

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ  
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
"ВОЛГОГРАДСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ"**

Институт приоритетных технологий

Кафедра информационной безопасности

**РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ**

Наименование **Языки программирования для автоматизации задач**  
дисциплины (модуля): **информационной безопасности**

Уровень ОПОП: Специалитет

Специальность: 10.05.01 Компьютерная безопасность

Специализация: Безопасность компьютерных систем и сетей (по отрасли или в  
сфере профессиональной деятельности)

Форма обучения: Очная

Срок обучения: 2024 - 2030 уч. г.

Программа составлена в соответствии с требованиями ФГОС ВО по  
направлению подготовки 10.05.01 Компьютерная безопасность (приказ № 1459  
от 26.11.2020 г.) и учебного плана, утвержденного Ученым советом (от  
26.05.2023 г., протокол № 9)

Разработчики:

Попов Г. А., старший преподаватель

Программа рассмотрена и утверждена на заседании кафедры, протокол № 08 от  
30.08.2023 года

Зав. кафедрой



Какорина О. А.

## 1. Цель и задачи изучения дисциплины

Цель изучения дисциплины - Формирование у студентов достаточных теоретических знаний и практических навыков по использованию методов алгоритмизации в профессиональной деятельности, в том и числе, при их программной реализации на компьютерах.

Задачи дисциплины:

- Изучение основ теории программирования, алгоритмизации, описания данных и средств описания действий языков программирования.
- Изучение практического программирования, современных сред создания программных продуктов.

## 2. Место дисциплины в структуре ОПОП ВО

Дисциплина «Языки программирования для автоматизации задач информационной безопасности» относится к обязательной части учебного плана.

Дисциплина изучается на 1 курсе.

## 3. Требования к результатам освоения дисциплины

Процесс изучения дисциплины направлен на формирование компетенций, определенных учебным планом в соответствии с ФГОС ВО.

Выпускник должен обладать следующими общепрофессиональными компетенциями (ОПК):

- **ОПК-7 Способен создавать программы на языках высокого и низкого уровня, применять методы и инструментальные средства программирования для решения профессиональных задач, осуществлять обоснованный выбор инструментария программирования и способов организации программ**

Знания, умения, навыки, формируемые по компетенции в рамках дисциплины

Студент должен знать:

Алгоритмические основы программирования на языках высокого и низкого уровня; языки программирования высокого и низкого уровня; методы, реализуемые в современных инструментальных средствах программирования

Студент должен уметь:

осуществлять обоснованный выбор способов организации программ и инструментария программирования при решении профессиональных задач

Студент должен владеть навыками:

разработки алгоритмов для последующего создания программ на языках общего назначения; навыками использования типовых инструментальных средств программирования для решения профессиональных задач

## 4. Объем дисциплины и виды учебной работы

Вид учебной работы	Всего часов	Первый семестр	Второй семестр
<b>Контактная работа (всего)</b>	<b>152</b>	<b>68</b>	<b>84</b>
Лабораторные	68	34	34
Лекции	84	34	50
<b>Самостоятельная работа (всего)</b>	<b>64</b>	<b>4</b>	<b>60</b>
<b>Виды промежуточной аттестации</b>	<b>72</b>	<b>36</b>	<b>36</b>
Экзамен	72	36	36
<b>Общая трудоемкость часы</b>	<b>288</b>	<b>108</b>	<b>180</b>
<b>Общая трудоемкость зачетные единицы</b>	<b>8</b>	<b>3</b>	<b>5</b>

## 5. Содержание дисциплины

## **5.1. Содержание дисциплины: Лекции (84 ч.)**

### **Первый семестр. (34 ч.)**

Тема 1. Перегрузка методов. Регулярные выражения. (2 ч.)

Передача параметров в метод. Параметры ref и out.

Тема 2. Перегрузка методов. Регулярные выражения. (2 ч.)

Что такое регулярные выражения, для чего используются.

Тема 3. Перегрузка методов. Регулярные выражения. (2 ч.)

Класс Regex. Методы класса Regex.

Тема 4. Абстрактные классы. (2 ч.)

Абстрактные члены классов.

Абстрактные методы.

Абстрактные свойства.

Тема 5. Абстрактные классы. (2 ч.)

Отказ от реализации абстрактных членов. Пример абстрактного класса.

Тема 6. Абстрактные классы. (2 ч.)

Назначение абстрактных классов.

Общая форма объявления абстрактного свойства

Общая форма объявления абстрактного индекса

Общая форма объявления абстрактного события

Тема 7. Интерфейсы. Множественное наследование. (2 ч.)

Интерфейсы.

Множественное наследование.

Объявление интерфейса. Реализация интерфейса.

В чем отличие интерфейсов от абстрактных классов?

Тема 8. Интерфейсы. Множественное наследование. (2 ч.)

Две стратегии реализации интерфейса. Преобразование к классу интерфейса

Тема 9. Интерфейсы. Множественное наследование. (2 ч.)

Проблемы множественного наследования. Наследование от общего предка. Встроенные интерфейсы.

Тема 10. Перегрузка операторов. (2 ч.)

Перегрузка операторов. Пример перегрузки оператора бинарного сложения.

Тема 11. Перегрузка операторов. (2 ч.)

Правила перегрузки операторов.

Тема 12. Перегрузка операторов. (2 ч.)

Перегрузка операторов внутри класса. Перегрузка операторов ввода-вывода.

Тема 13. Сетевая модель. (2 ч.)

Стандарты. Основные принципы. Уровни модели.

Тема 14. Сетевая модель. (2 ч.)

Соответствие модели OSI и других моделей сетевого взаимодействия.

Тема 15. Сетевая модель. (2 ч.)

Семейство TCP/IP. Семейство IPX/SPX. Значение NULL. Оператор ??. Синтаксис, пример использования.

Тема 16. Структуры. Перечисления. Универсальный класс. (2 ч.)

Структуры в C#. Синтаксис объявления, пример использования.

Перечисления (enum) в C#. Особенности использования.

Тема 17. Структуры. Перечисления. Универсальный класс. (2 ч.)

Структура — тип значений, класс — ссылочный тип

Структуры не поддерживают наследование

Когда использовать структуры (struct), а когда классы (class) в C#

### **Второй семестр. (50 ч.)**

Тема 18. Состав среды разработки Microsoft Visual Studio. (2 ч.)  
Общие сведения. Достоинства и недостатки.

Тема 19. Состав среды разработки Microsoft Visual Studio. (2 ч.)  
Интегрированная среда разработки Visual Studio. Поддерживаемые технологии и языки программирования.

Тема 20. Состав среды разработки Microsoft Visual Studio. (2 ч.)  
Интерфейс и простейшее приложение в среде разработки Visual Studio.

Тема 21. Алгоритм. (2 ч.)  
Блок-схема алгоритма. Основные алгоритмические конструкции.

Тема 22. Алгоритм. (2 ч.)  
Цикл с параметром. Цикл с предусловием. Цикл с постусловием. Оператор ветвления.

Тема 23. Алгоритм. (2 ч.)  
Алгоритмические конструкции. Следование, цикл и ветвление. Оператор цикла.

Тема 24. Арифметические и логические операторы в языке C#. (2 ч.)  
Условные операторы и оператор выбора switch в языке C#. Тернарный оператор.

Тема 25. Арифметические и логические операторы в языке C#. (2 ч.)  
Условные операторы if..else. Оператор выбора switch. Тернарный оператор.

Тема 26. Арифметические и логические операторы в языке C#. (2 ч.)  
Класс System.Convert  
Арифметические операции  
Класс Math  
Функции округления  
Тригонометрические функции  
Разное  
Преобразование числа в строку  
Преобразование строки в число

Тема 27. Циклические конструкции в языке C#. (2 ч.)  
Конструкции while, do while, for, foreach. Операторы break и continue.

Тема 28. Циклические конструкции в языке C#. (2 ч.)  
Конструкции while, do while, for, foreach. Операторы break и continue.

Тема 29. Циклические конструкции в языке C#. (2 ч.)  
Конструкции while, do while, for, foreach. Операторы break и continue.

Тема 30. Массивы в языке C#. Класс List. (2 ч.)  
Одномерные, многомерные, ступенчатые массивы. Класс List.

Тема 31. Массивы в языке C#. Класс List. (2 ч.)  
Свойство Length. Внутри List. Добавление элемента в список. Зачем нужно поле \_version?

Тема 32. Массивы в языке C#. Класс List. (2 ч.)  
Прочие особенности методов List. Преимущества задания Capacity

Тема 33. Система типов языка C#. (2 ч.)  
Задание типов в объявлениях переменных  
Встроенные типы  
Пользовательские типы  
Система общих типов CTS

Тема 34. Система типов языка C#. (2 ч.)  
Типы литеральных значений  
Универсальные типы  
Неявные типы, анонимные типы и типы, допускающие значение NULL

Тема 35. Система типов языка C#. (2 ч.)  
bool, byte, sbyte, short, ushort, int, uint, long, ulong, float, double, decimal, char, string, object.

Тема 36. Преобразования типов. (2 ч.)

Класс Convert и его методы.

Тема 37. Преобразования типов. (2 ч.)

Операция преобразования. Сужающие и расширяющие преобразования.

Тема 38. Преобразования типов. (2 ч.)

Явные и неявные преобразования. Потеря данных и ключевое слово checked.

Тема 39. Построение выражений на языке C#. Операторы языка C#. (2 ч.)

Арифметические операторы. Операторы сравнения. Логические операторы.

Тема 40. Построение выражений на языке C#. Операторы языка C#. (2 ч.)

Битовые операторы и операторы сдвига. Операторы равенства.

Тема 41. Построение выражений на языке C#. Операторы языка C#. (2 ч.)

Инициализация свойств. Объявление методов и свойств с помощью лямбда выражений. Интерполяция строк.

Тема 42. Операторы равенства. (2 ч.)

Равенство типов значений. Равенство ссылочных типов. Равенство строк.

## **5.2. Содержание дисциплины: Лабораторные (68 ч.)**

### **Первый семестр. (34 ч.)**

Тема 1. Основы синтаксиса C# (2 ч.)

Минимальный код консольного приложения на языке C#.

Тема 2. Основы синтаксиса C# (2 ч.)

Объявление переменных и инициализация. Модификация переменных.

Тема 3. Арифметические операции (2 ч.)

Выражение Math.Pow(x, y),

Выражение Math.Sqrt(x)

Выражение Math.Abs(x)

Выражения Math.Cos(x), Math.Sin(x) и Math.Tan

Выражение Math.Exp(x)

Тема 4. Арифметические операции (2 ч.)

Примитивные типы данных. Ввод и вывод текстовых данных с консоли.

Тема 5. Конверсия типов (2 ч.)

Неявная конверсия.

Явная конверсия.

Тема 6. Конверсия типов (2 ч.)

Оператор ветвления. Оператор switch-case. Организация циклов.

Тема 7. Структурирование кода с помощью функций (2 ч.)

Функции и процедуры. Аргументы функций.

Тема 8. Структурирование кода с помощью функций (2 ч.)

Возвращаемое значение. Ключевое слово return в теле функции. Тип возвращаемого значения функции.

Тема 9. Понятие стека вызовов и рекурсии (2 ч.)

Аргументы командной строки. Пути передачи программе аргументов.

Тема 10. Понятие стека вызовов и рекурсии (2 ч.)

Рекурсивная функция.

Тема 11. Работа со строками в C# (2 ч.)

Создание строковых значений. Управляющая последовательность (escape sequence). Интерполяция.

Тема 12. Работа со строками в C# (2 ч.)

Форматные строки. Функция string.Format().

Тема 13. Работа со строками в C# (2 ч.)

Операции над строками. Конкатенация. Чтение отдельных символов. Вычисление длины строки.

Тема 14. Сравнение строк (2 ч.)

Сравнение строк, не обращая внимания на регистр символов. Функции `.StartsWith()` и `.EndsWith()`.

Тема 15. Сравнение строк (2 ч.)

Простейший способ проверки эквивалентности двух строк. Функция для сравнения строк `string.Compare`.

Тема 16. Обрезка по краям строки (2 ч.)

Пример нормализации. Функции `.TrimStart()` и `.TrimEnd()`.

Тема 17. Обрезка по краям строки (2 ч.)

Изменение регистра. Функция `.ToLower()`. Функция `.ToUpper()`.

**Второй семестр. (34 ч.)**

Тема 18. Поиск внутри строки (2 ч.)

Функция `.Contains()`. Функция `.IndexOf()`. Функция `.IndexOfAny()`

Тема 19. Поиск внутри строки (2 ч.)

Функции `.LastIndexOf()` и `.LastIndexOfAny()`. Извлечение подстрок. Функция `.Substring()`

Тема 20. Поиск внутри строки (2 ч.)

Вставка, удаление и замена частей строки. Функция `.Insert()`. Функция `.Remove()`. Функция `.Replace()`.

Тема 21. Ленивая обработка строк (2 ч.)

Специальный тип `StringBuilder` библиотеки C#. Функция `.Append()`. Функция `.AppendFormat()`. Функция `.AppendJoin()`

Тема 22. Ленивая обработка строк (2 ч.)

Функция `.Insert()`. Функция `.Replace()`. Функция `.Remove()`

Тема 23. Ленивая обработка строк (2 ч.)

Недостатки `StringBuilder`.

Тема 24. Файловый текстовый ввод/вывод (2 ч.)

Функция `File.ReadAllText()`

Функция `File.ReadAllLines()`

Функция `File.WriteAllText()`

Тема 25. Файловый текстовый ввод/вывод (2 ч.)

Пространство имён `System.IO`.

Функция `File.ReadAllText()`

Функция `File.ReadAllLines()`

Функция `File.WriteAllText()`

Тема 26. Файловый текстовый ввод/вывод (2 ч.)

Функция `File.ReadAllText()`

Функция `File.ReadAllLines()`

Функция `File.WriteAllText()`

Тема 27. Поточковый ввод/вывод (2 ч.)

Концепция потокового I/O. Понятие курсора.

Тема 28. Поточковый ввод/вывод (2 ч.)

Функция `File.OpenText()`. Функция `.ReadLine()`.

Тема 29. Поточковый ввод/вывод (2 ч.)

Функции `.Read()`. Функция `.ReadToEnd()`. Тип `StreamWriter`.

Тема 30. Байты и кодировки (2 ч.)

Понятие кодировки. ASCII и Unicode.

Тема 31. Двоичные потоки (2 ч.)

Текстовые и двоичные данные. Двоичный ввод/вывод. Функция `.Seek()`.

Тема 32. Буферизация потоков (2 ч.)

Определение буферизации. Тип `BufferedStream`.

### Тема 33. Объявление пользовательских типов (2 ч.)

Пример использования пользовательского типа. Выражения, которые могут выступать в качестве инициализатора пользовательского типа.

### Тема 34. Ссылочные и значимые типы данных (2 ч.)

Целые числовые типы (int, short и т.д.), числовые типы с плавающей точкой (float, double), типы bool и char. Строки и массивы, тип object, а так же пользовательские типы, созданные с помощью ключевого слова class.

## 6. Виды самостоятельной работы студентов по дисциплине

### Первый семестр (4 ч.)

Вид СРС: Подготовка рефератов (4 ч.)

Тематика заданий СРС:

Реферат – письменная работа объемом 8–10 страниц. Это краткое и точное изложение сущности какого-либо вопроса, темы.

Тему реферата студент выбирает из предложенных преподавателем или может предложить свой вариант. В реферате нужны развернутые аргументы, рассуждения, сравнения. Содержание темы излагается объективно от имени автора.

Функции реферата. Информативная, поисковая, справочная, сигнальная, коммуникативная. Степень выполнения этих функций зависит от содержательных и формальных качеств реферата и целей.

Требования к языку реферата. Должен отличаться точностью, краткостью, ясностью и простотой.

Структура реферата.

1. Титульный лист.
2. Оглавление (на отдельной странице). Указываются названия всех разделов (пунктов плана) реферата и номера страниц, указывающие начало этих разделов в тексте реферата.
3. Введение. Аргументируется актуальность исследования, т.е. выявляется практическое и теоретическое значение данного исследования. Далее констатируется, что сделано в данной области предшественниками, перечисляются положения, которые должны быть обоснованы. Обязательно формулируются цель и задачи реферата.
4. Основная часть. Подчиняется собственному плану, что отражается в разделении текста на главы, параграфы, пункты. План основной части может быть составлен с использованием различных методов группировки материала. В случае если используется чья-либо неординарная мысль, идея, то обязательно нужно сделать ссылку на того автора, у кого взят данный материал.
5. Заключение. Последняя часть научного текста. В краткой и сжатой форме излагаются полученные результаты, представляющие собой ответ на главный вопрос исследования.
6. Приложение. Может включать графики, таблицы, расчеты.
7. Библиография (список литературы). Указывается реально использованная для написания реферата литература. Названия книг располагаются по алфавиту с указанием их выходных данных.

При проверке реферата оцениваются:

- знание фактического материала, усвоение общих представлений, понятий, идей;
- характеристика реализации цели и задач исследования;
- степень обоснованности аргументов и обобщений;
- качество и ценность полученных результатов;
- использование литературных источников;
- культура письменного изложения материала;
- культура оформления материалов работы.

Темы рефератов:

1. Жизненный цикл программного обеспечения.
2. Методология систем
3. Безопасность программного обеспечения.

## Второй семестр (60 ч.)

Вид СРС: Подготовка презентации на заданную тему (60 ч.)

Тематика заданий СРС:

Мультимедийная (электронная/учебная) презентация - это логически связанная последовательность слайдов, объединенных одной тематикой и общими принципами оформления. Мультимедийная презентация представляет сочетание компьютерной анимации, графики, видео, музыки и звукового ряда, которые организованы в единую среду. Чаще всего демонстрация презентации проецируется на большом экране, реже - раздается собравшимся как печатный материал.

Алгоритм самостоятельной работы по подготовке презентации на заданную тему:

- 1) Ознакомьтесь с предлагаемыми темами презентаций.
- 2) Ознакомьтесь со списком рекомендуемой литературы и источников и подготовьте их для работы.
- 3) Повторите лекционный материал по теме презентации (при наличии).
- 4) Изучите материал, касающийся темы презентации не менее чем по двум-трем рекомендованным источникам.
- 5) Составьте план-сценарий презентации, запишите его.
- 6) Проработайте найденный материал, выбирая только то, что раскрывает пункты плана презентации.
- 7) Составьте, наберите на компьютере и распечатайте текст своего устного выступления. При защите презентации он и будет являться сценарием презентации.
- 8) Продумайте дизайн презентации.
- 9) Подготовьте медиафрагменты (аудио-, видеоматериалы, текст и т.п.)
- 10) Оформите презентацию в соответствии с рекомендациями. Обязательно учтите возможные типичные ошибки и постарайтесь избежать их при создании своей презентации. Внимательно проверьте текст на отсутствие ошибок и опечаток.
- 11) Проверьте на работоспособность все элементы презентации.
- 12) Прочтите текст своего выступления медленно вслух, стараясь запомнить информацию.
- 13) Восстановите последовательность изложения текста сообщения, пересказав его устно.
- 14) Еще раз устно проговорите своё выступление в соответствии с планом, теперь уже сопровождая своё выступление демонстрацией слайдов на компьютере, делая в тексте пометки в тех местах, где нужна смена слайда.
- 15) Будьте готовы ответить на вопросы аудитории по теме Вашего сообщения.

К критериям оценки самостоятельной работы по подготовке презентации относятся:

Критерии оценки содержания презентации:

- соответствие материала презентации заданной теме;
- грамотное использование терминологии;
- обоснованное применение эффектов визуализации и анимации;
- общая грамотность;
- логичность изложения материала, доказательность, аргументированность.

Критерии оценки оформления презентации:

- творческий подход к оформлению презентации;
- прослеживается обоснованная последовательность слайдов и информации на слайдах;
- необходимое и достаточное количество фото- и видеоматериалов, учет особенностей восприятия графической (иллюстративной) информации, корректное сочетание фона и графики;
- дизайн презентации не противоречит ее содержанию;
- грамотное соотнесение устного выступления и компьютерного сопровождения, общее впечатление от мультимедийной презентации.

Темы презентаций:

1. Модульное программирование.
2. Верификация и тестирование
3. Программные ошибки.



## 7. Тематика курсовых работ(проектов)

- 1 Основные алгоритмические конструкции в языках программирования.
- 2 Правила составления блок-схем алгоритмов.
- 3 Алгоритмы сортировок и поиска.

## 8. Фонд оценочных средств. Оценочные материалы

### 8.1. Показатели и критерии оценивания компетенций, шкалы оценивания

В рамках изучаемой дисциплины студент демонстрирует уровни овладения компетенциями:

**Повышенный уровень:**

обучающийся демонстрирует глубокое знание учебного материала; способен использовать сведения из различных источников для успешного исследования и поиска решения в нестандартных ситуациях; способен анализировать, проводить сравнение и обоснование выбора методов решения практико-ориентированных заданий

**Базовый уровень:**

обучающийся способен понимать и интерпретировать освоенную информацию; демонстрирует осознанное владение учебным материалом и учебными умениями, навыками и способами деятельности, необходимыми для решения практико-ориентированных заданий

**Пороговый уровень:**

обучающийся обладает необходимой системой знаний и владеет некоторыми умениями; демонстрирует самостоятельность в применении знаний, умений и навыков к решению учебных заданий на репродуктивном уровне

**Уровень ниже порогового:**

система знаний, необходимая для решения учебных и практико-ориентированных заданий, не сформирована; обучающийся не владеет основными умениями, навыками и способами деятельности

Уровень сформированности компетенции	Шкала оценивания для промежуточной аттестации	Шкала оценивания по БРС
	Экзамен, зачет с оценкой	
Повышенный	5 (отлично)	91 и более
Базовый	4 (хорошо)	71 – 90
Пороговый	3 (удовлетворительно)	60 – 70
Ниже порогового	2 (неудовлетворительно)	Ниже 60

Критерии оценки знаний студентов по дисциплине

Оценка	Показатели
--------	------------

Отлично	<p>Обучающийся демонстрирует:</p> <p>систематизированные, глубокие и полные знания по всем разделам учебной дисциплины, а также по основным вопросам, выходящим за ее пределы;</p> <p>точное использование научной терминологии, грамотное, логически правильное изложение ответа на вопросы;</p> <p>безупречное владение инструментарием учебной дисциплины, умение его эффективно использовать в постановке и решении научных и профессиональных задач;</p> <p>выраженную способность самостоятельно и творчески решать сложные проблемы в нестандартной ситуации;</p> <p>полное и глубокое усвоение основной, и дополнительной литературы, по изучаемой учебной дисциплине;</p> <p>умение свободно ориентироваться в теориях, концепциях и направлениях по изучаемой учебной дисциплине и давать им аналитическую оценку, использовать научные достижения других дисциплин;</p> <p>творческую самостоятельную работу на учебных занятиях, активное творческое участие в групповых обсуждениях, высокий уровень культуры исполнения заданий.</p>
Хорошо	<p>Обучающийся демонстрирует:</p> <p>систематизированные, глубокие и полные знания по всем разделам учебной дисциплины;</p> <p>использование научной терминологии, грамотное, логически правильное изложение ответа на вопросы, умение делать обоснованные выводы и обобщения;</p> <p>владение инструментарием учебной дисциплины (методами комплексного анализа, техникой информационных технологий), умение его использовать в постановке и решении научных и профессиональных задач;</p> <p>способность решать сложные проблемы в рамках учебной дисциплины; свободное владение типовыми решениями;</p> <p>усвоение основной и дополнительной литературы, рекомендованной рабочей программой по учебной дисциплине;</p> <p>умение ориентироваться в теориях, концепциях и направлениях по изучаемой учебной дисциплине и давать им аналитическую оценку;</p> <p>активную самостоятельную работу на учебных занятиях, систематическое участие в групповых обсуждениях, высокий уровень культуры исполнения заданий.</p>
Удов-летвори-тельно	<p>Обучающийся демонстрирует:</p> <p>достаточные знания в объеме рабочей программы по учебной дисциплине;</p> <p>использование научной терминологии, грамотное, логически правильно изложение ответа на вопросы, умение делать выводы без существенных ошибок;</p> <p>владение инструментарием учебной дисциплины, умение его использовать в решении учебных и профессиональных задач;</p> <p>способность самостоятельно применять типовые решения в рамках изучаемой дисциплины;</p> <p>усвоение основной литературы, рекомендованной рабочей программой по дисциплине;</p> <p>умение ориентироваться в базовых теориях, концепциях и направлениях по дисциплине;</p> <p>работу на учебных занятиях под руководством преподавателя, фрагментарное участие в групповых обсуждениях, достаточный уровень культуры исполнения заданий.</p>

Неудов- летвори- тельно	Обучающийся демонстрирует: фрагментарные знания в рамках изучаемой дисциплины; знания отдельных литературных источников, рекомендованных рабочей программой по учебной дисциплине; неумение использовать научную терминологию учебной дисциплины, наличие в ответе грубых, логических ошибок; пассивность на занятиях или отказ от ответа, низкий уровень культуры исполнения заданий.
-------------------------------	---

## 8.2. Вопросы, задания текущего контроля

В целях освоения компетенций, указанных в рабочей программе дисциплины, предусмотрены следующие вопросы, задания текущего контроля:

**- ОПК-7 Способен создавать программы на языках высокого и низкого уровня, применять методы и инструментальные средства программирования для решения профессиональных задач, осуществлять обоснованный выбор инструментария программирования и способов организации программ**

Студент должен знать:

Алгоритмические основы программирования на языках высокого и низкого уровня; языки программирования высокого и низкого уровня; методы, реализуемые в современных инструментальных средствах программирования

Вопросы, задания:

1. В чем отличие компилятора языка программирования и IDE?
2. int.Parse - это явная или неявная конверсия типов?
3. С помощью какого выражения можно проверить что переменная x находится в промежутке [a;b)?

Студент должен уметь:

осуществлять обоснованный выбор способов организации программ и инструментария программирования при решении профессиональных задач

Задания:

1. Каким образом можно организовать бесконечный цикл?
2. Являются ли операторы if-else и switch-case взаимозаменяемыми?
3. Имея пользовательский тип данных с большим числом полей, и передавая его в качестве аргумента в некоторую функцию, что будет эффективнее - объявить этот тип с помощью class или с помощью struct?

Студент должен владеть навыками:

разработки алгоритмов для последующего создания программ на языках общего назначения; навыками использования типовых инструментальных средств программирования для решения профессиональных задач

Задания:

1. Каким образом можно вернуть несколько однотипных значений из функции?
2. Можно ли вызвать функцию с не-void возвращаемым значением, не сохраняя результат в какую-либо переменную?
3. Какую информацию содержит синтаксическая конструкция, называемая “сигнатура функции”?

## 8.3. Вопросы промежуточной аттестации

**Первый семестр (Экзамен)**

1. Перегрузка методов. Передача параметров в метод. Параметры ref и out.

2. Что такое регулярные выражения, для чего используются.
3. Класс Regex. Методы класса Regex.
4. Виртуальные методы. Переопределение виртуальных методов.
5. Статические методы. Вызов метода из базового класса.
6. Абстрактные классы. Зачем нужны абстрактные классы?
7. Абстрактные методы. Абстрактные свойства.
8. Интерфейсы. Множественное наследование.
9. Объявление интерфейса. Реализация интерфейса. В чем отличие интерфейсов от абстрактных классов?
10. Перегрузка операторов. Переопределение методов Equals и GetHashCode.

### **Второй семестр (Экзамен)**

1. Разница между Equals и «==». Оператор равенства «==».
2. Метод Equals. Перегрузка метода Equals. Метод GetHashCode.
3. Структуры в C#. Синтаксис объявления, пример использования.
4. Перечисления (enum) в C#. Особенности использования, пример кода.
5. Универсальный класс в C#. Особенности использования, пример кода.
6. Сетевая модель OSI. Реальная сетевая модель.
7. Стек протоколов TCP/IP. Значение NULL. Оператор ??. Синтаксис, пример использования.
8. Протокол HTTP. Протокол HTTPS.
9. Структура HTTP-сообщения.
10. HTTP-заголовки. Тело HTTP сообщения. Работа с HTTP-заголовками.
11. Классы HttpRequest и HttpResponse. Пример кода.

### **8.4. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности**

Промежуточная аттестация обучающихся ведется непрерывно и включает в себя: для дисциплин, завершающихся (согласно учебному плану) зачетом/зачетом с оценкой (дифференцированным зачетом), – текущую аттестацию (контроль текущей работы в семестре, включая оценивание промежуточных результатов обучения по дисциплине, – как правило, по трем модулям) и оценивание окончательных результатов обучения по дисциплине;

для дисциплин, завершающихся (согласно учебному плану) экзаменом, – текущую аттестацию (контроль текущей работы в семестре, включая оценивание промежуточных результатов обучения по дисциплине, – как правило, по трем модулям) и семестровую аттестацию (экзамен) – оценивание окончательных результатов обучения по дисциплине.

По дисциплинам, завершающимся зачетом/зачетом с оценкой, по обязательным формам текущего контроля студенту предоставляется возможность набрать в сумме не менее 100 баллов.

Оценивание окончательных результатов обучения по дисциплине ведется по 100-балльной шкале, оценка формируется автоматически как сумма количества баллов, набранных обучающимся за выполнение заданий обязательных форм текущего контроля.

По дисциплинам, завершающимся экзаменом, по обязательным формам текущего контроля студенту предоставляется возможность набрать в сумме не менее 60 баллов.

Оценивание окончательных результатов обучения по дисциплине ведется по 100-балльной шкале, оценка формируется автоматически как сумма количества баллов, набранных обучающимся за выполнение заданий обязательных форм текущего контроля и количества баллов, набранных на семестровой аттестации (экзамене).

Система оценивания.

В соответствии с Положением о балльно-рейтинговой системе оценки успеваемости обучающихся Волгоградского государственного университета предусмотрена возможность предоставления студентам выполнения дополнительных заданий повышенной сложности (не включаемых в перечень обязательных и, соответственно, в перечень обязательного текущего контроля успеваемости) и получения за выполнение таких заданий «премиальных» баллов, - для поощрения обучающихся, демонстрирующих выдающие способности.

Оценка качества освоения образовательной программы включает текущий контроль успеваемости, промежуточную аттестацию обучающихся и государственную итоговую аттестацию выпускников.

Текущий контроль представляет собой проверку усвоения учебного материала теоретического и практического характера, регулярно осуществляемую на протяжении семестра. К основным формам текущего контроля можно отнести:

Форма текущего контроля: Контрольная работа

контрольные работы применяются для оценки знаний, умений, навыков по дисциплине или ее части. Контрольная работа, как правило, состоит из небольшого количества средних по трудности вопросов, задач или заданий, требующих поиска обоснованного ответа. Может занимать часть или полное учебное занятие с разбором правильных решений на следующем занятии.

Форма текущего контроля: Устный опрос, собеседование

устный опрос, собеседование являются формой оценки знаний и предполагают специальную беседу преподавателя с обучающимся на темы, связанные с изучаемой дисциплиной. Процедуры направлены на выяснение объема знаний, обучающегося по определенному разделу, теме, проблеме и т.п.

Форма текущего контроля: Письменные задания или лабораторные работы

письменные задания являются формой оценки знаний и предполагают подготовка письменного ответа, решение специализированной задачи, выполнение теста. являются формами контроля и средствами применения и реализации полученных обучающимися знаний, умений и навыков в ходе выполнения учебно-практической задачи, связанной с получением значимого результата с помощью реальных средств деятельности. Рекомендуются для проведения в рамках тем (разделов), наиболее значимых в формировании компетенций. Тест является простейшей формой контроля, направленной на проверку владения терминологическим аппаратом, современными информационными технологиями и конкретными знаниями в области фундаментальных и прикладных дисциплин. Тест состоит из небольшого количества элементарных задач; может предоставлять возможность выбора из перечня ответов; занимает часть учебного занятия (10–30 минут); правильные решения разбираются на том же или следующем занятии; частота тестирования определяется преподавателем.

Промежуточная аттестация, как правило, осуществляется в конце семестра и может завершать изучение, как отдельной дисциплины, так и ее раздела (разделов) /модуля (модулей). Промежуточная аттестация помогает оценить более крупные совокупности знаний, умений и навыков, в некоторых случаях – даже формирование определенных компетенций.

К формам промежуточного контроля можно отнести:

Форма промежуточной аттестации: Экзамен

экзамен по дисциплине или ее части имеет цель оценить сформированность компетенций, теоретическую подготовку студента, его способность к творческому мышлению, приобретенные им навыки самостоятельной работы, умение синтезировать полученные знания и применять их при решении практических задач. Форма проведения, как правило, предусматривает ответы на вопросы экзаменационного билета, выполнение которых направленно на проверку сформированности компетенций по соответствующей учебной дисциплине.

Методика формирования результирующей оценки:

Первый семестр

1. Контрольная работа - от 0 до 30 баллов
2. Устный опрос, собеседование - от 0 до 30 баллов
3. Письменные задания или лабораторные работы - от 0 до 40 баллов
4. Экзамен - от 0 до 40 баллов

Второй семестр

1. Контрольная работа - от 0 до 30 баллов
2. Устный опрос, собеседование - от 0 до 30 баллов
3. Письменные задания или лабораторные работы - от 0 до 40 баллов
4. Экзамен - от 0 до 40 баллов

## **9. Перечень основной и дополнительной учебной литературы**

### **9.1 Основная литература**

1. Казанский А.А. Программирование на visual c# 2013 [Электронный ресурс]: - Профессиональное образование, 2018. - 191 с. - Режим доступа: <http://www.biblio-online.ru/book/A12DB344-78CA-4224-99E4-EDEB728A5578>
2. Гуриков, С. Р. Введение в программирование на языке Visual C# [Электронный ресурс]: учебное - Москва:Форум : ИНФРА-М, 2018. - 447 с. - Режим доступа: <http://znanium.com/catalog/product/967691>

### **9.2 Дополнительная литература**

1. А. А. Мерсов, А. М. Русаков, В. В. Филатов Языки программирования : методические рекомендации [Электронный ресурс]: - «Лань : электронно-библиотечная система» (Мерсов, А. А. Языки программирования : методические рекомендации / А. А. Мерсов, А. М. Русаков, В. В. Филатов. — Москва : РТУ МИРЭА, 2022. — 83 с. — Текст : электронный // Лань : электронно-библиотечная система, 2022. - 83 с. - Режим доступа: <https://e.lanbook.com/book/256697>

В качестве учебно-методического обеспечения могут быть использованы другие учебные, учебно-методические и научные источники по профилю дисциплины, содержащиеся в электронно-библиотечных системах, указанных в п. 11.2 «Электронно-библиотечные системы».

### **9.3. Перечень ресурсов информационно-телекоммуникационной сети «Интернет»**

1. <http://ibooks.ru/> - Электронная библиотечная система учебной и научной литературы
2. <http://window.edu.ru/library> - Федеральный образовательный портал. Библиотека. Единое окно доступа к образовательным ресурсам
3. <http://new.volsu.ru/umnik> - Образовательный портал Волгоградского государственного университета «УМНИК»

## **10.Методические указания по освоению дисциплины для лиц с ОВЗ и инвалидов**

При необходимости обучения студентов-инвалидов и лиц с ограниченными возможностями здоровья аудиторные занятия могут быть заменены или дополнены изучением полнотекстовых лекций, презентаций, видео- и аудиоматериалов в электронной

информационно-образовательной среде (ЭИОС) университета. Индивидуальные задания подбираются в адаптированных к ограничениям здоровья формах (письменно или устно, в форме презентаций). Выбор методов обучения зависит от их доступности для инвалидов и лиц с ограниченными возможностями здоровья.

В целях реализации индивидуального подхода к обучению студентов, осуществляющих учебный процесс по индивидуальной траектории в рамках индивидуального учебного плана (при необходимости), изучение данной дисциплины базируется на следующих возможностях:

- индивидуальные консультации преподавателя;
- максимально полная презентация содержания дисциплины в ЭИОС (в частности, полнотекстовые лекции, презентации, аудиоматериалы, тексты для перевода и анализа и т.п.).

## **11. Перечень информационных технологий**

В учебном процессе активно используются информационные технологии с применением современных средств телекоммуникации; электронные учебники и обучающие компьютерные программы. Каждый обучающийся обеспечен неограниченным доступом к электронной информационно-образовательной среде (ЭИОС) университета. ЭИОС предоставляет открытый доступ к учебным планам, рабочим программам дисциплин (модулей), практик, к электронным библиотечным системам и электронным образовательным ресурсам.

### **11.1 Перечень программного обеспечения**

**(обновление производится по мере появления новых версий программы)**

Лицензионное программное обеспечение:

1. Microsoft Windows 7 Professional, 11 лицензий, номер 60357707
  2. Microsoft Windows 7 Home Premium, 1 лицензия, OEM-лицензия
  3. Microsoft Windows 8.1 Home, 1 лицензия OEM-лицензия
  4. Microsoft Office 2007 Standart, 1 лицензия, номер 43847745
  5. Microsoft Office 2016, 1 лицензия, Сублицензионный договор No 31604241628 от 21.11.16
  6. LibreOffice 12 лицензий (свободно-распространяемое программное обеспечение)
  7. FreeBSD, 1 лицензия FreeBSD license свободное программное обеспечение
  8. Oracle VM VirtualBox, 14 лицензий GNU GPL свободное программное обеспечение
  9. Mozilla FireFox, 13 лицензий Mozilla Public License 2.0 (MPL) свободное программное обеспечение
  10. Visual Studio Community 2017, 13 лицензий, учебное программное обеспечение
  11. Python 2.7, 13 лицензий PSFL (свободно-распространяемое программное обеспечение)
1. 7-zip – свободно-распространяемое программное обеспечение;
  2. Microsoft Windows 7 – лицензия No 49487352;
  3. Microsoft Office 2007 – лицензия No 44414438;
  4. Антивирус Kaspersky – P/N: KL4863RAUFQ;
  5. Adobe Acrobat Reader – открытая лицензия

**11.2 Современные профессиональные базы данных и информационно-справочные системы, в т.ч. электронно-библиотечные системы (обновление выполняется еженедельно)**

Название	Краткое описание	URL-ссылка
Научная электронная библиотека	Крупнейший российский информационный портал в области науки, технологии, медицины и образования.	<a href="http://elibrary.ru/">http://elibrary.ru/</a>
ЭБС "Лань"	Электронно-библиотечная система	<a href="https://e.lanbook.com/">https://e.lanbook.com/</a>
ЭБС Znanium.com	Электронно-библиотечная система	<a href="https://znanium.com/">https://znanium.com/</a>
ЭБС BOOK.ru	Электронно-библиотечная система	<a href="https://www.book.ru/">https://www.book.ru/</a>
ЭБС Юрайт	Электронно-библиотечная система	<a href="https://www.biblio-online.ru/">https://www.biblio-online.ru/</a>

Scopus	Scopus – крупнейшая единая база данных, содержащая аннотации и информацию о цитируемости рецензируемой научной литературы, со встроенными инструментами отслеживания, анализа и визуализации данных. В базе содержится 23700 изданий от 5000 международных издателей, в области естественных, общественных и гуманитарных наук, техники, медицины и искусства.	<a href="http://www.scopus.com/">http://www.scopus.com/</a>
Web of Science	Наукометрическая реферативная база данных журналов и конференций. С платформой Web of Science вы можете получить доступ к непревзойденному объему исследовательской литературы мирового класса, связанной с тщательно отобранным списком журналов, и открыть для себя новую информацию при помощи скрупулезно записанных метаданных и ссылок.	<a href="https://apps.webofknowledge.com/">https://apps.webofknowledge.com/</a>
КонсультантПлюс	Информационно-справочная система	<a href="http://www.consultant.ru/">http://www.consultant.ru/</a>
Гарант	Информационно-справочная система по законодательству Российской Федерации	<a href="http://www.garant.ru/">http://www.garant.ru/</a>
Научная библиотека ВолГУ им О.В. Иншакова		<a href="http://library.volsu.ru/">http://library.volsu.ru/</a>

## 12. Материально-техническое обеспечение дисциплины

Учебные аудитории для проведения занятий лекционного типа представляют собой специальные помещения, в состав которых входят специализированная мебель и технические средства обучения.

Специализированная мебель:

парта со скамьей- 70 шт.

учебные места - 140 шт.

рабочее место преподавателя (стол и стул) – 1 шт.

доска аудиторная-1 шт.

Демонстрационное оборудование:

1. Доска (магнитная, маркерная)

2. Проектор BENQ MX722

3. Экран для проектора

Технические средства обучения:

Ноутбук ACER AspireES1-523-294D, 15.6", AMDE1 7010

1.5ГГц, 4ГБ, 500ГБ, AMDRadeonR2

Учебные аудитории для проведения практических работ представляют собой компьютерные классы или лаборатории, оснащенные лабораторным оборудованием, в зависимости от степени сложности.

Специализированная мебель:

1. Столы – 8 шт.

2. стулья – 16 шт.

3. парта со скамьей – 8 шт.

4. рабочее место преподавателя (стол и стул) – 1 шт.

Демонстрационное оборудование:

1. Проектор BenQ MX 505

2. Экран проекционный



3. Доска (магнитная, маркерная)

Рабочие места на базе вычислительной техники (18 шт):

1. Моноблок VPS 5000 (16 шт.);

2. Ноутбук Acer AS5738G;

3. Ноутбук HP Pavilion экран 15,6” Intel Pentium N3540.

Сетевое оборудование:

1. Wi-Fi роутер ASUS RT-N10

2. Концентратор.

3. Комплексе "Сетевое оборудование "Cisco" часть 1

Помещения для самостоятельной работы обучающихся оснащены компьютерной техникой с возможностью подключения к сети «Интернет» и обеспечением доступа в ЭИОС ВолГУ.