемые разделы (темы) дисциплины	Индекс контролируемой компетенции (или её части)	Оценочные средства по этапам формирования компетенций
		текущий контроль по дисциплине
Тема 1. Введение: особенности функционального программирования.	ОПК-4	контрольная работа
Тема 2. Основные понятия лямбда-исчисления.	ОПК-4	контрольная работа
Тема 3. Функциональные средства языка Лисп.	ОПК-4	контрольная работа
Тема 4. Доказательство свойств рекурсивных функций.	ОПК-4	контрольная работа
Тема 5. Базовые средства языка Хаскель.	ОПК-4	контрольная работа

## 8.2. Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений и навыков и (или) опыта деятельности в процессе текущего контроля

В качестве оценочных средств текущего контроля успеваемости предусмотрены:

контрольная работа

Контрольная работа № 1: состоит из 5-6 задач на составление рекурсивных функций на языке Лисп, включая разные виды рекурсии.

*Типовые задачи контрольной:*

- Определить функцию-предикат Double, которая в заданном списке дублирует каждый второй элемент верхнего уровня, например:  
(Double '(K(9)(45 G)()(W))=> (K(9)(9)(45 G)NIL NIL(W))
- Написать функцию-предикат Subset с двумя аргументами – множествами, записанными как списки из неповторяющихся атомов. Предикат определяет, является ли первое множество-аргумент подмножеством второго множества.  
Например: (Subset '(q r) '(p q s r))=> t .
- Определить функцию Subst0, заменяющую на всех уровнях заданного списочного выражения S все символьные (нечисловые) атомы, отличные от атома \* , на число 0: (Subst0 '((4 (Z(1))\*())Y))=> '((4 (0(1))\*())00).

Контрольная работа № 2: состоит из 5-6 задач на определение функций на языке Хаскель.

*Типовые задачи контрольной:*

- Объявить синоним типа для представления точки на плоскости и реализовать на языке Хаскель функцию, которая по четырем заданным точкам определяет, образуют ли они квадрат. Определить вспомогательные функции, если их использование улучшает наглядность решения.
- Используя функционалы, определить на Хаскеле функцию `joinLetters`, которая в заданном списке из кортежей-пар букв оставляет только пары с разными буквами, и преобразует эти пары в строки, вставляя между ними точку:  
`joinLetters [('A','S'),('S','S'),('N','V')] == ["A.S", "N.V"]`

### 8.3. Перечень вопросов и заданий для промежуточной аттестации по дисциплине

Вопросы к зачету

- Основные парадигмы (модели) вычислений. Функциональная парадигма и функциональные языки.
- Свойства функциональных языков: модульность, чистота (отсутствие побочных эффектов), прозрачность вычислений.
- Роль переменной в функциональном программировании. Достоинства и недостатки функциональной парадигмы.
- Синтаксис функциональных выражений в лямбда-исчислении. Операции абстракции и аппликации. Связанные и свободные переменные выражения. Понятие абстрактного синтаксиса.
- Правила вывода (редукции) в чистом лямбда-исчислении. Нормальный и аппликативный порядок редукции, их связь с энергичным и ленивым вычислением в функциональных языках. Реализация рекурсивных функций в лямбда-исчислении.
- Основные структуры данных языка Лисп: атомы и списки. Синтаксис функциональных выражений. Обычные и особые функции. Встроенные функции базового набора, средства определения новых функций, анонимные функции.
- Представление атомов и списков в памяти компьютера, понятие S-выражения. Виды рекурсивной обработки списков. Понятие функционала (функции высшего порядка). Основные встроенные функционалы.
- Проблема функционального аргумента, способы ее решения. Замыкание функционального аргумента. Функционалы с функциональным значением. Лисп: динамическое связывание и типизация.
- Простая и обобщенная математическая индукция. Вполне- и обоснованно-упорядоченные множества. Способы введения порядка на множествах списочных выражений.
- Структурная индукция. Примеры доказательства правильности рекурсивных функций обработки списков.

- Особенности языка Хаскель: встроенное каррирование функций, ленивые вычисления, полиморфизм, статическая типизация и автоматический вывод типов. Особенности синтаксиса выражений. Встроенные типы данных и операций: числовые, булевский, символьный, кортежи, списки.
- Охранные выражения. Конструкции LET и WHERE. Встроенные функционалы: отображения, фильтрация, левая и правая свертка. Анонимные функции. Полиморфные функции и типы.
- Классы типов. Синонимы типов. Конструкторы данных и конструкторы типов. Рекурсивные типы: списки, деревья. Списочные выражения, бесконечные списки.

#### 8.4. Планируемые уровни сформированности компетенций обучающихся и критерии оценивания

Уровни	Содержательное описание уровня	Основные признаки выделения уровня (этапы формирования компетенции, критерии оценки сформированности)	Пятибалльная шкала (академическая) оценка	Двухбалльная шкала, зачет	БРС, % освоения (рейтинговая оценка)
Повышенный	Творческая деятельность	<i>Включает нижестоящий уровень.</i> Умение самостоятельно принимать решение, решать проблему/задачу теоретического и прикладного характера на основе изученных методов, приемов, технологий	отлично	зачтено	86-100
Базовый	Применение знаний и умений в более широких контекстах учебной и профессиональной деятельности, нежели по образцу с большей степени самостоятельности и инициативы	<i>Включает нижестоящий уровень.</i> Способность собирать, систематизировать, анализировать и грамотно использовать информацию из самостоятельно найденных теоретических источников и иллюстрировать ими теоретические положения или обосновывать практику применения	хорошо		71-85
Удовлетворительный	Репродуктивный	Изложение в пределах	удовлетвор		55-70



тельный (достаточны й)	ая деятельность	задач теоретически практически контролируемого материала	курса и	ительно		
Недостаточн ый	Отсутствие признаков удовлетворительного уровня			неудовлетв орительно	не зачтено	Менее 55

## 9. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины.

### Основная литература

- Душкин, Р. В. Функциональное программирование на языке Haskell : практическое руководство / Р. В. Душкин. - 2-е изд. - Москва : ДМК Пресс, 2023. - 609 с. - ISBN 978-5-89818-623-4. - Текст : электронный. - URL: <https://znanium.com/catalog/product/2108510> (дата обращения: 20.11.2023). – Режим доступа: по подписке.
- Салмина, Н. Ю. Функциональное программирование и интеллектуальные системы : учебное пособие / Н. Ю. Салмина. - Томск : ФДО, ТУСУР, 2016. - 100 с. - Текст : электронный. - URL: <https://znanium.com/catalog/product/1845918> (дата обращения: 20.11.2023). – Режим доступа: по подписке.
  - Сенлорен, С. Введение в Elixir: введение в функциональное программирование / С. Сенлорен Д. Эйзенберг ; пер. с англ. А.Н. Киселева. - Москва : ДМК Пресс, 2017. - 262 с. - ISBN 978-5-97060-518-9. - Текст : электронный. - URL: <https://znanium.com/catalog/product/1027766> (дата обращения: 20.11.2023). – Режим доступа: по подписке.

### Дополнительная литература

- Абельсон Х., Сассман Дж. и др. Структура и интерпретация компьютерных программ. М.: Добросвет. 2010. 608 с.
- Харрисон Дж. Введение в функциональное программирование. Перевод с английского. Новосибирск. 2009. [PDF]. [<https://goo.gl/gDKPi5>]
- Филд А., Харрисон П. Функциональное программирование: Пер. с англ. – М.: Мир, 1993.
- Хювенен Э., Сеппянен Й. Мир Лиспа, Т.1. – М.: Наука, 1990.

## 10. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины (модуля).

- НЭБ Национальная электронная библиотека, диссертации и прочие издания
- eLIBRARY.RU Научная электронная библиотека, книги, статьи, тезисы докладов конференций
- Гребенников Электронная библиотека ИД журналы
- ЭБС Консультант студента

- ПРОСПЕКТ ЭБС
- ЭБС ZNANIUM.COM
- ЭБС IBOOKS.RU
- РГБ Информационное обслуживание по МБА
- БЕН РАН
- Электронно-библиотечная система (ЭБС) Кантиана (<https://elib.kantiana.ru/>)

## **11. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине.**

Программное обеспечение обучения включает в себя:

- система электронного образовательного контента БФУ им. И. Канта – [www.lms.kantiana.ru](http://www.lms.kantiana.ru), обеспечивающую разработку и комплексное использование электронных образовательных ресурсов;
- серверное программное обеспечение, необходимое для функционирования сервера и связи с системой электронного обучения через Интернет;
- корпоративная платформа
- установленное на рабочих местах обучающихся ПО: Microsoft Windows 10, Microsoft Office Standart 2010, антивирусное программное обеспечение Kaspersky Endpoint Security.
- GNU C++;
- Oracle Java;
- Deductor.
- среда разработки (JVE), компилятор (JVK) и виртуальная машина для исполнения кода (JVM) фирмы Oracle.
- Python 2.7.15 (Anaconda2 5.2.0 64-bit)
- Python 3.6.5 (Anaconda3 5.2.0 64-bit)

## **12. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине.**

Для проведения занятий лекционного типа, практических и семинарских занятий используются специальные помещения (учебные аудитории), оборудованные техническими средствами обучения – мультимедийной проекционной техникой. Для проведения занятий лекционного типа используются наборы демонстрационного оборудования.

Для проведения лабораторных работ, (практических занятий – при необходимости) используются специальные помещения (учебные аудитории), оснащенные специализированным лабораторным оборудованием: персональными компьютерами с возможностью выхода в интернет и с установленным программным обеспечением, заявленным в п.11.

Для проведения групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации используются специальные помещения (учебные аудитории), оборудованные специализированной мебелью (для обучающихся), меловой / маркерной доской.

Для организации самостоятельной работы обучающимся предоставляются помещения, оснащенные компьютерной техникой с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду университета.

Для обучения инвалидов и лиц с ограниченными возможностями здоровья университетом могут быть представлены специализированные средства обучения, в том числе технические средства коллективного и индивидуального пользования.

#### 4.11. Программа дисциплины: Тестирование и внедрение ПО.

##### Наименование дисциплины: «Тестирование и внедрение ПО»

**Целью** изучения дисциплины Тестирование и внедрение ПО является формирование у обучающихся компетенций, связанных с основными методами и технологиями тестирования программного обеспечения.

#### 2. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы

Код компетенции	Результаты освоения образовательной программы (ИДК)	Результаты обучения по дисциплине
ОПК-2 Способен применять компьютерные/ суперкомпьютерные методы, современное программное обеспечение, в том числе отечественного происхождения, для решения задач профессиональной деятельности	ОПК-2.1 Выбирает компьютерные/суперкомпьютерные методы для решения задач профессиональной деятельности;  ОПК-2.2 Использует современное программное обеспечение, в том числе отечественного происхождения, для решения задач профессиональной деятельности	<b>Знать:</b> теоретические основы технологии управления проектами по тестированию ПО, в том числе для решения задач анализа данных; теоретические основы технологии тестирования ПО. <b>Уметь:</b> использовать современные инструменты тестирования ПО в задачах анализа данных; использовать современные инструменты тестирования ПО и тестировать компьютерные программы. <b>Владеть:</b> навыком документирования технологии тестирования и создания собственных тест-кейсов

#### 3. Место дисциплины в структуре образовательной программы

«Тестирование и внедрение ПО» представляет собой дисциплину обязательной части (Б1.О.13) направления подготовки бакалавриата 02.03.02 «Фундаментальная информатика и информационные технологии», профиль «Программная инженерия в искусственном интеллекте».

#### 4. Виды учебной работы по дисциплине.

Виды учебной работы по дисциплине зафиксированы учебным планом основной профессиональной образовательной программы по указанному направлению и профилю, выражаются в академических часах. Часы контактной работы и самостоятельной работы студента и часы, отводимые на процедуры контроля, могут различаться в учебных планах ОПОП по формам обучения. Объем контактной работы включает часы контактной аудиторной работы (лекции/практические занятия/ лабораторные работы), контактной внеаудиторной работы (контроль самостоятельной работы), часы контактной работы в период аттестации. Контактная работа, в том числе может проводиться посредством электронной информационно-образовательной среды университета с использованием ресурсов сети Интернет и дистанционных технологий

#### 5. Содержание дисциплины, структурированное по темам (разделам)

Исходя из рамок, установленных учебным планом по трудоемкости и видам учебной работы по дисциплине, преподаватель самостоятельно выбирает тематику занятий по формам и количеству часов проведения контактной работы: лекции и иные учебные занятия, предусматривающие преимущественную передачу учебной информации преподавателем и (или) занятия семинарского типа (семинары, практические занятия, практикумы, лабораторные работы, коллоквиумы и иные аналогичные занятия), и (или) групповые консультации, и (или) индивидуальную работу обучающихся с преподавателем, в том числе индивидуальные консультации (по курсовым работам/проектам – при наличии курсовой работы/проекта по данной дисциплине в учебном плане). Рекомендуемая тематика занятий максимально полно реализуется в контактной работе со студентами очной формы обучения. В случае реализации образовательной программы в заочной / очно-заочной форме трудоемкость дисциплины сохраняется, однако объем учебного материала в значительной части осваивается студентами в форме самостоятельной работы. При этом требования к ожидаемым образовательным результатам студентов по данной дисциплине не зависят от формы реализации образовательной программы.

№	Наименование раздела	Содержание раздела
1	Тема 1. Основы технологии тестирования.	Основные понятия тестирования ПО. Правила проведения тестирования. История тестирования ПО. Релиз. Ведение статистики ошибок.
2	Тема 2. Документирование тестирования.	Определение тест-кейсов. Структура тест-кейса. Тест-кейсы, управляемые данными. Поддерживаемость тест-кейса. Количество идей, ожидаемых результатов в тест-кейсе. Проблемные тест-кейсы. Тест-комплекты. Состояния тест-кейса. Обзор тест-кейсов. Отчеты по тестированию. Идеи для написания тест-кейсов. Методология создания тест-

		кейсов. Методы генерирования тестов. Методы обзора тестов.
3	Тема 3. Виды тестирования, применяющиеся на различных этапах разработки.	Юнит-тестирование, модульное, интеграционное, системное, инсталляционное, статическое, юзабилити-тестирование, функциональное, альфа-, бета-тестирование, регрессионное, нагрузочное, производительности и др. Инструментальные средства поддержки технологии тестирования.
4	Тема 4. Технологии тестирования и этапы проекта разработки ПО.	Пути появления ошибок на различных этапах разработки. Цикл тестирования ПО и его связь с процессом разработки ПО. Планирование тестирования. Критерии начала и окончания тестирования. Управление тестированием. Исполнение тестирования. Документирование плана тестирования. Инструментальные средства составления плана тестирования.

#### **6. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине**

Рекомендуемая тематика учебных занятий *лекционного* типа (предусматривающих преимущественную передачу учебной информации преподавателями):

№	Наименование раздела	Темы лекций
1	Тема 1. Основы технологии тестирования.	Лекция 1. Основные понятия тестирования ПО.
2	Тема 2. Документирование тестирования.	Лекция 2. Структура тест-кейса.  Лекция 3. Методология создания тест-кейсов. Методы генерирования тестов. Методы обзора тестов.
3	Тема 3. Виды тестирования, применяющиеся на различных этапах разработки.	Лекция 4. Виды тестирования, применяющиеся на различных этапах разработки.  Лекция 5. Инструментальные средства поддержки технологии тестирования.
4	Тема 4. Технологии тестирования и этапы проекта разработки ПО.	Лекция 6. Технологии тестирования  Лекция 7. Этапы проекта разработки ПО

## Перечень тем лабораторных работ

1. Модульное тестирование
2. Тестирование производительности, нагрузочное тестирование
3. Статическое тестирование
4. Юзабилити-тестирование
5. Альфа- и бета-тестирование крупных комплексов программ
6. Разработка плана тестирования

## Требования к самостоятельной работе студентов

1. Работа с лекционным материалом, предусматривающая проработку конспекта лекций и учебной литературы, по всем темам из п. 6 настоящей рабочей программы.
2. Выполнение лабораторных работ, предусматривающих решение задач, по соответствующим темам из п. 6 настоящей рабочей программы.

Руководствуясь положениями статьи 47 и статьи 48 Федерального закона от 29 декабря 2012 г. N 273-ФЗ «Об образовании в Российской Федерации» научно-педагогические работники и иные лица, привлекаемые университетом к реализации данной образовательной программы, пользуются предоставленными академическими правами и свободами в части свободы преподавания, свободы от вмешательства в профессиональную деятельность; свободы выбора и использования педагогически обоснованных форм, средств, методов обучения и воспитания; права на творческую инициативу, разработку и применение авторских программ и методов обучения и воспитания в пределах реализуемой образовательной программы и отдельной дисциплины.

Исходя из рамок, установленных учебным планом по трудоемкости и видам учебной работы по дисциплине, преподаватель самостоятельно выбирает тематику занятий по формам и количеству часов проведения контактной работы: лекции и иные учебные занятия, предусматривающие преимущественную передачу учебной информации преподавателем и (или) занятия семинарского типа (семинары, практические занятия, практикумы, лабораторные работы, коллоквиумы и иные аналогичные занятия), и (или) групповые консультации, и (или) индивидуальную работу обучающихся с преподавателем, в том числе индивидуальные консультации (по курсовым работам/проектам – при наличии курсовой работы/проекта по данной дисциплине в учебном плане).

Рекомендуемая тематика занятий максимально полно реализуется в контактной работе со студентами очной формы обучения. В случае реализации образовательной программы в заочной / очно-заочной форме трудоемкость дисциплины сохраняется, однако объем учебного материала в значительной части осваивается студентами в форме самостоятельной работы. При этом требования к ожидаемым образовательным результатам студентов по данной дисциплине не зависят от формы реализации образовательной программы.

## 7. Методические рекомендации по видам занятий

Лекционные занятия.

В ходе лекционных занятий обучающимся рекомендуется выполнять следующие действия. Вести конспектирование учебного материала. Обращать внимание на категории, формулировки, раскрывающие содержание тех или иных явлений и процессов, научные выводы и практические рекомендации по их применению. Задавать преподавателю уточняющие вопросы с целью уяснения теоретических положений, разрешения спорных ситуаций.

Желательно оставить в рабочих конспектах поля, на которых во внеаудиторное время можно сделать пометки из рекомендованной литературы, дополняющие материал прослушанной лекции, а также подчеркивающие особую важность тех или иных теоретических положений.

Практические и семинарские занятия.

На практических и семинарских занятиях в зависимости от темы занятия выполняется поиск информации по решению проблем, практические упражнения, контрольные работы, выработка индивидуальных или групповых решений, итоговое обсуждение с обменом знаниями, участие в круглых столах, разбор конкретных ситуаций, командная работа, представление портфолио и т.п.

Самостоятельная работа.

Самостоятельная работа осуществляется в виде изучения литературы, эмпирических данных по публикациям и конкретных ситуаций из практики, подготовке индивидуальных работ, работа с лекционным материалом, самостоятельное изучение отдельных тем дисциплины; поиск и обзор литературы и электронных источников; чтение и изучение учебника и учебных пособий.

## 8. Фонд оценочных средств

### 8.1. Перечень компетенций с указанием этапов их формирования в процессе освоения образовательной программы в рамках учебной дисциплины

Основными этапами формирования указанных компетенций при изучении обучающимися дисциплины являются последовательное изучение содержательно связанных между собой тем учебных занятий. Изучение каждой темы предполагает овладение обучающимися необходимыми компетенциями. Результат аттестации обучающихся на различных этапах формирования компетенций показывает уровень освоения компетенций.

Контролируемые разделы (темы) дисциплины	Индекс контролируемой компетенции (или её части)	Оценочные средства по этапам формирования компетенций
		текущий контроль по дисциплине
Тема 1. Основы технологии тестирования.	ПКС-2	Письменный опрос, выполнение лабораторных работ
Тема 2. Документирование	ПКС-2	Письменный опрос, выполнение

Контролируемые разделы (темы) дисциплины	Индекс контролируемой компетенции (или её части)	Оценочные средства по этапам формирования компетенций
		текущий контроль по дисциплине
тестирования.		лабораторных работ
Тема 3. Виды тестирования, применяющиеся на различных этапах разработки.	ПКС-2	Письменный опрос, выполнение лабораторных работ
Тема 4. Технологии тестирования и этапы проекта разработки ПО.	ПКС-2	Письменный опрос, выполнение лабораторных работ

## 8.2. Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений и навыков и (или) опыта деятельности процессе текущего контроля Темы лабораторных работ

### 1. Модульное тестирование

#### Пример задания:

Выполнить инструментальным средством модульное тестирование программного обеспечения. Сформулировать спецификацию, которая будет проверяться данным тестированием, и список идей для тестирования данной спецификации. Сформировать тест-пакет, состоящий как минимум из пяти тест-кейсов. По разработанным тест-кейсам выполнить модульное тестирование с помощью инструментального средства. Сформировать отчеты по тестированию в виде стандартизованных бланков. По результатам тестирования сделать выводы и дать рекомендации.

### 2. Тестирование производительности, нагрузочное тестирование

#### Пример задания.

Выполнить инструментальным средством тестирование производительности, нагрузочное тестирование программного обеспечения. Сформулировать спецификацию, которая будет проверяться данным тестированием, и список идей для тестирования данной спецификации. Сформировать тест-пакет, состоящий как минимум из пяти тест-кейсов. По разработанным тест-кейсам выполнить тестирование производительности, нагрузочное тестирование с помощью инструментального средства. Сформировать отчеты по тестированию в виде стандартизованных бланков. По результатам тестирования сделать выводы и дать рекомендации.

### 3. Статическое тестирование

#### Пример задания:

Выполнить инструментальным средством статическое тестирование фрагментов кода известного программного продукта. Сформулировать спецификацию, которая будет проверяться данным тестированием, и список идей для тестирования данной спецификации. Сформировать тест-пакет, состоящий как минимум из пяти тест-кейсов. По разработанным тест-кейсам выполнить статическое тестирование с помощью инструментального средства. Сформировать отчеты по тестированию в виде стандартизованных бланков. По результатам тестирования сделать выводы, дать рекомендации, перечислить возможные способы устранения найденных ошибок.

### 4. Юзабилити-тестирование



### **Пример задания:**

Выполнить инструментальным средством юзабилити-тестирование приложения/сайта. Сформулировать спецификацию, которая будет проверяться данным тестированием, и список идей для тестирования данной спецификации. Сформировать тест-пакет, состоящий как минимум из пяти тест-кейсов. По разработанным тест-кейсам выполнить юзабилити-тестирование. Сформировать отчеты по тестированию в виде стандартизованных бланков. По результатам тестирования сделать выводы, предложить рекомендации разработчику приложения/сайта по улучшению эргономики.

### **5. Альфа- и бета-тестирование крупных комплексов программ**

#### **Пример задания:**

Изучить опыт альфа- и бета-тестирования крупных компаний – разработчиков ПО. Описать последовательность и результаты тестирования компаниями их программных продуктов. На примере своей программы провести альфа- и бета-тестирование и продемонстрировать поэтапно их ход выполнения. Сформировать отчеты по тестированию в виде бланков.

### **6. Разработка плана тестирования**

#### **Пример задания:**

Разработать план тестирования программного обеспечения, включающий: описание объекта тестирования, список функций и описание тестируемой системы и ее компонент в отдельности, окружение тестируемой системы (описание программно-аппаратных средств), стратегии тестирования (виды тестирования и их применение по отношению к объекту тестирования), последовательность проведения работ (подготовка, тестирование, анализ результатов в разрезе запланированных фаз разработки), критерии начала тестирования (готовность тестовой платформы, законченность разработки требуемого функционала, наличие всей необходимой документации), критерии окончания тестирования, необходимое для тестирования оборудование и программные средства (тестовый стенд и его конфигурация, программы для автоматизированного тестирования и т.д.), риски и пути разрешения.

## **Примерные вопросы к письменному опросу**

### **Тема 1. Основы технологии тестирования.**

Вопросы к опросу:

1. Основные понятия тестирования ПО.
2. Правила проведения тестирования.
3. История тестирования ПО.
4. Релиз.
5. Ведение статистики ошибок

### **Тема 2. Документирование тестирования.**

Вопросы к опросу:

1. Определение тест-кейсов.
2. Структура тест-кейса.
3. Тест-кейсы, управляемые данными.
4. Поддерживаемость тест-кейса.
5. Количество идей, ожидаемых результатов в тест-кейсе.

6. Проблемные тест-кейсы.
7. Тест-комплекты.
8. Состояния тест-кейса.
9. Обзор тест-кейсов.
10. Отчеты по тестированию.
11. Идеи для написания тест-кейсов.
12. Методология создания тест-кейсов.
13. Методы генерирования тестов.
14. Методы обзора тестов.

### **Тема 3. Виды тестирования, применяющиеся на различных этапах разработки.**

Вопросы к опросу:

1. Юнит-тестирование
2. Модульное тестирование
3. Интеграционное тестирование
4. Системное тестирование
5. Инсталляционное тестирование
6. Статическое тестирование
7. Юзабилити-тестирование
8. Функциональное тестирование
9. Альфа-, бета- тестирование
10. Регрессионное тестирование
11. Нагрузочное тестирование
12. Тестирование производительности.
13. Инструментальные средства поддержки технологии тестирования.

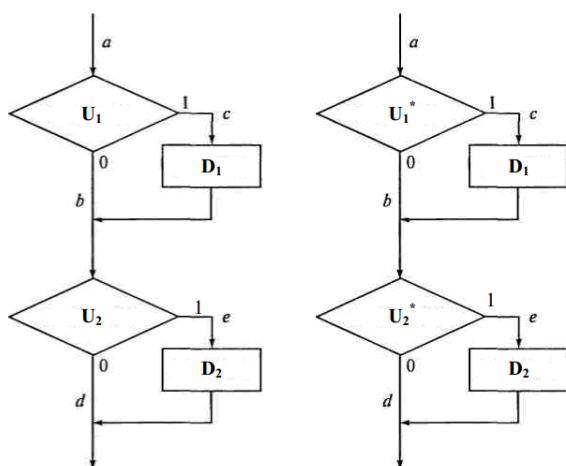
### **Тема 4. Технологии тестирования и этапы проекта разработки ПО.**

Вопросы к опросу:

1. Пути появления ошибок на различных этапах разработки.
2. Цикл тестирования ПО и его связь с процессом разработки ПО.
3. Планирование тестирования.
4. Критерии начала и окончания тестирования.
5. Управление тестированием.
6. Исполнение тестирования.
7. Документирование плана тестирования.
8. Инструментальные средства составления плана тестирования.

#### **Пример задания для контрольной работы.**

U1	U2	D1	D2	*U1	*U2
(A>1) and (B=0)	(A=2) or (Y>1)	X=X/A	X=X+1	A>1 or B=2	A=2 or Y<3



Пример алгоритма программы:

а) без ошибок; б) с ошибками

Выполнить покрытие решений.

### 8.3. Перечень вопросов и заданий для промежуточной аттестации по дисциплине

#### Перечень вопросов для промежуточного контроля (зачета).

1. Тестирование ПО. История возникновения и становления, цели и задачи тестирования.
2. Уровни тестирования ПО. Характеристика каждого уровня. Стоимость обнаружения ошибки на разных стадиях тестирования.
3. Тестирование методом черного ящика. Особенности, достоинства, недостатки.
4. Тестирование методом белого ящика. Особенности, достоинства, недостатки.
5. Тестирование методом серого ящика. Особенности, достоинства, недостатки.
6. Отладка, исключения. Механизм работы, обработка. Примеры использования, достоинства, недостатки.
7. Разработка через тестирование. Обоснованность использования.
8. Юнит тестирование. Механизмы автоматизации тестирования.
9. Механизмы оценки тестирования. Метрики.
10. Статические и динамическое тестирование. Плюсы и минусы.
11. Жизненный цикл бага.
12. Тестирование в экстремальном программировании.
13. Системы отслеживания ошибок. Принцип работы, основные функции.
14. Стресс тестирование, тестирование производительности и нагрузочное тестирование.
15. Дымовое тестирование, сборочное тестирование, регрессионное тестирование.
16. Санити (DOA) тесты. Основная задача, принципы написания санити (DOA) тестов.
17. Функциональное тестирование, системное тестирование. Отличие функционального тестирования от сборочного.
18. Полевое тестирование, тестирование удобства использования, обоснование необходимости этих видов тестирования.

19. Альфа и бета тестирование. Особенности, задачи, отличие от других видов тестирования.

20. Рефакторинг.

#### 8.4. Планируемые уровни сформированности компетенций обучающихся и критерии оценивания

Уровни	Содержательное описание уровня	Основные признаки выделения уровня (этапы формирования компетенции, критерии оценки сформированности)	Пятибалльная шкала (академическая) оценка	Диагностическая шкала, зачет	БРС, % освоения (рейтинговая оценка)
Повышенный	Творческая деятельность	<i>Включает нижестоящий уровень.</i> Умение самостоятельно принимать решение, решать проблему/задачу теоретического и прикладного характера на основе изученных методов, приемов, технологий	отлично	зачтено	86 -100
Базовый	Применение знаний и умений в более широких контекстах учебной и профессиональной деятельности, нежели по образцу с большей степени самостоятельности и инициативы	<i>Включает нижестоящий уровень.</i> Способность собирать, систематизировать, анализировать и грамотно использовать информацию из самостоятельно найденных теоретических источников и иллюстрировать ими теоретические положения или обосновывать практику применения	хорошо		71 -85
Удовлетворительный (достаточный)	Репродуктивная деятельность	Изложение в пределах задач курса теоретически и практически контролируемого материала	удовлетворительно		55 -70
Недостаточный	Отсутствие признаков удовлетворительного уровня		неудовлетворительно	не зачтено	Менее 55

		ЛЬНО		
--	--	------	--	--

## **9. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины.**

### **Основная литература**

1. Абрамов, Е. С. Машинно-ориентированное программирование: Учебное пособие / Абрамов Е.С., Сидоров И.Д. - Таганрог:Южный федеральный университет, 2016. - 87 с.: ISBN 978-5-9275-2065-7. - Текст : электронный. - URL: <https://znanium.com/catalog/product/991870> (дата обращения: 30.03.2022). – Режим доступа: по подписке.

### **Дополнительная литература**

1. Гуськова, О.И. Объектно ориентированное программирование в Java : учебное пособие / О. И. Гуськова. - Москва : МПГУ, 2018. - 240 с. - ISBN 978-5-4263-0648-6. - Текст : электронный. - URL: <https://znanium.com/catalog/product/1020593> (дата обращения: 30.03.2022). – Режим доступа: по подписке.

## **10. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины (модуля).**

- НЭБ Национальная электронная библиотека, диссертации и прочие издания
- eLIBRARY.RU Научная электронная библиотека, книги, статьи, тезисы докладовконференций
- Гребенников Электронная библиотека ИД журналы
- ЭБС Консультант студента
- ПРОСПЕКТ ЭБС
- ЭБС ZNANIUM.COM
- ЭБС IBOOKS.RU
- РГБ Информационное обслуживание по МБА
- БЕН РАН
- Электронно-библиотечная система (ЭБС) Кантиана (<https://elib.kantiana.ru/>)

## **11. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине.**

Программное обеспечение обучения включает в себя:

- система электронного образовательного контента БФУ им. И. Канта – [www.lms.kantiana.ru](http://www.lms.kantiana.ru), обеспечивающую разработку и комплексное использование электронных образовательных ресурсов;
- серверное программное обеспечение, необходимое для функционирования сервера и связи с системой электронного обучения через Интернет;
- корпоративная платформа;
- установленное на рабочих местах студентов ПО: Microsoft Windows 7, Microsoft Office Standart 2010, антивирусное программное обеспечение Kaspersky Endpoint Security.

- Microsoft Visual Studio 2010 Service Pack 1
- Microsoft Visual Studio Professional 2015 с обновлениями

## **12. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине.**

Для проведения занятий лекционного типа, практических и семинарских занятий используются специальные помещения (учебные аудитории), оборудованные техническими средствами обучения – мультимедийной проекционной техникой. Для проведения занятий лекционного типа используются наборы демонстрационного оборудования.

Для проведения лабораторных работ, (практических занятий – при необходимости) используются специальные помещения (учебные аудитории), оснащенные специализированным лабораторным оборудованием: персональными компьютерами с возможностью выхода в интернет и с установленным программным обеспечением, заявленным в п.11.

Для проведения групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации используются специальные помещения (учебные аудитории), оборудованные специализированной мебелью (для обучающихся), меловой / маркерной доской.

Для организации самостоятельной работы обучающимся предоставляются помещения, оснащенные компьютерной техникой с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду университета.

Для обучения инвалидов и лиц с ограниченными возможностями здоровья университетом могут быть представлены специализированные средства обучения, в том числе технические средства коллективного и индивидуального пользования.

### **5. Программа практики**

«Не предусмотрена».

### **6. Программа итоговой аттестации по модулю**

Определение результатов освоения модуля на основе вычисления оценки по каждому элементу модуля.

Оценка по модулю рассчитывается по формуле:

$$R_j^{\text{мод}} = \frac{k_1 R_1 + k_2 R_2 + k_3 R_3 + \dots + k_n R_n + k_{\text{пр}} R_{\text{пр}} + R_{\text{кур}}}{k_1 + k_2 + k_3 + \dots + k_{\text{пр}}}$$

Где:

$R_j^{\text{мод}}$  – оценка по модулю

$k_1, k_2, k_3, \dots, k_n$  – зачетные единицы дисциплин, входящих в модуль

$k_{\text{пр}}$  – зачетные единицы по практике

$R_1, R_2, R_3, \dots, R_n$  – оценки по дисциплинам модуля

$R_{\text{пр}}$  – оценка по практике

$R_{\text{кур}}$  – оценка по курсовой работе

В случае, если по дисциплине предусмотрен зачет без оценки, то за оценку по дисциплине принимается «5».

В случае, если по модулю применяется балльно-рейтинговая система, то

$R_1, R_2, R_3, \dots, R_n$  – рейтинговые баллы студента по дисциплинам модуля

$R_{\text{пр}}$  – рейтинговые баллы студента по практике

$R_{\text{кур}}$  – рейтинговые баллы студента по курсовой работе