

ЛИСТ ОДОБРЕНИЯ И СОГЛАСОВАНИЯ РАБОЧЕЙ ПРОГРАММЫ

Разработчик(и) программы:

доцент, к.ф.-м.н., Алпатов Алексей Викторович

Рецензент(ы):

(при наличии)

Рабочая программа дисциплины (модуля, практики)

Основы программирования

разработана в соответствии с ФГОС ВО:

Федеральный государственный образовательный стандарт высшего образования - бакалавриат по направлению подготовки 09.03.01 Информатика и вычислительная техника (приказ Минобрнауки России от 19.09.2017 г. № 929)

составлена на основании учебного плана:

09.03.01 Информатика и вычислительная техника

Профиль: Технологии разработки информационных систем обработки информации и управления

утвержденного учёным советом вуза от 31.05.2023 протокол № 10.

Рабочая программа одобрена на заседании кафедры:

Информатика и технология программирования

Зав. кафедрой, к.т.н., доцент Рыбанов Александр Александрович

СОГЛАСОВАНО:

Вечерний факультет

Председатель НМС факультета Лапшина С.В.

Протокол заседания НМС факультета № 1 от 30.08.2023 г.

Рабочая программа дисциплины (модуля, практики) актуализирована

1. ЦЕЛИ И ЗАДАЧИ ОСВОЕНИЯ ДИСЦИПЛИНЫ (МОДУЛЯ, ПРАКТИКИ). ВИД, ТИП ПРАКТИКИ, СПОСОБ И ФОРМА (ФОРМЫ) ЕЕ ПРОВЕДЕНИЯ.
Цель освоения дисциплины "Основы программирования": Углубление знаний о базовых понятиях и приемах программирования, применяемых на всех основных этапах разработки программ, методах программирования. Развитие и совершенствование у студентов умений и навыков использования как языков программирования, так и методов программирования. Формирование у студентов научного, творческого подхода к освоению технологий, методов и средств производства программного обеспечения.
Цели освоения учебной дисциплины соотносятся с общими целями ОП ВО.
Задачи изучения дисциплины: Изучение основ алгоритмического описания решения задач в различных формах представления. Изучение синтаксиса высокоуровневых языков программирования Си и Си++. Изучение процессов разработки программ на высокоуровневых языках программирования. Изучение типовых алгоритмов обработки данных в программах на высокоуровневом языке программирования. Изучение базовых принципов декомпозиции при разработке программных систем. Изучение инструментальной среды разработки и отладки приложений.
Дисциплина "Основы программирования" ориентирована на формирование знаний, умений и навыков, необходимых для выполнения обобщенной трудовой функции профессионального стандарта 06.035 – Разработчик Web и мультимедийных приложений (утвержден приказом Министерства труда и социальной защиты Российской Федерации от 18.01.2017 № 44н): С. Управление работами по созданию (модификации) и сопровождению информационных ресурсов (уровень квалификации 6).

2. МЕСТО ДИСЦИПЛИНЫ (МОДУЛЯ, ПРАКТИКИ) В СТРУКТУРЕ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ	
Цикл (раздел) ОП:	Б1.О
2.1	Требования к предварительной подготовке обучающегося:
2.1.1	Для изучения дисциплины “Основы программирования” необходимы знания, умения и навыки, полученные студентами в процессе изучения математики, основ информатики и алгоритмизации в рамках учебной программы средней школы.
2.2	Дисциплины (модули) и практики, для которых освоение данной дисциплины (модуля) необходимо как предшествующее:
2.2.1	Знания, умения и навыки, формируемые учебной дисциплиной "Основы программирования", необходимы для изучения следующих дисциплин: Аналитическое программное обеспечение, Базы данных, Компьютерная графика, Компьютерные методы обработки экспериментальных данных, Машинно-зависимые языки, Методы анализа нечеткой информации, Методы оптимизации, Метрология программного обеспечения, Многопоточные вычисления для автоматизированных систем обработки информации и управления, Мультимедийные технологии, Надежность и качество программного обеспечения, Операционные системы, Основы проектирования WEB-приложений, Основы системного программного обеспечения, Преддипломная практика, Проектирование и разработка программного обеспечения, Проектирование лингвистических систем, Проектирование человеко-машинного интерфейса, Математическая логика и теория сложности алгоритмов, Введение в проектирование автоматизированных систем обработки информации и управления, Разработка приложений для мобильных устройств, Учебная практика (ознакомительная практика), Учебная практика (эксплуатационная практика), Производственная практика (технологическая (проектно-технологическая) практика).
2.2.2	Архитектура ЭВМ
2.2.3	Электротехника и электроника
2.2.4	Сети и телекоммуникации
2.2.5	Задачи математической физики
2.2.6	Выполнение и защита выпускной квалификационной работы
3. КОМПЕТЕНЦИИ ОБУЧАЮЩЕГОСЯ, ФОРМИРУЕМЫЕ В РЕЗУЛЬТАТЕ ОСВОЕНИЯ ДИСЦИПЛИНЫ (МОДУЛЯ, ПРАКТИКИ)	
ОПК-2.1: Знать: современные информационные технологии и программные средства, в том числе отечественного производства, при решении задач профессиональной деятельности	
:	
Результаты обучения: знает: понятие алгоритма и классификацию алгоритмических языков и систем программирования; основные стадии жизненного цикла программного обеспечения представление о других (не процедурных) языках программирования; возможности современных операционных систем; технологию разработки алгоритмов и программ, методы отладки и решения задач на ЭВМ в различных режимах; языки программирования высокого уровня - Си и Си++; основы объектно-ориентированного подхода к программированию; визуальную среду, предназначенную для программирования на языках Си и Си++	
ОПК-2.2: Уметь: выбирать современные информационные технологии и программные средства, в том числе отечественного производства, при решении задач профессиональной деятельности	

:
<p>Результаты обучения: умеет: применять на практике современные технологии разработки алгоритмов и программ, языки программирования, методы тестирования, отладки и решения задач на ЭВМ; работать с современными системами программирования, включая объектно-ориентированные; программировать базовые алгоритмы на языке высокого уровня. разрабатывать программы в соответствии с выбранной моделью жизненного цикла использовать функции программного интерфейса операционных систем для управления ресурсами</p>
ОПК-2.3: Владеть: навыками применения современных информационных технологий и программных средств, в том числе отечественного производства, при решении задач профессиональной деятельности
:
<p>Результаты обучения: владеет: навыками алгоритмизации и программной реализации на языке высокого уровня решений практических задач навыками работы с языками процедурного и объектно-ориентированного программирования; навыками разработки и отладки программ не менее, чем на одном из алгоритмических процедурных языков программирования высокого уровня навыками разработки программ в водопадной модели жизненного цикла</p>
ОПК-5.1: Знать: основы системного администрирования, администрирования СУБД, современные стандарты информационного взаимодействия систем
:
<p>Результаты обучения: знает: понятие алгоритма и классификацию алгоритмических языков и систем программирования; основные стадии жизненного цикла программного обеспечения представление о других (не процедурных) языках программирования; возможности современных операционных систем; технологию разработки алгоритмов и программ, методы отладки и решения задач на ЭВМ в различных режимах; языки программирования высокого уровня - Си и Си++; основы объектно-ориентированного подхода к программированию; визуальную среду, предназначенную для программирования на языках Си и Си++</p>
ОПК-5.2: Уметь: выполнять параметрическую настройку ИС
:
<p>Результаты обучения: умеет: применять на практике современные технологии разработки алгоритмов и программ, языки программирования, методы тестирования, отладки и решения задач на ЭВМ; работать с современными системами программирования, включая объектно-ориентированные; программировать базовые алгоритмы на языке высокого уровня. разрабатывать программы в соответствии с выбранной моделью жизненного цикла использовать функции программного интерфейса операционных систем для управления ресурсами</p>
ОПК-5.3: Владеть: навыками инсталляции программного и аппаратного обеспечения информационных и автоматизированных систем
:
<p>Результаты обучения: владеет: навыками алгоритмизации и программной реализации на языке высокого уровня решений практических задач навыками работы с языками процедурного и объектно-ориентированного программирования; навыками разработки и отладки программ не менее, чем на одном из алгоритмических процедурных языков программирования высокого уровня навыками разработки программ в водопадной модели жизненного цикла навыками управления ресурсами вычислительной системы</p>
ОПК-8.1: Знать: основные языки программирования и работы с базами данных, операционные системы и оболочки, современные программные среды разработки информационных систем и технологий.
:
<p>Результаты обучения: знает: понятие алгоритма и классификацию алгоритмических языков и систем программирования; основные стадии жизненного цикла программного обеспечения представление о других (не процедурных) языках программирования; возможности современных операционных систем; технологию разработки алгоритмов и программ, методы отладки и решения задач на ЭВМ в различных режимах; языки программирования высокого уровня - Си и Си++; основы объектно-ориентированного подхода к программированию; визуальную среду, предназначенную для программирования на языках Си и Си++</p>
ОПК-8.2: Уметь: применять языки программирования и работы с базами данных, современные программные среды разработки информационных систем и технологий для автоматизации бизнес-процессов, решения прикладных задач различных классов, ведения баз данных и информационных хранилищ.

Результаты обучения: умеет: применять на практике современные технологии разработки алгоритмов и программ, языки программирования, методы тестирования, отладки и решения задач на ЭВМ;
 работать с современными системами программирования, включая объектно-ориентированные;
 программировать базовые алгоритмы на языке высокого уровня.
 разрабатывать программы в соответствии с выбранной моделью жизненного цикла
 использовать функции программного интерфейса операционных систем для управления ресурсами

ОПК-8.3: Владеть: навыками программирования, отладки и тестирования прототипов программно-технических комплексов задач

Результаты обучения: владеет: навыками алгоритмизации и программной реализации на языке высокого уровня решений практических задач
 навыками работы с языками процедурного и объектно-ориентированного программирования;
 навыками разработки и отладки программ не менее, чем на одном из алгоритмических процедурных языков программирования высокого уровня
 навыками разработки программ в водопадной модели жизненного цикла
 навыками управления ресурсами вычислительной системы

4. СТРУКТУРА, СОДЕРЖАНИЕ И ФОРМЫ КОНТРОЛЯ ДИСЦИПЛИНЫ (МОДУЛЯ, ПРАКТИКИ)

Код занятия	Наименование разделов и тем /вид занятия/	Семестр / Курс	Часов	Компетенции	Форма контроля (Наименование оценочного средства)
	Раздел 1. Лекционный курс				
1.1	Обзор языков программирования высокого уровня. Этапы решения задач на ЭВМ. Способы записи и требования к алгоритмам, базовые структуры. Теорема структуры и структурное программирование. Жизненный цикл программы. /Лек/	1	2	ОПК-2.1 ОПК-5.1 ОПК-8.1 ОПК-8.2 ОПК-8.3 ОПК-5.2 ОПК-5.3 ОПК-2.2 ОПК-2.3	экзамен
1.2	Структура программы на языке C++. Типы данных. Переменные. /Лек/	1	2	ОПК-2.1 ОПК-5.1 ОПК-8.1 ОПК-8.2 ОПК-8.3 ОПК-5.2 ОПК-5.3 ОПК-2.2 ОПК-2.3	экзамен
1.3	Операции в C++, приоритеты операций /Лек/	1	2	ОПК-2.1 ОПК-5.1 ОПК-8.1 ОПК-8.2 ОПК-8.3 ОПК-5.2 ОПК-5.3 ОПК-2.2 ОПК-2.3	экзамен
1.4	Ввод и вывод в стиле C и C++. /Лек/	1	2	ОПК-2.1 ОПК-5.1 ОПК-8.1 ОПК-8.2 ОПК-8.3 ОПК-5.2 ОПК-5.3 ОПК-2.2 ОПК-2.3	экзамен

1.5	Программирование разветвляющихся алгоритмов на языке C++ /Лек/	1	2	ОПК-2.1 ОПК-5.1 ОПК-8.1 ОПК-8.2 ОПК-8.3 ОПК-5.2 ОПК-5.3 ОПК-2.2 ОПК-2.3	экзамен
1.6	Программирование циклических алгоритмов на языке C++. Оператор goto. /Лек/	1	2	ОПК-2.1 ОПК-5.1 ОПК-8.1 ОПК-8.2 ОПК-8.3 ОПК-5.2 ОПК-5.3 ОПК-2.2 ОПК-2.3	экзамен
1.7	Компиляторы. Интегрированные среды разработки. Отладка программ. Препробессор. Макросы. /Лек/	1	2	ОПК-2.1 ОПК-5.1 ОПК-8.1 ОПК-8.2 ОПК-8.3 ОПК-5.2 ОПК-5.3 ОПК-2.2 ОПК-2.3	экзамен
1.8	Указатели. Арифметика указателей. Динамическое выделение памяти. /Лек/	1	2	ОПК-2.1 ОПК-5.1 ОПК-8.1 ОПК-8.2 ОПК-8.3 ОПК-5.2 ОПК-5.3 ОПК-2.2 ОПК-2.3	экзамен
1.9	Организация памяти на C++. Время жизни и область видимости переменных. /Ср/	1	2	ОПК-2.1 ОПК-5.1 ОПК-8.1 ОПК-8.2 ОПК-8.3 ОПК-5.2 ОПК-5.3 ОПК-2.2 ОПК-2.3	контрольная работа
1.10	Одномерные массивы. Связь массивов и указателей. Динамическое выделение памяти под одномерные массивы. Генерация псевдослучайных чисел. /Ср/	1	2	ОПК-2.1 ОПК-5.1 ОПК-8.1 ОПК-8.2 ОПК-8.3 ОПК-5.2 ОПК-5.3 ОПК-2.2 ОПК-2.3	контрольная работа
1.11	Сортировка массивов: методы вставки, выбора, пузырьком. Сложность алгоритмов сортировки. /Ср/	1	2	ОПК-2.1 ОПК-5.1 ОПК-8.1 ОПК-8.2 ОПК-8.3 ОПК-5.2 ОПК-5.3 ОПК-2.2 ОПК-2.3	контрольная работа

1.12	Двумерные массивы. Динамическое выделение памяти под двумерные массивы. /Ср/	1	2	ОПК-2.1 ОПК-5.1 ОПК-8.1 ОПК-8.2 ОПК-8.3 ОПК-5.2 ОПК-5.3 ОПК-2.2 ОПК-2.3	контрольная работа
1.13	Символьные массивы. Строки класса std::string. Алгоритмы обработки строк. /Ср/	1	2	ОПК-2.1 ОПК-5.1 ОПК-8.1 ОПК-8.2 ОПК-8.3 ОПК-5.2 ОПК-5.3 ОПК-2.2 ОПК-2.3	контрольная работа
1.14	Функции. Вызов функции. Передача параметров в функцию. Параметры по умолчанию. Указатель на функцию. Функции обратного вызова. /Ср/	1	2	ОПК-2.1 ОПК-5.1 ОПК-8.1 ОПК-8.2 ОПК-8.3 ОПК-5.2 ОПК-5.3 ОПК-2.2 ОПК-2.3	контрольная работа
1.15	Ссылки. Передача аргументов по ссылке. Встраиваемые функции. Главная функция программы. Перегрузка функций. Функции с переменным числом параметров. /Ср/	1	2	ОПК-2.1 ОПК-5.1 ОПК-8.1 ОПК-8.2 ОПК-8.3 ОПК-5.2 ОПК-5.3 ОПК-2.2 ОПК-2.3	контрольная работа
1.16	Рекурсия. Алгоритм Хоара сортировки массива. Тестирование функций. Макрос assert. Ключевое слово static_assert /Ср/	1	2	ОПК-2.1 ОПК-5.1 ОПК-8.1 ОПК-8.2 ОПК-8.3 ОПК-5.2 ОПК-5.3 ОПК-2.2 ОПК-2.3	контрольная работа
1.17	Структуры. Битовые поля структур. Выравнивание структурных переменных. /Ср/	1	2	ОПК-2.1 ОПК-5.1 ОПК-8.1 ОПК-8.2 ОПК-8.3 ОПК-5.2 ОПК-5.3 ОПК-2.2 ОПК-2.3	контрольная работа
1.18	Основы объектно-ориентированного подхода к программированию: понятия класса, объекта. /Ср/	1	2	ОПК-2.1 ОПК-5.1 ОПК-8.1 ОПК-8.2 ОПК-8.3 ОПК-5.2 ОПК-5.3 ОПК-2.2 ОПК-2.3	контрольная работа

1.19	Конструкторы, деструкторы. Перегрузка методов и операций. /Ср/	1	2	ОПК-2.1 ОПК-5.1 ОПК-8.1 ОПК-8.2 ОПК-8.3 ОПК-5.2 ОПК-5.3 ОПК-2.2 ОПК-2.3	контрольная работа
1.20	Наследование. Виртуальные функции. Абстрактные классы /Ср/	1	2	ОПК-2.1 ОПК-5.1 ОПК-8.1 ОПК-8.2 ОПК-8.3 ОПК-5.2 ОПК-5.3 ОПК-2.2 ОПК-2.3	контрольная работа
1.21	Объединения. Перечисления. Создание псевдонимов с помощью ключевого слова typedef и директивы using. /Ср/	1	2	ОПК-2.1 ОПК-5.1 ОПК-8.1 ОПК-8.2 ОПК-8.3 ОПК-5.2 ОПК-5.3 ОПК-2.2 ОПК-2.3	контрольная работа
1.22	Потоки данных. Методы для записи в файл. Методы для чтения из файлов /Ср/	1	2	ОПК-2.1 ОПК-5.1 ОПК-8.1 ОПК-8.2 ОПК-8.3 ОПК-5.2 ОПК-5.3 ОПК-2.2 ОПК-2.3	контрольная работа
1.23	Бинарный режим чтения и записи. Работа с каталогами /Ср/	1	2	ОПК-2.1 ОПК-5.1 ОПК-8.1 ОПК-8.2 ОПК-8.3 ОПК-5.2 ОПК-5.3 ОПК-2.2 ОПК-2.3	контрольная работа
1.24	Основные понятия теории графов. Деревья. Динамическое программирование /Ср/	1	2	ОПК-2.1 ОПК-5.1 ОПК-8.1 ОПК-8.2 ОПК-8.3 ОПК-5.2 ОПК-5.3 ОПК-2.2 ОПК-2.3	контрольная работа
1.25	Динамические структуры данных. Односвязные и двусвязные линейные списки. /Ср/	1	2	ОПК-2.1 ОПК-5.1 ОПК-8.1 ОПК-8.2 ОПК-8.3 ОПК-5.2 ОПК-5.3 ОПК-2.2 ОПК-2.3	контрольная работа

1.26	Циклический список. Дек. Очередь. Бинарные деревья /Ср/	1	2	ОПК-2.1 ОПК-5.1 ОПК-8.1 ОПК-8.2 ОПК-8.3 ОПК-5.2 ОПК-5.3 ОПК-2.2 ОПК-2.3	контрольная работа
1.27	Многофайловые проекты. Пространство имен. Препроцессор /Ср/	1	2	ОПК-2.1 ОПК-5.1 ОПК-8.1 ОПК-8.2 ОПК-8.3 ОПК-5.2 ОПК-5.3 ОПК-2.2 ОПК-2.3	контрольная работа
1.28	Приложения с графическим интерфейсом. Windows Forms /Ср/	1	2	ОПК-2.1 ОПК-5.1 ОПК-8.1 ОПК-8.2 ОПК-8.3 ОПК-5.2 ОПК-5.3 ОПК-2.2 ОПК-2.3	контрольная работа
1.29	Шаблоны функций и классов. Функторы. Лямбда-функции /Ср/	1	2	ОПК-2.1 ОПК-5.1 ОПК-8.1 ОПК-8.2 ОПК-8.3 ОПК-5.2 ОПК-5.3 ОПК-2.2 ОПК-2.3	контрольная работа
1.30	Обработка исключительных ситуаций. Умные указатели /Ср/	1	2	ОПК-2.1 ОПК-5.1 ОПК-8.1 ОПК-8.2 ОПК-8.3 ОПК-5.2 ОПК-5.3 ОПК-2.2 ОПК-2.3	контрольная работа
1.31	Стандартная библиотека шаблонов STL. Линейные контейнеры. /Ср/	1	2	ОПК-2.1 ОПК-5.1 ОПК-8.1 ОПК-8.2 ОПК-8.3 ОПК-5.2 ОПК-5.3 ОПК-2.2 ОПК-2.3	контрольная работа
1.32	Стандартная библиотека шаблонов STL. Ассоциативные контейнеры. Итераторы. /Ср/	1	2	ОПК-2.1 ОПК-5.1 ОПК-8.1 ОПК-8.2 ОПК-8.3 ОПК-5.2 ОПК-5.3 ОПК-2.2 ОПК-2.3	контрольная работа
Раздел 2. Практические занятия					

2.1	Разработка алгоритмов линейных и разветвляющихся вычислительных процессов /Пр/	1	2	ОПК-2.1 ОПК-5.1 ОПК-8.1 ОПК-8.2 ОПК-8.3 ОПК-5.2 ОПК-5.3 ОПК-2.2 ОПК-2.3	контрольная работа
2.2	Разработка алгоритмов циклических вычислительных процессов /Пр/	1	2	ОПК-2.1 ОПК-5.1 ОПК-8.1 ОПК-8.2 ОПК-8.3 ОПК-5.2 ОПК-5.3 ОПК-2.2 ОПК-2.3	контрольная работа
2.3	Типы данных, операции. /Пр/	1	2	ОПК-2.1 ОПК-5.1 ОПК-8.1 ОПК-8.2 ОПК-8.3 ОПК-5.2 ОПК-5.3 ОПК-2.2 ОПК-2.3	контрольная работа
2.4	Программирование циклических и разветвляющихся алгоритмов. /Пр/	1	2	ОПК-2.1 ОПК-5.1 ОПК-8.1 ОПК-8.2 ОПК-8.3 ОПК-5.2 ОПК-5.3 ОПК-2.2 ОПК-2.3	контрольная работа
2.5	Сегменты памяти. Указатели /Ср/	1	2	ОПК-2.1 ОПК-5.1 ОПК-8.1 ОПК-8.2 ОПК-8.3 ОПК-5.2 ОПК-5.3 ОПК-2.2 ОПК-2.3	контрольная работа
2.6	Программирование типовых алгоритмов обработки одномерных массивов /Ср/	1	2	ОПК-2.1 ОПК-5.1 ОПК-8.1 ОПК-8.2 ОПК-8.3 ОПК-5.2 ОПК-5.3 ОПК-2.2 ОПК-2.3	контрольная работа
2.7	Программирование типовых алгоритмов обработки двумерных массивов. /Ср/	1	2	ОПК-2.1 ОПК-5.1 ОПК-8.1 ОПК-8.2 ОПК-8.3 ОПК-5.2 ОПК-5.3 ОПК-2.2 ОПК-2.3	контрольная работа

2.8	Обработка строковых данных. /Ср/	1	2	ОПК-2.1 ОПК-5.1 ОПК-8.1 ОПК-8.2 ОПК-8.3 ОПК-5.2 ОПК-5.3 ОПК-2.2 ОПК-2.3	контрольная работа
2.9	Пользовательские функции. Рекурсивный вызов функции /Ср/	1	2	ОПК-2.1 ОПК-5.1 ОПК-8.1 ОПК-8.2 ОПК-8.3 ОПК-5.2 ОПК-5.3 ОПК-2.2 ОПК-2.3	контрольная работа
2.10	Обработка данных структурного типа в программах на языке C++. /Ср/	1	2	ОПК-2.1 ОПК-5.1 ОПК-8.1 ОПК-8.2 ОПК-8.3 ОПК-5.2 ОПК-5.3 ОПК-2.2 ОПК-2.3	контрольная работа
2.11	Основы объектно-ориентированного программирования /Ср/	1	2	ОПК-2.1 ОПК-5.1 ОПК-8.1 ОПК-8.2 ОПК-8.3 ОПК-5.2 ОПК-5.3 ОПК-2.2 ОПК-2.3	контрольная работа
2.12	Наследование /Ср/	1	2	ОПК-2.1 ОПК-5.1 ОПК-8.1 ОПК-8.2 ОПК-8.3 ОПК-5.2 ОПК-5.3 ОПК-2.2 ОПК-2.3	контрольная работа
2.13	Динамическое программирование. /Ср/	1	2	ОПК-2.1 ОПК-5.1 ОПК-8.1 ОПК-8.2 ОПК-8.3 ОПК-5.2 ОПК-5.3 ОПК-2.2 ОПК-2.3	контрольная работа
2.14	Динамические структуры данных /Ср/	1	2	ОПК-2.1 ОПК-5.1 ОПК-8.1 ОПК-8.2 ОПК-8.3 ОПК-5.2 ОПК-5.3 ОПК-2.2 ОПК-2.3	контрольная работа

2.15	Работа с файлами в C++ /Ср/	1	2	ОПК-2.1 ОПК-5.1 ОПК-8.1 ОПК-8.2 ОПК-8.3 ОПК-5.2 ОПК-5.3 ОПК-2.2 ОПК-2.3	контрольная работа
2.16	Шаблоны функций и классов. STL /Ср/	1	2	ОПК-2.1 ОПК-5.1 ОПК-8.1 ОПК-8.2 ОПК-8.3 ОПК-5.2 ОПК-5.3 ОПК-2.2 ОПК-2.3	контрольная работа
Раздел 3. Лабораторные работы					
3.1	Структура программы на языке C++. Встроенные типы данных. /Лаб/	1	4	ОПК-2.1 ОПК-5.1 ОПК-8.1 ОПК-8.2 ОПК-8.3 ОПК-5.2 ОПК-5.3 ОПК-2.2 ОПК-2.3	отчет по лабораторной работе
3.2	Операции на языке C++. Форматированный ввод и вывод. /Лаб/	1	4	ОПК-2.1 ОПК-5.1 ОПК-8.1 ОПК-8.2 ОПК-8.3 ОПК-5.2 ОПК-5.3 ОПК-2.2 ОПК-2.3	отчет по лабораторной работе
3.3	Программирование разветвляющихся алгоритмов. /Лаб/	1	4	ОПК-2.1 ОПК-5.1 ОПК-8.1 ОПК-8.2 ОПК-8.3 ОПК-5.2 ОПК-5.3 ОПК-2.2 ОПК-2.3	отчет по лабораторной работе
3.4	Программирование циклических алгоритмов. /Лаб/	1	4	ОПК-2.1 ОПК-5.1 ОПК-8.1 ОПК-8.2 ОПК-8.3 ОПК-5.2 ОПК-5.3 ОПК-2.2 ОПК-2.3	отчет по лабораторной работе
3.5	Знакомство с системами программирования на языках С и C++ /Лаб/	1	4	ОПК-2.1 ОПК-5.1 ОПК-8.1 ОПК-8.2 ОПК-8.3 ОПК-5.2 ОПК-5.3 ОПК-2.2 ОПК-2.3	отчет по лабораторной работе

3.6	Программирование алгоритмов обработки одномерных массивов /Лаб/	1	4	ОПК-2.1 ОПК-5.1 ОПК-8.1 ОПК-8.2 ОПК-8.3 ОПК-5.2 ОПК-5.3 ОПК-2.2 ОПК-2.3	отчет по лабораторной работе
3.7	Программирование алгоритмов обработки двумерных массивов /Ср/	1	4	ОПК-2.1 ОПК-5.1 ОПК-8.1 ОПК-8.2 ОПК-8.3 ОПК-5.2 ОПК-5.3 ОПК-2.2 ОПК-2.3	контрольная работа
3.8	Обработка строковых данных /Ср/	1	4	ОПК-2.1 ОПК-5.1 ОПК-8.1 ОПК-8.2 ОПК-8.3 ОПК-5.2 ОПК-5.3 ОПК-2.2 ОПК-2.3	контрольная работа
3.9	Программирование с использованием функций, определяемых пользователем. /Ср/	1	4	ОПК-2.1 ОПК-5.1 ОПК-8.1 ОПК-8.2 ОПК-8.3 ОПК-5.2 ОПК-5.3 ОПК-2.2 ОПК-2.3	контрольная работа
3.10	Программирование рекурсивных алгоритмов на языке С++ /Ср/	1	4	ОПК-2.1 ОПК-5.1 ОПК-8.1 ОПК-8.2 ОПК-8.3 ОПК-5.2 ОПК-5.3 ОПК-2.2 ОПК-2.3	контрольная работа
3.11	Использование структурного типа данных в программах на языке С++ /Ср/	1	4	ОПК-2.1 ОПК-5.1 ОПК-8.1 ОПК-8.2 ОПК-8.3 ОПК-5.2 ОПК-5.3 ОПК-2.2 ОПК-2.3	контрольная работа
3.12	Основы объектно-ориентированного программирования. /Ср/	1	4	ОПК-2.1 ОПК-5.1 ОПК-8.1 ОПК-8.2 ОПК-8.3 ОПК-5.2 ОПК-5.3 ОПК-2.2 ОПК-2.3	контрольная работа

3.13	Программирование с использованием динамических структур данных /Ср/	1	4	ОПК-2.1 ОПК-5.1 ОПК-8.1 ОПК-8.2 ОПК-8.3 ОПК-5.2 ОПК-5.3 ОПК-2.2 ОПК-2.3	контрольная работа
3.14	Работа с текстовыми файлами на языке С++ /Ср/	1	4	ОПК-2.1 ОПК-5.1 ОПК-8.1 ОПК-8.2 ОПК-8.3 ОПК-5.2 ОПК-5.3 ОПК-2.2 ОПК-2.3	контрольная работа
3.15	Разработка многофайловых проектов /Ср/	1	4	ОПК-2.1 ОПК-5.1 ОПК-8.1 ОПК-8.2 ОПК-8.3 ОПК-5.2 ОПК-5.3 ОПК-2.2 ОПК-2.3	контрольная работа
3.16	Шаблоны функций и классов /Ср/	1	4	ОПК-2.1 ОПК-5.1 ОПК-8.1 ОПК-8.2 ОПК-8.3 ОПК-5.2 ОПК-5.3 ОПК-2.2 ОПК-2.3	контрольная работа
Раздел 4. Решение задач с использованием различных типов данных языка С++ (контрольная работа)					
4.1	Программирование линейных и разветвляющихся вычислительных процессов /Ср/	1	20	ОПК-2.1 ОПК-5.1 ОПК-8.1 ОПК-8.2 ОПК-8.3 ОПК-5.2 ОПК-5.3 ОПК-2.2 ОПК-2.3	контрольная работа
4.2	Программирование циклических вычислительных процессов /Ср/	1	24	ОПК-2.1 ОПК-5.1 ОПК-8.1 ОПК-8.2 ОПК-8.3 ОПК-5.2 ОПК-5.3 ОПК-2.2 ОПК-2.3	контрольная работа
4.3	Программирование с использованием одно- и двумерных массивов /Ср/	1	24	ОПК-2.1 ОПК-5.1 ОПК-8.1 ОПК-8.2 ОПК-8.3 ОПК-5.2 ОПК-5.3 ОПК-2.2 ОПК-2.3	контрольная работа

4.4	Программирование с использованием строкового типа данных /Ср/	1	24	ОПК-2.1 ОПК-5.1 ОПК-8.1 ОПК-8.2 ОПК-8.3 ОПК-5.2 ОПК-5.3 ОПК-2.2 ОПК-2.3	контрольная работа
4.5	Подготовка к промежуточной итоговой аттестации /Экзамен/	1	4	ОПК-2.1 ОПК-5.1 ОПК-8.1 ОПК-8.2 ОПК-8.3 ОПК-5.2 ОПК-5.3 ОПК-2.2 ОПК-2.3	экзамен
Раздел 5. Разработка СУБД средствами языка Си (контрольная работа)					
5.1	Описание форматов представления и хранения данных в программе /Ср/	1	16	ОПК-2.1 ОПК-5.1 ОПК-8.1 ОПК-8.2 ОПК-8.3 ОПК-5.2 ОПК-5.3 ОПК-2.2 ОПК-2.3	контрольная работа
5.2	Структурная декомпозиция программной системы с описанием модулей и функций /Ср/	1	22	ОПК-2.1 ОПК-5.1 ОПК-8.1 ОПК-8.2 ОПК-8.3 ОПК-5.2 ОПК-5.3 ОПК-2.2 ОПК-2.3	контрольная работа
5.3	Разработка программы, реализующей решение задания /Ср/	1	26	ОПК-2.1 ОПК-5.1 ОПК-8.1 ОПК-8.2 ОПК-8.3 ОПК-5.2 ОПК-5.3 ОПК-2.2 ОПК-2.3	контрольная работа
5.4	Подготовка к промежуточной итоговой аттестации /Экзамен/	1	4	ОПК-2.1 ОПК-5.1 ОПК-8.1 ОПК-8.2 ОПК-8.3 ОПК-5.2 ОПК-5.3 ОПК-2.2 ОПК-2.3	экзамен

Примечание. Формы контроля: Эк – экзамен, К- контрольная работа, Ко- контрольный опрос, Сз- семестровое задание, З-зачет, ОП -отчет по практике.

5. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

Оценочные средства планируемых результатов обучения представлены в виде фондов оценочных средств (ФОС), разработанных в соответствии с локальным нормативным актом университета. ФОС может быть представлен в Приложении к рабочей программе.

Контрольные вопросы для проведения текущего контроля и промежуточной аттестации по итогам освоения дисциплины:
Рабочая программа дисциплины обеспечена фондом оценочных средств для проведения текущего контроля и

промежуточной аттестации (см. приложение).

Вопросы к промежуточной аттестации по итогам освоения дисциплины:

Первый семестр

1. История развития языков программирования. Основные парадигмы программирования.
2. История создания языков C и C++. Стандарты языка C++. Стандартная библиотека C++.
3. Жизненный цикл программного обеспечения. Этапы решения задач на ЭВМ.
4. Алгоритмы: понятие, свойства и описание. Линейные, разветвляющиеся и циклические алгоритмы. Представление с помощью блок-схем.
5. Структура программы на языке C++. Главная функция. Препроцессор. Оформление комментариев в программе. Трансляторы. Компиляторы и интерпретаторы. Их различия. Этапы компиляции. Компиляторы языков C/C++.
7. IDE для программирования на языках C и C++. Основы работы в Visual Studio: создание консольного приложения, отладка программы.
8. Понятие переменной. Требования к идентификатору. Объявление, определение, инициализация.
9. Основные типы данных на языке C++. Целочисленные типы данных. Модификаторы. Тип данных с плавающей точкой. Логический тип данных. Символьный тип данных. Константный тип данных.
10. Арифметические операции. Явное и неявное приведение типов. Операции присваивания. Приоритет операций.
11. Логические операции и операции сравнения. Тернарный оператор.
12. Ввод и вывод в стиле C++. Флаги. Манипуляторы.
13. Ввод и вывод в стиле C. Функции printf(), scanf().
14. Реализация разветвляющихся алгоритмов с помощью операторов if и else.
15. Оператор выбора switch. Особенности применения.
16. Оператор безусловного перехода goto. Особенности применения. Критика goto.
17. Реализация циклических алгоритмов. Операторы continue и break.
18. Создание макросов. Отличие макросов от функций.
19. Понятие указателя. Назначение указателей. Инициализация указателя. Операции с указателями.
20. Указатель общего назначения. Особенности его применения. Указатель и ключевое слово const. Многоуровневая адресация.
21. Динамическое выделение памяти. Понятие кучи (heap). Особенности кучи.
22. Организация памяти программы на C++. Куча, стек, bss, data.
23. Особенности хранения переменных в оперативной памяти: область видимости и время жизни переменной, связывание.
24. Представление целых и вещественных чисел в оперативной памяти.
25. Побитовые операции.
26. Обработка времени и даты. Типы данных и функции из заголовочного файла <ctime> стандартной библиотеки C++.
27. Понятие массива. Создание статического одномерного массива. Инициализация.
28. Генерация псевдослучайных чисел. Функции rand() и srand(). Вихрь Мерсена. Заполнение случайными числами массивов.
29. Связь массива (одномерного и двумерного) с указателем. Константный массив.
30. Классификация алгоритмов обработки массивов. Алгоритмы поиска максимального и минимального элемента в массиве. Нахождение суммы элементов в массиве.
31. Двумерный массив. Инициализация двумерного массива. Вывод в консоль массива по строкам и столбцам.
32. Понятие вычислительной сложности алгоритма. Оценка сложности для различных способов сортировки массивов.
33. Сортировка одномерного массива методом выбора.
34. Сортировка одномерного массива методом вставки.
35. Сортировка одномерного массива методом «пузырька».
36. Динамическое выделение памяти под двумерные массивы. Освобождение памяти.
37. Пользовательские функции. Объявление, определение и прототип функции. Вызов функции.
38. Строки в стиле Си: определение, инициализация, кодирование символов. Терминатор строки, его роль при работе со строками.
39. Функции для работы со строками в стиле Си: взятие размера, объединения и копирования строк. Примеры.
40. Функции для работы со строками в стиле Си: сравнение строк, поиск подстроки. Примеры использования.
41. Реализация строк с помощью класса std::string. Инициализации строк. Конкатенация строк и сравнение строк.
42. Основные методы и функции для работы со строками string. Методы, которые изменяют размеры строки. Методы, которые изменяют содержание строки.

Второй семестр

1. Понятие функции в C++. Объявление и определение функции. Прототип функции. Вызов функции.
2. Понятие функции в C++. Порядок вызова функции. Порядок возврата функции.
3. Переменные в функции. Локальные и статические переменные.
4. Формальные и фактические параметры. Механизм передачи параметров в функцию. Передача параметров по значению и по адресу.
5. Понятие функции в C++. Указатель на функцию. Передача указателя на функцию аргументом. Пример использования.
6. Главная функция программы. Аргументы главной функции.
7. Ссылки: назначение, определение, использование. Ограничения при работе со ссылками.
8. Механизм передачи параметров в функцию. Передача параметров по константной и неконстантной ссылке.
9. Функции с произвольным числом параметров. Функции со значениями параметров по умолчанию.
10. Тестирование функций с помощью макроса assert. Применение утверждения static_assert.
11. Перегрузка функций. Вызов перегруженных функций. Ограничения на перегрузку функций.
12. Рекурсивный вызов функции: прямой и косвенный. Глубина рекурсии. Пример применения рекурсии.
13. Рекурсивный вызов функции. Алгоритм Хоара. Функция qsort().

14. Понятие структуры. Определение структуры. Определение переменной структурного типа. Инициализация переменной структурного типа. Выравнивание полей структуры.
15. Понятие структуры. Указатель на структуру. Массив из экземпляров структуры. Динамическое выделение памяти под массив. Получение доступа к полям структуры. Копирование структур.
16. Вложенные структуры. Объем памяти выделяемой под экземпляр структуры. Битовые поля структур. Ограничения при использовании битовых полей.
17. Объединения. Создание экземпляров типа union. Объем памяти, выделяемой при определении экземпляров. Инициализация полей.
18. Перечисляемый тип данных. Перечисления с областью видимости.
19. Оператор typedef. Создание псевдонимов типов с помощью using декларации.
20. Односвязный линейный список. Основные алгоритмы работы со списком данных: создание элемента, добавление, удаление.
21. Двусвязный линейный список. Основные алгоритмы работы со списком данных: создание элемента, добавление, удаление.
22. Стек. Применение стека. Дек.
23. Очередь. Применение очереди. Циклический список.
24. Деревья. Двоичные деревья. Сортировка с помощью двоичного дерева.
25. Объектно-ориентированное программирование. Основные принципы.
26. Понятие класса. Поля класса. Методы. Объекты.
27. Ограничение доступности компонент класса. Область видимости компонент.
28. Методы класса. Определение методов. Дружественные функции.
29. Конструкторы и деструкторы.
30. Ключевое слово const и классы. Константные методы. Константные ссылки.
31. Перегрузка методов и операций.
32. Статические компоненты класса. Статические методы. Указатель this.
33. Механизм наследования классов C++. Обращение к компонентам класса при наследовании. Видимость компонент при наследовании.
34. Множественное наследование.
35. Виртуальные функции. Абстрактные классы.
36. Базовый класс ios. Его наследники. Основные классы для файлового ввода и вывода.
37. Методы для чтения из файла и запись в файл.
38. Объектный ввод и вывод в файл. Методы read() и write().
39. Многофайловые проекты. Причины использования многофайловых проектов. Создание многофайловых проектов в MS Visual Studio.
40. Пространство имен. Стандартное пространство имен.
41. Препроцессор. Основные стадии препроцессорной обработки. Директивы препроцессора.
42. Основные директивы препроцессора: #include, #define, #if ...#else, #pragma.
43. Обработка исключительных ситуаций.
44. Шаблоны функций. Шаблоны классов.
45. Умные указатели: auto_ptr, unique_ptr и shared_ptr.
46. Функции. Лямбда-выражение.
47. Стандартная библиотека шаблонов STL. Контейнеры, алгоритмы, итераторы.
48. STL. Последовательные контейнеры: vector, list. Методы при работе с линейными контейнерами.
49. STL. Ассоциативные контейнеры: set, map. Методы для работы с ассоциативными контейнерами.
50. STL. Алгоритмы при работе с контейнерами.

В рамках освоения дисциплины «Основы программирования» используются следующие критерии оценивания знаний студентов по оценочным средствам:

Студент в результате выполнения и сдачи оценочного средства может получить следующие оценки.

Отлично

Полностью и правильно выполнено, и оформлено задание.

При отчете студент дал полные и правильные ответы на 90-100% задаваемых вопросов по теме работы.

Хорошо

Полностью и с небольшими неточностями выполнено и оформлено задание.

При отчете студент дал не полные и с небольшими ошибками ответы на все задаваемые вопросы по теме работы или доля правильных ответов составила 70 – 89%.

Удовлетворительно

Не полностью и с ошибками выполнено и оформлено задание.

При отчете студент дал не полные ответы и не на все задаваемые вопросы по теме работы. Доля правильных ответов составила 50 – 69%.

Неудовлетворительно

Студент не выполнил задание. Доля правильных ответов составила менее 50%.

Оценивание компетенций при изучении дисциплины «Основы программирования»

Исходя из 100-балльной (пятибалльной) системы оценивания системы оценки успеваемости студентов, в ходе освоения изучаемой дисциплины студент получает итоговую оценку, по которой оценивается уровень освоения компетенций.

90-100 баллов (отлично) повышенный уровень
Студент демонстрирует сформированность компетенций на повышенном уровне, обнаруживает всестороннее, систематическое и глубокое знание учебного материала, усвоил основную литературу и знаком с дополнительной литературой, рекомендованной программой, умеет свободно выполнять практические задания, предусмотренные программой, свободно оперирует приобретенными знаниями, умениями и навыками, применяет их в ситуациях повышенной сложности.

76-89 баллов (хорошо) базовый уровень

Студент демонстрирует сформированность дисциплинарной компетенций на базовом уровне: основные знания, умения и навыки освоены, но допускаются незначительные ошибки, неточности, затруднения при аналитических операциях, переносе знаний, умений и навыков на новые, нестандартные ситуации.

61-75 баллов (удовлетворительно) пороговый уровень

Студент демонстрирует сформированность компетенций на пороговом уровне: в ходе контрольных мероприятий допускаются значительные ошибки, проявляется отсутствие отдельных знаний, умений, навыков по компетенциям, студент испытывает значительные затруднения при оперировании знаниями, умениями и навыками при их переносе на новые ситуации

0-60 баллов (неудовлетворительно) уровень освоения компетенций ниже порогового

Компетенции не сформированы. Проявляется недостаточность знаний, умений, навыков.

6. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ, ПРАКТИКИ)

6.1. Рекомендуемая литература

	Авторы, составители	Заглавие	Издательство,	Электронный адрес
Л.1	Лясин Дмитрий Николаевич, Фадеева Марина Викторовна	Функции, определяемые пользователем, в языках Си и Си ++: Сборник «Методические указания». Выпуск 5	Волжский: ВПИ (филиал) ВолгГТУ, 2011	
Л.2	Зверева, И. Е., Савченко В. Ф.	Разработка программ с использованием строкового типа данных на языках С++ и Pascal: методические указания - http://lib.volpi.ru	Волжский: ВПИ (филиал) ВолгГТУ, 2011	http://lib.volpi.ru
Л.3	Лясин Д.Н., Абрамова О.Ф.	Использование рекурсивных вызовов в программах на языке Си: Сборник «Методические указания». Выпуск 5	Волгоград: ВолгГТУ, 2012	
Л.4	Лясин, Д.Н., Абрамова, О.Ф.	Знакомство с системами программирования на языках Си и Си++. Вып. 1 [Электронный ресурс] : методические указания - http://library.volpi.ru	Волгоград: ВолгГТУ, 2013	http://library.volpi.ru
Л.5	Лясин, Д.Н., Абрамова, О.Ф.	Работа с массивами в языках Си и Си++. Вып.3 [Электронный ресурс] : методические указания - http://library.volpi.ru	Волгоград: ВолгГТУ, 2013	http://library.volpi.ru
Л.6	Лясин, Д.Н., Абрамова, О.Ф.	Динамические структуры данных. Вып. 1 [Электронный ресурс] : методические указания - http://library.volpi.ru	Волгоград: ВолгГТУ, 2014	http://library.volpi.ru
Л.7	Лясин Д.Н., Фадеева М.В.	Потоковый ввод-вывод на языках Си и С++: методические указания к лабораторным работам	Волжский: , 2016	
Л.8	Лясин Д.Н., Фадеева М.В.	Основы программирования: метод. указания к контрольным работам	Волжский, 2016	
Л.9	Абрамова, О.Ф., Лясин, Д.Н.	Программирование и основы алгоритмизации [Электронный ресурс] : методические указания - http://library.volpi.ru	Волжский: [Б. и.], 2017	http://library.volpi.ru
Л.10	Лясин Д.Н., Фадеева М.В.	Практикум по алгоритмизации решения задач. Основы программирования на языках Си и Си++ : Учебное пособие	Волгоград: ВолгГТУ, 2016	
Л.11	Дьюхэрст, С.К.	Скользкие места С++. Как избежать проблем при проектировании и компиляции ваших программ [Электронный ресурс]: учебное пособие - https://e.lanbook.com/book/1229	М.: ДМК Пресс, 2009	https://e.lanbook.com/book/1229
Л.12	Потопахин, В.	Искусство алгоритмизации [Электронный ресурс] : https://e.lanbook.com/book/1269	М.: ДМК Пресс, 2011	https://e.lanbook.com/book/1269
Л.13	Дейл Н., Уимз Ч., Хедингтон М.	Программирование на С++ [Электронный ресурс]: https://e.lanbook.com/book/1219	Москва : ДМК Пресс, 2007	https://e.lanbook.com/book/1219
Л.14	Подбельский В.В., Фомин С.С.	Курс программирования на языке Си [Электронный ресурс]: учебник - https://e.lanbook.com/book/4148	Москва : ДМК Пресс, 2012	https://e.lanbook.com/book/4148

6.2. Перечень ресурсов информационно-телекоммуникационной сети "Интернет"

Э1	Электронная информационная образовательная среда по дисциплине "Основы программирования". - URL: https://eos2.vstu.ru
Э2	Электронно-библиотечная система ВПИ (филиал) ВолгГТУ - http://lib.volpi.ru:57772/csp/lib/StartEBS.csp?p=1
Э3	Электронно-библиотечная система ВолгГТУ. - URL: http://library.vstu.ru/ebsvstustaticpage?command=search
Э4	Электронно-библиотечная система "Лань". - URL: https://e.lanbook.com/
Э5	Электронно-библиотечная система "BOOK.RU". - URL: https://www.book.ru/
Э6	Научный журнал "Программирование" [электронный ресурс]. URL: http://www.ispras.ru/programming/
Э7	Научная электронная библиотека открытого доступа "КиберЛенинка". - URL: https://cyberleninka.ru/
Э8	Библиографическая и реферативная база данных "Scopus". - URL: https://www.scopus.com
Э9	Библиографическая и реферативная база данных "Web of Science". - URL: http://wokinfo.com/
Э10	Научная электронная библиотека "Elibrary.ru". - URL: https://elibrary.ru/defaultx.asp

6.3 Перечень программного обеспечения

6.3.1.1	Для успешного освоения дисциплины студент использует следующие программные средства:
6.3.1.2	MS Visual Studio 2010 (подписка Microsoft Imagine Premium ID df8605e9-c758-42d6-a856-ae0ba9714cc4, лицензионный договор № Tr000150654 от 07.07.2017г. (подписка на 2017-2018гг), лицензионный договор № КИС-193-2016 от 25.04.2016г. (подписка на 2016-2017гг), лицензионный договор № КИС-108-2015 от 07.04.2015г. (подписка на 2015-2016гг), лицензионный договор № КИС-099-2014 от 08.04.2014г. (подписка на 2014-2015гг), лицензионный договор № Tr018575 от 01.04.2013г. (подписка на 2013-2014гг), ежегодное продление);
6.3.1.3	QT Creator (GNU General Public License, https://ru.wikipedia.org/wiki/Qt_Creator)

6.4 Перечень информационных справочных систем и электронных библиотечных систем (ЭБС)

6.3.2.1	Единый реестр российских программ для электронных вычислительных машин и баз данных (бесплатный доступ). - url: https://reestr.minsvyaz.ru . Реестр создан в соответствии со статьей 12.1 Федерального закона «Об информации, информационных технологиях и о защите информации» в целях расширения использования российских программ для электронных вычислительных машин и баз данных, подтверждения их происхождения из Российской Федерации, а также в целях оказания правообладателям программ для электронных вычислительных машин или баз данных мер государственной поддержки.
6.3.2.2	Информационно-поисковая система федерального государственного учреждения «Федеральный институт промышленной собственности (бесплатный доступ). – url: http://www1.fips.ru/wps/wcm/connect/content_ru/ru/inform_resources/inform_retrieval_system . В информационно-поисковой системе возможен поиск по изобретениям, рефератам патентных документов на русском и английском языках, перспективным изобретениям, полезным моделям, товарным знакам, общеизвестным товарным знакам, наименованиям мест происхождения товаров, промышленным образцам, программам для ЭВМ, базам данных, топологиям интегральных микросхем, классификаторам и документам официальных бюллетеней за последний месяц.
6.3.2.3	Информационно-справочная система "Консультант Плюс" - http://www.consultant.ru/online/ (Общество с ограниченной ответственностью «Инженеры информации». Договор №207-К об оказании информационных услуг с использованием экземпляров Системы "Консультант Плюс");
6.3.2.4	Информационно-поисковая система всемирной организации по интеллектуальной собственности (бесплатный доступ).- url: https://patentscope.wipo.int/search/en/search.jsf
6.3.2.5	Информационно-справочная система Европейской патентной организации (бесплатный доступ). - url: http://www.espacenet.com/access/index.en.html . Позволяет произвести поиск патентных документов: Европейской патентной организации (ЕПО), Всемирной организации интеллектуальной собственности (WIPO), Японии, Австрии, Бельгии, Кипра, Дании, Финляндии, Франции, Германии, Греции, Ирландии, Италии, Лихтенштейна, Люксембурга, Монако, Нидерландов, Португалии, Испании, Швеции, Швейцарии, Англии.

7. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ, ПРАКТИКИ) /ОБОРУДОВАНИЕ

7.1	ВПИ (филиал) ВолгГТУ располагает материально-технической базой, соответствующей действующим противопожарным правилам и нормам, и обеспечивающей проведение всех видов учебной деятельности, предусмотренных учебным планом по дисциплине.
7.2	Специальные помещения представляют собой учебные аудитории для проведения занятий лекционного типа, занятий семинарского типа, курсового проектирования (выполнения курсовых работ), групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации, а также помещения для самостоятельной работы и помещения для хранения и профилактического обслуживания учебного оборудования.
7.3	Специальные помещения укомплектованы специализированной мебелью (учебная доска, посадочные места по количеству обучающихся, рабочее место преподавателя) и техническими средствами обучения, служащими для представления учебной информации большой аудитории
7.4	При проведении занятий лекционного типа, занятий семинарского типа, и групповых консультаций используется презентационное оборудование (плазменная панель (проектор), ноутбук) и комплект презентации, обеспечивающие тематические иллюстрации по темам рабочей программы дисциплины.

7.5	Помещения для самостоятельной работы обучающихся оснащены компьютерной техникой с возможностью подключения к сети "Интернет" и обеспечением доступа в электронную информационно-образовательную среду организации.
7.6	Электронно-библиотечная система ВПИ (филиал) ВолгГТУ обеспечивает возможность одновременного доступа не менее 25 процентов обучающихся по программе бакалавриата из любой точки, в которой имеется доступ к информационно-телекоммуникационной сети "Интернет", как на территории ВПИ (филиал) ВолгГТУ, так и вне его.
7.7	В корпусах А (ул.Энгельса 42а), №1 (пр. Ленина 72) и 2 (пр. Ленина 70) развернута сеть Wi-Fi, обеспечивающая свободный доступ студентам к ресурсам сети Интернет и локальным Интернет - ресурсам ВПИ.
7.8	Аудитория 1-303. Учебная аудитория для проведения занятий лекционного и семинарского типа, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации. Корпус «1», пр. Ленина 72: 42 посадочных места; рабочее место преподавателя; учебная доска; учебная мебель; LCD телевизор.
7.9	Аудитория 1-311. Учебная аудитория для проведения занятий лекционного и семинарского типа, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации. Корпус «1», пр. Ленина 72: 42 посадочных места; рабочее место преподавателя; учебная доска; учебная мебель; LCD телевизор.
7.10	Аудитория 1-302. Лаборатория "Математическое обеспечение" для проведения занятий лекционного и семинарского типа, лабораторных работ, курсового проектирования (выполнения курсовых работ), групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации, самостоятельной работы, подготовки к процедуре защиты и процедуры защиты выпускных квалификационных работ. Корпус «1», пр. Ленина 72: 24 посадочных места; рабочее место преподавателя; учебная доска; учебная мебель; компьютеры 12 шт. с доступом к электронной информационно-образовательной среде ВПИ и выходом в сеть Internet; экран на штативе Keydo KSC-TR 125*125; ноутбук Toshiba Satellite L300; коммутатор 16 PORT D-LINK DES-1016D; мультимедиапроектор NEC NP 210.
7.11	Аудитория 1-510. Лаборатория "Программное обеспечение" для проведения занятий лекционного и семинарского типа, лабораторных работ, курсового проектирования (выполнения курсовых работ), групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации, самостоятельной работы, выполнения выпускных квалификационных работ. Корпус пр. Ленина 72: 26 посадочных мест; рабочее место преподавателя; учебная доска; учебная мебель; компьютеры 13 шт. с доступом к электронной информационно-образовательной среде ВПИ и выходом в сеть Internet; плазменная панель LG 42; сплиттер ATEN VS 92A VGA*2.
7.12	Аудитория 1-502. Лаборатория "Компьютерные технологии в науке и образовании" для проведения занятий лекционного и семинарского типа, лабораторных работ, курсового проектирования (выполнения курсовых работ), групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации, самостоятельной работы, выполнения выпускных квалификационных работ. Корпус «1», пр. Ленина 72: 26 посадочных мест; рабочее место преподавателя; учебная доска; учебная мебель; видеопроектор Acer Projector P134w; компьютеры 13 шт. с доступом к электронной информационно-образовательной среде ВПИ и выходом в сеть Internet; кронштейн ARM Media Projector-3; экран настенный Lumien Master 244*244.
7.13	Помещения для хранения и профилактического обслуживания учебного оборудования:
7.14	Аудитория 1-304. Кафедра "Информатика и технология программирования". Корпус «1», пр. Ленина 72,
7.15	Аудитория А-22. Информационно-вычислительный центр. Корпус «А», улица Энгельса, 42а.

8. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ (МОДУЛЯ, ПРАКТИКИ)

Планирование и организация времени, необходимого для изучения дисциплины:

Важным условием успешного освоения дисциплины является создание студентом системы правильной организации своего труда, позволяющей распределить учебную нагрузку равномерно в соответствии с графиком образовательного процесса. Большую помощь в этом может оказать составление плана работы на семестр, месяц, неделю, день. Его наличие позволит подчинить свободное время целям учебы, трудиться более успешно и эффективно. Нужно осуществлять самоконтроль, который является необходимым условием успешной учебы. Все задания к практическим занятиям и лабораторным работам, а также задания, вынесенные на самостоятельную работу, рекомендуется выполнять непосредственно после соответствующей темы лекционного курса, что способствует лучшему усвоению материала, позволяет своевременно выявить и устранить «пробелы» в знаниях, систематизировать ранее пройденный материал, на его основе приступить к овладению новыми знаниями, умениями и навыками.

Методические указания к лекционным занятиям:

Подготовка к лекционному занятию включает выполнение всех видов заданий, рекомендованных к каждой лекции, т.е. задания выполняются еще до лекционного занятия по соответствующей теме.

В ходе лекционных занятий необходимо вести конспектирование учебного материала, обращать внимание на категории, формулировки, раскрывающие содержание тех или иных явлений и процессов, научные выводы и практические рекомендации. Желательно оставить в рабочих конспектах поля, на которых делать пометки из рекомендованной литературы, дополняющие материал прослушанной лекции, а также подчеркивающие особую важность тех или иных теоретических положений.

Необходимо задавать преподавателю уточняющие вопросы с целью уяснения теоретических положений, разрешения спорных ситуаций. Целесообразно дорабатывать свой конспект лекции, делая в нем соответствующие записи из литературы, рекомендованной преподавателем и предусмотренной учебной программой.

Подготовка к практическим занятиям и лабораторным работам:

Практические занятия и лабораторные работы позволяют развивать у студентов творческое теоретическое мышление, умение самостоятельно изучать литературу, анализировать практику; учат четко формулировать мысль, вести дискуссию, то есть имеют исключительно важное значение в развитии самостоятельного мышления.

Подготовка к практическому занятию или лабораторной работе включает два этапа. На первом этапе студент планирует свою самостоятельную работу, которая включает: уяснение задания на самостоятельную работу; подбор рекомендованной литературы; составление плана работы, в котором определяются основные пункты предстоящей подготовки. Составление плана дисциплинирует и повышает организованность в работе. Второй этап включает непосредственную подготовку к практическому занятию или лабораторной работе. Начинать надо с изучения рекомендованной литературы. Необходимо помнить, что на лекции обычно рассматривается не весь материал, а только его часть. Остальная его часть восполняется в процессе самостоятельной работы. В связи с этим работа с рекомендованной литературой обязательна. Особое внимание при этом необходимо обратить на содержание основных положений и выводов, объяснение явлений и фактов, уяснение практического приложения рассматриваемых теоретических вопросов.

Готовясь к практическому занятию или лабораторной работе, студент может обращаться за методической помощью к преподавателю. Идя на консультацию, необходимо хорошо продумать вопросы, которые требуют разъяснения.

Методические указания к самостоятельной работе:

Самостоятельная работа студента является основным средством овладения учебным материалом во время, свободное от обязательных учебных занятий. Самостоятельная работа студента над усвоением учебного материала по учебной дисциплине может выполняться в библиотеке университета, учебных кабинетах, компьютерных классах, а также в домашних условиях. Содержание самостоятельной работы студента определяется учебной программой дисциплины, методическими материалами, заданиями и указаниями преподавателя. Самостоятельная работа может осуществляться в аудиторной и внеаудиторной формах. Самостоятельная работа в аудиторное время может включать:

- 1) конспектирование (составление тезисов) лекций;
- 2) решение задач;
- 3) работу со справочной и методической литературой;
- 4) выступления с докладами, сообщениями на практических занятиях;
- 5) защиту выполненных работ;
- 6) участие в текущем опросе по отдельным темам изучаемой дисциплины;
- 7) участие в собеседованиях, деловых (ролевых) играх, дискуссиях, круглых столах, конференциях;
- 8) участие в тестировании и др.

Самостоятельная работа во внеаудиторное время может состоять из:

- 1) повторение лекционного материала;
- 2) изучения учебной и научной литературы;
- 3) выделение наиболее сложных и проблемных вопросов по изучаемой теме, получение разъяснений и рекомендаций по данным вопросам с преподавателями кафедры на их консультациях;
- 4) проведение самоконтроля путем ответов на вопросы текущего контроля знаний, решения представленных в учебно-методических материалах кафедры задач, тестов по отдельным вопросам изучаемой темы;
- 5) подготовки к практическим занятиям и лабораторным работам;
- 6) решения задач, выданных на практических занятиях и лабораторных работах;
- 7) выполнения контрольной работы, предусмотренной учебным планом.

Наиболее важным моментом самостоятельной работы является выполнение контрольной работы. Теоретическая часть контрольной работы выполняется по установленным темам с использованием учебно-методических материалов.

К каждой теме контрольной работы рекомендуется примерный перечень узловых вопросов, список необходимой литературы. Необходимо изучить литературу, рекомендуемую для выполнения контрольной работы. Чтобы полнее раскрыть тему, студенту следует выявить дополнительные источники и материалы. При написании контрольной работы необходимо ознакомиться с публикациями по теме, опубликованными в журналах.

Необходимо изложить собственные соображения по существу излагаемых вопросов и решаемых задач, внести свои предложения. Общие положения должны быть подкреплены и пояснены конкретными примерами. Излагаемый материал при необходимости следует проиллюстрировать таблицами, схемами, программным кодом, диаграммами и т.д.

Рекомендации по работе с литературой:

Работу с литературой целесообразно начать с изучения общих работ по теме, а также учебников и учебных пособий. Далее рекомендуется перейти к анализу монографий и статей, рассматривающих отдельные аспекты проблем, изучаемых в рамках курса, а также официальных материалов и неопубликованных документов (научно-исследовательские работы, диссертации), в которых могут содержаться основные вопросы изучаемой проблемы.

Работу с источниками надо начинать с ознакомительного чтения, т.е. просмотреть текст, выделяя его структурные единицы. При ознакомительном чтении закладками отмечаются те страницы, которые требуют более внимательного изучения.

В зависимости от результатов ознакомительного чтения выбирается дальнейший способ работы с источником. Если для разрешения поставленной задачи требуется изучение некоторых фрагментов текста, то используется метод выборочного чтения. Если в книге нет подробного оглавления, следует обратить внимание ученика на предметные и именные указатели. Избранные фрагменты или весь текст (если он целиком имеет отношение к теме) требуют вдумчивого, неторопливого чтения с «мысленной проработкой» материала. Такое чтение предполагает выделение: 1) главного в тексте; 2) основных аргументов; 3) выводов. Особое внимание следует обратить на то, вытекает тезис из аргументов или нет.

Необходимо также проанализировать, какие из утверждений автора несут проблематичный, гипотетический характер, и уловить скрытые вопросы.

Таким образом, при работе с источниками и литературой важно уметь:

- 1) сопоставлять, сравнивать, классифицировать, группировать, систематизировать информацию в соответствии с определенной учебной задачей;
- 2) обобщать полученную информацию, оценивать прослушанное и прочитанное;
- 3) фиксировать основное содержание сообщений; формулировать, устно и письменно, основную идею сообщения; составлять план, формулировать тезисы;
- 4) готовить и презентовать развернутые сообщения типа доклада;
- 5) работать в разных режимах (индивидуально, в паре, в группе), взаимодействуя друг с другом;
- 6) пользоваться реферативными и справочными материалами;
- 7) контролировать свои действия и действия своих товарищей, объективно оценивать свои действия;
- 8) обращаться за помощью, дополнительными разъяснениями к преподавателю, другим студентам.

Подготовка к промежуточной аттестации по дисциплине:

При подготовке к промежуточной аттестации целесообразно:

- 1) внимательно изучить перечень вопросов к промежуточной аттестации по дисциплине и определить, в каких источниках находятся сведения, необходимые для ответа на них;
- 2) внимательно прочитать рекомендованную литературу;
- 3) составить краткие конспекты ответов (планы ответов).

При обучении используются следующие образовательные технологии:

1) Интерактивные методы обучения:

1.1) Интерактивная форма в рамках лекционных занятий:

Каждая лекция заканчивается диагностическим блоком в виде интерактивного элемента. На плазменную панель выводятся тестовые задания вопросы по теме проведенной лекции, студенты отвечают на вопросы. Преподаватель организует дискуссию. Идет групповое обсуждение ответа. После обсуждения преподаватель выделяет на слайде правильный вариант ответа. Если ответ студентов был неправильным, либо они затруднились с ответом, преподаватель поясняет, почему на данный вопрос надо отвечать именно так.

1.2) Интерактивная форма в рамках лабораторных работ:

В качестве интерактивного метода обучения при проведении лабораторных работ используется «метод анализа конкретной ситуации» (case-study). Каждая лабораторная работа начинается с разбора конкретных ситуаций, методов и подходов, связанных с алгоритмическими и программными решениями по тематике лабораторной работы. Метод case-study инициирует совместное погружение студентов в проблемное поле решаемой в рамках лабораторной работы задачи, т. е. включение в единое творческое пространство, а также обеспечивает согласованность в выборе средств и методов программной реализации решения задачи. Совместная деятельность в процессе разбора конкретных ситуаций, методов, и подходов, связанных с алгоритмическими и программными решениями по тематике лабораторной работы означает, что каждый студент вносит свой особый индивидуальный вклад, идет обмен знаниями, идеями и способами решения задач разработки математического, алгоритмического и программного обеспечения.

1.3) Интерактивная форма в рамках практических занятий:

В качестве интерактивного метода обучения, при проведении практических занятий в составе учебной группы, используется метод «мозгового штурма». Каждое практическое занятие начинается с интерактивного элемента - решение задачи по теме практического занятия. Цель интерактивного элемента – раскрепощение и активизация познавательной деятельности студентов, пробуждение интереса к рассматриваемой теме практического занятия. Методика: На слайде дано условие задачи. Студенты предлагают свои варианты решения и коллективно обсуждают результаты предложенных вариантов. Преподаватель комментирует обсуждение, побуждает аудиторию найти правильное решение задачи.

2) Технология модульного обучения – предусматривает деление содержания дисциплины на достаточно автономные разделы (модули), интегрированные в общий курс.

3) Технология использования компьютерных программ – позволяет эффективно дополнить процесс обучения на всех уровнях. Мультимедийные программы предназначены как для аудиторной, так и самостоятельной работы студентов.

4) Интернет-технологии – предоставляют широкие возможности для поиска информации и ведения научных исследований.

5) Технология индивидуализации обучения – помогает реализовывать личностно-ориентированный подход, учитывая индивидуальные особенности и потребности учащихся.

6) Технология тестирования – используется для контроля уровня усвоения знаний в рамках модуля на определённом этапе обучения. Данная технология позволяет преподавателю выявить и систематизировать аспекты, требующие дополнительной проработки.

7) Технология развития критического мышления – способствует формированию разносторонней личности, способной критически относиться к информации, умению отбирать информацию для решения поставленной задачи.

Комплексное использование в учебном процессе всех вышеназванных технологий стимулируют личностную, интеллектуальную активность, развивают познавательные процессы, способствуют формированию компетенций, которыми должен обладать будущий специалист.

Методические рекомендации по обучению лиц с ограниченными возможностями здоровья и инвалидов

В соответствии с методическими рекомендациями Минобрнауки РФ (утв. 8 апреля 2014 г. № АК-44/05вн) в курсе предполагается использовать социально-активные и рефлексивные методы обучения, технологии социокультурной реабилитации с целью оказания помощи в установлении полноценных межличностных отношений с другими студентами, создании комфортного психологического климата в студенческой группе. Подбор и разработка учебных материалов производится с учетом предоставления материала в различных формах: аудиальной, визуальной, с использованием

специальных технических средств и информационных систем.

Освоение дисциплины лицами с ОВЗ осуществляется с использованием средств обучения общего и специального назначения (персонального и коллективного использования). Материально-техническое обеспечение предусматривает приспособление аудиторий к нуждам лиц с ОВЗ.

Форма проведения аттестации для студентов-инвалидов устанавливается с учетом индивидуальных психофизических особенностей.

Для студентов с ОВЗ предусматривается доступная форма предоставления заданий оценочных средств, а именно:

- в печатной или электронной форме (для лиц с нарушениями опорно-двигательного аппарата);
- в печатной форме или электронной форме с увеличенным шрифтом и контрастностью (для лиц с нарушениями слуха, речи, зрения);
- методом чтения ассистентом задания вслух (для лиц с нарушениями зрения).

Студентам с инвалидностью увеличивается время на подготовку ответов на контрольные вопросы. Для таких студентов предусматривается доступная форма предоставления ответов на задания, а именно:

- письменно на бумаге или набором ответов на компьютере (для лиц с нарушениями слуха, речи);
- выбором ответа из возможных вариантов с использованием услуг ассистента (для лиц с нарушениями опорно-двигательного аппарата);
- устно (для лиц с нарушениями зрения, опорно-двигательного аппарата).

При необходимости для обучающихся с инвалидностью процедура оценивания результатов обучения может проводиться в несколько этапов.