

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)
«ОБЪЕКТНО-ОРИЕНТИРОВАННОЕ ПРОГРАММИРОВАНИЕ»**

Уровень высшего образования: бакалавриат

Направление подготовки: 38.03.05 Бизнес-информатика

Направленность (профиль) подготовки: Управление цифровой трансформацией бизнеса

Квалификация (степень) выпускника: бакалавр

Формы обучения: очная, очно-заочная

Год набора (приема на обучение): 2026

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

Объем: в зачетных единицах: 5 з.е.
в академических часах: 180 ак.ч.

Разработчики:

Иванова Е.А.

Рабочая программа дисциплины (модуля) составлена в соответствии с требованиями ФГОС ВО по направлению подготовки 38.03.05 Бизнес-информатика, утвержденного приказом Минобрнауки 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	13.04.2026, № 8

1. Цель и задачи освоения дисциплины (модуля)

Цель освоения дисциплины - формирование комплекса знаний об организационных, научных и методических основах анализа и проектирования приложений с использованием объектно-ориентированного подхода

Задачи изучения дисциплины:

- обучение теоретическим и практическим основам знаний в области технологии объектно-ориентированного программирования, включая методы программирования, стандарты и инструментальные средства программирования;
- формирование у обучающихся практических навыков технологии объектно-ориентированного программирования, работы на персональном компьютере с целью составления моделей для решения прикладных экономических задач, предусмотренных для освоения на лабораторных занятиях.

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) «Объектно-ориентированное программирование» относится к обязательной части образовательной программы и изучается в семестре(ах): Очная форма обучения - 3, 4, Очно-заочная форма обучения - 4, 5.

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)
Третий семестр	72	2	61	1		30	30		11	Зачет

Четвертый семестр	108	3	56	6		26	18	6	25	Курсовой проект Экзамен (27)
Всего	180	5	117	7		56	48	6	36	27

Очно-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)
Четвертый семестр	72	2	15	1		8	6		57	Зачет
Пятый семестр	108	3	28	6		6	10	6	80	Курсовой проект Экзамен
Всего	180	5	43	7		14	16	6	137	

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 с результатам освоения программы
Раздел 1. Разработка алгоритмов и программных средств с использованием объектно-ориентированного подхода	146		56	48	6	36	ОПК-3.1 ОПК-3.2 ОПК-3.3 ОПК-3.4
Тема 1.1. Основы объектно-ориентированного программирования	3			2		1	
Тема 1.2. Перегрузка функций, шаблоны функций	9		4	4		1	
Тема 1.3. Класс как абстрактный тип данных	12		4	6		2	

Тема 1.4. Перегрузка стандартных операций	9		4	4		1	
Тема 1.5. Наследование классов и виртуальные функции	10		4	4		2	
Тема 1.6. Шаблоны классов	7		4	2		1	
Тема 1.7. Обработка исключительных ситуаций	9		4	4		1	
Тема 1.8. Работа со стандартной библиотекой шаблонов STL	12		6	4		2	
Тема 1.9. Объектно-ориентированное проектирование с помощью UML	12			2	6	4	
Тема 1.10. Дополнительные средства объектно-ориентированных языков программирования	11		4	4		3	
Тема 1.11. Библиотека классов .NET	5			2		3	
Тема 1.12. Работа с коллекциями	9		4	2		3	
Тема 1.13. Ввод, вывод и сериализация объектов	11		6	2		3	
Тема 1.14. Разработка Windows приложений	9		4	2		3	
Тема 1.15. Классы для работы с потоками	9		4	2		3	
Тема 1.16. Разработка сетевых приложений	9		4	2		3	
Раздел 2. Промежуточная аттестация	4	4					ОПК-3.1 ОПК-3.2
Тема 2.1. Зачет	1	1					ОПК-3.3
Тема 2.2. Экзамен	3	3					ОПК-3.4
Раздел 3. Курсовой проект	3	3					ОПК-3.1 ОПК-3.2
Тема 3.1. Защита курсового проекта	3	3					ОПК-3.3 ОПК-3.4
Итого	153	7	56	48	6	36	

Очно-заочная форма обучения

Наименование раздела, темы	Всего	Внеаудиторная контактная работа	Лабораторные занятия	Лекционные занятия	Практические занятия	Самостоятельная работа	Планируемые результаты обучения, соответствующие с результатам освоения программы
----------------------------	-------	---------------------------------	----------------------	--------------------	----------------------	------------------------	---

Раздел 1. Разработка алгоритмов и программных средств с использованием объектно-ориентированного подхода	180	7	14	16	6	137	ОПК-3.1 ОПК-3.2 ОПК-3.3 ОПК-3.4
Тема 1.1. Основы объектно-ориентированного программирования	17	3	2			12	
Тема 1.2. Перегрузка функций, шаблоны функций	21	1		8		12	
Тема 1.3. Класс как абстрактный тип данных	12					12	
Тема 1.4. Перегрузка стандартных операций	15				3	12	
Тема 1.5. Наследование классов и виртуальные функции	22			5		17	
Тема 1.6. Шаблоны классов	20	3	5			12	
Тема 1.7. Обработка исключительных ситуаций	12					12	
Тема 1.8. Работа со стандартной библиотекой шаблонов STL	17		2		3	12	
Тема 1.9. Объектно-ориентированное проектирование с помощью UML	15			3		12	
Тема 1.10. Дополнительные средства объектно-ориентированных языков программирования	8					8	
Тема 1.11. Библиотека классов .NET	13		5			8	
Тема 1.12. Работа с коллекциями	4					4	
Тема 1.13. Ввод, вывод и сериализация объектов							
Тема 1.14. Разработка Windows приложений	4					4	
Тема 1.15. Классы для работы с потоками							
Тема 1.16. Разработка сетевых приложений							
Раздел 2. Промежуточная аттестация							ОПК-3.1 ОПК-3.2 ОПК-3.3 ОПК-3.4
Тема 2.1. Зачет							ОПК-3.3
Тема 2.2. Экзамен							ОПК-3.4
Раздел 3. Курсовой проект							ОПК-3.1 ОПК-3.2 ОПК-3.3 ОПК-3.4
Тема 3.1. Защита курсового проекта							ОПК-3.3 ОПК-3.4
Итого	180	7	14	16	6	137	

5.2. Содержание разделов, тем дисциплин

Раздел 1. Разработка алгоритмов и программных средств с использованием объектно-ориентированного подхода

(Очно-заочная: Внеаудиторная контактная работа - 7ч.; Лабораторные занятия - 14ч.; Лекционные занятия - 16ч.; Практические занятия - 6ч.; Самостоятельная работа - 137ч.; Очная: Лабораторные занятия - 56ч.; Лекционные занятия - 48ч.; Практические занятия - 6ч.; Самостоятельная работа - 36ч.)

Тема 1.1. Основы объектно-ориентированного программирования

(Очно-заочная: Внеаудиторная контактная работа - 3ч.; Лабораторные занятия - 2ч.; Самостоятельная работа - 12ч.; Очная: Лекционные занятия - 2ч.; Самостоятельная работа - 1ч.)

Цели и задачи курса. Обзор методов и моделей объектно-ориентированного программирования. Основные принципы объектно-ориентированного программирования: классы и объекты. Понятие инкапсуляции, полиморфизма, наследования. Объектно-ориентированные языки программирования.

Тема 1.2. Перегрузка функций, шаблоны функций

(Очно-заочная: Внеаудиторная контактная работа - 1ч.; Лекционные занятия - 8ч.; Самостоятельная работа - 12ч.; Очная: Лабораторные занятия - 4ч.; Лекционные занятия - 4ч.; Самостоятельная работа - 1ч.)

Параметрический полиморфизм. Цель и способы перегрузки функций. Правила передачи параметров в перегруженные функции. Понятие и основные свойства шаблонов функций. Правила использования шаблонов и перегруженных функций.

Тема 1.3. Класс как абстрактный тип данных

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

Понятие и объявление классов. Создание объектов класса. Конструкторы, деструкторы и доступность компонентов класса. Инкапсуляция данных. Компонентные данные и компонентные функции. Статические и константные компоненты классов. Указатель this. Дружбы классов.

Тема 1.4. Перегрузка стандартных операций

(Очная: Лабораторные занятия - 4ч.; Лекционные занятия - 4ч.; Самостоятельная работа - 1ч.; Очно-заочная: Практические занятия - 3ч.; Самостоятельная работа - 12ч.)

Цель и способы перегрузки стандартных операций. Функции-операторы как глобальные, дружественные и компонентные функции классов. Перегрузка префиксных и постфиксных операций. Ввод-вывод для типов, определенных пользователем.

Тема 1.5. Наследование классов и виртуальные функции

(Очная: Лабораторные занятия - 4ч.; Лекционные занятия - 4ч.; Самостоятельная работа - 2ч.; Очно-заочная: Лекционные занятия - 5ч.; Самостоятельная работа - 17ч.)

Определение базовых и производных классов. Статусы доступа при наследовании. Создание объектов базовых и производных классов. Вызов методов базовых классов из производных. Понятие множественного наследования. Варианты его использования. Виртуальные базовые классы. Динамический полиморфизм. Понятие виртуальных функций. Чистые виртуальные функции и абстрактные классы, их использование.

Тема 1.6. Шаблоны классов

(Очно-заочная: Внеаудиторная контактная работа - 3ч.; Лабораторные занятия - 5ч.; Самостоятельная работа - 12ч.; Очная: Лабораторные занятия - 4ч.; Лекционные занятия - 2ч.; Самостоятельная работа - 1ч.)

Параметрический полиморфизм в объектно-ориентированном программировании. Понятие шаблона класса и способы его объявления. Методы создания объектов классов, порождаемых шаблонами. Формальные параметры шаблонов.

Тема 1.7. Обработка исключительных ситуаций

(Очная: Лабораторные занятия - 4ч.; Лекционные занятия - 4ч.; Самостоятельная работа - 1ч.; Очно-заочная: Самостоятельная работа - 12ч.)

Понятие исключительных ситуаций и способы их обработки. Примеры исключений. Понятие контролируемых блоков. Обработчики исключений. Методы создания классов исключений и их объектов. Способы формирования исключений в программе.

Тема 1.8. Работа со стандартной библиотекой шаблонов STL

(Очная: Лабораторные занятия - 6ч.; Лекционные занятия - 4ч.; Самостоятельная работа - 2ч.; Очно-заочная: Лабораторные занятия - 2ч.; Практические занятия - 3ч.; Самостоятельная работа - 12ч.)

Назначение стандартной библиотеки шаблонов языка C++. Структура библиотеки. Объекты типа string: конструкторы, операции, методы. Основные виды контейнеров. Понятие и типы итераторов. Понятие и вызов алгоритмов STL.

Тема 1.9. Объектно-ориентированное проектирование с помощью UML

(Очная: Лекционные занятия - 2ч.; Практические занятия - 6ч.; Самостоятельная работа - 4ч.; Очно-заочная: Лекционные занятия - 3ч.; Самостоятельная работа - 12ч.)

Назначение языка UML. Способы использования UML. Базовые типы диаграмм. Границы применения UML. Практические примеры.

Тема 1.10. Дополнительные средства объектно-ориентированных языков программирования

(Очная: Лабораторные занятия - 4ч.; Лекционные занятия - 4ч.; Самостоятельная работа - 3ч.; Очно-заочная: Самостоятельная работа - 8ч.)

Понятие и описание свойств классов. Аксессуары. Назначение и способы создания индексаторов. Описание и реализация интерфейсов. Делегаты и события.

Тема 1.11. Библиотека классов .NET

(Очно-заочная: Лабораторные занятия - 5ч.; Самостоятельная работа - 8ч.; Очная: Лекционные занятия - 2ч.; Самостоятельная работа - 3ч.)

Общие принципы функционирования приложений в среде .NETFramework. Структура библиотеки .NET. Иерархия классов .NET

Тема 1.12. Работа с коллекциями

(Очная: Лабораторные занятия - 4ч.; Лекционные занятия - 2ч.; Самостоятельная работа - 3ч.; Очно-заочная: Самостоятельная работа - 4ч.)

Интерфейсы коллекций. Классы коллекций, их свойства и методы

Тема 1.13. Ввод, вывод и сериализация объектов

(Лабораторные занятия - 6ч.; Лекционные занятия - 2ч.; Самостоятельная работа - 3ч.)

Пространство имен System.IO. Классы потоков. Текстовые файлы. Двоичные файлы. Сериализация/десериализация объектов

Тема 1.14. Разработка Windows приложений

(Очная: Лабораторные занятия - 4ч.; Лекционные занятия - 2ч.; Самостоятельная работа - 3ч.; Очно-заочная: Самостоятельная работа - 4ч.)

Общие принципы функционирования Windows-приложений. Виды Windows-приложений. Работа с формами. Передача данных между окнами приложения. Разработка MDI-приложений

Тема 1.15. Классы для работы с потоками

(Лабораторные занятия - 4ч.; Лекционные занятия - 2ч.; Самостоятельная работа - 3ч.)

Общие сведения по работе с потоками. Синхронизация потоков. Взаимодействие потоков

Тема 1.16. Разработка сетевых приложений

(Лабораторные занятия - 4ч.; Лекционные занятия - 2ч.; Самостоятельная работа - 3ч.)

Использование Internet в приложениях .NET. Компоненты пространства имен System.Net. Универсальные идентификаторы ресурсов. Классы запросов и ответов. Обработка сетевых ошибок

Раздел 2. Промежуточная аттестация

(Внеаудиторная контактная работа - 4ч.)

Тема 2.1. Зачет

(Внеаудиторная контактная работа - 1ч.)

Проведение промежуточной аттестации в форме зачета

Тема 2.2. Экзамен

(Внеаудиторная контактная работа - 3ч.)

Проведение промежуточной аттестации в форме экзамена

Раздел 3. Курсовой проект

(Внеаудиторная контактная работа - 3ч.)

Тема 3.1. Защита курсового проекта

(Внеаудиторная контактная работа - 3ч.)

Защита курсового проекта

6. Оценочные материалы текущего контроля

Раздел 1. Разработка алгоритмов и программных средств с использованием объектно-ориентированного подхода

Форма контроля/оценочное средство: Задача

Вопросы/Задания:

1. К базовым принципам ООП относятся:

- А. инкапсуляция
- Б. легкость и простота использования
- В. наследование
- Г. полиморфизм

2. Напишите верный ответ (в нижнем регистре).

Выполнение методом с одним и тем же именем различных действий в зависимости от контекста, в частности, от принадлежности тому или другому классу – это ...

3. Действия, которые можно выполнять над объектами, в объектно-ориентированном программировании называют ...

- А. методами
- Б. свойствами
- В. классами
- Г. перечислениями

4. Перегруженные функции различаются компилятором с помощью

- А. разного кода записанного в теле функций
- Б. сигнатуры функций
- В. типа возвращаемого функцией значения

Г. имени функции

5. Определена перегруженная функция

```
double prim(double x) {return x*x*x;}
```

```
double prim(double x, double y) {return x*y*y;}
```

Правильные обращения к функции prim():

А. prim(0.5, 1.8, 2.6)

Б. prim(0.5, ABC)

В. prim(0.5, 1.8)

Г. prim(2.6e-6)

6. В качестве ключа класса в языке C++ может использоваться ключевое слово

А. class

Б. struct

В. union

Г. main

7. Открытые компоненты класса описываются с помощью служебного слова ...

А. public

Б. private

В. protected

Г. opened

8. Правильное описание дружественной функции класса MyClass:

А. void friend void Func(MyClass, char)

Б. friend void Func(MyClass, char)

В. MyClass::friend void Func(MyClass, char)

Г. MyClass::void friend void Func(MyClass, char)

9. Напишите верный ответ (в нижнем регистре).

Метод класса, выполняющий инициализацию его компонентных данных - это ...

10. Правильное объявление конструктора класса Student

А. ~имя_конструктора

Б. Student (список_формальных_параметров) {тело_конструктора}

В. ~Student (список_формальных_параметров) {тело_конструктора}

Г. void имя_конструктора (Student) {тело_конструктора}

11. Правильное объявление деструктора класса Student:

А. ~Student()

Б. void ~имя_деструктора(Student)

В. int Student()

Г. ~имя_деструктора(Student) {тело_деструктора}

12. Раздел protected предназначен для

А. объявлений, которые доступны для внешнего использования

Б. объявления полей и функций, используемых только внутри данного класса и недоступных извне

В. объявления полей и функций, используемых только внутри данного класса и доступных извне

Г. объявлений, доступных только для объявляемого класса и его потомков

13. Верный формат перегрузки операции «Поместить в поток ввода»:

А. istream& operator >>(istream& in, имя_класса& имя)

Б. ostream& operator <<(ostream& out, имя_класса имя)

В. istream& operator >>(istream in, имя_класса имя)

Г. ostream& operator <<(список_параметров)

14. Напишите верный ответ (в нижнем регистре).

Процесс получения производным классом данных и методов от другого, уже определенного базового класса - это ...

15. Определение производного класса Circle от базового класса Shape:

- A. class Shape:public Circle
- Б. class Circle::public Shape
- В. class Circle:public Shape
- Г. class Shape::public Circle

16. Чистая (пустая) виртуальная функция – это

- A. компонентная функция, для которой не указана реализация
- Б. функция получения производным классом данных и методов от другого, уже определенного базового класса
- В. процесс, при котором в производный класс передаются параметры ранее определенного базового класса
- Г. функция, описанная в базовом классе, которая может по-разному выполняться в производных классах

17. Абстрактный класс может использоваться только в качестве

- A. производного класса
- Б. базового для производных классов
- В. дружественного класса
- Г. производного класса, имеющего несколько базовых классов

18. Основное свойство абстрактных классов:

- A. от него нельзя наследовать другие классы
- Б. нельзя создавать объекты данного класса
- В. он не может содержать методов
- Г. он не может содержать компонентных данных

19. Общий формат описания шаблона классов:

- A. template <список_параметров_шаблона> определение_класса
- Б. template (список_параметров_шаблона) определение_класса
- В. <список_параметров_шаблона> определение_класса
- Г. template <имена_производных_классов> определение_класса

20. Соответствие между операторами обработки исключений языка C++ и их назначением:

1. try - г) завершает работу программы
2. throw - в) содержит обработчик исключения
3. catch - б) генерирует объект-исключение
4. terminate - а) выделяет контролируемый блок в тексте программы

21. Для создания стека символов, в котором в качестве контейнера используется динамический массив, в библиотеке STL используется конструкция вида

- A. stack <char, vector<char> > st
- Б. queue <char, vector<char> > st
- В. vector <char, stack<char> > st
- Г. stack <char, vector<stack> > st

22. Вызовет ли данный код ошибку компиляции?

```
class Rectangle
{
public:
int a, b;
int sum ();
int square ();
~ Rect ();
};
```

- A. Ошибки нет, все записано верно.
- Б. Ошибка является: имя деструктора должно совпадать с именем класса. +
- В. Ошибка является: имя деструктора не может начинаться с маленькой буквы.
- Г. Ошибка является: никакой идентификатор в C ++ не может начинаться со знака «~».

23. Укажите правильное объявление виртуального метода, который принимает одно целочисленное значение и ничего не возвращает.

- A. virtual void SomeFunction (int x);
- Б. void SomeFunction (int x) virtual;
- В. virtual SomeFunction (int x);
- Г. virtual void SomeFunction (int * x);

24. В программе описан абстрактный класс А и производный от него класс А1. Какая из записей заведомо не верна?

- A. A * a = new A;
- Б. A1 a1;
- В. A1 a1; A & a = a1;
- Г. A1 a1; A1 a2;

25. Укажите правильный вариант доступа к членам объектов (язык C ++), описанных следующим образом:

```
class my
{
char s;
public:
double Z;
int f (int c, int d)
{return c + d;};
} T1, T2;
```

- A. T1.Z = 23.1;
- Б. T2-> f (2, 1);
- В. T1.s = '#';
- Г. my.T2-> s = 'L';

Раздел 2. Промежуточная аттестация

Форма контроля/оценочное средство:

Вопросы/Задания:

.

Раздел 3. Курсовой проект

Форма контроля/оценочное средство:

Вопросы/Задания:

.

7. Оценочные материалы промежуточной аттестации

Очная форма обучения, Третий семестр, Зачет

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

Вопросы/Задания:

1. Понятие абстракции и абстрактных типов данных
2. Базовые термины и определения объектно-ориентированного программирования
3. Основные принципы объектно-ориентированного программирования
4. Классификация языков программирования по степени поддержки объектно-ориентированного подхода

5. Объектно-ориентированные языки программирования
6. Параметрический полиморфизм. Перегрузка функций
7. Особенности перегрузки функций со значениями параметров по умолчанию
8. Параметрический полиморфизм. Шаблоны функций
9. Основные свойства параметров шаблонов функций
10. Класс как абстрактный тип данных
11. Создание объектов классов
12. Доступ к компонентам класса через объект и через указатель на него
13. Конструкторы классов. Понятие, способы описания, вызов
14. Деструкторы классов. Понятие, способы описания, вызов
15. Доступность компонентов класса
16. Компоненты классов: данные и методы
17. Статические компоненты классов
18. Константные компоненты классов
19. Указатель this
20. Дружественные функции классов
21. Дружественность функций и классов при наследовании
22. Перегрузка стандартных операций. Понятие и ограничения
23. Перегрузка стандартных операций. Описание и вызов операций-функций
24. Перегрузка префиксных и постфиксных операций
25. Перегрузка операций ввода-вывода для типов, определенных пользователем
26. Наследование классов. Базовые и производные классы
27. Переопределение функций при наследовании классов
28. Статусы доступа при наследовании
29. Динамический полиморфизм. Виртуальные функции

30. Чистые виртуальные функции и абстрактные классы
31. Множественное наследование и виртуальные базовые классы
32. Локальные классы
33. Параметрический полиморфизм. Описание шаблонов классов
34. Параметрический полиморфизм. Создание объектов шаблонов
35. Понятие и примеры исключительных ситуаций
36. Контролируемые блоки программ
37. Способы генерации и перехвата исключений. Обработчики исключений
38. Создание классов исключений
39. Вложенные try-блоки и ретрансляция исключений
40. Стандартная библиотека шаблонов STL. Назначение и общие характеристики
41. Стандартная библиотека шаблонов STL. Создание строк типа string
42. Стандартная библиотека шаблонов STL. Особенности ввода строк типа string
43. Стандартная библиотека шаблонов STL. Операции со строками типа string
44. Стандартная библиотека шаблонов STL. Основные функции работы со строками типа string
45. Стандартная библиотека шаблонов STL и ее структура
46. Стандартная библиотека шаблонов STL. Понятие и виды итераторов
47. Стандартная библиотека шаблонов STL. Последовательные контейнеры
48. Стандартная библиотека шаблонов STL. Ассоциативные контейнеры
49. Стандартная библиотека шаблонов STL. Контейнеры-адаптеры
50. Стандартная библиотека шаблонов STL. Псевдоконтейнеры
51. Стандартная библиотека шаблонов STL. Работа со списками
52. Стандартная библиотека шаблонов STL. Работа с динамическими массивами
53. Стандартная библиотека шаблонов STL. Стеки и очереди

54. Стандартная библиотека шаблонов STL. Работа с множествами

55. Особенности организации конструкторов, деструкторов и основных операций с контейнерами STL

56. Алгоритмы стандартной библиотеки шаблонов STL

Очная форма обучения, Четвертый семестр, Курсовой проект

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

Вопросы/Задания:

1. Разработка объектно-ориентированного приложения «Стек»
2. Разработка объектно-ориентированного приложения «Очередь»
3. Разработка объектно-ориентированного приложения «Односвязный список»
4. Разработка объектно-ориентированного приложения «Двусвязный список»
5. Разработка объектно-ориентированного приложения «Множество»
6. Разработка объектно-ориентированного приложения «Кольцевой односвязный список»
7. Разработка объектно-ориентированного приложения «Бинарное дерево»
8. Разработка объектно-ориентированного приложения «Ориентированный граф»
9. Разработка объектно-ориентированного приложения «Неориентированный граф»
10. Разработка объектно-ориентированного приложения «Матрица»
11. Разработка объектно-ориентированного приложения «Составное арифметическое выражение»
12. Разработка объектно-ориентированного приложения «Лифт»
13. Разработка объектно-ориентированного приложения «Банкомат»
14. Разработка объектно-ориентированного приложения «Секундомер»
15. Разработка объектно-ориентированного приложения «Нечеткое множество»
16. Разработка объектно-ориентированного приложения «Нечеткое отношение»
17. Разработка объектно-ориентированного приложения «Часы с будильником»
18. Разработка объектно-ориентированного приложения «Система линейных алгебраических уравнений»

19. Разработка объектно-ориентированного приложения «График функции»

20. Разработка объектно-ориентированного приложения «Столбчатая диаграмма»

Очная форма обучения, Четвертый семестр, Экзамен

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

Вопросы/Задания:

1. Цели и способы использования языка UML
2. Базовые типы диаграмм UML
3. Границы применения UML
4. Диаграмма классов: правила построения и основные компоненты
5. Отношения наследования и генерализации. Понятие и изображение на языке UML
6. Отношения агрегации и композиции. Понятие и изображение на языке UML
7. Назначение и способы создания свойств классов
8. Аксессоры доступа: set и get. Сущность и использование
9. Понятие индексатора
10. Одномерные и двумерные индексаторы: создание и вызов
11. Интерфейсы классов: понятие и назначение
12. Реализация интерфейсов классов
13. Делегаты: сущность, способы описания и использования
14. Понятие, назначение и методы описания событий
15. Функционирование приложений в среде .NET
16. Среда выполнения .NET
17. Понятие сборки
18. Общая система типов .NET. Правила создания типов
19. Составные элементы библиотеки .NET
20. Иерархия классов .NET
21. Иерархия интерфейсов коллекций

22. Интерфейс ICollection: базовые свойства и методы
23. Интерфейс IList: базовые свойства и методы
24. Интерфейс IDictionary: базовые свойства и методы
25. Интерфейсы IEnumerable и IEnumerator: базовые свойства и методы
26. Классы коллекций: общий обзор
27. Класс ArrayList: конструкторы, базовые свойства и методы
28. Класс Hashtable: конструкторы, базовые свойства и методы
29. Класс SortedList: конструкторы, базовые свойства и методы
30. Класс Stack: конструкторы, базовые свойства и методы
31. Класс Queue: конструкторы, базовые свойства и методы
32. Обзор классов пространства имен System.IO
33. Иерархия классов потоков
34. Свойства, методы и исключения класса Stream
35. Класс FileStream. Назначение и принципы работы
36. Классы для работы с текстовыми файлами
37. Классы для работы с двоичными файлами
38. Сериализация/десериализация объектов
39. Функционирование Windows-приложений: понятие цикла обработки сообщений
40. Виды Windows-приложений: однодокументный и многодокументный интерфейс
41. Работа с формами: иерархия классов
42. Работа с формами: основные свойства и события
43. Работа с формами: передача данных между формами
44. Разработка MDI-приложений: родительские и дочерние окна
45. Разработка MDI-приложений: упорядочивание дочерних окон, доступ к активному документу

46. Понятие и виды многозадачности
47. Потоки: понятие, типы, состояния
48. Класс Thread и его базовые компоненты
49. Блокировка и синхронизация потоков
50. Средства межпоточного взаимодействия
51. Пространства имен для программирования сетевых приложений
52. Универсальные идентификаторы ресурсов
53. Классы WebRequest: базовые свойства и методы
54. Классы WebResponse: базовые свойства и методы
55. Виды сетевых исключений и их обработка
56. Класс Uri: конструкторы и базовые свойства
57. Доступ к дополнительной HTTP-информации
58. Доступ к заголовкам HTTP-ответа
59. Доступ к cookie-данным HTTP-ответа
60. Класс WebClient: базовые свойства и методы

Очно-заочная форма обучения, Четвертый семестр, Зачет
Контролируемые ИДК: ОПК-3.1 ОПК-3.2 ОПК-3.3 ОПК-3.4
Вопросы/Задания:

1. Понятие абстракции и абстрактных типов данных
2. Базовые термины и определения объектно-ориентированного программирования
3. Основные принципы объектно-ориентированного программирования
4. Классификация языков программирования по степени поддержки объектно-ориентированного подхода
5. Объектно-ориентированные языки программирования
6. Параметрический полиморфизм. Перегрузка функций
7. Особенности перегрузки функций со значениями параметров по умолчанию

8. Параметрический полиморфизм. Шаблоны функций
9. Основные свойства параметров шаблонов функций
10. Класс как абстрактный тип данных
11. Создание объектов классов
12. Доступ к компонентам класса через объект и через указатель на него
13. Конструкторы классов. Понятие, способы описания, вызов
14. Деструкторы классов. Понятие, способы описания, вызов
15. Доступность компонентов класса
16. Компоненты классов: данные и методы
17. Статические компоненты классов
18. Константные компоненты классов
19. Указатель this
20. Дружественные функции классов
21. Дружественность функций и классов при наследовании
22. Перегрузка стандартных операций. Понятие и ограничения
23. Перегрузка стандартных операций. Описание и вызов операций-функций
24. Перегрузка префиксных и постфиксных операций
25. Перегрузка операций ввода-вывода для типов, определенных пользователем
26. Наследование классов. Базовые и производные классы
27. Переопределение функций при наследовании классов
28. Статусы доступа при наследовании
29. Динамический полиморфизм. Виртуальные функции
30. Чистые виртуальные функции и абстрактные классы
31. Множественное наследование и виртуальные базовые классы
32. Локальные классы

33. Параметрический полиморфизм. Описание шаблонов классов
34. Параметрический полиморфизм. Создание объектов шаблонов
35. Понятие и примеры исключительных ситуаций
36. Контролируемые блоки программ
37. Способы генерации и перехвата исключений. Обработчики исключений
38. Создание классов исключений
39. Вложенные try-блоки и ретрансляция исключений
40. Стандартная библиотека шаблонов STL. Назначение и общие характеристики
41. Стандартная библиотека шаблонов STL. Создание строк типа string
42. Стандартная библиотека шаблонов STL. Особенности ввода строк типа string
43. Стандартная библиотека шаблонов STL. Операции со строками типа string
44. Стандартная библиотека шаблонов STL. Основные функции работы со строками типа string
45. Стандартная библиотека шаблонов STL и ее структура
46. Стандартная библиотека шаблонов STL. Понятие и виды итераторов
47. Стандартная библиотека шаблонов STL. Последовательные контейнеры
48. Стандартная библиотека шаблонов STL. Ассоциативные контейнеры
49. Стандартная библиотека шаблонов STL. Контейнеры-адаптеры
50. Стандартная библиотека шаблонов STL. Псевдоконтейнеры
51. Стандартная библиотека шаблонов STL. Работа со списками
52. Стандартная библиотека шаблонов STL. Работа с динамическими массивами
53. Стандартная библиотека шаблонов STL. Стеки и очереди
54. Стандартная библиотека шаблонов STL. Работа с множествами
55. Особенности организации конструкторов, деструкторов и основных операций с контейнерами STL
56. Алгоритмы стандартной библиотеки шаблонов STL

Вопросы/Задания:

1. Разработка объектно-ориентированного приложения «Стек»
2. Разработка объектно-ориентированного приложения «Очередь»
3. Разработка объектно-ориентированного приложения «Односвязный список»
4. Разработка объектно-ориентированного приложения «Двусвязный список»
5. Разработка объектно-ориентированного приложения «Множество»
6. Разработка объектно-ориентированного приложения «Кольцевой односвязный список»
7. Разработка объектно-ориентированного приложения «Бинарное дерево»
8. Разработка объектно-ориентированного приложения «Ориентированный граф»
9. Разработка объектно-ориентированного приложения «Неориентированный граф»
10. Разработка объектно-ориентированного приложения «Матрица»
11. Разработка объектно-ориентированного приложения «Составное арифметическое выражение»
12. Разработка объектно-ориентированного приложения «Лифт»
13. Разработка объектно-ориентированного приложения «Банкомат»
14. Разработка объектно-ориентированного приложения «Секундомер»
15. Разработка объектно-ориентированного приложения «Нечеткое множество»
16. Разработка объектно-ориентированного приложения «Нечеткое отношение»
17. Разработка объектно-ориентированного приложения «Часы с будильником»
18. Разработка объектно-ориентированного приложения «Система линейных алгебраических уравнений»
19. Разработка объектно-ориентированного приложения «График функции»
20. Разработка объектно-ориентированного приложения «Столбчатая диаграмма»

Вопросы/Задания:

1. Цели и способы использования языка UML
2. Базовые типы диаграмм UML
3. Границы применения UML
4. Диаграмма классов: правила построения и основные компоненты
5. Отношения наследования и генерализации. Понятие и изображение на языке UML
6. Отношения агрегации и композиции. Понятие и изображение на языке UML
7. Назначение и способы создания свойств классов
8. Аксессоры доступа: set и get. Сущность и использование
9. Понятие индексатора
10. Одномерные и двумерные индексаторы: создание и вызов
11. Интерфейсы классов: понятие и назначение
12. Реализация интерфейсов классов
13. Делегаты: сущность, способы описания и использования
14. Понятие, назначение и методы описания событий
15. Функционирование приложений в среде .NET
16. Среда выполнения .NET
17. Понятие сборки
18. Общая система типов .NET. Правила создания типов
19. Составные элементы библиотеки .NET
20. Иерархия классов .NET
21. Иерархия интерфейсов коллекций
22. Интерфейс ICollection: базовые свойства и методы
23. Интерфейс IList: базовые свойства и методы
24. Интерфейс IDictionary: базовые свойства и методы

25. Интерфейсы IEnumerable и IEnumerator: базовые свойства и методы
26. Классы коллекций: общий обзор
27. Класс ArrayList: конструкторы, базовые свойства и методы
28. Класс Hashtable: конструкторы, базовые свойства и методы
29. Класс SortedList: конструкторы, базовые свойства и методы
30. Класс Stack: конструкторы, базовые свойства и методы
31. Класс Queue: конструкторы, базовые свойства и методы
32. Обзор классов пространства имен System.IO
33. Иерархия классов потоков
34. Свойства, методы и исключения класса Stream
35. Класс FileStream. Назначение и принципы работы
36. Классы для работы с текстовыми файлами
37. Классы для работы с двоичными файлами
38. Сериализация/десериализация объектов
39. Функционирование Windows-приложений: понятие цикла обработки сообщений
40. Виды Windows-приложений: однодокументный и многодокументный интерфейс
41. Работа с формами: иерархия классов
42. Работа с формами: основные свойства и события
43. Работа с формами: передача данных между формами
44. Разработка MDI-приложений: родительские и дочерние окна
45. Разработка MDI-приложений: упорядочивание дочерних окон, доступ к активному документу
46. Понятие и виды многозадачности
47. Потоки: понятие, типы, состояния
48. Класс Thread и его базовые компоненты

- 49. Блокировка и синхронизация потоков
- 50. Средства межпоточного взаимодействия
- 51. Пространства имен для программирования сетевых приложений
- 52. Универсальные идентификаторы ресурсов
- 53. Классы WebRequest: базовые свойства и методы
- 54. Классы WebResponse: базовые свойства и методы
- 55. Виды сетевых исключений и их обработка
- 56. Класс Uri: конструкторы и базовые свойства
- 57. Доступ к дополнительной HTTP-информации
- 58. Доступ к заголовкам HTTP-ответа
- 59. Доступ к cookie-данным HTTP-ответа
- 60. Класс WebClient: базовые свойства и методы

8. Материально-техническое и учебно-методическое обеспечение дисциплины

8.1. Перечень основной и дополнительной учебной литературы

Основная литература

1. Мейер,, Б. Основы объектно-ориентированного проектирования: учебник / Б. Мейер,. - Основы объектно-ориентированного проектирования - Москва: Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2025. - 751 с. - 978-5-4497-0885-4. - Текст: электронный // IPR SMART: [сайт]. - URL: <https://www.iprbookshop.ru/146365.html> (дата обращения: 08.10.2025). - Режим доступа: по подписке
2. Комлев, Н.Ю. Объектно Ориентированное Программирование. Настольная книга программиста: Практическое пособие / Н.Ю. Комлев. - 1 - Москва: Издательство "СОЛОН-Пресс", 2022. - 298 с. - 978-5-91359-276-7. - Текст: электронный // Общество с ограниченной ответственностью «ЗНАНИУМ»: [сайт]. - URL: <https://znanium.ru/cover/2185/2185094.jpg> (дата обращения: 08.09.2025). - Режим доступа: по подписке
3. 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: 12.05.2026). - Режим доступа: по подписке

Дополнительная литература

1. Володько Л. П. Объектно-ориентированное программирование: учебное пособие / Володько Л. П., Николаенко В. Л., Николаенко Д. В.. - Пинск: ПолесГУ, 2020. - 164 с. - 978-985-516-649-9. - Текст: электронный. // RuSpLAN: [сайт]. - URL: <https://e.lanbook.com/img/cover/book/448427.jpg> (дата обращения: 08.10.2025). - Режим доступа: по подписке

2. Хорев, П. Б. Объектно-ориентированное программирование с примерами на C#: Учебное пособие / П. Б. Хорев. - 1 - Москва: Издательство "ФОРУМ", 2023. - 200 с. - 978-5-16-015117-5. - Текст: электронный // Общество с ограниченной ответственностью «ЗНАНИУМ»: [сайт]. - URL: <https://znanium.com/cover/1926/1926392.jpg> (дата обращения: 08.09.2025). - Режим доступа: по подписке

3. ИВАНОВА Е. А. Объектно-ориентированное программирование: метод. указания / ИВАНОВА Е. А., Ефанова Н. В.. - Краснодар: КубГАУ, 2022. - 34 с. - Текст: электронный. // : [сайт]. - URL: <https://edu.kubsau.ru/mod/resource/view.php?id=11422> (дата обращения: 12.05.2026). - Режим доступа: по подписке

4. ИВАНОВА Е. А. Объектно-ориентированное программирование: практикум / ИВАНОВА Е. А., Ефанова Н. В.. - Краснодар: КубГАУ, 2022. - 233 с. - Текст: электронный. // : [сайт]. - URL: <https://edu.kubsau.ru/mod/resource/view.php?id=11425> (дата обращения: 12.05.2026). - Режим доступа: по подписке

5. Мейер,, Б. Почувствуй класс. Учимся программировать хорошо с объектами и контрактами: учебник / Б. Мейер,; под редакцией В. А. Биллига. - Почувствуй класс. Учимся программировать хорошо с объектами и контрактами - Москва: Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2024. - 775 с. - 978-5-4497-3332-0. - Текст: электронный // IPR SMART: [сайт]. - URL: <https://www.iprbookshop.ru/142295.html> (дата обращения: 08.10.2025). - 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. <http://www.iprbookshop.ru/> - IPRbook
3. <https://znanium.com/> - Znanium.com
4. <https://e.lanbook.com/> - Lanbook
5. <https://intuit.ru/> - Материалы Национального Открытого Университета «Интуит»
6. <https://learn.microsoft.com/ru-ru/> - Материалы портала для разработчиков Microsoft

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.

402эк

Персональный компьютер IRU 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 объем 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 зачетных 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 АООП 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,

тлости 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 плоскостную информацию;
- наличие возможности использовать индивидуальные звукоусиливающие устройства и сурдотехнические средства, позволяющие осуществлять приём и передачу информации; осуществлять взаимобратный 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)

Дисциплина "Объектно-ориентированное программирование" ведётся в соответствии с календарным учебным планом и расписанием занятий по неделям. Темы проведения занятий определяются тематическим планом рабочей программы дисциплины