

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



УТВЕРЖДЕНО

Декан

Замотайлова Д.А.

Протокол от 23.04.2026 № 6

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

Уровень высшего образования: бакалавриат

Направление подготовки: 09.03.03 Прикладная информатика

Направленность (профиль) подготовки: Менеджмент ИТ-проектов, управление жизненным циклом информационных систем

Квалификация (степень) выпускника: бакалавр

Формы обучения: очная, заочная

Год набора (приема на обучение): 2026

Срок получения образования: Очная форма обучения – 4 года
Заочная форма обучения – 4 года 8 месяца(-ев)

Объем: в зачетных единицах: 6 з.е.
в академических часах: 216 ак.ч.

Разработчики:

Доцент, кафедра системного анализа и обработки информации Крамаренко Т.А.

Рабочая программа дисциплины (модуля) составлена в соответствии с требованиями ФГОС ВО по направлению подготовки 09.03.03 Прикладная информатика, утвержденного приказом Минобрнауки от 19.09.2017 № 922, с учетом трудовых функций профессиональных стандартов: "Специалист по информационным системам", утвержден приказом Минтруда России от 13.07.2023 № 586н; "Руководитель проектов в области информационных технологий", утвержден приказом Минтруда России от 27.04.2023 № 369н; "Руководитель проектов в области информационных технологий", утвержден приказом Минтруда России от 18.11.2014 № 893н; "Специалист по информационным системам", утвержден приказом Минтруда России от 18.11.2014 № 896н.

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	13.04.2026, № 8

1. Цель и задачи освоения дисциплины (модуля)

Цель освоения дисциплины - В изучение основных особенностей разработки алгоритмов и реализация их на одном из языков программирования высокого уровня, основы объектноориентированной методологии программирования. Преподавание дисциплины строится исходя из требуемого уровня базовой подготовки в области разработки алгоритмов и программ на языках высокого уровня. Конечная цель изучения дисциплины - формирование твердых теоретических знаний и практических навыков по составлению алгоритмов и объектноориентированных программ.

Задачи изучения дисциплины:

- Сформировать у обучающихся навыки работы с современными информационными технологиями для проектирования и разработки программных продуктов и их прототипов;
- Сформировать у обучающихся навыки адаптации прикладного программного обеспечения в соответствии с заданием.

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

Компетенции, индикаторы и результаты обучения

ОПК-2 Способен использовать современные информационные технологии и программные средства, в том числе отечественного производства, при решении задач профессиональной деятельности

ОПК-2.1 Знает современные информационные технологии и программные средства, в том числе отечественного производства при решении задач профессиональной деятельности

Знать:

ОПК-2.1/Зн1 Современные информационные технологии для решения задач профессиональной деятельности

ОПК-2.1/Зн2 Современные программные средства, в том числе отечественного производства при решении задач профессиональной деятельности

ОПК-2.2 Умеет выбирать современные информационные технологии и программные средства, в том числе отечественного производства при решении задач профессиональной деятельности

Уметь:

ОПК-2.2/Ум1 Выбирать современные информационные технологии для решения задач профессиональной деятельности

ОПК-2.2/Ум2 Выбирать современные программные средства, в том числе отечественного производства для решения задач профессиональной деятельности

ОПК-2.3 Владеет навыками применения современных информационных технологий и программных средств, в том числе отечественного производства, при решении задач профессиональной деятельности

Владеть:

ОПК-2.3/Нв1 Навыками применения современных информационных технологий для решения задач профессиональной деятельности

ОПК-2.3/Нв2 Навыками применения современных программных средств, в том числе отечественного производства для решения задач профессиональной деятельности

ОПК-3 Способен решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности

ОПК-3.1 Знает принципы, методы и средства решения стандартных задач профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности

Знать:

ОПК-3.1/Зн1 Принципы решения стандартных задач профессиональной деятельности

ОПК-3.1/Зн2 Методы решения стандартных задач профессиональной деятельности

ОПК-3.1/Зн3 Средства решения стандартных задач профессиональной деятельности

ОПК-3.2 Умеет решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности

Уметь:

ОПК-3.2/Ум1 Решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий

ОПК-3.2/Ум2 Решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с учетом требований информационной безопасности

ОПК-3.3 Владеет навыками подготовки обзоров, аннотаций, составления рефератов, научных докладов, публикаций, и библиографии по научно-исследовательской работе с учетом требований информационной безопасности

Владеть:

ОПК-3.3/Нв1 Навыками подготовки аннотаций с учетом требований информационной безопасности

ОПК-3.3/Нв2 Навыками составления рефератов с учетом требований информационной безопасности

ОПК-3.3/Нв3 Навыками составления научных докладов с учетом требований информационной безопасности

ОПК-3.3/Нв4 Навыками составления библиографии по научно-исследовательской работе с учетом требований информационной безопасности

ОПК-7 Способен разрабатывать алгоритмы и программы, пригодные для практического применения

ОПК-7.1 Знает основные языки программирования и работы с базами данных, операционные системы и оболочки, современные программные среды разработки информационных систем и технологий

Знать:

ОПК-7.1/Зн1 Основные языки программирования

ОПК-7.1/Зн2 Основные языки работы с базами данных

ОПК-7.1/Зн3 Современные программные среды разработки информационных систем и технологий

ОПК-7.2 Умеет 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

Уметь:

ОПК-7.2/Ум1 Применять языки программирования

ОПК-7.2/Ум2 Применять языки работы с базами данных

ОПК-7.2/Ум3 Применять современные программные среды разработки информационных систем и технологий

ОПК-7.3 Владеет навыками программирования, отладки и тестирования прототипов программно-технических комплексов задач

Владеть:

ОПК-7.3/Нв1 Навыками программирования прототипов программно-технических комплексов задач

ОПК-7.3/Нв2 Навыками отладки прототипов программно-технических комплексов задач

ОПК-7.3/Нв3 Навыками тестирования прототипов программно-технических комплексов задач

ПК-П2 Способен разрабатывать и адаптировать прикладное программное обеспечение

ПК-П2.1 Знает теоретическое и практическое содержание этапов процесса внедрения, адаптации и настройки прикладного программного обеспечения

Знать:

ПК-П2.1/Зн1 Основы управления изменениями

ПК-П2.1/Зн3 Возможности ис

Уметь:

ПК-П2.1/Ум1 Работать с записями по качеству (в том числе с корректирующими действиями, предупреждающими действиями, запросами на исправление несоответствий)

Владеть:

ПК-П2.1/Нв1 Контроль фактического внесения изменений в элементы ис

ПК-П2.1/Нв2 Изменение статуса проверенных запросов на изменение в системе учета требований проекта в области ит

ПК-П2.2 Умеет организовывать и управлять процессом внедрения, адаптации и настройки прикладного программного обеспечения, в т.ч., распределять работы, выделять ресурсы, контролировать исполнение

Знать:

ПК-П2.2/Зн1 Языки программирования и работы с базами данных

ПК-П2.2/Зн2 Инструменты и методы модульного тестирования

ПК-П2.2/Зн3 Инструменты и методы тестирования нефункциональных и функциональных характеристик ис

ПК-П2.2/Зн4 Инструменты и методы прототипирования пользовательского интерфейса

ПК-П2.2/Зн5 Возможности типовой ис

ПК-П2.2/Зн6 Предметная область автоматизации

ПК-П2.2/Зн10 Основы современных субд
ПК-П2.2/Зн11 Устройство и функционирование современных ис
ПК-П2.2/Зн14 Основы программирования
ПК-П2.2/Зн15 Современные объектно-ориентированные языки программирования
ПК-П2.2/Зн16 Современные структурные языки программирования
ПК-П2.2/Зн19 Современные стандарты информационного взаимодействия систем
ПК-П2.2/Зн21 Современные подходы и стандарты автоматизации организации
ПК-П2.2/Зн25 Лучшие практики создания (модификации) и сопровождения ис в экономике

Уметь:

ПК-П2.2/Ум1 Кодировать на языках программирования в рамках выполнения работ по созданию (модификации) и сопровождению ис
ПК-П2.2/Ум2 Тестировать результаты прототипирования ис в рамках выполнения работ по созданию (модификации) и сопровождению ис
ПК-П2.2/Ум3 Проводить презентации в рамках выполнения работ по созданию (модификации) и сопровождению ис

Владеть:

ПК-П2.2/Нв1 Разработка прототипа ис на базе типовой ис в соответствии с требованиями заказчика к ис
ПК-П2.2/Нв2 Тестирование прототипа ис на корректность архитектурных решений
ПК-П2.2/Нв3 Обработка результатов тестирования прототипа ис на корректность архитектурных решений

ПК-П2.3 Владеет навыками внедрения, адаптации и настройки прикладного программного обеспечения, обеспечивая соответствие и контроль разработанного кода и процесса кодирования принятым в организации регламентам и стандартам

Знать:

ПК-П2.3/Зн1 Инструменты и методы верификации структуры программного кода
ПК-П2.3/Зн2 Регламенты кодирования на языках программирования
ПК-П2.3/Зн3 Возможности ис
ПК-П2.3/Зн4 Предметная область автоматизации
ПК-П2.3/Зн5 Источники информации, необходимой для профессиональной деятельности при выполнении работ и управлении работами по созданию (модификации) и сопровождению ис
ПК-П2.3/Зн6 Лучшие практики создания (модификации) и сопровождения ис в экономике

Уметь:

ПК-П2.3/Ум1 Распределять работы и выделять ресурсы в рамках выполнения работ и управления работами по созданию (модификации) и сопровождению ис
ПК-П2.3/Ум2 Контролировать исполнение поручений в рамках выполнения работ и управления работами по созданию (модификации) и сопровождению ис

Владеть:

ПК-П2.3/Нв1 Обеспечение соответствия разработанного кода ис и процесса создания программного кода ис принятым в организации или проекте стандартам и технологиям в рамках выполнения работ и управления работами по созданию (модификации) и сопровождению ис
ПК-П2.3/Нв2 Назначение и распределение ресурсов в рамках выполнения работ и управления работами по созданию (модификации) и сопровождению ис

ПК-П2.3/Нв3 Контроль соответствия разработанного кода ис и процесса создания программного кода ис принятым в организации или проекте стандартам и технологиям в рамках выполнения работ и управления работами по созданию (модификации) и сопровождению ис

3. Место дисциплины в структуре ОП

Дисциплина (модуль) «Алгоритмизация и программирование» относится к обязательной части образовательной программы и изучается в семестре(ах): Очная форма обучения - 1, 2, Заочная форма обучения - 1, 2.

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

4. Объем дисциплины (модуля) и виды учебной работы

Очная форма обучения

Период обучения	Общая трудоемкость (часы)	Общая трудоемкость (ЗЕТ)	Контактная работа (часы, всего)	Внеаудиторная контактная работа (часы)	Зачет (часы)	Лабораторные занятия (часы)	Лекционные занятия (часы)	Практические занятия (часы)	Самостоятельная работа (часы)	Промежуточная аттестация (часы)
Первый семестр	72	2	61	1		30	30		11	Зачет
Второй семестр	144	4	69	5		16	32	16	48	Курсовая работа Экзамен (27)
Всего	216	6	130	6		46	62	16	59	27

Заочная форма обучения

Период обучения	Общая трудоемкость (часы)	Общая трудоемкость (ЗЕТ)	Контактная работа (часы, всего)	Внеаудиторная контактная работа (часы)	Зачет (часы)	Лабораторные занятия (часы)	Лекционные занятия (часы)	Практические занятия (часы)	Самостоятельная работа (часы)	Промежуточная аттестация (часы)
Первый семестр	72	2	7	1		4	2		65	Зачет
Второй семестр	144	4	17	5		4	2	6	127	Курсовая работа Экзамен
Всего	216	6	24	6		8	4	6	192	

5. Содержание дисциплины (модуля)
5.1. Разделы, темы дисциплины и виды занятий
(часы промежуточной аттестации не указываются)

Очная форма обучения

Наименование раздела, темы	Всего	Внеаудиторная контактная работа	Лабораторные занятия	Лекционные занятия	Практические занятия	Самостоятельная работа	Планируемые результаты обучения, соответствующие результатам освоения программы
Раздел 1. Разработка алгоритмов и процедурно-ориентированных программ	71		30	30		11	ОПК-2.1 ОПК-2.2 ОПК-2.3
Тема 1.1. Алгоритмизация процессов обработки данных. Введение в программирование	5		2	2		1	
Тема 1.2. Особенности разработки программы на C++. Лексические основы языка	13		6	6		1	
Тема 1.3. Программирование разветвлений на C++	5		2	2		1	
Тема 1.4. Реализация алгоритмов с циклической структурой на C++	14		8	4		2	
Тема 1.5. Массивы одномерные и многомерные в C++. Строки как массив символов	16		6	8		2	
Тема 1.6. Структуры. Объединения. Перечисления	6		2	2		2	
Тема 1.7. Программирование функций	12		4	6		2	
Раздел 2. Промежуточная аттестация	1	1					ОПК-2.1 ОПК-2.2 ОПК-2.3 ОПК-3.1 ОПК-3.2 ОПК-3.3
Тема 2.1. Зачет	1	1					ОПК-7.1 ОПК-7.2 ОПК-7.3 ПК-П2.1 ПК-П2.2 ПК-П2.3

Раздел 3. Использование указателей, динамической памяти и файлового ввода-вывода при разработке программ	30		4	10	4	12	ОПК-7.1 ОПК-7.2 ОПК-7.3
Тема 3.1. Указатели. Динамические переменные	16		2	6	2	6	
Тема 3.2. Файловый ввод-вывод	14		2	4	2	6	
Раздел 4. Разработка приложений с графическим интерфейсом	28		4	8	4	12	ПК-П2.1 ПК-П2.2 ПК-П2.3
Тема 4.1. Основы объектно-ориентированного программирования	8			2		6	
Тема 4.2. Основы разработки windows-приложений с использованием инструментальной среды разработки и отладки программ в Microsoft Visual Studio 20XX	20		4	6	4	6	
Раздел 5. Разрабока собственных классов и обработка исключительных ситуаций	54		8	14	8	24	ОПК-3.1 ОПК-3.2 ОПК-3.3
Тема 5.1. Классы и инкапсуляция	16		2	4	4	6	
Тема 5.2. Наследование	14		2	4	2	6	
Тема 5.3. Полиморфизм	14		2	4	2	6	
Тема 5.4. Обработка исключительных ситуаций	10		2	2		6	
Раздел 6. Курсовая работа	2	2					ОПК-2.1 ОПК-2.2 ОПК-2.3 ОПК-3.1 ОПК-3.2 ОПК-3.3
Тема 6.1. Защита курсовой работы	2	2					ОПК-7.1 ОПК-7.2 ОПК-7.3 ПК-П2.1 ПК-П2.2 ПК-П2.3
Раздел 7. Промежуточная аттестация	3	3					ОПК-2.1 ОПК-2.2 ОПК-2.3 ОПК-3.1 ОПК-3.2 ОПК-3.3

Тема 7.1. Экзамен	3	3					ОПК-7.1 ОПК-7.2 ОПК-7.3 ПК-П2.1 ПК-П2.2 ПК-П2.3
Итого	189	6	46	62	16	59	

Заочная форма обучения

Наименование раздела, темы	Всего	Внеаудиторная контактная работа	Лабораторные занятия	Лекционные занятия	Практические занятия	Самостоятельная работа	Планируемые результаты обучения, соответствующие результатам освоения программы
Раздел 1. Разработка алгоритмов и процедурно-ориентированных программ	71		4	2		65	ОПК-2.1 ОПК-2.2 ОПК-2.3
Тема 1.1. Алгоритмизация процессов обработки данных. Введение в программирование	8					8	
Тема 1.2. Особенности разработки программы на C++. Лексические основы языка	10					10	
Тема 1.3. Программирование разветвлений на C++	8					8	
Тема 1.4. Реализация алгоритмов с циклической структурой на C++	10					10	
Тема 1.5. Массивы одномерные и многомерные в C++. Строки как массив символов	13		2	1		10	
Тема 1.6. Структуры. Объединения. Перечисления	8					8	
Тема 1.7. Программирование функций	14		2	1		11	
Раздел 2. Промежуточная аттестация	1	1					ОПК-2.1 ОПК-2.2 ОПК-2.3 ОПК-3.1 ОПК-3.2 ОПК-3.3
Тема 2.1. Зачет	1	1					ОПК-7.1 ОПК-7.2 ОПК-7.3 ПК-П2.1 ПК-П2.2 ПК-П2.3

Раздел 3. Использование указателей, динамической памяти и файлового ввода-вывода при разработке программ	35		4	1	2	28	ОПК-7.1 ОПК-7.2 ОПК-7.3
Тема 3.1. Указатели. Динамические переменные	19		4	1		14	
Тема 3.2. Файловый ввод-вывод	16				2	14	
Раздел 4. Разработка приложений с графическим интерфейсом	43			1	2	40	ПК-П2.1 ПК-П2.2 ПК-П2.3
Тема 4.1. Основы объектно-ориентированного программирования	15			1		14	
Тема 4.2. Основы разработки windows-приложений с использованием инструментальной среды разработки и отладки программ в Microsoft Visual Studio 20XX	28				2	26	
Раздел 5. Разрабока собственных классов и обработка исключительных ситуаций	61				2	59	ОПК-3.1 ОПК-3.2 ОПК-3.3
Тема 5.1. Классы и инкапсуляция	20				2	18	
Тема 5.2. Наследование	12					12	
Тема 5.3. Полиморфизм	12					12	
Тема 5.4. Обработка исключительных ситуаций	17					17	
Раздел 6. Курсовая работа	2	2					ОПК-2.1 ОПК-2.2 ОПК-2.3 ОПК-3.1 ОПК-3.2 ОПК-3.3
Тема 6.1. Защита курсовой работы	2	2					ОПК-7.1 ОПК-7.2 ОПК-7.3 ПК-П2.1 ПК-П2.2 ПК-П2.3
Раздел 7. Промежуточная аттестация	3	3					ОПК-2.1 ОПК-2.2 ОПК-2.3 ОПК-3.1 ОПК-3.2 ОПК-3.3

Тема 7.1. Экзамен	3	3					ОПК-7.1 ОПК-7.2 ОПК-7.3 ПК-П2.1 ПК-П2.2 ПК-П2.3
Итого	216	6	8	4	6	192	

5.2. Содержание разделов, тем дисциплин

Раздел 1. Разработка алгоритмов и процедурно-ориентированных программ

(Заочная: Лабораторные занятия - 4ч.; Лекционные занятия - 2ч.; Самостоятельная работа - 65ч.; Очная: Лабораторные занятия - 30ч.; Лекционные занятия - 30ч.; Самостоятельная работа - 11ч.)

Тема 1.1. Алгоритмизация процессов обработки данных. Введение в программирование

(Очная: Лабораторные занятия - 2ч.; Лекционные занятия - 2ч.; Самостоятельная работа - 1ч.; Заочная: Самостоятельная работа - 8ч.)

Алгоритмизация процессов обработки данных. Введение в программирование

Тема 1.2. Особенности разработки программы на C++. Лексические основы языка

(Очная: Лабораторные занятия - 6ч.; Лекционные занятия - 6ч.; Самостоятельная работа - 1ч.; Заочная: Самостоятельная работа - 10ч.)

Особенности разработки программы на C++. Лексические основы языка

Тема 1.3. Программирование разветвлений на C++

(Очная: Лабораторные занятия - 2ч.; Лекционные занятия - 2ч.; Самостоятельная работа - 1ч.; Заочная: Самостоятельная работа - 8ч.)

Программирование разветвлений на C++

Тема 1.4. Реализация алгоритмов с циклической структурой на C++

(Очная: Лабораторные занятия - 8ч.; Лекционные занятия - 4ч.; Самостоятельная работа - 2ч.; Заочная: Самостоятельная работа - 10ч.)

Реализация алгоритмов с циклической структурой на C++

Тема 1.5. Массивы одномерные и многомерные в C++. Строки как массив символов

(Заочная: Лабораторные занятия - 2ч.; Лекционные занятия - 1ч.; Самостоятельная работа - 10ч.; Очная: Лабораторные занятия - 6ч.; Лекционные занятия - 8ч.; Самостоятельная работа - 2ч.)

Массивы одномерные и многомерные в C++. Строки как массив символов

Тема 1.6. Структуры. Объединения. Перечисления

(Очная: Лабораторные занятия - 2ч.; Лекционные занятия - 2ч.; Самостоятельная работа - 2ч.; Заочная: Самостоятельная работа - 8ч.)

Структуры. Объединения. Перечисления

Тема 1.7. Программирование функций

(Заочная: Лабораторные занятия - 2ч.; Лекционные занятия - 1ч.; Самостоятельная работа - 11ч.; Очная: Лабораторные занятия - 4ч.; Лекционные занятия - 6ч.; Самостоятельная работа - 2ч.)

Программирование функций

Раздел 2. Промежуточная аттестация

(Заочная: Внеаудиторная контактная работа - 1ч.; Очная: Внеаудиторная контактная работа - 1ч.)

Тема 2.1. Зачет

(Заочная: Внеаудиторная контактная работа - 1ч.; Очная: Внеаудиторная контактная работа - 1ч.)

Проведение промежуточной аттестации в форме зачета

Раздел 3. Использование указателей, динамической памяти и файлового ввода-вывода при разработке программ

(Заочная: Лабораторные занятия - 4ч.; Лекционные занятия - 1ч.; Практические занятия - 2ч.; Самостоятельная работа - 28ч.; Очная: Лабораторные занятия - 4ч.; Лекционные занятия - 10ч.; Практические занятия - 4ч.; Самостоятельная работа - 12ч.)

Тема 3.1. Указатели. Динамические переменные

(Заочная: Лабораторные занятия - 4ч.; Лекционные занятия - 1ч.; Самостоятельная работа - 14ч.; Очная: Лабораторные занятия - 2ч.; Лекционные занятия - 6ч.; Практические занятия - 2ч.; Самостоятельная работа - 6ч.)

Указатели. Динамические переменные

Тема 3.2. Файловый ввод-вывод

(Очная: Лабораторные занятия - 2ч.; Лекционные занятия - 4ч.; Практические занятия - 2ч.; Самостоятельная работа - 6ч.; Заочная: Практические занятия - 2ч.; Самостоятельная работа - 14ч.)

Файловый ввод-вывод

Раздел 4. Разработка приложений с графическим интерфейсом

(Очная: Лабораторные занятия - 4ч.; Лекционные занятия - 8ч.; Практические занятия - 4ч.; Самостоятельная работа - 12ч.; Заочная: Лекционные занятия - 1ч.; Практические занятия - 2ч.; Самостоятельная работа - 40ч.)

Тема 4.1. Основы объектно-ориентированного программирования

(Заочная: Лекционные занятия - 1ч.; Самостоятельная работа - 14ч.; Очная: Лекционные занятия - 2ч.; Самостоятельная работа - 6ч.)

Основы объектно-ориентированного программирования

Тема 4.2. Основы разработки windows-приложений с использованием инструментальной среды разработки и отладки программ в Microsoft Visual Studio 20XX

(Очная: Лабораторные занятия - 4ч.; Лекционные занятия - 6ч.; Практические занятия - 4ч.; Самостоятельная работа - 6ч.; Заочная: Практические занятия - 2ч.; Самостоятельная работа - 26ч.)

Основы разработки windows-приложений с использованием инструментальной среды разработки и отладки программ в Microsoft Visual Studio 20XX

Раздел 5. Разработка собственных классов и обработка исключительных ситуаций

(Очная: Лабораторные занятия - 8ч.; Лекционные занятия - 14ч.; Практические занятия - 8ч.; Самостоятельная работа - 24ч.; Заочная: Практические занятия - 2ч.; Самостоятельная работа - 59ч.)

Тема 5.1. Классы и инкапсуляция

(Очная: Лабораторные занятия - 2ч.; Лекционные занятия - 4ч.; Практические занятия - 4ч.; Самостоятельная работа - 6ч.; Заочная: Практические занятия - 2ч.; Самостоятельная работа - 18ч.)

Классы и инкапсуляция

Тема 5.2. Наследование

(Очная: Лабораторные занятия - 2ч.; Лекционные занятия - 4ч.; Практические занятия - 2ч.; Самостоятельная работа - 6ч.; Заочная: Самостоятельная работа - 12ч.)

Классы и наследование

Тема 5.3. Полиморфизм

(Очная: Лабораторные занятия - 2ч.; Лекционные занятия - 4ч.; Практические занятия - 2ч.; Самостоятельная работа - 6ч.; Заочная: Самостоятельная работа - 12ч.)

Полиморфизм

Тема 5.4. Обработка исключительных ситуаций

(Очная: Лабораторные занятия - 2ч.; Лекционные занятия - 2ч.; Самостоятельная работа - 6ч.; Заочная: Самостоятельная работа - 17ч.)

Обработка исключительных ситуаций

Раздел 6. Курсовая работа

(Заочная: Внеаудиторная контактная работа - 2ч.; Очная: Внеаудиторная контактная работа - 2ч.)

Тема 6.1. Защита курсовой работы

(Заочная: Внеаудиторная контактная работа - 2ч.; Очная: Внеаудиторная контактная работа - 2ч.)

Защита курсовой работы

Раздел 7. Промежуточная аттестация

(Заочная: Внеаудиторная контактная работа - 3ч.; Очная: Внеаудиторная контактная работа - 3ч.)

Тема 7.1. Экзамен

(Заочная: Внеаудиторная контактная работа - 3ч.; Очная: Внеаудиторная контактная работа - 3ч.)

Проведение промежуточной аттестации в форме экзамена

6. Оценочные материалы текущего контроля

Раздел 1. Разработка алгоритмов и процедурно-ориентированных программ

Форма контроля/оценочное средство: Задача

Вопросы/Задания:

1. Установите последовательность этапов разработки программы
1. Анализ задачи - А
- 2.Разработка алгоритма - Б
- 3.Написание кода - В
- 4.Тестирование - Г

2. Соотнесите языки программирования с их основным применением

1. Python - В. Data Science

2. C++ - А. Системное программирование

3. SQL - Б. Работа с базами данных

3. Какой язык программирования относится к отечественным разработкам

А) Java

Б) 1С

В) Python

4. Укажите правильный порядок обработки исключений

1. Возникновение исключения - А

2. Перехват исключения - Б

3. Обработка исключения - В

5. Соотнесите IDE с поддерживаемыми языками

1. PyCharm - А. Python

2. Visual Studio - В. C#

3. Qt Creator - Б. C++

6. Прочитайте задание и запишите развернутый, обоснованный ответ

Назовите два преимущества использования современных IDE

7. Прочитайте задание и запишите развернутый, обоснованный ответ

Какие критерии важны при выборе языка программирования

8. Прочитайте задание и запишите развернутый, обоснованный ответ

Почему важно использовать системы контроля версий

9. Прочитайте задание и запишите развернутый, обоснованный ответ

Какие преимущества даёт использование фреймворков

10. Прочитайте задание и запишите развернутый, обоснованный ответ

Как современные технологии влияют на производительность разработки

11. Какой фактор наиболее важен при выборе инструмента разработки

А) Цвет интерфейса

Б) Соответствие задачам проекта

В) Популярность среди друзей

12. Какие технологии повышают эффективность разработки

А) Системы контроля версий

Б) Ручное тестирование без автоматизации

В) CI/CD системы

13. Почему важно учитывать кроссплатформенность при разработке

А) Чтобы усложнить поддержку

Б) Для увеличения аудитории пользователей

В) Чтобы уменьшить функциональность

14. Какие методы улучшают качество кода

- А) Регулярный рефакторинг
- Б) Использование "магических чисел"
- В) Написание unit-тестов

15. Какой подход к разработке обеспечивает наибольшую гибкость

- А) Waterfall
- Б) Agile
- В) Ad-hoc

Раздел 2. Промежуточная аттестация

Форма контроля/оценочное средство:

Вопросы/Задания:

.

Раздел 3. Использование указателей, динамической памяти и файлового ввода-вывода при разработке программ

Форма контроля/оценочное средство: Задача

Вопросы/Задания:

1. Установите последовательность разработки алгоритма
1. Постановка задачи - А
2. Разработка блок-схемы - Б
3. Реализация на языке программирования - В
4. Оптимизация - Г
2. Соотнесите структуры данных с их характеристиками
1. Массив - А. Быстрый доступ по индексу
2. Связный список - В. Динамическое изменение размера
3. Хеш-таблица - Б. Эффективный поиск по ключу
3. Какой алгоритм сортировки самый быстрый в среднем случае
- А) Пузырьковая
- Б) Быстрая
- В) Выбором
4. Укажите порядок работы с динамической памятью
1. Выделение памяти - А
2. Использование памяти - Б
3. Освобождение памяти - В
5. Соотнесите парадигмы программирования с их описанием
1. ООП - В. Работа с объектами
2. Функциональное - А. Работа с функциями без состояния

3.Процедурное - Б. Последовательность команд

6. Прочитайте задание и запишите развернутый, обоснованный ответ
Назовите два преимущества ООП

7. Прочитайте задание и запишите развернутый, обоснованный ответ
Какие принципы важны при разработке алгоритмов

8. Прочитайте задание и запишите развернутый, обоснованный ответ
Почему важно освобождать динамическую память

9. Прочитайте задание и запишите развернутый, обоснованный ответ
Какие преимущества дают рекурсивные алгоритмы

10. Прочитайте задание и запишите развернутый, обоснованный ответ
Как документация помогает в разработке алгоритмов

11. Какой метод сортировки эффективен для больших данных
А) Пузырьковая

Б) Слиянием

В) Вставками

12. Какие структуры данных эффективны для поиска
А) Хеш-таблица

Б) Стек

В) Бинарное дерево поиска

13. Почему важно учитывать сложность алгоритмов
А) Для увеличения времени выполнения

Б) Для оптимизации ресурсов

В) Чтобы усложнить код

14. Какие методы улучшают производительность кода
А) Кэширование

Б) Глобальные переменные

В) Оптимизация циклов

15. Какой подход к разработке алгоритмов наиболее эффективен
А) Сначала писать код, потом думать

Б) Сначала проектировать, потом реализовывать

В) Копировать чужой код без понимания

Раздел 4. Разработка приложений с графическим интерфейсом

Форма контроля/оценочное средство: Задача

Вопросы/Задания:

1. Установите последовательность разработки GUI-приложения

1.Проектирование интерфейса - А

2.Реализация логики - Б

3.Тестирование - В

4.Развёртывание - Г

2. Соотнесите элементы GUI с их назначением

1. Кнопка - В. Выполнение действия

2. Текстовое поле - А. Ввод текста

3. Чекбокс - Б. Выбор опции

3. Какой инструмент используется для создания GUI на Python

А) Tkinter

Б) Pandas

В) NumPy

4. Укажите порядок обработки событий в GUI

1. Возникновение события - А

2. Перехват события - Б

3. Обработка события - В

5. Соотнесите паттерны проектирования с их применением

1. MVC - А. Разделение логики и представления

2. Singleton - В. Гарантия единственного экземпляра

3. Observer - Б. Уведомление об изменениях

6. Прочитайте задание и запишите развернутый, обоснованный ответ
Назовите два преимущества использования MVC

7. Прочитайте задание и запишите развернутый, обоснованный ответ
Какие принципы важны при разработке GUI

8. Прочитайте задание и запишите развернутый, обоснованный ответ
Почему важно обрабатывать исключения в GUI

9. Прочитайте задание и запишите развернутый, обоснованный ответ
Какие преимущества даёт использование шаблонов проектирования

10. Прочитайте задание и запишите развернутый, обоснованный ответ
Как документация помогает в адаптации ПО

11. Какой подход к разработке GUI наиболее эффективен
А) Сначала дизайн, потом логика

Б) Смешивать логику и интерфейс

В) Писать код без планирования

12. Какие методы улучшают пользовательский опыт

А) Интуитивный интерфейс

Б) Сложные меню

В) Быстрая реакция на действия

13. Почему важно тестировать GUI на разных устройствах

А) Чтобы увеличить количество багов

Б) Для обеспечения кросс-платформенности

В) Чтобы усложнить процесс разработки

14. Какие технологии упрощают адаптацию ПО

А) Контейнеризация

Б) Жёсткая привязка к ОС

В) Использование стандартных библиотек

15. Какой метод наиболее эффективен для поддержки разных версий ПО

А) Полный отказ от обновлений

Б) Система ветвления

В) Переписывание кода для каждой версии

Раздел 5. Разработка собственных классов и обработка исключительных ситуаций

Форма контроля/оценочное средство: Задача

Вопросы/Задания:

1. Установите последовательность действий при поиске информации

1.Формулировка запроса - А

2.Выбор источника - Б

3.Анализ информации - В

4.Проверка достоверности - Г

2. Соотнесите угрозы безопасности с методами защиты

1.SQL-инъекция - А. Валидация входных данных

2.XSS - В. Экранирование HTML

3.Фишинг - Б. Использование HTTPS

3. Какой метод защиты предотвращает перехват данных

А) Шифрование

Б) Увеличение шрифта

В) Удаление комментариев

4. Укажите порядок обработки персональных данных

1.Сбор - А

2.Хранение - Б

3.Уничтожение - В

5. Соотнесите лицензии ПО с их характеристиками

1.GPL - Б. Обязательное открытие исходного кода

2.MIT - А. Свободное использование с указанием автора

3.Проприетарная - В. Ограниченное использование

6. Прочитайте задание и запишите развернутый, обоснованный ответ

Назовите два принципа защиты информации

7. Прочитайте задание и запишите развернутый, обоснованный ответ

Какие методы позволяют проверить достоверность источника

8. Прочитайте задание и запишите развернутый, обоснованный ответ

Почему важно использовать лицензионное ПО

9. Прочитайте задание и запишите развернутый, обоснованный ответ

Какие риски возникают при использовании публичных Wi-Fi

10. Прочитайте задание и запишите развернутый, обоснованный ответ

Как организовать безопасное хранение паролей

11. Какой метод аутентификации наиболее безопасен

А) Пароль "12345"

Б) Биометрия

В) Один пароль для всех сервисов

12. Какие меры повышают безопасность приложений

А) Регулярные обновления

Б) Хранение логов в открытом доступе

В) Шифрование данных

13. Почему важно вести документацию по безопасности

А) Для усложнения работы

Б) Для быстрого реагирования на инциденты

В) Чтобы увеличить объём отчетности

14. Какие технологии защищают от DDoS-атак

А) CDN

Б) Отключение сервера

В) Межсетевые экраны

15. Какой стандарт регулирует защиту персональных данных

А) GDPR

Б) ISO 9001

В) HTML5

Раздел 6. Курсовая работа

Форма контроля/оценочное средство:

Вопросы/Задания:

.

Раздел 7. Промежуточная аттестация

Форма контроля/оценочное средство:

Вопросы/Задания:

.

7. Оценочные материалы промежуточной аттестации

Очная форма обучения, Первый семестр, Зачет

Вопросы/Задания:

1. Этапы процесса создания программ на C++ (схема получения исполнимого модуля программы в интегрированной среде программирования)

2. Синтаксическая структура программы на языке СИ/СИ++: типовые части исходного модуля

3. Определение главной функции

4. Рекомендации по написанию синтаксически правильной программы

5. Алфавит языка C++

6. Особенности языка C++

7. Типы данных в языке C++

8. Понятие операции, выражения, оператора в C++

9. Основные арифметические операции. Примеры

10. Операция присваивания. Виды операции. Примеры

11. Операции инкремента и декремента. Примеры

12. Побитовые или булевские операции

13. Объявление и инициализация переменных

14. Константы. Способы объявления констант в C++

15. Реализация ввода-вывода в C

16. Реализация ввода-вывода в C++

17. Битовые поля

18. Условие в программировании. Простые и составные условия

19. Операции отношения (сравнения), логические операции

20. Условный оператор

21. Условная операция ? : в C++

22. Оператор множественного выбора switch() в C++

23. Особенности использования оператора break
24. Структура как тип данных. Синтаксис описания типа. Объявление структурных переменных
25. Доступ к полям структуры. Инициализация полей структуры
26. Массив структур
27. Вложенные структуры
28. Перечисление. Синтаксис описания типа. Объявление переменных
29. Объединение. Синтаксис описания типа. Объявление переменных
30. Основные понятия функций
31. Особенности функций
32. Определение функций
33. Объявление (прототипы) функций
34. Вызов функции
35. Технология вставки функции пользователя в одном файле с основной программой
36. Технология вставки функции пользователя в отдельный файл
37. Библиотечные функции
38. Использование формальных и фактических параметров (аргументов) при обмене информацией с функцией
39. Передача констант и значений переменных в функцию
40. Передача аргументов по умолчанию (default argument) в функцию
41. Использование массивов в качестве параметров (аргументов) функции
42. Структурные переменные в качестве аргументов функций
43. Рекурсивные функции. Примеры
44. Перегруженная функция
45. Шаблоны функций

46. Встраиваемые функции
47. Основные понятия функций
48. Особенности функций
49. Определение функций
50. Объявление (прототипы) функций
51. Вызов функции
52. Технология вставки функции пользователя в одном файле с основной программой
53. Технология вставки функции пользователя в отдельный файл
54. Библиотечные функции
55. Использование формальных и фактических параметров (аргументов) при обмене информацией с функцией
56. Передача констант и значений переменных в функцию.
57. Передача аргументов по умолчанию (default argument) в функцию
58. Использование массивов в качестве параметров (аргументов) функции
59. Структурные переменные в качестве аргументов функций
60. Рекурсивные функции. Примеры
61. Перегруженная функция
62. Шаблоны функций
63. Встраиваемые функции

Очная форма обучения, Второй семестр, Курсовая работа

Контролируемые ИДК: ОПК-2.1 ОПК-3.1 ОПК-7.1 ОПК-2.2 ОПК-3.2 ОПК-7.2 ОПК-2.3 ОПК-3.3 ОПК-7.3 ПК-П2.1 ПК-П2.2 ПК-П2.3

Вопросы/Задания:

1. Программирование приложения «Решение систем линейных алгебраических уравнений методом Гаусса»

2. Программирование приложения численного решения нелинейных и трансцендентных уравнений

3. Программирование приложения численного решения дифференциальных уравнений
4. Программирование приложения нахождения уравнения регрессии
5. Программирование приложения численного решения оптимизационной задачи (нахождения экстремума функции)
6. Программирование приложения решения оптимизационной задачи линейного программирования (Симплексный метод)
7. Программирование графического приложения «Маятник». Получить изображение движущегося математического маятника. Длина маятника и начальное положение задаются. Учесть сопротивление воздуха
8. Программирование игрового приложения «Усложненные крестики-нолики». Реализовать процесс игры с компьютером. Предусмотреть возможность различных настроек: размер игрового поля, право первого хода и т.п
9. Программирование игрового приложения "Пятнашки"
10. Программирование игрового приложения «Точки». Реализовать процесс игры с компьютером. Предусмотреть возможность различных настроек: размер игрового поля, количество игроков, право первого хода и т.п.
11. Программирование игрового приложения «Парные картинки»
12. Программирование приложения «Телефонный справочник»
13. Программирование приложения для демонстрации операций над множествами (множества задаются графически или в виде набора элементов)
14. Программирование приложения "Магический квадрат Пифагора"
15. Программирование приложения «Графическая модель движения тела в пространстве под силой тяжести»
16. Программирование приложения "Табулирование функции"
17. Программирование приложения ведения базы данных «Учет успеваемости студентов»
18. Программирование приложения ведения базы данных «Учет телефонных переговоров»
19. Программирование приложения ведения базы данных «Учет работы автовокзала»
20. Программирование приложения ведения базы данных «Учет сотрудников факультета»

21. Программирование приложения ведения базы данных «Учет ремонта станков»
22. Программирование приложения ведения базы данных «Журнал академической группы»
23. Программирование приложения ведения базы данных «Учет работы нотариальной конторы»
24. Программирование приложения ведения базы данных «Страховая компания»
25. Программирование приложения ведения базы данных «Учет деятельности классного руководителя»
26. Программирование приложения ведения базы данных «Учет деятельности строительной фирмы»
27. Программирование приложения ведения базы данных «Учет проведения спортивных соревнований» (командных, индивидуальных) по определенному виду спорта
28. Программирование приложения ведения базы данных «Учет оборудования в компьютерных аудиториях»
29. Программирование приложения ведения базы данных «Учет работы аптеки»
30. Программирование приложения ведения базы данных «Учет учета работы менеджера турфирмы»
31. Программирование приложения ведения базы данных «Учет распределения дополнительных обязанностей сотрудников»

Очная форма обучения, Второй семестр, Экзамен

Контролируемые ИДК: ОПК-2.1 ОПК-3.1 ОПК-7.1 ОПК-2.2 ОПК-3.2 ОПК-7.2 ОПК-2.3 ОПК-3.3 ОПК-7.3 ПК-П2.1 ПК-П2.2 ПК-П2.3

Вопросы/Задания:

1. Указатели на массивы и их использование
2. Указатели на функции и их использование
3. Указатель на строку, объявленную как массив символов
4. Указатели (массивы указателей) на строковые константы
5. Определение одномерного массива строк через массивы указателей на строки
6. Указатели на строки как аргументы функций

7. Функции модуля `stdlib.h`, используемых для преобразования строковых данных в числовые

8. Функции модуля `stdlib.h`, используемых для преобразования числовых данных в строковые

9. Функции модуля `string.h`

10. Массивы указателей на строки

11. Указатели на структуры и их использование

12. Указатель на `void` и его использование

13. Технология использования указателей на указатели

14. Свободная память. Операции управления `new` и `delete`

15. Оператор `new` и `delete` при работе с простыми типами

16. Оператор `new` и `delete` при работе с одномерными динамическими массивами

17. Многомерные динамические массивы. Алгоритм выделения, алгоритм освобождения памяти

18. Понятие файлов в C++. Текстовый формат записи в файл

19. Понятие файлов в C++. Двоичный формат записи в файл. Особенности сохранения чисел в двоичном формате

20. Запись информации в текстовый файл

21. Чтение информации из текстового файла

22. Режимы файлов

23. Запись данных в двоичный файл

24. Чтение данных из двоичного файла

25. Дополнительные функции файлового ввода / вывода: функция `rdbuf()`

26. Дополнительные функции файлового ввода / вывода: функции `write`(символьный массив, число символов), `read`(символьный массив, число символов), `gcount()`, `put`(параметр типа `char`)

27. Дополнительные функции файлового ввода / вывода: функция `get()`. Варианты функции `get()`

28. Произвольный доступ к файлам: функции реализации «передвижения» по файлу в определенную точку файла. Пример реализации

29. Произвольный доступ к файлам: функции для проверки текущей позиции файлового указателя

30. Работа с файлом, как с базой данных. Пример реализации

31. Копирование файлов. Пример реализации

32. Этапы программирования в объектно-ориентированном стиле

33. Указатель this. Примеры

34. Дружественные функции. Правила описания и особенности дружественных функций. Примеры

35. Дружественный класс. Примеры

36. Перегрузка операций

37. Наследование классов. Объявление производного класса. Примеры

38. Множественное наследование. Дублирование классов. Примеры

39. Виртуальные классы. Примеры

40. Полиморфизм. Переопределение функций. Примеры

41. Виртуальные функции. Примеры

42. Абстрактные классы

43. Локальные классы

44. Шаблоны классов. Примеры

45. Понятие исключительной ситуации. Реализации механизма обработки исключений в языке C++. Примеры

46. Форматы обработчиков исключений

47. Форматы генерации исключений

48. Основные понятия объектно-ориентированного программирования: объект, класс, поле

49. Основные понятия объектно-ориентированного программирования: метод, свойства, сообщение, событие и т. д

- 50. Понятие инкапсуляции в объектно-ориентированного программировании
- 51. Понятие наследования в объектно-ориентированного программировании
- 52. Понятие полиморфизма в объектно-ориентированного программировании
- 53. Этапы программирования в объектно-ориентированном стиле
- 54. Визуальное программирование. Основные понятия
- 55. Использование диалогов для выбора файлов в графическом приложении. Примеры.
- 56. Работа с файлами с использованием функций компонент в графическом приложении. Примеры
- 57. .Работа с датами в графическом приложении
- 58. Построение графиков с помощью компонента TChart. Примеры
- 59. Построение графиков с помощью свойства объекта TForm – Canvas. Примеры
- 60. Работа с базами данных в интегрированной среде программирования. Механизмы доступа
- 61. Работа с базами данных в интегрированной среде программирования. Подключение базы данных. Доступ к объектам базы данных
- 62. Описание класса в C++. Синтаксис, примеры
- 63. Создание и использование объектов в C++
- 64. Размещение классов в различных файлах. Примеры
- 65. Понятие конструктора. Основные свойства конструкторов
- 66. Синтаксис описание конструктора. Примеры
- 67. Конструктор копирования. Примеры
- 68. Деструкторы. Синтаксис определения деструктора в описании класса. Примеры
- 69. Особенности деструкторов

70. Указатель this. Примеры

71. Дружественные функции. Правила описания и особенности дружественных функций. Примеры

72. Дружественный класс. Примеры

73. Перегрузка операций

74. Наследование классов. Объявление производного класса. Примеры

75. Множественное наследование. Дублирование классов. Примеры

76. Виртуальные классы. Примеры

77. Полиморфизм. Переопределение функций. Примеры

78. Виртуальные функции. Примеры

79. Абстрактные классы

80. Локальные классы

81. Шаблоны классов. Примеры

82. Понятие исключительной ситуации. Реализации механизма обработки исключений в языке C++. Примеры

83. Форматы обработчиков исключений

84. Форматы генерации исключений

Заочная форма обучения, Первый семестр, Зачет

Контролируемые ИДК: ОПК-2.1 ОПК-3.1 ОПК-7.1 ОПК-2.2 ОПК-3.2 ОПК-7.2 ОПК-2.3 ОПК-3.3 ОПК-7.3 ПК-П2.1 ПК-П2.2 ПК-П2.3

Вопросы/Задания:

1. Этапы процесса создания программ на C++ (схема получения исполнимого модуля программы в интегрированной среде программирования)

2. Синтаксическая структура программы на языке СИ/СИ++: типовые части исходного модуля

3. Определение главной функции

4. Рекомендации по написанию синтаксически правильной программы

5. Алфавит языка C++

6. Особенности языка C++
7. Типы данных в языке C++
8. Понятие операции, выражения, оператора в C++
9. Основные арифметические операции. Примеры
10. Операция присваивания. Виды операции. Примеры
11. Операции инкремента и декремента. Примеры
12. Побитовые или булевские операции
13. Объявление и инициализация переменных
14. Константы. Способы объявления констант в C++
15. Реализация ввода-вывода в C
16. Реализация ввода-вывода в C++
17. Битовые поля
18. Условие в программировании. Простые и составные условия
19. Операции отношения (сравнения), логические операции
20. Условный оператор
21. Условная операция ? : в C++
22. Оператор множественного выбора switch() в C++
23. Особенности использования оператора break
24. Структура как тип данных. Синтаксис описания типа. Объявление структурных переменных
25. Доступ к полям структуры. Инициализация полей структуры
26. Массив структур
27. Вложенные структуры
28. Перечисление. Синтаксис описания типа. Объявление переменных
29. Объединение. Синтаксис описания типа. Объявление переменных

30. Основные понятия функций
31. Особенности функций
32. Определение функций
33. Объявление (прототипы) функций
34. Вызов функции
35. Технология вставки функции пользователя в одном файле с основной программой
36. Технология вставки функции пользователя в отдельный файл
37. Библиотечные функции
38. Использование формальных и фактических параметров (аргументов) при обмене информацией с функцией
39. Передача констант и значений переменных в функцию
40. Передача аргументов по умолчанию (default argument) в функцию
41. Использование массивов в качестве параметров (аргументов) функции
42. Структурные переменные в качестве аргументов функций
43. Рекурсивные функции. Примеры
44. Перегруженная функция
45. Шаблоны функций
46. Встраиваемые функции
47. Основные понятия функций
48. Особенности функций
49. Определение функций
50. Объявление (прототипы) функций
51. Вызов функции
52. Технология вставки функции пользователя в одном файле с основной программой

- 53. Технология вставки функции пользователя в отдельный файл
- 54. Библиотечные функции
- 55. Использование формальных и фактических параметров (аргументов) при обмене информацией с функцией
- 56. Передача констант и значений переменных в функцию.
- 57. Передача аргументов по умолчанию (default argument) в функцию
- 58. Использование массивов в качестве параметров (аргументов) функции
- 59. Структурные переменные в качестве аргументов функций
- 60. Рекурсивные функции. Примеры
- 61. Перегруженная функция
- 62. Шаблоны функций
- 63. Встраиваемые функции

Заочная форма обучения, Второй семестр, Курсовая работа

Контролируемые ИДК: ОПК-2.1 ОПК-3.1 ОПК-7.1 ОПК-2.2 ОПК-3.2 ОПК-7.2 ОПК-2.3 ОПК-3.3 ОПК-7.3 ПК-П2.1 ПК-П2.2 ПК-П2.3

Вопросы/Задания:

1. Программирование приложения «Решение систем линейных алгебраических уравнений методом Гаусса»
2. Программирование приложения численного решения нелинейных и трансцендентных уравнений
3. Программирование приложения численного решения дифференциальных уравнений
4. Программирование приложения нахождения уравнения регрессии
5. Программирование приложения численного решения оптимизационной задачи (нахождения экстремума функции)
6. Программирование приложения решения оптимизационной задачи линейного программирования (Симплексный метод)
7. Программирование графического приложения «Маятник». Получить изображение движущегося математического маятника. Длина маятника и начальное положение задаются. Учесть сопротивление воздуха

8. Программирование игрового приложения «Усложненные крестики-нолики». Реализовать процесс игры с компьютером. Предусмотреть возможность различных настроек: размер игрового поля, право первого хода и т.п

9. Программирование игрового приложения "Пятнашки"

10. Программирование игрового приложения «Точки». Реализовать процесс игры с компьютером. Предусмотреть возможность различных настроек: размер игрового поля, количество игроков, право первого хода и т.п.

11. Программирование игрового приложения «Парные картинки»

12. Программирование приложения «Телефонный справочник»

13. Программирование приложения для демонстрации операций над множествами (множества задаются графически или в виде набора элементов)

14. Программирование приложения "Магический квадрат Пифагора"

15. Программирование приложения «Графическая модель движения тела в пространстве под силой тяжести»

16. Программирование приложения "Табулирование функции"

17. Программирование приложения ведения базы данных «Учет успеваемости студентов»

18. Программирование приложения ведения базы данных «Учет телефонных переговоров»

19. Программирование приложения ведения базы данных «Учет работы автовокзала»

20. Программирование приложения ведения базы данных «Учет сотрудников факультета»

21. Программирование приложения ведения базы данных «Учет ремонта станков»

22. Программирование приложения ведения базы данных «Журнал академической группы»

23. Программирование приложения ведения базы данных «Учет работы нотариальной конторы»

24. Программирование приложения ведения базы данных «Страховая компания»

25. Программирование приложения ведения базы данных «Учет деятельности классного руководителя»

26. Программирование приложения ведения базы данных «Учет деятельности строительной фирмы»

27. Программирование приложения ведения базы данных «Учет проведения спортивных соревнований» (командных, индивидуальных) по определенному виду спорта

28. Программирование приложения ведения базы данных «Учет оборудования в компьютерных аудиториях»

29. Программирование приложения ведения базы данных «Учет работы аптеки»

30. Программирование приложения ведения базы данных «Учет учета работы менеджера турфирмы»

31. Программирование приложения ведения базы данных «Учет распределения дополнительных обязанностей сотрудников»

Заочная форма обучения, Второй семестр, Экзамен

Контролируемые ИДК: ОПК-2.1 ОПК-3.1 ОПК-7.1 ОПК-2.2 ОПК-3.2 ОПК-7.2 ОПК-2.3 ОПК-3.3 ОПК-7.3 ПК-П2.1 ПК-П2.2 ПК-П2.3

Вопросы/Задания:

1. Указатели на массивы и их использование
2. Указатели на функции и их использование
3. Указатель на строку, объявленную как массив символов
4. Указатели (массивы указателей) на строковые константы
5. Определение одномерного массива строк через массивы указателей на строки
6. Указатели на строки как аргументы функций
7. Функции модуля `stdlib.h`, используемых для преобразования строковых данных в числовые
8. Функции модуля `stdlib.h`, используемых для преобразования числовых данных в строковые
9. Функции модуля `string.h`
10. Массивы указателей на строки
11. Указатели на структуры и их использование
12. Указатель на `void` и его использование

13. Технология использования указателей на указатели
14. Свободная память. Операции управления new и delete
15. Оператор new и delete при работе с простыми типами
16. Оператор new и delete при работе с одномерными динамическими массивами
17. Многомерные динамические массивы. Алгоритм выделения, алгоритм освобождения памяти
18. Понятие файлов в C++. Текстовый формат записи в файл
19. Понятие файлов в C++. Двоичный формат записи в файл. Особенности сохранения чисел в двоичном формате
20. Запись информации в текстовый файл
21. Чтение информации из текстового файла
22. Режимы файлов
23. Запись данных в двоичный файл
24. Чтение данных из двоичного файла
25. Дополнительные функции файлового ввода / вывода: функция `rdbuf()`
26. Дополнительные функции файлового ввода / вывода: функции `write(символьный массив, число символов)`, `read(символьный массив, число символов)`, `gcount()`, `put(параметр типа char)`
27. Дополнительные функции файлового ввода / вывода: функция `get()`. Варианты функции `get()`
28. Произвольный доступ к файлам: функции реализации «перемещения» по файлу в определенную точку файла. Пример реализации
29. Произвольный доступ к файлам: функции для проверки текущей позиции файлового указателя
30. Работа с файлом, как с базой данных. Пример реализации
31. Копирование файлов. Пример реализации
32. Этапы программирования в объектно-ориентированном стиле
33. Указатель `this`. Примеры

34. Дружественные функции. Правила описания и особенности дружественных функций. Примеры

35. Дружественный класс. Примеры

36. Перегрузка операций

37. Наследование классов. Объявление производного класса. Примеры

38. Множественное наследование. Дублирование классов. Примеры

39. Виртуальные классы. Примеры

40. Полиморфизм. Переопределение функций. Примеры

41. Виртуальные функции. Примеры

42. Абстрактные классы

43. Локальные классы

44. Шаблоны классов. Примеры

45. Понятие исключительной ситуации. Реализации механизма обработки исключений в языке C++. Примеры

46. Форматы обработчиков исключений

47. Форматы генерации исключений

48. Основные понятия объектно-ориентированного программирования: объект, класс, поле

49. Основные понятия объектно-ориентированного программирования: метод, свойства, сообщение, событие и т. д

50. Понятие инкапсуляции в объектно-ориентированного программировании

51. Понятие наследования в объектно-ориентированного программировании

52. Понятие полиморфизма в объектно-ориентированного программировании

53. Этапы программирования в объектно-ориентированном стиле

54. Визуальное программирование. Основные понятия

55. Использование диалогов для выбора файлов в графическом приложении. Примеры.

56. Работа с файлами с использованием функций компонент в графическом приложении. Примеры

57. Работа с датами в графическом приложении

58. Построение графиков с помощью компонента TChart. Примеры

59. Построение графиков с помощью свойства объекта TForm – Canvas. Примеры

60. Работа с базами данных в интегрированной среде программирования. Механизмы доступа

61. Работа с базами данных в интегрированной среде программирования. Подключение базы данных. Доступ к объектам базы данных

62. Описание класса в C++. Синтаксис, примеры

63. Создание и использование объектов в C++

64. Размещение классов в различных файлах. Примеры

65. Понятие конструктора. Основные свойства конструкторов

66. Синтаксис описание конструктора. Примеры

67. Конструктор копирования. Примеры

68. Деструкторы. Синтаксис определения деструктора в описании класса. Примеры

69. Особенности деструкторов

70. Указатель this. Примеры

71. Дружественные функции. Правила описания и особенности дружественных функций. Примеры

72. Дружественный класс. Примеры

73. Перегрузка операций

74. Наследование классов. Объявление производного класса. Примеры

75. Множественное наследование. Дублирование классов. Примеры

76. Виртуальные классы. Примеры

77. Полиморфизм. Переопределение функций. Примеры

78. Виртуальные функции. Примеры

79. Абстрактные классы

80. Локальные классы

81. Шаблоны классов. Примеры

82. Понятие исключительной ситуации. Реализации механизма обработки исключений в языке C++. Примеры

83. Форматы обработчиков исключений

84. Форматы генерации исключений

8. Материально-техническое и учебно-методическое обеспечение дисциплины

8.1. Перечень основной и дополнительной учебной литературы

Основная литература

1. Назаренко,, П. А. Алгоритмы и структуры данных: учебное пособие / П. А. Назаренко,. - Алгоритмы и структуры данных - Самара: Поволжский государственный университет телекоммуникаций и информатики, 2015. - 130 с. - 2227-8397. - Текст: электронный // IPR SMART: [сайт]. - URL: <https://www.iprbookshop.ru/71819.html> (дата обращения: 08.10.2025). - Режим доступа: по подписке

2. Букунов,, С. В. Основы программирования на языке C++: учебное пособие / С. В. Букунов,. - Основы программирования на языке C++ - Санкт-Петербург: Санкт-Петербургский государственный архитектурно-строительный университет, ЭБС АСВ, 2015. - 201 с. - 978-5-9227-0619-3. - Текст: электронный // IPR SMART: [сайт]. - URL: <https://www.iprbookshop.ru/63631.html> (дата обращения: 08.10.2025). - Режим доступа: по подписке

3. ПРОГРАММИРОВАНИЕ на языке Си ++: учеб. пособие / Краснодар: КубГАУ, 2016. - 185 с. - 978-5-00097-176-5. - Текст: непосредственный.

Дополнительная литература

1. ИВАНОВА Е. А. Алгоритмизация и программирование: метод. рекомендации / ИВАНОВА Е. А., Крамаренко Т. А.. - Краснодар: КубГАУ, 2020. - 32 с. - Текст: электронный. // : [сайт]. - URL: <https://edu.kubsau.ru/mod/resource/view.php?id=9097> (дата обращения: 12.05.2026). - Режим доступа: по подписке

2. КРЕЙМЕР А. С. Веб-дизайн и интернет-программирование: метод. рекомендации / КРЕЙМЕР А. С., Крепышев Д. А.. - Краснодар: КубГАУ, 2020. - 35 с. - Текст: электронный. // : [сайт]. - URL: <https://edu.kubsau.ru/mod/resource/view.php?id=9247> (дата обращения: 12.05.2026). - Режим доступа: по подписке

3. КРАМАРЕНКО Т. А. Алгоритмизация и программирование: практикум / КРАМАРЕНКО Т. А., Иванова Е. А.. - Краснодар: КубГАУ, 2020. - 178 с. - Текст: электронный. // : [сайт]. - URL: <https://edu.kubsau.ru/mod/resource/view.php?id=9066> (дата обращения: 12.05.2026). - Режим доступа: по подписке

4. Павловская, Т. А. Программирование на языке высокого уровня С#: учебное пособие / Т. А. Павловская,. - Программирование на языке высокого уровня С# - Москва: Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2025. - 245 с. - 978-5-4497-0862-5. - Текст: электронный // IPR SMART: [сайт]. - URL: <https://www.iprbookshop.ru/146386.html> (дата обращения: 08.10.2025). - Режим доступа: по подписке

5. ИВАНОВА Е. А. Алгоритмизация и программирование: метод. рекомендации / ИВАНОВА Е. А., Ефанова Н. В., Крамаренко Т. А.. - Краснодар: КубГАУ, 2020. - 67 с. - Текст: электронный. // : [сайт]. - URL: <https://edu.kubsau.ru/mod/resource/view.php?id=9387> (дата обращения: 12.05.2026). - 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://learn.microsoft.com/ru-ru/> - Материалы портала для разработчиков Microsoft
2. <http://www.iprbookshop.ru/> - IPRbook
3. <https://edu.kubsau.ru/> - Образовательный портал КубГАУ
4. <https://znanium.com/> - Znanium.com

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)

1. Индиго;

*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.

Лекционный зал

310эк

- 0 шт.

Компьютерный класс

401эк

Персональный компьютер IRU i5/16Gb/512GbSSD/23.8 - 1 шт.

404эк

Персональный компьютер UNIVERSALD1 i5/16Gb/512GbSSD/23.8 - 1 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.

Практические занятия

Форма организации обучения, проводимая под руководством преподавателя и служащая для детализации, анализа, расширения, углубления, закрепления, применения (или выполнения) разнообразных практических работ, упражнений) и контроля усвоения полученной на лекциях учебной информации. Практические занятия проводятся с использованием учебно-методических изданий, 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, прочитать 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», «прожектор» 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 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 плоскочечную 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)