

МИНИСТЕРСТВО СЕЛЬСКОГО ХОЗЯЙСТВА РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«КУБАНСКИЙ ГОСУДАРСТВЕННЫЙ АГРАРНЫЙ УНИВЕРСИТЕТ
имени И.Т. ТРУБИЛИНА»

Факультет прикладной информатики
Системного анализа и обработки информации

**РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)
«ПРОГРАММИРОВАНИЕ»**

Уровень высшего образования: бакалавриат

Направление подготовки: 38.03.05 Бизнес-информатика

Направленность (профиль) подготовки: Анализ, моделирование и формирование интегрального представления стратегий и целей, бизнес-процессов и информационно-логической инфраструктуры предпри

Квалификация (степень) выпускника: бакалавр

Форма обучения: очная

Год набора: 2024

Срок получения образования: 4 года

Объем:
в зачетных единицах: 8 з.е.
в академических часах: 288 ак.ч.

2024

Разработчики:

Доцент, кафедра системного анализа и обработки информации Самойленкова В.А.

Рабочая программа дисциплины (модуля) составлена в соответствии с требованиями ФГОС ВО по направлению подготовки Направление подготовки: 38.03.05 Бизнес-информатика, утвержденного приказом Минобрнауки России от 29.07.2020 №838, с учетом трудовых функций профессиональных стандартов: "Менеджер по информационным технологиям", утвержден приказом Минтруда России от 30.08.2021 № 588н; "Специалист по информационным системам", утвержден приказом Минтруда России от 13.07.2023 № 586н; "Системный аналитик", утвержден приказом Минтруда России от 27.04.2023 № 367н.

Согласование и утверждение

№	Подразделение или коллегиальный орган	Ответственное лицо	ФИО	Виза	Дата, протокол (при наличии)
1	Системного анализа и обработки информации	Заведующий кафедрой, руководитель подразделения, реализующего ОП	Барановская Т.П.	Согласовано	08.04.2024, № 8

1. Цель и задачи освоения дисциплины (модуля)

Цель освоения дисциплины - «Программирование» является формирование комплекса знаний об организационных, научных и методических основах разработки алгоритмов и реализация их на одном из языков программирования высокого уровня с использованием процедурного подхода разработки программ

Задачи изучения дисциплины:

- приобретение системы знаний об алгоритмизации и программировании как одной из функций деятельности информатика, позволяющей принять соответствующие решения при разработке программ;
- организация информационной системы для широкого круга внутренних и внешних пользователей;
- формирование и представление информации, удовлетворяющей требованиям различных пользователей программного обеспечения;
- изучение методов и способов разработки алгоритмов и программ с использованием языка высокого уровня C++, C#;;
- технические и программные средства реализации информационных процессов.

2. Планируемые результаты обучения по дисциплине (модулю), соотнесенные с планируемыми результатами освоения образовательной программы

Компетенции, индикаторы и результаты обучения

ОПК-3 Способен управлять процессами создания и использования продуктов и услуг в сфере информационно-коммуникационных технологий, в том числе разрабатывать алгоритмы и программы для их практической реализации

ОПК-3.1 Правильно использует методы формализованного описания решения поставленной задачи

Знать:

ОПК-3.1/Зн1 Знает методы формализованного описания решения поставленной задачи

Уметь:

ОПК-3.1/Ум1 Умеет правильно использовать методы формализованного описания решения поставленной задачи

Владеть:

ОПК-3.1/Нв1 Владеет знаниями использования методов формализованного описания решения поставленной задачи

ОПК-3.2 Способен проводить реализацию алгоритма на языках программирования, в том числе объектно-ориентированных с использованием современных инструментальных средств разработки

Знать:

ОПК-3.2/Зн1 Знает способы реализации алгоритма на языках программирования, в том числе объектно-ориентированных с использованием современных инструментальных средств разработки

Уметь:

ОПК-3.2/Ум1 Умеет проводить реализацию алгоритма на языках программирования, в том числе объектно-ориентированных с использованием современных инструментальных средств разработки

Владеть:

ОПК-3.2/Нв1 Владеет знаниями реализации алгоритма на языках программирования, в том числе объектно-ориентированных с использованием современных инструментальных средств разработки

ОПК-3.3 Применяет на практике современные методы программирования, в том числе командной разработки, проводит сборку программного продукта и контроль качества выполненных работ

Знать:

ОПК-3.3/Зн1 Знает методы программирования, в том числе командной разработки, проведения сборки программного продукта и контроля качества выполненных работ

Уметь:

ОПК-3.3/Ум1 Умеет применять современные методы программирования, в том числе командной разработки, проводить сборку программного продукта и контроль качества выполненных работ

Владеть:

ОПК-3.3/Нв1 Применяет на практике современные методы программирования, в том числе командной разработки, проводит сборку программного продукта и контроль качества выполненных работ

ОПК-3.4 Знает основы тестирования и отладки программного продукта и способен применять их на практике

Знать:

ОПК-3.4/Зн1 Знает основы тестирования и отладки программного продукта

Уметь:

ОПК-3.4/Ум1 Умеет проводить тестирование и отладку программного продукта

Владеть:

ОПК-3.4/Нв1 Владеет знаниями тестирования и отладки программного продукта и способен применять их на практике

3. Место дисциплины в структуре ОП

Дисциплина (модуль) «Программирование» относится к обязательной части образовательной программы и изучается в семестре(ах): 1, 2.

В процессе изучения дисциплины студент готовится к видам профессиональной деятельности и решению профессиональных задач, предусмотренных ФГОС ВО и образовательной программой.

4. Объем дисциплины и виды учебной работы

Период обучения	Общая трудоемкость (часы)	Общая трудоемкость (ЗЕТ)	Контактная работа (часы, всего)	Внеаудиторная контактная работа (часы)	Лабораторные занятия (часы)	Лекционные занятия (часы)	Самостоятельная работа (часы)	Промежуточная аттестация (часы)
Первый семестр	144	4	65	3	30	32	52	Экзамен (27)

Второй семестр	144	4	97	3	60	34	20	Экзамен (27)
Всего	288	8	162	6	90	66	72	54

5. Содержание дисциплины

5.1. Разделы, темы дисциплины и виды занятий

(часы промежуточной аттестации не указываются)

Наименование раздела, темы	Всего	Внеаудиторная контактная работа	Лабораторные занятия	Лекционные занятия	Самостоятельная работа	Планируемые результаты обучения, соответствующие результатам освоения программы
Раздел 1. Основы алгоритмизации и программирования	12			4	8	ОПК-3.1 ОПК-3.2 ОПК-3.3 ОПК-3.4
Тема 1.1. Алгоритмизация процессов обработки данных	6			2	4	
Тема 1.2. Введение в программирование	6			2	4	
Раздел 2. Программирование на C++	102		30	28	44	ОПК-3.1 ОПК-3.2 ОПК-3.3 ОПК-3.4
Тема 2.1. Особенности разработки программы на C++. Лексические основы языка	16		4	6	6	
Тема 2.2. Программирование разветвлений на C++	18		6	4	8	
Тема 2.3. Реализация алгоритмов с циклической структурой на C++	18		6	4	8	
Тема 2.4. Массивы одномерные и двумерные в C++. Строки как массив символов	20		6	6	8	
Тема 2.5. Сортировка массивов	14		4	2	8	
Тема 2.6. Структуры. Объединения. Перечисления	16		4	6	6	
Раздел 3. Промежуточная аттестация	3	3				ОПК-3.1 ОПК-3.2 ОПК-3.3 ОПК-3.4
Тема 3.1. Экзамен	3	3				
Раздел 4. Разработка приложений	114		60	34	20	ОПК-3.1 ОПК-3.2 ОПК-3.3 ОПК-3.4
Тема 4.1. Программирование функций	20		12	4	4	
Тема 4.2. Указатели. Динамические переменные	24		12	8	4	

Тема 4.3. Разработка приложений с использованием файлового ввода-вывода	22		12	6	4	
Тема 4.4. Основы разработки графических приложений	24		12	8	4	
Тема 4.5. Основы разработки приложений баз данных	24		12	8	4	
Раздел 5. Промежуточная аттестация	3	3				ОПК-3.1 ОПК-3.2
Тема 5.1. Экзамен	3	3				ОПК-3.3 ОПК-3.4
Итого	234	6	90	66	72	

5.2. Содержание разделов, тем дисциплин

Раздел 1. Основы алгоритмизации и программирования (Лекционные занятия - 4ч.; Самостоятельная работа - 8ч.)

Тема 1.1. Алгоритмизация процессов обработки данных (Лекционные занятия - 2ч.; Самостоятельная работа - 4ч.)

Основные понятия. Свойства алгоритмов. Способы записи алгоритмов. Запись алгоритмов в виде блок-схем. Этапы решения прикладных задач с использованием компьютера

Тема 1.2. Введение в программирование (Лекционные занятия - 2ч.; Самостоятельная работа - 4ч.)

Основные понятия программирования. История и принципы развития языков программирования. Типы языков программирования. Виды языков высокого уровня. Системы программирования

Раздел 2. Программирование на C++ (Лабораторные занятия - 30ч.; Лекционные занятия - 28ч.; Самостоятельная работа - 44ч.)

Тема 2.1. Особенности разработки программы на C++. Лексические основы языка (Лабораторные занятия - 4ч.; Лекционные занятия - 6ч.; Самостоятельная работа - 6ч.)

Общая характеристика языка. Технология разработки программ. Примерная структура исходного модуля программы в процедурно-ориентированном стиле. Препроцессор. Директивы препроцессора. Определение главной функции. Обзор сред программирования. Алфавит, особенности языка. Классификация типов данных. Операции, выражения и операторы. Объявление переменных. Задание констант. Операторы ввода-вывода. Область видимости и время жизни переменных. Преобразование типов. Библиотека math

Тема 2.2. Программирование разветвлений на C++ (Лабораторные занятия - 6ч.; Лекционные занятия - 4ч.; Самостоятельная работа - 8ч.)

Условие в программировании. Простые и составные условия. Условный оператор if(). Условная операция ? : в C++. Оператор множественного выбора switch() в C++. Особенности использования оператора break

Тема 2.3. Реализация алгоритмов с циклической структурой на C++ (Лабораторные занятия - 6ч.; Лекционные занятия - 4ч.; Самостоятельная работа - 8ч.)

Понятие циклических вычислительных процессов. Цикл for(): синтаксис, особенности. Цикл с предусловием while(): синтаксис, особенности. Цикл с постусловием do(): синтаксис, особенности. Итерационные циклы. Особенности использования и способы задания условия в итерационном цикле. Вложенные циклы. Бесконечный цикл. Операторы continue и break в циклах. Функция exit().

*Тема 2.4. Массивы одномерные и двумерные в C++. Строки как массив символов
(Лабораторные занятия - 6ч.; Лекционные занятия - 6ч.; Самостоятельная работа - 8ч.)*

Массив как тип данных. Определение и инициализация одномерного массива. Доступ к элементам. Определение и инициализация двумерного массива. Доступ к элементам. Строка как массив символов типа char. Строковые переменные и константы: объявление и инициализация, вывод на экран. Одномерный массив строк. Операции со строками: копирование, сравнение, поиск, добавление строки к строке, определение длины, изменение регистра букв, кодировки букв

*Тема 2.5. Сортировка массивов
(Лабораторные занятия - 4ч.; Лекционные занятия - 2ч.; Самостоятельная работа - 8ч.)*

Понятие сортировки. Методы сортировки массивов. Алгоритм пузырьковой сортировки. Достоинства и недостатки метода. Реализация на C++ пузырьковой сортировки числовых массивов и строк

*Тема 2.6. Структуры. Объединения. Перечисления
(Лабораторные занятия - 4ч.; Лекционные занятия - 6ч.; Самостоятельная работа - 6ч.)*

Понятие структуры. Особенности структур. Определение структуры. Способы описания вложенных структур. Объявление структурных переменных. Использование операции доступа к полю структуры, в том числе вложенной. Способы инициализации полей структуры, в том числе вложенной. Объявление и инициализация массива структур. Действия со структурными переменными. Объявление типа перечисление. Объявление переменной типа перечисление. Объявление типа объединение. Объявление переменной типа объединение. Присваивание объединений друг другу

Раздел 3. Промежуточная аттестация (Внеаудиторная контактная работа - 3ч.)

*Тема 3.1. Экзамен
(Внеаудиторная контактная работа - 3ч.)*
Экзамен

Раздел 4. Разработка приложений (Лабораторные занятия - 60ч.; Лекционные занятия - 34ч.; Самостоятельная работа - 20ч.)

*Тема 4.1. Программирование функций
(Лабораторные занятия - 12ч.; Лекционные занятия - 4ч.; Самостоятельная работа - 4ч.)*

Понятие, назначение, особенности функций. Определение, объявление (прототипы) функций. Вызов функции пользователя. Способы описания функций в C++. Использование формальных и фактических параметров (аргументов) при обмене информацией с функцией. Рекурсивные функции

*Тема 4.2. Указатели. Динамические переменные
(Лабораторные занятия - 12ч.; Лекционные занятия - 8ч.; Самостоятельная работа - 4ч.)*

Указатель как тип данных. Синтаксис и семантика типа данных. Получение адреса переменной, доступ к значению переменной по адресу. Адресная арифметика. Связь указателей с массивами, в том числе строками, как массивами, функциями, структурами. Использование указателей на указатели. Свободная память. Действия с динамическими объектами: создание, уничтожение, присваивание значения. Динамические массивы: объявление, обработка

Тема 4.3. Разработка приложений с использованием файлового ввода-вывода

(Лабораторные занятия - 12ч.; Лекционные занятия - 6ч.; Самостоятельная работа - 4ч.)

Основные понятия. Текстовые и двоичные файлы. Файловый ввод/вывод с помощью потоков. Дополнительные функции файлового ввода / вывода. Режимы файлов. Произвольный доступ к файлам

Тема 4.4. Основы разработки графических приложений

(Лабораторные занятия - 12ч.; Лекционные занятия - 8ч.; Самостоятельная работа - 4ч.)

Разработка проекта приложения на основе стандартов управления проектами ISO/IEC 12207. Структура проекта. Интегрированная среда разработки программ. Компонент форма - TForm. Обработка события создания формы. Обработка событий «мыши». Обработка событий нажатия клавиш клавиатуры. Использование компонентов TLabel, Tedit, Tmemo, TmainMenu, TpopupMenu. Функции перевода данных из одного типа в другой. Функции выдачи сообщений на экран. Функции ввода данных из диалоговых окон в программу. Работа с файлами с использованием функций компонент. Работа с датами. Использование средств обработки графической информации

Тема 4.5. Основы разработки приложений баз данных

(Лабораторные занятия - 12ч.; Лекционные занятия - 8ч.; Самостоятельная работа - 4ч.)

Работа с базами данных: подключение к приложению, реализация ввода, редактирования, удаления данных в таблицах. Визуализация данных таблиц. Реализация импорта/экспорта данных в MS Word и MS Excel

Раздел 5. Промежуточная аттестация

(Внеаудиторная контактная работа - 3ч.)

Тема 5.1. Экзамен

(Внеаудиторная контактная работа - 3ч.)

Промежуточная аттестация

6. Оценочные материалы текущего контроля

Раздел 1. Основы алгоритмизации и программирования

Форма контроля/оценочное средство: Задача

Вопросы/Задания:

1. Графическое представление алгоритма из отдельных геометрических фигур – блоков, обозначающих то или иное действие, соединенных направленными линиями, которые показывают последовательность перехода от одного блока к следующему, называется ...

блочной схемой

графиком

блок-схемой

диаграммой

2. Программная форма представления алгоритмов – это ...

тексты на языках программирования

запись на естественном языке

изображения из графических символов

полуформализованные описания алгоритмов на условном алгоритмическом языке

3. Словесная форма представления алгоритмов – это ...

тексты на языках программирования
запись на естественном языке
изображения из графических символов
полуформализованные описания алгоритмов на условном алгоритмическом языке

4. Укажите базовые алгоритмические структуры

Линейная структура
Разветвляющаяся структура
Повторяющаяся структура
Программная структура
Циклическая структура

5. Любой язык программирования включает:

Алфавит языка
Лексемы (слова)
Предложения
Выражения
Цифры

6. Расположите в правильном порядке этапы решения прикладных задач с использованием ЭВМ

Постановка задачи
Анализ и исследование задачи, модели
Разработка алгоритма
Программирование
Анализ результатов решения задачи и уточнение в случае необходимости математической модели с повторным выполнением этапов 2-5
Тестирование и отладка
Сопровождение программы

7. Следующее определение «Материальный или мысленно представляемый объект, используемый вместо объекта или явления (процесса) при его исследовании, сохраняющий информацию о некоторых важных для данного исследования типичных чертах и свойствах оригинала» – это описание ... модели

Математической
Информационной
Компьютерной
Имитационной

8. Конечный набор правил (последовательность действий), расположенных в определенном логическом порядке, позволяющий исполнителю решать любую конкретную задачу из некоторого класса однотипных задач – это ...

закон
алгоритм
нормативный документ
схема

9. Алгоритм должен отвечать определенным требованиям (условиям)

однозначность
конечность
своевременность
эффективность
объемность
общность
корректность

10. Главными элементами любой программы в процедурно-ориентированном стиле являются

переменные

числа
константы
символы
выражения
операторы

11. Предложение языка программирования, которое задает полное описание действий, которые необходимо выполнить - это
оператор (без учета регистра)

12. Транслятор – это ...
преобразователь программы на языке C++ в программу на языке Visual Basic
специальная программа, которая преобразует программу, написанную на одном из языков высокого уровня, в программу на языке C++
программа-переводчик, преобразует программу, написанную на одном из языков высокого уровня, в программу, состоящую из машинных команд

13. ... – читает всю программу целиком, делает ее перевод и создает законченный вариант программы на машинном языке, который затем и выполняется.
компилятор

Раздел 2. Программирование на C++

Форма контроля/оценочное средство: Задача

Вопросы/Задания:

1. Язык программирования C++ является ... языком программирования
Процедурным
Функциональным
Логическим
Объектно-ориентированным

2. Язык программирования C++ разработал
Бьерн Страуструп
Кен Томпсон
Никлаус Вирт
Дональд Кнут

3. Расположите в правильном порядке этапы процесса создания программ на C++ в среде программирования:

Написание и редактирование исходного текста программы с сохранением ее в виде исходного файла или модуля.

Компиляция программы и получение ее на определенном промежуточном языке с сохранением в виде объектного файла или модуля.

Построение исполнимого файла или модуля путем объединения (компоновки) полученного объектного модуля программы с другими объектными модулями стандартных и специальных библиотек.

Отладка программы, которую можно проводить с помощью специального средства (отладчика), облегчающего обнаружение ошибок.

4. Укажите, какую часть исходного модуля программа на C++ должна включать обязательно

подключение заголовочных файлов библиотек (с функциями, константами и др.), определение пространства имен

объявления пользовательских символических констант, вспомогательных функций, классов и типов данных, глобальных переменных

определение главной функции

определение пользовательских вспомогательных функций и методов классов

5. Точкой входа в программу консольного приложения C++ является функция ...
main

6. Определение главной функции в C++ состоит из следующих основных частей:

заголовка определения функции
методов функции
тела определения функции
заголовка тела функции
полей функции

7. Файл, имеющий имя "test_file.cpp", это:

динамически загружаемая библиотека
файл исходного текста языка C++
исполняемый файл
заголовочный файл

8. Файлы с текстом программ на языке C++ имеют расширение ...

cpp
c
cs
c#

9. Заголовочные файлы в C++ имеют расширение ...

h
cpp
obj
exe
c

10. Выберите правильный синтаксис заголовка главной функции программы на C++

int main()
void main(<список формальных параметров>)
int main(<список формальных параметров>)
void main()

11. Пространства имен помогают

избегать конфликтов имен (функций, переменных и так далее)
сократить код
ускорить доступ к объектам (переменным, функциям и так далее)

12. Групповой проект в MS VS 20XX называется

решение (без учета регистра)

Раздел 3. Промежуточная аттестация

Форма контроля/оценочное средство: Задача

Вопросы/Задания:

1. Выберите интегрированные среды программирования, которые поддерживают разработку программ на C++

MICROSOFT VISUAL STUDIO
ECLIPSE
CODE GEAR RAD STUDIO
EMBARCADERO RAD STUDIO
NETBEANS

2. Среда программирования Microsoft Visual Studio 20XX позволяет создавать следующие проекты VISUAL C++:

Стандартное приложение Win32
Консольное приложение Win32
Приложение Windows Form
Проект Windows Form

3. Язык программирования ... не относится к языкам высокого уровня

C++
Pascal
Assembler

4. Укажите языки, которые не относятся к языкам высокого уровня

Процедурные
Функциональные
Машинно-ориентированные
Логические
Объектно-ориентированные

5. Среда разработки программного обеспечения – это ...

компилятор кода
система программных средств, используемая для разработки программного обеспечения
программа, предназначенная для запуска других программ
программа, предназначенная для написания кода программ

6. Языки высокого уровня ...

имитируют естественные языки, используя некоторые слова разговорного языка и общепринятые математические символы
не требуют знания основ программирования
требуют указания мелких деталей процесса обработки данных
это графические языки

7. Языки низкого уровня требуют ...

указания средних деталей процесса обработки данных
указания мелких деталей процесса обработки данных
указания крупных деталей процесса обработки данных
описания алгоритмов

8. Укажите характеристики (атрибуты) переменной

имя переменной
псевдоним переменной
значение переменной
адрес переменной
область вызова
область видимости

9. Виды проектов Win32 Console Application в MS VS 20XX

Пустой
Шаблон
Стандартный
Простой
Общий

10. Скомпилированные файлы проекта в MS VS 20XX хранятся в каталогах

Debug
Bin
Win32
Release

11. Алфавит языка C++ включает:

латинские буквы
арабские цифры
русские буквы
специальные символы
управляющие символы
римские цифры

12. Установите соответствие между типом данных и его записью в языке C++

(1) Логический
(2) Символьный
(3) Вещественный
(4) Целочисленный

Раздел 4. Разработка приложений

Форма контроля/оценочное средство: Задача

Вопросы/Задания:

1. Команда, позволяющая переменной A присвоить значение 38:

A=38;

A==38;

A:=38;

нет верного ответа

2. Переменная D после выполнения команд: D=3; D*=D; D*=D; примет значение...

81

3. Переменная C после выполнения команд: C=14; C%=3;

2

4. Переменная X после выполнения команды X= pow(4.0, 2)/4*2; примет значение ...

8

5. Фиксированная величина, которая не может быть изменена в программе, обозначается как ...

static

const

private

protected

6. Текст в операторе вывода должен быть заключен в ... (напишите слово)

кавычки

7. Укажите правильную последовательность команд, позволяющих ввести значение радиуса, вычислить и вывести длину окружности:

cout << " R= ";

cin >> R;

L=2*3.14*R;

cout << " L= " << L;

8. Укажите правильную последовательность команд, позволяющих ввести стороны треугольника A, B, C и вывести его площадь S

cin >> A >> B >> C;

P=(A+B+C)/2;

S=sqrt(P*(P-A)*(P-B)*(P-C));

cout << " S= " << S;

9. Выполните сопоставление знаков операций их названиям

(1) Арифметические

(2) Отношения

(3) Логические

(4) Присваивания

(5) Адресные

(6) Поразрядные логические

(7) Условная операция

(8) Операция запятая

10. Какому зарезервированному слову программа передаёт управление в случае, если значение переменной или выражения оператора switch не совпадает ни с одним константным выражением?

other

contingency

else

default

11. Сопоставьте название оператора с его синтаксисом на C++.

(1) Оператор присваивания

(2) Условный оператор

(3) Составной оператор

(4) Условная операция

12. Укажите правильные варианты записи условного оператора в языке C++:

if (a>0) a=1;

if (a>0) then a:=1;

if a>0 else a=1;

if (a>0) a=1 else a=0;

if (a>0) a:=1 else a:=0;

Раздел 5. Промежуточная аттестация

Форма контроля/оценочное средство: Задача

Вопросы/Задания:

1. После выполнения программы при n=1 на экран будет выведено число ...

```
int a=7;
int b=3;
int n;
cin >> n;
switch(n)
{
    case 1: cout << a+b; break;
    case 2: cout << a-b; break;
    case 3: cout << a*b; break;
    default: cout << a/b; break;
}
```

После выполнения программы при n=1 на экран будет выведено число ...

```
int a=7;
int b=3;
int n;
cin >> n;
switch(n)
{
    case 1: cout << a+b; break;
    case 2: cout << a-b; break;
    case 3: cout << a*b; break;
    default: cout << a/b; break;
}
```

2. После выполнения программы при n=5 на экран будет выведено число ...

```
int a=7;
int b=3;
int n;
cin >> n;
switch(n)
{
    case 1: cout << a+b; break;
    case 2: cout << a-b; break;
    case 3: cout << a*b; break;
    default: cout << a/b; break;
}
```

После выполнения программы при n=5 на экран будет выведено число ...

```
int a=7;
int b=3;
int n;
cin >> n;
switch(n)
{
```

```

    case 1: cout << a+b; break;
    case 2: cout << a-b; break;
    case 3: cout << a*b; break;
    default: cout << a/b; break;
}

```

3. После выполнения программы при n=5 на экран будет выведено число...

```

int a=7;
int b=3;
int n;
cin >> n;
if (n<5)
    cout << a+b;
else
    cout << a-b;

```

После выполнения программы при n=5 на экран будет выведено число...

```

int a=7;
int b=3;
int n;
cin >> n;
if (n<5)
    cout << a+b;
else
    cout << a-b;

```

4. После выполнения программы при n=5 на экран будет выведено число ...

```

int a=7;
int b=4;
int n;
cin >> n;
if (n<=5)
    if (a+b==10)cout << a+b;
    else cout << a-b;
else cout << a*b;

```

После выполнения программы при n=5 на экран будет выведено число ...

```

int a=7;
int b=4;
int n;
cin >> n;
if (n<=5)
    if (a+b==10)cout << a+b;
    else cout << a-b;
else cout << a*b;

```

5. После выполнения программы при n=6 на экран будет выведено число ...

```

int a=7;
int b=4;
int n;
cin >> n;
if (n<=5)
    if (a+b==10)cout << a+b;
    else cout << a-b;
else cout << a*b;

```

После выполнения программы при n=6 на экран будет выведено число ...

```

int a=7;
int b=4;
int n;

```



```

cin >> n;
if (n<=5)
    if (a+b==10)cout << a+b;
    else cout << a-b;
else cout << a*b;

```

6. Какой вариант оператора-переключателя является НЕверным?

1. switch(n)


```

{
    case 1 : cout << a+b; break;
    case 2 : cout << a-b; break;
    default: cout << a*b; break;
}

```
2. switch(n)


```

{
    case 1: case 2: cout << a+b; break;
    default: cout << a-b; break;
}

```
3. switch(n)


```

{
    case 1 : cout << a+b; break;
    case 2 : cout << a-b; break;
}

```
4. switch(n)


```

{
case 1, 2 : cout << a+b; break;
    default: cout << a*b; break;
}

```

Какой вариант оператора-переключателя является НЕверным?

1. switch(n)


```

{
    case 1 : cout << a+b; break;
    case 2 : cout << a-b; break;
    default: cout << a*b; break;
}

```
2. switch(n)


```

{
    case 1: case 2: cout << a+b; break;
    default: cout << a-b; break;
}

```
3. switch(n)


```

{
    case 1 : cout << a+b; break;
    case 2 : cout << a-b; break;
}

```
4. switch(n)


```

{
case 1, 2 : cout << a+b; break;
    default: cout << a*b; break;
}

```

7. Установите правильную последовательность элементов, составляющих условный оператор для выбора максимального из чисел А и В

- 1 if
- 2 (A>B)
- 3 max=A;

```
4 else
5 max=B;
```

8. Установите правильную последовательность команд, позволяющих ввести числа A, B, C и выбрать из них минимальное

```
1 cin >> A >> B >> C;
2 if (A<B) min=A; else min=B;
3 if (C<min) min=C;
4 cout << "min=" << min;
```

9. После выполнения программы при b=2, n=3 на экран будет выведено число...

```
int a=3;
int b, n;
cin >> b >> n;
if (n>b)
cout << a+b;
else
switch(n)
{
case 5: cout << a+b; break;
case 6: cout << a-b; break;
case 7: cout << a*b; break;
default: cout << a/b; break;
}
```

После выполнения программы при b=2, n=3 на экран будет выведено число...

```
int a=3;
int b, n;
cin >> b >> n;
if (n>b)
cout << a+b;
else
switch(n)
{
case 5: cout << a+b; break;
case 6: cout << a-b; break;
case 7: cout << a*b; break;
default: cout << a/b; break;
}
```

10. Тело любого цикла в C++ выполняется до тех пор, пока его условие ...

у цикла нет условия

истинно, или ложно при применении некоторых операторов цикла

ложно

ложно, или истинно при применении некоторых операторов цикла

истинно

11. Выберите правильные записи заголовка цикла for

```
for (int i = 1; i <= 10; i++)....
for (j = 10; j >= 1; j--)....
for (i = 1; j=10; i<= 10, j>=1; i++; j-=2)....
for (int i = 1; i <= 10; i = i + 1)....
for (char a = a; a <= z; a++)....
for (float b = 0.1; b <= 10; i+=0.2)....
for (i = 1, j=10; i<= 10, j>=1; i++, j-=2)....
```

12. Оператор break производит выход:

только из цикла наибольшей глубины вложенности;

только из ветвления switch наибольшей глубины вложенности;

из всех вложенных циклов и ветвлений;

из цикла или ветвления наибольшей глубины вложенности.

7. Оценочные материалы промежуточной аттестации

Первый семестр, Экзамен

Контролируемые ИДК: ОПК-3.1 ОПК-3.2 ОПК-3.3 ОПК-3.4

Вопросы/Задания:

1. Понятие информационной, математической, компьютерной модели
2. Этапы решения прикладных задач с использованием ЭВМ
3. Базовые алгоритмические структуры (конструкции)
4. Понятие программы
5. Структура языка программирования.
6. Понятие системы программирования. Транслятор
7. Интегрированная среда программирования
8. Языки программирования низкого уровня
9. Виды языков программирования высокого уровня
10. Этапы процесса создания программ на C++
11. Синтаксическая структура программы на языке C++: типовые части исходного модуля
12. Алфавит языка C++
13. Особенности языка C++
14. Типы данных в языке C++
15. Понятие операции, выражения, оператора в C++
16. Основные арифметические операции. Примеры
17. Операции присваивания. Виды операции. Примеры
18. Операции инкремента и декремента. Примеры
19. Побитовые или булевские операции
20. Объявление и инициализация переменных

21. Константы. Способы объявления констант в C++
22. Реализация ввода-вывода в C++
23. Битовые поля
24. Условие в программировании. Простые и составные условия
25. Операции отношения (сравнения), логические операции
26. Условный оператор. Примеры
27. Условная операция ? : в C++. Примеры
28. Оператор множественного выбора switch() в C++. Примеры
29. Особенности использования оператора break.
30. Цикл FOR: синтаксис, особенности.
31. Цикл с предусловием: синтаксис, особенности.
32. Цикл с постусловием: синтаксис, особенности.
33. Вложенные циклы.
34. Бесконечный цикл.
35. Операторы continue и break в циклах. Примеры.
36. Массив как тип данных. Определение и инициализация одномерного массива. Доступ к элементам
37. Определение и инициализация двумерного массива. Доступ к элементам.

Второй семестр, Экзамен

Контролируемые ИДК: ОПК-3.1 ОПК-3.2 ОПК-3.3 ОПК-3.4

Вопросы/Задания:

38. Строка как массив символов типа char.
39. Строковые переменные и константы: объявление и инициализация, вывод на экран.
40. Операции со строками: копирование, сравнение, поиск, добавление строки к строке, определение длины.
41. Указатели на массивы и их использование.
42. Указатели на функции и их использование.

43. Указатель на строку, объявленную как массив символов.
44. Указатели (массивы указателей) на строковые константы.
45. Определение одномерного массива строк через массивы указателей на строки.
46. Указатели на строки как аргументы функций.
47. Массивы указателей на строки.
48. Указатели на структуры и их использование.
49. Указатель на void и его использование.
50. Технология использования указателей на указатели.
51. Свободная память. Операции управления new и delete.
52. Оператор new и delete при работе с простыми типами.
53. Оператор new и delete при работе с одномерными динамическими массивами.
54. Многомерные динамические массивы. Алгоритм выделения, алгоритм освобождения памяти.
55. Понятие файлов в C++. Текстовый формат записи в файл.
56. Понятие файлов в C++. Двоичный формат записи в файл. Особенности сохранения чисел в двоичном формате.
57. Режимы файлов.
58. Запись данных в двоичный файл. Чтение данных из двоичного файла.
59. Файловые потоки. Открытие и закрытие
60. Чтение и запись текстовых файлов
61. Дополнительные функции файлового ввода / вывода: функция rdbuf(), write(символьный массив, число символов), read(символьный массив, число символов), gcount(), функция get(). Варианты функции get().
62. Произвольный доступ к файлам: функции реализации «передвижения» по файлу в определенную точку файла. Пример реализации
63. Использование формальных и фактических параметров (аргументов) при обмене информацией с функцией.
64. Передача констант и значений переменных в функцию

65. Передача аргументов по умолчанию (default argument) в функцию
66. Использование массивов в качестве параметров (аргументов) функции.
67. Структурные переменные в качестве аргументов функций.
68. Рекурсивные функции. Примеры.
69. Основные понятия объектно-ориентированного программирования: объект, класс, поле, метод, свойства, сообщение, событие.
70. Визуальное программирование. Основные понятия.
71. Компонент форма - TForm. Добавление к проекту, исключение из проекта, реализация связи между формами. Режимы запуска формы на выполнение. TForm. Свойства формы. Методы формы.
72. Компонент TLabel. Свойства компонента.
73. Компонент TEdit. Свойства, события и методы компонента.
74. Компонент TMemo.
75. Функции перевода данных из одного типа в другой. Примеры.
76. Функции выдачи сообщений на экран. Примеры.
77. Компоненты для работы с массивами
78. Статическая типизация и преобразования типов
79. Перегрузка операторов
80. Перегрузка функций

8. Материально-техническое и учебно-методическое обеспечение дисциплины

8.1. Перечень основной и дополнительной учебной литературы

Основная литература

1. Белева,, Л. Ф. Программирование на языке C++: учебное пособие / Л. Ф. Белева,. - Программирование на языке C++ - Москва: Ай Пи Ар Медиа, 2024. - 81 с. - 978-5-4497-2560-8. - Текст: электронный. // IPR SMART: [сайт]. - URL: <https://www.iprbookshop.ru/134887.html> (дата обращения: 20.02.2024). - Режим доступа: по подписке
2. ИВАНОВА Е. А. Объектно-ориентированное программирование: метод. указания / ИВАНОВА Е. А., Ефанова Н. В.. - Краснодар: КубГАУ, 2020. - 27 с. - Текст: электронный. // : [сайт]. - URL: <https://edu.kubsau.ru/mod/resource/view.php?id=8425> (дата обращения: 02.05.2024). - Режим доступа: по подписке

3. Скворцова Л. А. Объектно-ориентированное программирование на языке C++: учебное пособие / Скворцова Л. А.. - Москва: РТУ МИРЭА, 2020. - 246 с. - Текст: электронный. // RuSpLAN: [сайт]. - URL: <https://e.lanbook.com/img/cover/book/163862.jpg> (дата обращения: 21.02.2024). - Режим доступа: по подписке

4. КРАМАРЕНКО Т. А. Программирование: практикум / КРАМАРЕНКО Т. А., Иванова Е. А.. - Краснодар: КубГАУ, 2020. - 172 с. - Текст: электронный. // : [сайт]. - URL: <https://edu.kubsau.ru/mod/resource/view.php?id=9067> (дата обращения: 13.01.2025). - Режим доступа: по подписке

5. ИВАНОВА Е. А. Объектно-ориентированное программирование: учеб. пособие / ИВАНОВА Е. А., Ефанова Н. В., Крамаренко Т. А.. - Краснодар: КубГАУ, 2018. - 86 с. - 978-5-00097-655-5. - Текст: электронный. // : [сайт]. - URL: <https://edu.kubsau.ru/mod/resource/view.php?id=5142> (дата обращения: 13.01.2025). - Режим доступа: по подписке

Дополнительная литература

1. Ермина,, М. А. Информатика. Алгоритмизация и программирование вычислительных задач: учебное пособие / М. А. Ермина,, Д. А. Ермин,. - Информатика. Алгоритмизация и программирование вычислительных задач - Санкт-Петербург: Санкт-Петербургский государственный университет промышленных технологий и дизайна, 2017. - 103 с. - 978-5-7937-1479-2. - Текст: электронный. // IPR SMART: [сайт]. - URL: <https://www.iprbookshop.ru/102422.html> (дата обращения: 20.02.2024). - Режим доступа: по подписке

2. Немцова, Т.И. Программирование на языке высокого уровня. Программирование на языке C++: Учебное пособие / Т.И. Немцова, С.Ю. Голова, А.И. Терентьев; ООО "ИННОВАЦИЯ" структурное подразделение "Центр Компьютерного Обучения и Дополнительного Образовани.; ООО "ИННОВАЦИЯ" структурное подразделение "Центр Компьютерного Обучения и Дополнительного Образовани. - 1 - Москва: Издательский Дом "ФОРУМ", 2024. - 512 с. - 978-5-16-013214-3. - Текст: электронный. // Общество с ограниченной ответственностью «ЗНАНИУМ»: [сайт]. - URL: <https://znanium.com/cover/2083/2083383.jpg> (дата обращения: 20.02.2024). - Режим доступа: по подписке

3. Хиценко,, В. П. Основы программирования: учебное пособие / В. П. Хиценко,. - Основы программирования - Новосибирск: Новосибирский государственный технический университет, 2015. - 83 с. - 978-5-7782-2706-4. - Текст: электронный. // IPR SMART: [сайт]. - URL: <https://www.iprbookshop.ru/91642.html> (дата обращения: 20.02.2024). - Режим доступа: по подписке

4. Анищик Т. А. Основы алгоритмического программирования на языке Паскаль: учебное пособие / Анищик Т. А.. - Краснодар: КубГАУ, 2017. - 183 с. - 978-5-00097-413-1. - Текст: электронный. // RuSpLAN: [сайт]. - URL: <https://e.lanbook.com/img/cover/book/254267.jpg> (дата обращения: 21.02.2024). - Режим доступа: по подписке

5. Беляков,, С. Л. Основы разработки программ на языке C++ для систем информационной безопасности: учебное пособие / С. Л. Беляков,, А. В. Боженюк,, М. В. Петряева,. - Основы разработки программ на языке C++ для систем информационной безопасности - Ростов-на-Дону, Таганрог: Издательство Южного федерального университета, 2020. - 152 с. - 978-5-9275-3521-7. - Текст: электронный. // IPR SMART: [сайт]. - URL: <https://www.iprbookshop.ru/107970.html> (дата обращения: 20.02.2024). - Режим доступа: по подписке

6. Типовые алгоритмы и их программирование: учебное пособие / Гущин А. Н., Лазарева Т. И., Мартынова И. В., Палехова О. А.. - Санкт-Петербург: БГТУ "Военмех" им. Д.Ф. Устинова, 2016. - 128 с. - 978-5-85546-983-7. - Текст: электронный. // RuSpLAN: [сайт]. - URL: <https://e.lanbook.com/img/cover/book/98227.jpg> (дата обращения: 21.02.2024). - Режим доступа: по подписке

7. Корнеев, В.И. Программирование графики на С++. Теория и примеры: Учебное пособие / В.И. Корнеев, Л.Г. Гагарина, М.В. Корнеева. - 1 - Москва: ООО "Научно-издательский центр ИНФРА-М", 2024. - 517 с. - 978-5-16-106928-8. - Текст: электронный. // Общество с ограниченной ответственностью «ЗНАНИУМ»: [сайт]. - URL: <https://znanium.com/cover/2111/2111934.jpg> (дата обращения: 20.02.2024). - Режим доступа: по подписке

8. Язык С++ и основы технологии объектноориентированного программирования. Часть II: учебно-методическое пособие по специальности ПРИКЛАДНАЯ МАТЕМАТИКА И ИНФОРМАТИКА (01.03.02) / Воронеж: ВГУ, 2017. - 56 с. - Текст: электронный. // RuSpLAN: [сайт]. - URL: <https://e.lanbook.com/img/cover/book/154785.jpg> (дата обращения: 21.02.2024). - Режим доступа: по подписке

9. Воронцова, Е.А. Программирование на С++ с погружением: практические задания и примеры кода: Учебное пособие / Е.А. Воронцова. - 1 - Москва: ООО "Научно-издательский центр ИНФРА-М", 2016. - 80 с. - 978-5-16-105159-7. - Текст: электронный. // Общество с ограниченной ответственностью «ЗНАНИУМ»: [сайт]. - URL: <https://znanium.com/cover/0563/563294.jpg> (дата обращения: 20.02.2024). - Режим доступа: по подписке

10. Рысин М. Л. Основы программирования на языке С++: учебное пособие / Рысин М. Л., Сартаков М. В., Макеева О. В.. - Москва: РТУ МИРЭА, 2022. - 118 с. - Текст: электронный. // RuSpLAN: [сайт]. - URL: <https://e.lanbook.com/img/cover/book/239957.jpg> (дата обращения: 21.02.2024). - Режим доступа: по подписке

8.2. Профессиональные базы данных и ресурсы «Интернет», к которым обеспечивается доступ обучающихся

Профессиональные базы данных

Не используются.

Ресурсы «Интернет»

1. <https://edu.kubsau.ru/> - Образовательный портал КубГАУ
2. <https://znanium.com/> - Znanium.com
3. <http://www.iprbookshop.ru/> - IPRbook

8.3. Программное обеспечение и информационно-справочные системы, используемые при осуществлении образовательного процесса по дисциплине

Информационные технологии, используемые при осуществлении образовательного процесса по дисциплине позволяют:

- обеспечить взаимодействие между участниками образовательного процесса, в том числе синхронное и (или) асинхронное взаимодействие посредством сети «Интернет»;
- фиксировать ход образовательного процесса, результатов промежуточной аттестации по дисциплине и результатов освоения образовательной программы;
- организовать процесс образования путем визуализации изучаемой информации посредством использования презентаций, учебных фильмов;
- контролировать результаты обучения на основе компьютерного тестирования.

Перечень лицензионного программного обеспечения:

1 Microsoft Windows - операционная система.

2 Microsoft Office (включает Word, Excel, Power Point) - пакет офисных приложений.

Перечень профессиональных баз данных и информационных справочных систем:

1 Гарант - правовая, <https://www.garant.ru/>

2 Консультант - правовая, <https://www.consultant.ru/>

3 Научная электронная библиотека eLibrary - универсальная, <https://elibrary.ru/>

Доступ к сети Интернет, доступ в электронную информационно-образовательную среду университета.

Перечень программного обеспечения

(обновление производится по мере появления новых версий программы)

Не используется.

Перечень информационно-справочных систем

(обновление выполняется еженедельно)

Не используется.

8.4. Специальные помещения, лаборатории и лабораторное оборудование

Университет располагает на праве собственности или ином законном основании материально-техническим обеспечением образовательной деятельности (помещениями и оборудованием) для реализации программы бакалавриата, специалитета, магистратуры по Блоку 1 "Дисциплины (модули)" и Блоку 3 "Государственная итоговая аттестация" в соответствии с учебным планом.

Каждый обучающийся в течение всего периода обучения обеспечен индивидуальным неограниченным доступом к электронной информационно-образовательной среде университета из любой точки, в которой имеется доступ к информационно-телекоммуникационной сети "Интернет", как на территории университета, так и вне его. Условия для функционирования электронной информационно-образовательной среды могут быть созданы с использованием ресурсов иных организаций.

Компьютерный класс

222гл

Интерактивная панель Samsung - 1 шт.

Компьютер персональный Aquarius i5/4Gb/500Gb/21,5" - 1 шт.

010зоо

Интерактивная панель Samsung - 1 шт.

Персональный компьютер iRU I5/16GB/512GbSSD - 1 шт.

Лекционный зал

Проектор Epson EH-TW650, белый с креплением и кабелем HDMI - 0 шт.

Сплит-система RODA RS/RU-A12F - 0 шт.

9. Методические указания по освоению дисциплины (модуля)

Учебная работа по направлению подготовки осуществляется в форме контактной работы с преподавателем, самостоятельной работы обучающегося, текущей и промежуточной аттестаций, иных формах, предлагаемых университетом. Учебный материал дисциплины структурирован и его изучение производится в тематической последовательности. Содержание методических указаний должно соответствовать требованиям Федерального государственного образовательного стандарта и учебных программ по дисциплине. Самостоятельная работа студентов может быть выполнена с помощью материалов, размещенных на портале поддержки Moodle.

Методические указания по формам работы

Лекционные занятия

Передача значительного объема систематизированной информации в устной форме достаточно большой аудитории. Дает возможность экономно и систематично излагать учебный материал. Обучающиеся изучают лекционный материал, размещенный на портале поддержки обучения Moodle.

Лабораторные занятия

Практическое освоение студентами научно-теоретических положений изучаемого предмета, овладение ими техникой экспериментирования в соответствующей отрасли науки. Лабораторные занятия проводятся с использованием методических указаний, размещенных на образовательном портале университета.

Описание возможностей изучения дисциплины лицами с ОВЗ и инвалидами

Для инвалидов и лиц с ОВЗ может изменяться объём дисциплины (модуля) в часах, выделенных на контактную работу обучающегося с преподавателем (по видам учебных занятий) и на самостоятельную работу обучающегося (при этом не увеличивается количество зачётных единиц, выделенных на освоение дисциплины).

Фонды оценочных средств адаптируются к ограничениям здоровья и восприятия информации обучающимися.

Основные формы представления оценочных средств – в печатной форме или в форме электронного документа.

Формы контроля и оценки результатов обучения инвалидов и лиц с ОВЗ с нарушением зрения:

– устная проверка: дискуссии, тренинги, круглые столы, собеседования, устные коллоквиумы и др.;

– с использованием компьютера и специального ПО: работа с электронными образовательными ресурсами, тестирование, рефераты, курсовые проекты, дистанционные формы, если позволяет острота зрения - графические работы и др.;

– при возможности письменная проверка с использованием рельефно-точечной системы Брайля, увеличенного шрифта, использование специальных технических средств (тифлотехнических средств): контрольные, графические работы, тестирование, домашние задания, эссе, отчеты и др.

Формы контроля и оценки результатов обучения инвалидов и лиц с ОВЗ с нарушением слуха:

– письменная проверка: контрольные, графические работы, тестирование, домашние задания, эссе, письменные коллоквиумы, отчеты и др.;

- с использованием компьютера: работа с электронными образовательными ресурсами, тестирование, рефераты, курсовые проекты, графические работы, дистанционные формы и др.;

- при возможности устная проверка с использованием специальных технических средств (аудиосредств, средств коммуникации, звукоусиливающей аппаратуры и др.): дискуссии, тренинги, круглые столы, собеседования, устные коллоквиумы и др.

Формы контроля и оценки результатов обучения инвалидов и лиц с ОВЗ с нарушением опорно-двигательного аппарата:

- письменная проверка с использованием специальных технических средств (альтернативных средств ввода, управления компьютером и др.): контрольные, графические работы, тестирование, домашние задания, эссе, письменные коллоквиумы, отчеты и др.;

- устная проверка, с использованием специальных технических средств (средств коммуникаций): дискуссии, тренинги, круглые столы, собеседования, устные коллоквиумы и др.;

- с использованием компьютера и специального ПО (альтернативных средств ввода и управления компьютером и др.): работа с электронными образовательными ресурсами, тестирование, рефераты, курсовые проекты, графические работы, дистанционные формы предпочтительнее обучающимся, ограниченным в передвижении и др.

Адаптация процедуры проведения промежуточной аттестации для инвалидов и лиц с ОВЗ.

В ходе проведения промежуточной аттестации предусмотрено:

- предъявление обучающимся печатных и (или) электронных материалов в формах, адаптированных к ограничениям их здоровья;

- возможность пользоваться индивидуальными устройствами и средствами, позволяющими адаптировать материалы, осуществлять приём и передачу информации с учетом их индивидуальных особенностей;

- увеличение продолжительности проведения аттестации;

- возможность присутствия ассистента и оказания им необходимой помощи (занять рабочее место, передвигаться, прочесть и оформить задание, общаться с преподавателем).

Формы промежуточной аттестации для инвалидов и лиц с ОВЗ должны учитывать индивидуальные и психофизические особенности обучающегося/обучающихся по АОПОП ВО (устно, письменно на бумаге, письменно на компьютере, в форме тестирования и т.п.).

Специальные условия, обеспечиваемые в процессе преподавания дисциплины студентам с нарушениями зрения:

- предоставление образовательного контента в текстовом электронном формате, позволяющем переводить плоскостную информацию в аудиальную или тактильную форму;

- возможность использовать индивидуальные устройства и средства, позволяющие адаптировать материалы, осуществлять приём и передачу информации с учетом индивидуальных особенностей и состояния здоровья студента;

- предоставление возможности предкурсового ознакомления с содержанием учебной дисциплины и материалом по курсу за счёт размещения информации на корпоративном образовательном портале;

- использование чёткого и увеличенного по размеру шрифта и графических объектов в мультимедийных презентациях;

- использование инструментов «лупа», «проектор» при работе с интерактивной доской;

- озвучивание визуальной информации, представленной обучающимся в ходе занятий;

- обеспечение раздаточным материалом, дублирующим информацию, выводимую на экран;

- наличие подписей и описания у всех используемых в процессе обучения рисунков и иных графических объектов, что даёт возможность перевести письменный текст в аудиальный;

- обеспечение особого речевого режима преподавания: лекции читаются громко, разборчиво, отчётливо, с паузами между смысловыми блоками информации, обеспечивается интонирование, повторение, акцентирование, профилактика рассеивания внимания;

- минимизация внешнего шума и обеспечение спокойной аудиальной обстановки;

- возможность вести запись учебной информации студентами в удобной для них форме (аудиально, аудиовизуально, на ноутбуке, в виде пометок в заранее подготовленном тексте);

- увеличение доли методов социальной стимуляции (обращение внимания, апелляция к

ограничениям по времени, контактные виды работ, групповые задания и др.) на практических и лабораторных занятиях;

- минимизирование заданий, требующих активного использования зрительной памяти и зрительного внимания;
- применение поэтапной системы контроля, более частый контроль выполнения заданий для самостоятельной работы.

Специальные условия, обеспечиваемые в процессе преподавания дисциплины студентам с нарушениями опорно-двигательного аппарата (маломобильные студенты, студенты, имеющие трудности передвижения и патологию верхних конечностей):

- возможность использовать специальное программное обеспечение и специальное оборудование и позволяющее компенсировать двигательное нарушение (коляски, ходунки, трости и др.);
- предоставление возможности предкурсового ознакомления с содержанием учебной дисциплины и материалом по курсу за счёт размещения информации на корпоративном образовательном портале;
- применение дополнительных средств активизации процессов запоминания и повторения;
- опора на определенные и точные понятия;
- использование для иллюстрации конкретных примеров;
- применение вопросов для мониторинга понимания;
- разделение изучаемого материала на небольшие логические блоки;
- увеличение доли конкретного материала и соблюдение принципа от простого к сложному при объяснении материала;
- наличие чёткой системы и алгоритма организации самостоятельных работ и проверки заданий с обязательной корректировкой и комментариями;
- увеличение доли методов социальной стимуляции (обращение внимания, апелляция к ограничениям по времени, контактные виды работ, групповые задания др.);
- обеспечение беспрепятственного доступа в помещения, а также пребывания в них;
- наличие возможности использовать индивидуальные устройства и средства, позволяющие обеспечить реализацию эргономических принципов и комфортное пребывание на месте в течение всего периода учёбы (подставки, специальные подушки и др.).

Специальные условия, обеспечиваемые в процессе преподавания дисциплины студентам с нарушениями слуха (глухие, слабослышащие, позднооглохшие):

- предоставление образовательного контента в текстовом электронном формате, позволяющем переводить аудиальную форму лекции в плоскпечатную информацию;
- наличие возможности использовать индивидуальные звукоусиливающие устройства и сурдотехнические средства, позволяющие осуществлять приём и передачу информации; осуществлять взаимообратный перевод текстовых и аудиофайлов (блокнот для речевого ввода), а также запись и воспроизведение зрительной информации;
- наличие системы заданий, обеспечивающих систематизацию вербального материала, его схематизацию, перевод в таблицы, схемы, опорные тексты, глоссарий;
- наличие наглядного сопровождения изучаемого материала (структурно-логические схемы, таблицы, графики, концентрирующие и обобщающие информацию, опорные конспекты, раздаточный материал);
- наличие чёткой системы и алгоритма организации самостоятельных работ и проверки заданий с обязательной корректировкой и комментариями;
- обеспечение практики опережающего чтения, когда студенты заранее знакомятся с материалом и выделяют незнакомые и непонятные слова и фрагменты;
- особый речевой режим работы (отказ от длинных фраз и сложных предложений, хорошая артикуляция; четкость изложения, отсутствие лишних слов; повторение фраз без изменения слов и порядка их следования; обеспечение зрительного контакта во время говорения и чуть более медленного темпа речи, использование естественных жестов и мимики);
- чёткое соблюдение алгоритма занятия и заданий для самостоятельной работы (называние темы, постановка цели, сообщение и запись плана, выделение основных понятий и методов их изучения, указание видов деятельности студентов и способов проверки усвоения материала, словарная работа);

- соблюдение требований к предъявляемым учебным текстам (разбивка текста на части; выделение опорных смысловых пунктов; использование наглядных средств);
- минимизация внешних шумов;
- предоставление возможности соотносить вербальный и графический материал; комплексное использование письменных и устных средств коммуникации при работе в группе;
- сочетание на занятиях всех видов речевой деятельности (говорения, слушания, чтения, письма, зрительного восприятия с лица говорящего).

Специальные условия, обеспечиваемые в процессе преподавания дисциплины студентам с прочими видами нарушений (ДЦП с нарушениями речи, заболевания эндокринной, центральной нервной и сердечно-сосудистой систем, онкологические заболевания):

- наличие возможности использовать индивидуальные устройства и средства, позволяющие осуществлять приём и передачу информации;
- наличие системы заданий, обеспечивающих систематизацию вербального материала, его схематизацию, перевод в таблицы, схемы, опорные тексты, глоссарий;
- наличие наглядного сопровождения изучаемого материала;
- наличие чёткой системы и алгоритма организации самостоятельных работ и проверки заданий с обязательной корректировкой и комментариями;
- обеспечение практики опережающего чтения, когда студенты заранее знакомятся с материалом и выделяют незнакомые и непонятные слова и фрагменты;
- предоставление возможности соотносить вербальный и графический материал; комплексное использование письменных и устных средств коммуникации при работе в группе;
- сочетание на занятиях всех видов речевой деятельности (говорения, слушания, чтения, письма, зрительного восприятия с лица говорящего);
- предоставление образовательного контента в текстовом электронном формате;
- предоставление возможности предкурсового ознакомления с содержанием учебной дисциплины и материалом по курсу за счёт размещения информации на корпоративном образовательном портале;
- возможность вести запись учебной информации студентами в удобной для них форме (аудиально, аудиовизуально, в виде пометок в заранее подготовленном тексте);
- применение поэтапной системы контроля, более частый контроль выполнения заданий для самостоятельной работы;
- стимулирование выработки у студентов навыков самоорганизации и самоконтроля;
- наличие пауз для отдыха и смены видов деятельности по ходу занятия.

10. Методические рекомендации по освоению дисциплины (модуля)