

Рабочая программа дисциплины

Объектно-ориентированное программирование

<i>Направление подготовки</i>	Информационные системы и технологии
<i>Код</i>	09.03.02
<i>Направленность (профиль)</i>	Информационные системы и технологии в экономике и управлении
<i>Квалификация выпускника</i>	бакалавр

1. Перечень кодов компетенций, формируемых дисциплиной в процессе освоения образовательной программы

Группа компетенций	Категория компетенций	Код
Профессиональные	-	ПК-5

2. Компетенции и индикаторы их достижения

Код компетенции	Формулировка компетенции	Индикаторы достижения компетенции
ПК-5	Способен разрабатывать алгоритмы и программы на базе языков программирования и пакетов прикладных программ, пригодные для практического применения.	ПК-5.1 Кодирование на языках программирования. ПК-5.2 Разработка кода ИС и баз данных ИС. ПК-5.3 Верификация кода ИС и баз данных ИС относительно дизайна ИС и структуры баз данных ИС. ПК-5.4 Разработка структуры программного кода ИС. ПК-5.5 Верификация структуры программного кода ИС относительно архитектуры ИС. ПК-5.6 Организационное и технологическое обеспечение кодирования на языках программирования.

3. Описание планируемых результатов обучения по дисциплине

3.1. Описание планируемых результатов обучения по дисциплине

Планируемые результаты обучения по дисциплине представлены дескрипторами (знания, умения, навыки).

Дескрипторы по дисциплине	Знать	Уметь	Владеть
Код компетенции	ПК-5		
	-современные методы разработки и реализации алгоритмов на базе языков программирования и пакетов прикладных программ; - основы современных систем управления базами данных, теории баз данных; - формальные	-разрабатывать алгоритмы и программы на базе языков программирования и пакетов прикладных программ, пригодные для практического применения; - кодировать на языках программирования; - тестировать результаты кодирования;	- приемами разработки алгоритмов и программ на базе языков программирования и пакетов прикладных программ, пригодных для практического применения.

	<p>методы, технологии и инструменты разработки программного обеспечения и баз данных;</p> <p>- основы программирования, современные объектно-ориентированные языки программирования;</p> <p>современные структурные языки программирования, языки современных бизнес-приложений;</p> <p>- современные методики тестирования разрабатываемых ИС: инструменты и методы модульного тестирования, инструменты и методы тестирования нефункциональных и функциональных характеристик ИС.</p>		
--	---	--	--

4. Место дисциплины (модуля) в структуре образовательной программы

Дисциплина относится к части, формируемой участниками образовательных отношений учебного плана ОПОП.

Данная дисциплина взаимосвязана с другими дисциплинами, такими как «Спецсеминар (компьютерный практикум)», «Спецсеминар (математический практикум)», «Методы и компьютерные технологии имитационного моделирования», «Компьютерная графика», «Разработка программных приложений», «Программная инженерия».

В рамках освоения программы бакалавриата выпускники готовятся к решению задач профессиональной деятельности следующих типов: научно-исследовательский, производственно-технологический, организационно-управленческий, проектный.

Профиль (направленность) программы установлена путем ее ориентации на сферу профессиональной деятельности выпускников: Информационные системы и технологии в экономике и управлении.

5. Объем дисциплины

<i>Виды учебной работы</i>	<i>Формы обучения</i>
	<i>Очная</i>
Общая трудоемкость: зачетные единицы/часы	11/396

Контактная работа:	
Занятия лекционного типа	76
Занятия семинарского типа	152
Промежуточная аттестация: зачет с оценкой, курсовая работа, экзамен	36.15
Самостоятельная работа (СРС)	131.85

6. Содержание дисциплины (модуля), структурированное по темам / разделам с указанием отведенного на них количества академических часов и видов учебных занятий

6.1.Распределение часов по разделам/темам и видам работы

6.1.1.Очная форма обучения

№ п/п	Раздел/тема	Виды учебной работы (в часах)						Самостоятельная работа
		Контактная работа						
		Занятия лекционного типа		Занятия семинарского типа				
		Лекции	Иные учебные занятия	Практические занятия	Семинары	Лабораторные работы	Иные	
1.	Объекты ООП.	4		8				8
2.	Пространства имен.	4		8				8
3.	Модификаторы.	4		8				8
4.	Наследование.	4		8				8
5.	Абстрактные классы.	4		8				8
6.	Обработка исключений.	4		8				8
7.	Делегаты, события и лямбды.	4		8				8
8.	Интерфейсы.	4		8				8
9.	Дополнительные возможности ООП в C#.	4		8				8
10.	Коллекции	4		8				8
11.	Работа со строками	4		8				8
12.	Дополнительные классы и структуры .NET	4		8				6
13.	Многопоточность	4		8				6
14.	Асинхронное программирование	4		8				6
15.	LINQ	4		8				6
16.	Рефлексия. Dynamic Language Runtime. Сборка мусора, управление памятью и указатели.	4		8				6
17.	Работа с потоками и файловой системой	4		8				6
18.	Сериализация.	4		8				6
19.	Процессы и домены приложения. Валидация	4		8				1,85

	модели.						
	Промежуточная аттестация	36.15					
	Итого	76		152			131.85

6.2 Программа дисциплины, структурированная по темам / разделам

6.2.1 Содержание лекционного курса

№ п/п	Наименование темы (раздела) дисциплины	Содержание лекционного занятия
1.	Объекты ООП.	Классы и объекты. Конструкторы, инициализаторы и деконструкторы.
2.	Пространства имен.	Пространства имен. Глобальные пространства имен.
3.	Модификаторы.	Модификаторы доступа. Свойства. Перегрузка методов. Поля и структуры для чтения.
4.	Наследование.	Наследование. Преобразование типов. Перегрузка операций преобразования типов. Виртуальные методы и свойства.
5.	Абстрактные классы.	Абстрактные классы. Обобщенные типы. Ограничения обобщений.
6.	Обработка исключений.	Исключения и фильтры исключений. Типы исключений. Генерация исключения.
7.	Делегаты, события и лямбды.	Делегаты. Анонимные методы. Лямбды. События. Ковариантность и контравариантность делегатов. Замыкания.
8.	Интерфейсы.	Определение интерфейсов. Применение интерфейсов. Явная реализация интерфейсов. Реализация интерфейсов в базовых и производных классах. Наследование интерфейсов. Интерфейсы в обобщениях. Копирование объектов.
9.	Дополнительные возможности ООП в C#.	Переменные-ссылки и возвращение ссылки. Методы расширения. Частичные классы и методы. Анонимные типы. Кортежи.
10.	Коллекции	Список. Двухсвязный список. Очередь. Стек. Словарь. Итераторы.
11.	Работа со строками	Операции со строками. Форматирование и интерполяция строк. Регулярные выражения. Работа с датами и временем.
12.	Дополнительные классы и структуры .NET	Отложенная инициализация. Математические вычисления. Преобразование типов. Индексы и диапазоны.
13.	Многопоточность	Введение в многопоточность. Создание потоков. Потоки с параметрами. Синхронизация потоков. Мониторы. Мьютексы. Семафоры. Таймеры. Задачи и класс Task. Задачи продолжения. Класс

		Parallel. Отмена задач и параллельных операций.
14.	Асинхронное программирование	Асинхронные методы. Возвращение результата из асинхронного метода. Последовательный и параллельный вызов асинхронных операций.
15.	LINQ	Фильтрация выборки и проекция. Сортировка. Объединение, пересечение и разность коллекций. Агрегатные операции. Группировка. Соединение коллекций. Отложенное и немедленное выполнение LINQ. Делегаты и анонимные методы в запросах LINQ.
16.	Рефлексия. Dynamic Language Runtime. Сборка мусора, управление памятью и указатели.	Введение в рефлексию. Применение рефлексии и исследование типов. Динамическая загрузка сборок и позднее связывание. Сборщик мусора в C#. Финализируемые объекты. Указатели. Указатели на структуры, члены классов и массивы.
17.	Работа с потоками и файловой системой	Работа с дисками. Работа с каталогами. Работа с файлами. Чтение и запись файла. Чтение и запись текстовых файлов. Бинарные файлы. Бинарная сериализация. Архивация и сжатие файлов.
18.	Сериализация.	Сериализация в JSON. XML-Документы. Изменение XML-документа. Создание Xml-документа.
19.	Процессы и домены приложения. Валидация модели.	Процессы. Домены приложений. Основы валидации модели. Атрибуты валидации.

6.2.2 Содержание практических занятий

№ п/п	Наименование темы (раздела) дисциплины	Содержание практического занятия
1.	Объекты ООП.	Создание объекта класса. Обращение к функциональности класса. Создание конструкторов. Класс Program и метод Main. Программы верхнего уровня.
2.	Пространства имен.	Подключение пространств имен по умолчанию. Создание библиотеки классов.
3.	Модификаторы.	Модификаторы в рамках текущего проекта. Модификаторы в рамках сборок. Определение свойств. Перегрузка методов. Перегрузка методов.
4.	Наследование.	Доступ к членам базового класса из класса-наследника. Конструкторы в производных классах.
5.	Абстрактные классы.	Класс System.Object и его методы. Обобщенные типы. Ограничения обобщений. Наследование обобщенных типов.
6.	Обработка исключений.	Конструкция try..catch..finally. Блок catch и фильтры исключений. Класс Exception. Поиск блока catch при обработке исключений. Оператор throw.
7.	Делегаты, события и лямбды.	Делегаты. Лямбды. События. Делегаты Action, Predicate и Func. Замыкания.
8.	Интерфейсы.	Явная реализация интерфейсов. Реализация интерфейсов в базовых и производных классах.

		Наследование интерфейсов. Интерфейсы в обобщениях. Интерфейс ICloneable. Интерфейс IComparable.
9.	Дополнительные возможности ООП в C#.	Null и ссылочные типы. Null и значимые типы. Проверка на null, операторы ? и ?. Частичные классы и методы. Анонимные типы. Кортежи. Patternmatching. Records.
10.	Коллекции	ArrayList. Список List<T>. Двухсвязный список LinkedList<T>. Очередь Queue<T>. Стек Stack<T>. Словарь Dictionary<T, V>. Класс ObservableCollection. Интерфейсы IEnumerable и IEnumerator. Итераторы и оператор yield.
11.	Работа со строками	Строки и класс System.String. Операции со строками. Форматирование и интерполяция строк. Класс StringBuilder. Работа с датами и временем.
12.	Дополнительные классы и структуры .NET	Отложенная инициализация и тип Lazy. Математические вычисления и класс Math. Преобразование типов и класс Convert. Класс Array и массивы. Span.
13.	Многопоточность	Класс Thread. Делегат ThreadStart. Поток с параметрами ParameterizedThreadStart. Синхронизация потоков. Класс AutoResetEvent. Семафоры. Таймеры. Задачи и класс Task. Работа с классом Task. Класс Parallel. CancellationToken.
14.	Асинхронное программирование	Асинхронные методы, async и await. Обработка ошибок в асинхронных методах. Отмена асинхронных операций. Асинхронные стримы.
15.	LINQ	Основы LINQ. Фильтрация выборки и проекция. Сортировка. Методы Skip и Take. Метод Join, GroupJoin и Zip. Методы All и Any. Делегаты и анонимные методы в запросах LINQ.
16.	Рефлексия. Dynamic Language Runtime. Сборка мусора, управление памятью и указатели.	Класс System.Type. Атрибуты в .NET. DLR в C#. Ключевое слово dynamic. DynamicObject и ExpandoObject. Использование IronPython в .NET. Метод Dispose. Конструкция using. Указатели. Указатели на структуры, члены классов и массивы.
17.	Работа с потоками и файловой системой	Классы File и FileInfo. FileStream. StreamReader и StreamWriter. BinaryWriter и BinaryReader. BinaryFormatter. Архивация и сжатие файлов.
18.	Сериализация.	JsonSerializer. Работа с XML с помощью System.Xml. XPath. Linq to Xml. Создание Xml-документа. Выборка элементов в Linq to XML. XmlSerializer.
19.	Процессы и домены приложения. Валидация модели.	Процессы. AssemblyLoadContext и динамическая загрузка и выгрузка сборок. Создание своих атрибутов валидации. Самовалидация модели.

6.2.3 Содержание самостоятельной работы

№	Наименование темы	Содержание самостоятельной работы
---	-------------------	-----------------------------------

п/п	(раздела) дисциплины	
1.	Объекты ООП.	Классы и объекты. Конструкторы, инициализаторы и деконструкторы. Класс Program и метод Main. Программы верхнего уровня.
2.	Пространства имен.	Пространства имен. Глобальные пространства имен. Подключение пространств имен по умолчанию. Создание библиотеки классов.
3.	Модификаторы.	Модификаторы доступа. Свойства. Перегрузка методов. Статические члены и модификатор static. Поля и структуры для чтения. Перегрузка операторов. Индексаторы.
4.	Наследование.	Наследование. Преобразование типов. Перегрузка операций преобразования типов. Виртуальные методы и свойства. Соккрытие методов. Различие переопределения и сокращения методов.
5.	Абстрактные классы.	Абстрактные классы. Класс System.Object и его методы. Обобщенные типы. Ограничения обобщений. Наследование обобщенных типов.
6.	Обработка исключений.	Конструкция try..catch..finally. Блок catch и фильтры исключений. Типы исключений. Класс Exception. Создание классов исключений. Поиск блока catch при обработке исключений. Генерация исключения и оператор throw.
7.	Делегаты, события и лямбды.	Делегаты. Применение делегатов. Анонимные методы. Лямбды. События. Ковариантность и контравариантность делегатов. Делегаты Action, Predicate и Func. Замыкания.
8.	Интерфейсы.	Определение интерфейсов. Применение интерфейсов. Явная реализация интерфейсов. Реализация интерфейсов в базовых и производных классах. Наследование интерфейсов. Интерфейсы в обобщениях. Копирование объектов. Интерфейс ICloneable. Сортировка объектов. Интерфейс IComparable. Ковариантность и контравариантность обобщенных интерфейсов
9.	Дополнительные возможности ООП в C#.	Null и ссылочные типы. Null и значимые типы. Проверка на null, операторы ?. и ??. Переменные-ссылки и возвращение ссылки. Методы расширения. Частичные классы и методы. Анонимные типы. Кортежи. Patternmatching. Records.
10	Коллекции	ArrayList. Список List<T>. Двухсвязный список LinkedList<T>. Очередь Queue<T>. Стек Stack<T>. Словарь Dictionary<T, V>. Класс ObservableCollection. Интерфейсы IEnumerable и IEnumerator. Итераторы и оператор yield.
11	Работа со строками	Работа со строками. Строки и класс System.String. Операции со строками. Форматирование и интерполяция строк. Класс StringBuilder. Регулярные выражения. Работа с датами и временем.
12	Дополнительные классы и структуры .NET	Отложенная инициализация и тип Lazy. Математические вычисления и класс Math.

		Преобразование типов и класс Convert. Класс Array и массивы. Span. Индексы и диапазоны.
13	Многопоточность	Введение в многопоточность. Класс Thread. Создание потоков. Делегат ThreadStart. Потоки с параметрами и ParameterizedThreadStart. Синхронизация потоков. Мониторы. Класс AutoResetEvent. Мьютексы. Семафоры. Таймеры. Задачи и класс Task. Работа с классом Task. Задачи продолжения. Класс Parallel. Отмена задач и параллельных операций. CancellationToken.
14	Асинхронное программирование	Асинхронные методы, async и await. Возвращение результата из асинхронного метода. Последовательный и параллельный вызов асинхронных операций. Обработка ошибок в асинхронных методах. Отмена асинхронных операций. Асинхронные стримы.
15	LINQ	Основы LINQ. Фильтрация выборки и проекция. Сортировка. Объединение, пересечение и разность коллекций. Агрегатные операции. Методы Skip и Take. Группировка. Соединение коллекций. Метод Join, GroupJoin и Zip. Методы All и Any. Отложенное и немедленное выполнение LINQ. Делегаты и анонимные методы в запросах LINQ.
16	Рефлексия. Dynamic Language Runtime. Сборка мусора, управление памятью и указатели.	Введение в рефлексю. Класс System.Type. Применение рефлексии и исследование типов. Динамическая загрузка сборок и позднее связывание. Атрибуты в .NET. DLR в C#. Ключевое слово dynamic. DynamicObject и ExpandoObject. Использование IronPython в .NET. Сборщик мусора в C#. Финализируемые объекты. Метод Dispose. Конструкция using. Указатели. Указатели на структуры, члены классов и массивы.
17	Работа с потоками и файловой системой	Работа с дисками. Работа с каталогами. Работа с файлами. Классы File и FileInfo. FileStream. Чтение и запись файла. Чтение и запись текстовых файлов. StreamReader и StreamWriter. Бинарные файлы. BinaryWriter и BinaryReader. Бинарная сериализация. BinaryFormatter. Архивация и сжатие файлов.
18	Сериализация.	Сериализация в JSON. JsonSerializer. XML-Документы. Работа с XML с помощью System.Xml. Изменение XML-документа. XPath. LinqtoXml. Создание Xml-документа. Выборка элементов в LINQ to XML. Изменение документа в LINQ to XML. Сериализация в XML. XmlSerializer.
19	Процессы и домены приложения. Валидация модели.	Процессы. Домены приложений. AssemblyLoadContext и динамическая загрузка и выгрузка сборок. Основы валидации модели. Атрибуты валидации. Создание своих атрибутов валидации. Самовалидация модели.

7. Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине (модулю)

Предусмотрены следующие виды контроля качества освоения конкретной дисциплины:

- текущий контроль успеваемости
- промежуточная аттестация обучающихся по дисциплине

Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине оформлен в **ПРИЛОЖЕНИИ** к РАБОЧЕЙ ПРОГРАММЕ ДИСЦИПЛИНЫ

Текущий контроль успеваемости обеспечивает оценивание хода освоения дисциплины в процессе обучения.

7.1 Паспорт фонда оценочных средств для проведения текущей аттестации по дисциплине (модулю)

№ п/п	Контролируемые разделы (темы)	Наименование оценочного средства
1.	Объекты ООП.	Опрос, проблемно-аналитическое задание, тестирование.
2.	Пространства имен.	Опрос, проблемно-аналитическое задание, тестирование.
3.	Модификаторы.	Опрос, проблемно-аналитическое задание, тестирование.
4.	Наследование.	Опрос, проблемно-аналитическое задание, тестирование.
5.	Абстрактные классы.	Опрос, проблемно-аналитическое задание, тестирование.
6.	Обработка исключений.	Опрос, проблемно-аналитическое задание, тестирование.
7.	Делегаты, события и лямбды.	Опрос, проблемно-аналитическое задание, тестирование.
8.	Интерфейсы.	Опрос, проблемно-аналитическое задание, тестирование.
9.	Дополнительные возможности ООП в C#.	Опрос, проблемно-аналитическое задание, тестирование.
10.	Коллекции	Опрос, проблемно-аналитическое задание, тестирование.
11.	Работа со строками	Опрос, проблемно-аналитическое задание, тестирование.
12.	Дополнительные классы и структуры .NET	Опрос, проблемно-аналитическое задание, тестирование.

13.	Многопоточность	Опрос, проблемно-аналитическое задание, тестирование.
14.	Асинхронное программирование	Опрос, проблемно-аналитическое задание, тестирование.
15.	LINQ	Опрос, проблемно-аналитическое задание, тестирование.
16.	Рефлексия. Dynamic Language Runtime. Сборка мусора, управление памятью и указатели.	Опрос, проблемно-аналитическое задание, тестирование.
17.	Работа с потоками и файловой системой	Опрос, проблемно-аналитическое задание, тестирование.
18.	Сериализация.	Опрос, проблемно-аналитическое задание, тестирование.
19.	Процессы и домены приложения. Валидация модели.	Опрос, проблемно-аналитическое задание, тестирование.

7.2 Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности в процессе текущего контроля

Типовые вопросы

1. Объекты ООП.
2. Пространства имен.
3. Модификаторы.
4. Наследование.
5. Абстрактные классы.
6. Обработка исключений.
7. Делегаты, события и лямбды.
8. Интерфейсы.
9. Дополнительные возможности ООП в C#.
10. Коллекции
11. Работа со строками
12. Дополнительные классы и структуры .NET
13. Многопоточность
14. Асинхронное программирование
15. LINQ
16. Рефлексия. Dynamic Language Runtime. Сборка мусора, управление памятью и указатели.
17. Работа с потоками и файловой системой
18. Сериализация.
19. Процессы и домены приложения. Валидация модели.

Типовые проблемно-аналитические задания

1. Создать класс Персона, содержащий поля Имя и Возраст. Создать наследуемый класс Студент с дополнительным полем Группа. Создать функционал консольного ввода и вывода базы данных студентов ВУЗа по группам.
2. Создать класс Человек, содержащий поля Идентификатор, Фамилия, Имя, Отчество, Дата Рождения, Отец, Мать, список детей. Создать функционал консольного ввода и вывода базы данных генеалогического дерева. Реализовать функции вывода детей и вывода родителей определенного человека из базы данных.
3. Создать анализатор JSON, получаемый с помощью запроса погоды через открытые API. Реализовать функционал вывода доступных городов и отображение погоды в определенном городе по запросу.

Темы исследовательских, творческих проектов

1. Объектно-ориентированный подход к программированию.
2. Объектно-ориентированное программирование и программная инженерия.
3. Объектно-ориентированные языки программирования.
4. Основные требования к объектно-ориентированной системе.
5. Компонентное программирование.
6. Класс-ориентированное программирование.
7. История и эволюция Java.
8. Версии языка Java. Средства разработки.
9. Значение ССJ для программистов.
10. Application Programming Interface – интерфейс прикладного программирования.
11. Принципы объектно-ориентированного дизайна классов.
12. Полиморфизм, абстрактные классы и интерфейсы.
13. Язык программирования Python.
14. Технология Microsoft .NET.
15. Язык программирования C++.
16. Язык программирования C#.
17. Язык программирования Objective-C.
18. Язык программирования Ruby.
19. Язык программирования Swift.
20. Инструментальные средства разработки программ.

Типовые задания к интерактивным занятиям

1. Дан объект: цвет в формате RGB, который является элементом для набора. Элемент состоит из компонент: красная [0...255], зеленая [0...255], синяя [0...255], которые хранятся в нём как структура битовых полей. Реализовать сортировку по компонентам и целиком, пересчёт в СΥΜΚ.
2. Дан объект: двумерный вектор. Реализовать операции: (+ и +=) – сложение, (- и -=) – вычитание, (* и *=) – умножение, (!) – проверка на нулевой вектор, (== и !=) – сравнение, (-) – противоположный вектор, (~) – единичный вектор, (=) – копирование, (^) – вычисление угла, (double) – вычисление длины вектора, (cin>>) – ввод из потока, (cout<<) – вывод в поток.
3. Дан объект: двумерная фигура окружность. Реализовать описание свойств фигуры: ограничивающая область: $\{E = (x_0 - R, y_0 - R)_{\min} \rightarrow (x_0 + R, y_0 + R)_{\max}\}$, длина окружности

$\{L = 2\pi R\}$, площадь окружности $\{S = \pi R^2\}$. Интерфейс и классы реализуются в одном модуле. Для редактирования фигуры разработать функцию `ModifyFigure()`, которая должна получать ссылку на экземпляр фигуры и предоставлять интерактивный консольный интерфейс для работы с ним.

Типовые тесты

1. Из приведенных ниже высказываний укажите все истинные высказывания:
 - 1) модификатор (спецификатор) `protected` определяет доступность вложенного класса коду производных классов
 - 2) модификатор (спецификатор) `protected` определяет, где доступен не вложенный класс или любой другой не вложенный тип
 - 3) модификатор (спецификатор) `protected` определяет доступность любого члена класса в производных классах в любом коде
 - 4) модификатор (спецификатор) `protected` определяет доступность любого члена класса в производных классах только в пределах включающей его сборки
 - 5) модификатор (спецификатор) `protected` нельзя применять к событиям
2. Определяемый программистом тип может быть
 - 1) классом
 - 2) одномерным массивом
 - 3) делегатом
 - 4) строкой символов
 - 5) событием
3. Определяемый программистом тип может быть
 - 1) файлом
 - 2) структурой
 - 3) перечислением
 - 4) указателем
 - 5) многомерным массивом
4. Из приведенных ниже высказываний укажите истинное высказывание:
 - 1) если метод описывается вне класса, то в классе должен быть указан его прототип
 - 2) если метод не возвращает значение, то он должен иметь модификатор (спецификатор) `virtual`
 - 3) для вызова метода можно не создавать экземпляр (объект) класса только в том случае, если метод вызывается внутри класса
 - 4) метод может иметь пустое тело
 - 5) метод должен возвращать какое-либо значение
5. В заголовке определяемого не вложенного типа программист может указать доступность типа с помощью модификатора:
 - 1) `internal`
 - 2) `protected`
 - 3) `private`
 - 4) `static`
 - 5) `abstract`
6. В заголовке определяемого не вложенного типа программист может указать доступность типа
 - 1) с помощью модификатора `override`

- 2) с помощью модификатора `protected`
 - 3) с помощью модификатора `private`
 - 4) с помощью модификатора `new`
 - 5) по умолчанию
7. Из приведенных ниже высказываний укажите все истинные высказывания:
- 1) для перебора элементов коллекции с помощью оператора `foreach` необходимо реализовать интерфейс `IEnumerable`
 - 2) класс `Queue` имеет метод, изменяющий порядок следования элементов всего списка или его части на обратный
 - 3) коллекция `SortedList` хранит элементы, отсортированные по значению
 - 4) коллекция класса `ListDictionary` является связным (однонаправленным) списком
 - 5) упаковка объектов - это процесс сжатия данных
8. Из приведенных ниже высказываний укажите все истинные высказывания:
- 1) удаление элементов из коллекции `SortedList` осуществляется по принципу "первым пришел - первым обслужен"
 - 2) удаление элементов из коллекции `Queue` осуществляется по принципу "последним пришел - первым обслужен"
 - 3) если коллекция наследует интерфейс `IList`, то с коллекцией можно работать, используя индексатор
 - 4) коллекция класса `BitArray` хранит последовательность байтов (со значениями ноль или один), каждый из которых рассматривается как соответствующий бит с тем же значением
 - 5) все объекты, записываемые в коллекцию, автоматически приводятся к типу `System.Object`
9. Из приведенных ниже высказываний укажите все истинные высказывания:
- 1) доступ к элементам коллекции типа `ArrayList` может выполняться по индексу элемента
 - 2) значение свойства `Count` применительно к коллекции типа `Stack` соответствует количеству элементов в коллекции в данный момент
 - 3) значение свойства `Count` применительно к коллекции типа `Queue` доступно для записи и может быть установлено с помощью операции присваивания
 - 4) метод `ToArray()` применительно к коллекции типа `Stack` выполняет преобразование этой коллекции в коллекцию типа `ArrayList`
 - 5) коллекция типа `ListDictionary` не может хранить объекты классов, определяемых программистом
10. Из приведенных ниже высказываний укажите все истинные высказывания:
- 1) значение свойства `Capacity` применительно к коллекции типа `ArrayList` соответствует максимально возможному количеству элементов в коллекции
 - 2) метод `Clear()` применительно к коллекции типа `Hashtable` выполняет установку всех элементов коллекции в значение по умолчанию
 - 3) коллекция типа `Queue` поддерживает упорядоченность элементов в порядке их занесения в коллекцию
 - 4) коллекция типа `Stack` поддерживает упорядоченность элементов в порядке, обратном их занесения в коллекцию
 - 5) коллекция типа `ArrayList` может хранить объекты только типов значений

Типовые темы для курсовой работы

1. Моделирование пассажиропотока
2. Настольная игра.
3. Кадровый состав предприятия
4. Система ведения результатов успеваемости студентов
5. Подход к построению распределенной виртуальной машины
6. Исследование алгоритма поиска простого пути на графе
7. Приложение «Машина Тьюринга»
8. Уровень платформера
9. Игра «Морской бой»
10. Оценка знаний по дисциплине на основе коллекции ребусов
11. Визуализация алгоритмов сортировки массива
12. Автоматическое реферирование текста
13. Двоичное дерево поиска
14. Коллекция игр
15. Многофункциональный калькулятор с использованием графических средств
16. Приложение «Игра в экономику»
17. Приложение «Сортировщик фотографий»
18. Игра «Судоку»
19. Исследование алгебраических и вероятностных свойств криптографических примитивов
20. Игра «Стрелялки»
21. Игра «Жизнь»
22. Приложение «Формирование заказа клиентов»
23. Генерация (детерминистическая) больших простых с подбором разложения функции Эйлера
24. Приложение «Проверка орфографии»
25. Приложение «Планировщик помещений» /
26. Шахматы
27. Моделирование жизни в муравейнике
28. Создание ребусов по школьному предмету
29. Проектирование игры жанра Tower Defence. Ч1.

30. Проектирование тестирования знаний по дисциплине «Алгоритмизация и программирование»
31. Детская игра «Тамагочи»
32. Проектирование игры «Кости»
33. Тренажер для обучения печатания на клавиатуре
34. Графический редактор
35. Проектирование программы шифрования и дешифрования текста
36. Программа редактирования текстового файла
37. Построение диаграмм в различном виде
38. Пакет математических операций с матрицами
39. Приложение «Машина Поста»
40. Проектирование игры Tower Defence. Ч2.
41. Алгоритм Хаффмана
42. Приложение «Построение графиков функций»
43. Игра «Сапер»
44. Плеер
45. Шифр Цезаря и Виженера
46. Графический редактор

7.3 Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности

Все задания, используемые для текущего контроля формирования компетенций условно можно разделить на две группы:

1. задания, которые в силу своих особенностей могут быть реализованы только в процессе обучения на занятиях (например, дискуссия, круглый стол, диспут, мини-конференция);
2. задания, которые дополняют теоретические вопросы (практические задания, проблемно-аналитические задания, тест).

Выполнение всех заданий является необходимым для формирования и контроля знаний, умений и навыков. Поэтому, в случае невыполнения заданий в процессе обучения, их необходимо «отработать» до зачета (экзамена). Вид заданий, которые необходимо выполнить для ликвидации «задолженности» определяется в индивидуальном порядке, с учетом причин невыполнения.

1. Требование к теоретическому устному ответу

Оценка знаний предполагает дифференцированный подход к студенту, учет его индивидуальных способностей, степень усвоения и систематизации основных понятий и категорий по дисциплине. Кроме того, оценивается не только глубина знаний поставленных

вопросов, но и умение использовать в ответе практический материал. Оценивается культура речи, владение навыками ораторского искусства.

Критерии оценивания: последовательность, полнота, логичность изложения, анализ различных точек зрения, самостоятельное обобщение материала, использование профессиональных терминов, культура речи, навыки ораторского искусства. Изложение материала без фактических ошибок.

Оценка «отлично» ставится в случае, когда материал излагается исчерпывающе, последовательно, грамотно и логически стройно, при этом раскрываются не только основные понятия, но и анализируются точки зрения различных авторов. Обучающийся не затрудняется с ответом, соблюдает культуру речи.

Оценка «хорошо» ставится, если обучающийся твердо знает материал, грамотно и по существу излагает его, знает практическую базу, но при ответе на вопрос допускает несущественные погрешности.

Оценка «удовлетворительно» ставится, если обучающийся освоил только основной материал, но не знает отдельных деталей, допускает неточности, недостаточно правильные формулировки, нарушает последовательность в изложении материала, затрудняется с ответами, показывает отсутствие должной связи между анализом, аргументацией и выводами.

Оценка «неудовлетворительно» ставится, если обучающийся не отвечает на поставленные вопросы.

2. Творческие задания

Эссе – это небольшая по объему письменная работа, сочетающая свободные, субъективные рассуждения по определенной теме с элементами научного анализа. Текст должен быть легко читаем, но необходимо избегать нарочито разговорного стиля, сленга, шаблонных фраз. Объем эссе составляет примерно 2 – 2,5 стр. 12 шрифтом с одинарным интервалом (без учета титульного листа).

Критерии оценивания - оценка учитывает соблюдение жанровой специфики эссе, наличие логической структуры построения текста, наличие авторской позиции, ее научность и связь с современным пониманием вопроса, адекватность аргументов, стиль изложения, оформление работы. Следует помнить, что прямое заимствование (без оформления цитат) текста из Интернета или электронной библиотеки недопустимо.

Оценка «отлично» ставится в случае, когда определяется: наличие логической структуры построения текста (вступление с постановкой проблемы; основная часть, разделенная по основным идеям; заключение с выводами, полученными в результате рассуждения); наличие четко определенной личной позиции по теме эссе; адекватность аргументов при обосновании личной позиции, стиль изложения.

Оценка «хорошо» ставится, когда в целом определяется: наличие логической структуры построения текста (вступление с постановкой проблемы; основная часть, разделенная по основным идеям; заключение с выводами, полученными в результате рассуждения); но не прослеживается наличие четко определенной личной позиции по теме эссе; не достаточно аргументов при обосновании личной позиции.

Оценка «удовлетворительно» ставится, когда в целом определяется: наличие логической структуры построения текста (вступление с постановкой проблемы; основная часть, разделенная по основным идеям; заключение). Но не прослеживаются четкие выводы, нарушается стиль изложения.

Оценка «неудовлетворительно» ставится, если не выполнены никакие требования.

3. Требование к решению ситуационной, проблемной задачи (кейс-измерители)

Студент должен уметь выделить основные положения из текста задачи, которые требуют анализа и служат условиями решения. Исходя из поставленного вопроса в задаче,

попытаться максимально точно определить проблему и соответственно решить ее.

Задачи должны решаться студентами письменно. При решении задач также важно правильно сформулировать и записать вопросы, начиная с более общих и, кончая частными.

Критерии оценивания – оценка учитывает методы и средства, использованные при решении ситуационной, проблемной задачи.

Оценка «отлично» ставится в случае, когда обучающийся выполнил задание (решил задачу), используя в полном объеме теоретические знания и практические навыки, полученные в процессе обучения.

Оценка «хорошо» ставится, если обучающийся в целом выполнил все требования, но не совсем четко определяется опора на теоретические положения, изложенные в научной литературе по данному вопросу.

Оценка «удовлетворительно» ставится, если обучающийся показал положительные результаты в процессе решения задачи.

Оценка «неудовлетворительно» ставится, если обучающийся не выполнил все требования.

4. Интерактивные задания

Механизм проведения диспут-игры (ролевой (деловой) игры).

Необходимо разбиться на несколько команд, которые должны поочередно высказать свое мнение по каждому из заданных вопросов. Мнение высказывающейся команды засчитывается, если противоположная команда не опровергнет его контраргументами. Команда, чье мнение засчитано как верное (не получило убедительных контраргументов от противоположных команд), получает один балл. Команда, опровергнувшая мнение противоположной команды своими контраргументами, также получает один балл. Побеждает команда, получившая максимальное количество баллов.

Ролевая игра как правило имеет фабулу (ситуацию, казус), распределяются роли, подготовка осуществляется за 2-3 недели до проведения игры.

Критерии оценивания – оцениваются действия всех участников группы. Понимание проблемы, высказывания и действия полностью соответствуют заданным целям. Соответствие реальной действительности решений, выработанных в ходе игры. Владение терминологией, демонстрация владения учебным материалом по теме игры, владение методами аргументации, умение работать в группе (умение слушать, конструктивно вести беседу, убеждать, управлять временем, бесконфликтно общаться), достижение игровых целей, (соответствие роли – при ролевой игре). Ясность и стиль изложения.

Оценка «отлично» ставится в случае, выполнения всех критериев.

Оценка «хорошо» ставится, если обучающиеся в целом демонстрируют понимание проблемы, высказывания и действия полностью соответствуют заданным целям. Решения, выработанные в ходе игры, полностью соответствуют реальной действительности. Но некоторые объяснения не совсем аргументированы, нарушены нормы общения, нарушены временные рамки, нарушен стиль изложения.

Оценка «удовлетворительно» ставится, если обучающиеся в целом демонстрируют понимание проблемы, высказывания и действия в целом соответствуют заданным целям. Однако, решения, выработанные в ходе игры, не совсем соответствуют реальной действительности. Некоторые объяснения не совсем аргументированы, нарушены временные рамки, нарушен стиль изложения.

Оценка «неудовлетворительно» ставится, если обучающиеся не понимают проблему, их высказывания не соответствуют заданным целям.

5. Комплексное проблемно-аналитическое задание

Задание носит проблемно-аналитический характер и выполняется в три этапа. На первом из них необходимо ознакомиться со специальной литературой.

Целесообразно также повторить учебные материалы лекций и семинарских занятий

по темам, в рамках которых предлагается выполнение данного задания.

На втором этапе выполнения работы необходимо сформулировать проблему и изложить авторскую версию ее решения, на основе полученной на первом этапе информации.

Третий этап работы заключается в формулировке собственной точки зрения по проблеме. Результат третьего этапа оформляется в виде аналитической записки (объем: 2-2,5 стр.; 14 шрифт, 1,5 интервал).

Критерий оценивания- оценка учитывает: понимание проблемы, уровень раскрытия поставленной проблемы в плоскости теории изучаемой дисциплины, умение формулировать и аргументировано представлять собственную точку зрения, выполнение всех этапов работы.

Оценка «отлично» ставится в случае, когда обучающийся демонстрирует полное понимание проблемы, все требования, предъявляемые к заданию выполнены.

Оценка «хорошо» ставится, если обучающийся демонстрирует значительное понимание проблемы, все требования, предъявляемые к заданию выполнены.

Оценка «удовлетворительно» ставится, если обучающийся, демонстрирует частичное понимание проблемы, большинство требований, предъявляемых к заданию, выполнены

Оценка «неудовлетворительно» ставится, если обучающийся демонстрирует непонимание проблемы, многие требования, предъявляемые к заданию, не выполнены.

6. Исследовательский проект

Исследовательский проект – проект, структура которого приближена к формату научного исследования и содержит доказательство актуальности избранной темы, определение научной проблемы, предмета и объекта исследования, целей и задач, методов, источников, историографии, обобщение результатов, выводы.

Результаты выполнения исследовательского проекта оформляется в виде реферата (объем: 12-15 страниц; 14 шрифт, 1,5 интервал).

Критерии оценивания - поскольку структура исследовательского проекта максимально приближена к формату научного исследования, то при выставлении учитывается доказательство актуальности темы исследования, определение научной проблемы, объекта и предмета исследования, целей и задач, источников, методов исследования, выдвижение гипотезы, обобщение результатов и формулирование выводов, обозначение перспектив дальнейшего исследования.

Оценка «отлично» ставится в случае, когда обучающийся демонстрирует полное понимание проблемы, все требования, предъявляемые к заданию выполнены.

Оценка «хорошо» ставится, если обучающийся демонстрирует значительное понимание проблемы, все требования, предъявляемые к заданию выполнены.

Оценка «удовлетворительно» ставится, если обучающийся, демонстрирует частичное понимание проблемы, большинство требований, предъявляемых к заданию, выполнены

Оценка «неудовлетворительно» ставится, если обучающийся демонстрирует непонимание проблемы, многие требования, предъявляемые к заданию, не выполнены.

7. Информационный проект (презентация):

Информационный проект – проект, направленный на стимулирование учебно-познавательной деятельности студента с выраженной эвристической направленностью (поиск, отбор и систематизация информации об объекте, оформление ее для презентации). Итоговым продуктом проекта может быть письменный реферат, электронный реферат с иллюстрациями, слайд-шоу, мини-фильм, презентация и т.д.

Информационный проект отличается от исследовательского проекта, поскольку представляет собой такую форму учебно-познавательной деятельности, которая отличается ярко выраженной эвристической направленностью.

Критерии оценивания - при выставлении оценки учитывается самостоятельный поиск, отбор и систематизация информации, раскрытие вопроса (проблемы), ознакомление

студенческой аудитории с этой информацией (представление информации), ее анализ и обобщение, оформление, полные ответы на вопросы аудитории с примерами.

Оценка *«отлично»* ставится в случае, когда обучающийся полностью раскрывает вопрос (проблему), представляет информацию систематизировано, последовательно, логично, взаимосвязано, использует более 5 профессиональных терминов, широко использует информационные технологии, ошибки в информации отсутствуют, дает полные ответы на вопросы аудитории с примерами.

Оценка *«хорошо»* ставится, если обучающийся раскрывает вопрос (проблему), представляет информацию систематизировано, последовательно, логично, взаимосвязано, использует более 2 профессиональных терминов, достаточно использует информационные технологии, допускает не более 2 ошибок в изложении материала, дает полные или частично полные ответы на вопросы аудитории.

Оценка *«удовлетворительно»* ставится, если обучающийся, раскрывает вопрос (проблему) не полностью, представляет информацию не систематизировано и не совсем последовательно, использует 1-2 профессиональных термина, использует информационные технологии, допускает 3-4 ошибки в изложении материала, отвечает только на элементарные вопросы аудитории без пояснений.

Оценка *«неудовлетворительно»* ставится, если вопрос не раскрыт, представленная информация логически не связана, не используются профессиональные термины, допускает более 4 ошибок в изложении материала, не отвечает на вопросы аудитории.

8. Дискуссионные процедуры

Круглый стол, дискуссия, полемика, диспут, дебаты, мини-конференции являются средствами, позволяющими включить обучающихся в процесс обсуждения спорного вопроса, проблемы и оценить их умение аргументировать собственную точку зрения. Задание дается заранее, определяется круг вопросов для обсуждения, группы участников этого обсуждения.

Дискуссионные процедуры могут быть использованы для того, чтобы студенты:

– лучше поняли усвояемый материал на фоне разнообразных позиций и мнений, не обязательно достигая общего мнения;

– смогли постичь смысл изучаемого материала, который иногда чувствуют интуитивно, но не могут высказать вербально, четко и ясно, или конструировать новый смысл, новую позицию;

– смогли согласовать свою позицию или действия относительно обсуждаемой проблемы.

Критерии оценивания – оцениваются действия всех участников группы. Понимание проблемы, высказывания и действия полностью соответствуют заданным целям. Соответствие реальной действительности решений, выработанных в ходе игры. Владение терминологией, демонстрация владения учебным материалом по теме игры, владение методами аргументации, умение работать в группе (умение слушать, конструктивно вести беседу, убеждать, управлять временем, бесконфликтно общаться), достижение игровых целей, (соответствие роли – при ролевой игре). Ясность и стиль изложения.

Оценка *«отлично»* ставится в случае, когда все требования выполнены в полном объеме.

Оценка *«хорошо»* ставится, если обучающиеся в целом демонстрируют понимание проблемы, высказывания и действия полностью соответствуют заданным целям. Решения, выработанные в ходе игры, полностью соответствуют реальной действительности. Но некоторые объяснения не совсем аргументированы, нарушены нормы общения, нарушены временные рамки, нарушен стиль изложения.

Оценка *«удовлетворительно»* ставится, если обучающиеся в целом демонстрируют понимание проблемы, высказывания и действия в целом соответствуют заданным целям. Однако, решения, выработанные в ходе игры, не совсем соответствуют реальной

действительности. Некоторые объяснения не совсем аргументированы, нарушены временные рамки, нарушен стиль изложения.

Оценка «*неудовлетворительно*» ставится, если обучающиеся не понимают проблему, их высказывания не соответствуют заданным целям.

9. Тестирование

Является одним из средств контроля знаний, обучающихся по дисциплине.

Критерии оценивания – правильный ответ на вопрос.

Оценка «*отлично*» ставится в случае, если правильно выполнено 90-100% заданий.

Оценка «*хорошо*» ставится, если правильно выполнено 70-89% заданий.

Оценка «*удовлетворительно*» ставится в случае, если правильно выполнено 50-69% заданий.

Оценка «*неудовлетворительно*» ставится, если правильно выполнено менее 50% заданий.

10. Требование к письменному опросу (контрольной работе)

Оценивается не только глубина знаний поставленных вопросов, но и умение изложить письменно.

Критерии оценивания: последовательность, полнота, логичность изложения, анализ различных точек зрения, самостоятельное обобщение материала. Изложение материала без фактических ошибок.

Оценка «*отлично*» ставится в случае, когда соблюдены все критерии.

Оценка «*хорошо*» ставится, если обучающийся твердо знает материал, грамотно и по существу излагает его, знает практическую базу, но допускает несущественные погрешности.

Оценка «*удовлетворительно*» ставится, если обучающийся освоил только основной материал, но не знает отдельных деталей, допускает неточности, недостаточно правильные формулировки, нарушает последовательность в изложении материала, затрудняется с ответами, показывает отсутствие должной связи между анализом, аргументацией и выводами.

Оценка «*неудовлетворительно*» ставится, если обучающийся не отвечает на поставленные вопросы.

11. Требование к курсовой работе / курсовому проекту

Курсовая работа – одна из форм текущей аттестации знаний, полученных студентами при изучении дисциплины «Объектно-ориентированное программирование». Тематика курсовых работ утверждается кафедрой.

Курсовая работа содержит, как правило, теоретическую часть — изложение позиций и подходов, сложившихся в науке по данному вопросу, и аналитическую (практическую часть) — содержащую анализ проблемы на примере различных организаций России.

Курсовая работа в обязательном порядке включает: оглавление (содержание), введение, теоретический раздел, практический раздел, заключение, список литературы (не менее 10 источников, изданных преимущественно в течение последних 5 лет).

Объем курсовой работы - 25-30 страниц. Работа должна быть напечатана на компьютере в текстовом редакторе Microsoft Word, шрифт Times New Roman, кегль 14, межстрочный интервал - 1,5. Выравнивание - «по ширине».

Подробные требования к содержанию, объему, структуре, оформлению курсовой работы содержатся в «Методических указаниях по выполнению курсовой работы».

8. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины (модуля)

8.1. Основная учебная литература

1. Объектно-ориентированное программирование : лабораторный практикум / . — Ставрополь : Северо-Кавказский федеральный университет, 2018. — 111 с. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: <https://www.iprbookshop.ru/92712.html>

2. Объектно-ориентированное программирование на C++ : учебник / И.В. Баранова [и др.]. — Красноярск : Сибирский федеральный университет, 2019. — 288 с. — ISBN 978-5-7638-4034-6. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: <https://www.iprbookshop.ru/100067.html>

8.2. Дополнительная учебная литература:

1. Лебедева Т.Н. Теория и практика объектно-ориентированного программирования : учебное пособие / Лебедева Т.Н.. — Челябинск, Саратов : Южно-Уральский институт управления и экономики, Ай Пи Эр Медиа, 2019. — 221 с. — ISBN 978-5-4486-0663-2. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: <https://www.iprbookshop.ru/81498.html>

2. Литвиненко В.А. Основы объектно-ориентированного программирования задач на графах : учебное пособие / Литвиненко В.А.. — Ростов-на-Дону, Таганрог : Издательство Южного федерального университета, 2019. — 133 с. — ISBN 978-5-9275-3472-2. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: <https://www.iprbookshop.ru/107969.html>

3. Мейер Б. Объектно-ориентированное программирование и программная инженерия / Мейер Б.. — Москва : Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Эр Медиа, 2019. — 285 с. — ISBN 978-5-4486-0513-0. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: <https://www.iprbookshop.ru/79706.html>

8.3. Периодические издания

1. Журнал «Математическое моделирование и численные методы». [Математическое моделирование и численные методы \(bmstu.ru\)](http://mathnet.ru)

2. [Вестник Московского Университета. Математика, Механика \(msu.ru\)](http://msu.ru)

3. Дискретная математика. Discrete Mathematics and Applications. mathnet.ru

9. Перечень ресурсов информационно-телекоммуникационной сети "Интернет" (далее - сеть "Интернет"), необходимых для освоения дисциплины (модуля)

1. Федеральный портал «Российское образование». <http://www.edu.ru/>

2. Электронно-библиотечная система «Научная электронная библиотека eLIBRARY.RU» <https://www.elibrary.ru> /

3. Электронно-библиотечная система ЛАНЬ <https://e.lanbook.com/>

4. Электронно-библиотечная система IPR BOOKS <https://www.iprbookshop.ru>

5. <https://www.rsl.ru> - Российская Государственная Библиотека (ресурсы открытого доступа)

6. <https://link.springer.com> - Международная реферативная база данных научных изданий Springerlink (ресурсы открытого доступа)

7. <https://zbmath.org> - Международная реферативная база данных научных изданий zbMATH (ресурсы открытого доступа)

8. <https://openedu.ru> - «Национальная платформа открытого образования» (ресурсы открытого доступа)

10. Методические указания для обучающихся по освоению дисциплины (модуля)

Успешное освоение данного курса базируется на рациональном сочетании нескольких видов учебной деятельности – лекций, семинарских занятий, самостоятельной работы. При этом самостоятельную работу следует рассматривать одним из главных звеньев полноценного высшего образования, на которую отводится значительная часть учебного времени.

Самостоятельная работа студентов складывается из следующих составляющих:

1. работа с основной и дополнительной литературой, с материалами интернета и конспектами лекций;
2. внеаудиторная подготовка к контрольным работам, выполнение докладов, рефератов и курсовых работ;
3. выполнение самостоятельных практических работ;
4. подготовка к экзаменам (зачетам) непосредственно перед ними.

Для правильной организации работы необходимо учитывать порядок изучения разделов курса, находящихся в строгой логической последовательности. Поэтому хорошее усвоение одной части дисциплины является предпосылкой для успешного перехода к следующей. Задания, проблемные вопросы, предложенные для изучения дисциплины, в том числе и для самостоятельного выполнения, носят междисциплинарный характер и базируются, прежде всего, на причинно-следственных связях между компонентами окружающего нас мира. В течение семестра, необходимо подготовить рефераты (проекты) с использованием рекомендуемой основной и дополнительной литературы и сдать рефераты для проверки преподавателю. Важным составляющим в изучении данного курса является решение ситуационных задач и работа над проблемно-аналитическими заданиями, что предполагает знание соответствующей научной терминологии и т.д.

Для лучшего запоминания материала целесообразно использовать индивидуальные особенности и разные виды памяти: зрительную, слуховую, ассоциативную. Успешному запоминанию также способствует приведение ярких свидетельств и наглядных примеров. Учебный материал должен постоянно повторяться и закрепляться.

При выполнении докладов, творческих, информационных, исследовательских проектов особое внимание следует обращать на подбор источников информации и методику работы с ними.

Для успешной сдачи экзамена (зачета) рекомендуется соблюдать следующие правила:

1. Подготовка к экзамену (зачету) должна проводиться систематически, в течение всего семестра.
2. Интенсивная подготовка должна начаться не позднее, чем за месяц до экзамена.
3. Время непосредственно перед экзаменом (зачетом) лучше использовать таким образом, чтобы оставить последний день свободным для повторения курса в целом, для систематизации материала и доработки отдельных вопросов.

На экзамене высокую оценку получают студенты, использующие данные, полученные в процессе выполнения самостоятельных работ, а также использующие собственные выводы на основе изученного материала.

Учитывая значительный объем теоретического материала, студентам рекомендуется регулярное посещение и подробное конспектирование лекций.

11. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине (модулю), включая перечень программного обеспечения и информационных справочных систем (при необходимости)

1. Microsoft Windows Server;
2. Семейство ОС Microsoft Windows;
3. Libre Office свободно распространяемый офисный пакет с открытым исходным кодом;

4. Информационно-справочная система: Система КонсультантПлюс (КонсультантПлюс);
5. Информационно-правовое обеспечение Гарант: Электронный периодический справочник «Система ГАРАНТ» (Система ГАРАНТ);

Перечень используемого программного обеспечения указан в п.12 данной рабочей программы дисциплины.

12. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине (модулю)

12.1. Учебная аудитория для проведения учебных занятий, предусмотренных программой бакалавриата, оснащенная оборудованием и техническими средствами обучения.

Специализированная мебель:

Комплект учебной мебели (стол, стул) по количеству обучающихся; комплект мебели для преподавателя; доска (маркерная).

Технические средства обучения:

Компьютер в сборе для преподавателя; компьютеры в сборе для обучающихся; наушники; телевизор.

Перечень лицензионного программного обеспечения, в том числе отечественного производства и свободно распространяемого программного обеспечения:

Windows Server 2016, Windows 10, Microsoft Office, КонсультантПлюс, Система ГАРАНТ, Kaspersky Endpoint Security, Microsoft Windows Server, Microsoft Project, Spider Project, EclipseIDEforJavaEEDevelopers, AndroidStudio, IntelliJIDEA, Adobe Acrobat Reader DC, Google Chrome, LibreOffice, Skype, Gimp, Paint.net, AnyLogic, Inkscape, Microsoft Visual Studio Community, Denver, GNU Octave, PostgreSQL, Ramus.

Подключение к сети «Интернет» и обеспечение доступа в электронную информационно-образовательную среду ММУ.

12.2. Помещение для самостоятельной работы обучающихся.

Специализированная мебель:

Комплект учебной мебели (стол, стул) по количеству обучающихся; комплект мебели для преподавателя; доска (маркерная).

Технические средства обучения:

Компьютер в сборе для преподавателя; компьютеры в сборе для обучающихся; колонки; проектор, экран.

Перечень лицензионного программного обеспечения, в том числе отечественного производства:

Windows Server 2016, Windows 10, Microsoft Office, КонсультантПлюс, Система ГАРАНТ, Kaspersky Endpoint Security.

Перечень свободно распространяемого программного обеспечения:

Adobe Acrobat Reader DC, Google Chrome, LibreOffice, Skype, Zoom, Gimp, Paint.net, AnyLogic, Inkscape.

Помещение для самостоятельной работы обучающихся оснащено компьютерной техникой с возможностью подключения к сети "Интернет" и обеспечением доступа в электронную информационно-образовательную среду ММУ.

13. Образовательные технологии, используемые при освоении дисциплины

Для освоения дисциплины используются как традиционные формы занятий – лекции (типы лекций – установочная, вводная, текущая, заключительная, обзорная; виды лекций – проблемная, визуальная, лекция конференция, лекция консультация); и семинарские (практические) занятия, так и активные и интерактивные формы занятий - деловые и ролевые игры, решение ситуационных задач и разбор конкретных ситуаций.

На учебных занятиях используются технические средства обучения мультимедийной аудитории: компьютер, монитор, колонки, настенный экран, проектор, микрофон, пакет программ Microsoft Office для демонстрации презентаций и медиафайлов, видеопроектор для демонстрации слайдов, видеосюжетов и др. Тестирование обучаемых может осуществляться с использованием компьютерного оборудования университета.

13.1. В освоении учебной дисциплины используются следующие традиционные образовательные технологии:

- чтение проблемно-информационных лекций с использованием доски и видеоматериалов;
- семинарские занятия для обсуждения, дискуссий и обмена мнениями;
- контрольные опросы;
- консультации;
- самостоятельная работа студентов с учебной литературой и первоисточниками;
- подготовка и обсуждение рефератов (проектов), презентаций (научно-исследовательская работа);
- тестирование по основным темам дисциплины.

13.2. Активные и интерактивные методы и формы обучения

Из перечня видов: (*«мозговой штурм», анализ НПА, анализ проблемных ситуаций, анализ конкретных ситуаций, инциденты, имитация коллективной профессиональной деятельности, разыгрывание ролей, творческая работа, связанная с освоением дисциплины, ролевая игра, круглый стол, диспут, беседа, дискуссия, мини-конференция и др.*) используются следующие:

- диспут
- анализ проблемных, творческих заданий, ситуационных задач
- ролевая игра;
- круглый стол;
- мини-конференция
- дискуссия
- беседа.

13.3. Особенности обучения инвалидов и лиц с ограниченными возможностями здоровья (ОВЗ)

При организации обучения по дисциплине учитываются особенности организации взаимодействия с инвалидами и лицами с ограниченными возможностями здоровья (далее – инвалиды и лица с ОВЗ) с целью обеспечения их прав. При обучении учитываются особенности их психофизического развития, индивидуальные возможности и при необходимости обеспечивается коррекция нарушений развития и социальная адаптация указанных лиц.

Выбор методов обучения определяется содержанием обучения, уровнем методического и материально-технического обеспечения, особенностями восприятия учебной информации студентов-инвалидов и студентов с ограниченными возможностями здоровья и т.д. В образовательном процессе используются социально-активные и рефлексивные методы обучения, технологии социокультурной реабилитации с целью оказания помощи в установлении полноценных межличностных отношений с другими студентами, создании комфортного психологического климата в студенческой группе.

При обучении лиц с ограниченными возможностями здоровья электронное обучение и дистанционные образовательные технологии предусматривают возможность приема-передачи информации в доступных для них формах.

Обучающиеся из числа лиц с ограниченными возможностями здоровья обеспечены печатными и электронными образовательными ресурсами в формах, адаптированных к ограничениям их здоровья.