

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Нигматуллина Танзиля Алтафовна
Должность: Директор
Дата подписания: 14.02.2024 08:11:41
Уникальный программный ключ:
72a47dccbea51ad439ebc42366ed030bf219f69a



**Образовательное учреждение профсоюзов
высшего образования
«АКАДЕМИЯ ТРУДА И СОЦИАЛЬНЫХ
ОТНОШЕНИЙ»**



**БАШКИРСКИЙ ИНСТИТУТ СОЦИАЛЬНЫХ
ТЕХНОЛОГИЙ (филиал)**

**РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ
Б1.В.13 ОБЪЕКТИВНО-ОРИЕНТИРОВАННЫЕ
ПРОГРАММИРОВАНИЕ**

Направление подготовки
09.03.03 Прикладная информатика
(код и наименование направления подготовки)

Профиль (программа) подготовки
Прикладная информатика
(направленность (профиль) (уровень бакалавриата))

Квалификация выпускника
Бакалавр

1. Целью дисциплины является:

- развитие компетенций:

ПК-2. Способен разрабатывать проекты и адаптировать прикладное программное обеспечение

Индикаторы достижения:

ПК-2.1 Проводит анализ рынка программно-технических средств и осуществляет выбор средств для разработки информационных систем

ПК-2.2 Контролирует разработку алгоритмов и проектов компьютерных программ, пригодных для практического применения.

ПК-2.3 Выполняет разработку проектов компьютерных программ, пригодных для практического применения

2. Трудоемкость учебной дисциплины зафиксирована учебным планом соответствующей основной профессиональной образовательной программы, выражается в зачетных единицах. Одна зачетная единица равна 36 академическим часам продолжительностью 45 минут (27 астрономическим часам по 60 минут) и включает часы контактной работы и часы самостоятельной работы студента, в том числе часы, отводимые на процедуры контроля и подготовку к ним.

3. Результаты освоения образовательной программы:

В результате изучения дисциплины обучающийся должен:

Знать:

- основные виды прикладного программного обеспечения;
- средства повышения эффективности применения прикладного программного обеспечения;
- уровни применения прикладного программного обеспечения, определенные международными стандартами;

Уметь:

- определять средства прикладных систем, обеспечивающие повышение эффективности применения данных систем;
- применять прикладное обеспечение на различных уровнях;

Владеть:

- средствами повышения эффективности применения прикладного программного обеспечения.

4. Место дисциплины (модуля) в структуре образовательной программы:

Дисциплина «Объективно-ориентированные программирование» относится к вариативной части учебного плана.

5. Содержание дисциплины

№	Наименование раздела дисциплины	Содержание раздела
1.	Тема 1. Основные идеи ООП.	Причины появления и актуальность объектно-ориентированного программирования (ООП) как направления современной теории и практики создания программных систем. Основные области применения ООП: разработка программ, создание баз данных, построение прикладных систем. Основные преимущества применения ООП при создании программных систем. Предмет дисциплины, краткая характеристика содержания курса, место дисциплины в ряду дисциплин учебного плана. Факторы, обусловившие появление и содержание концепции ООП. Основные идеи ООП: использование объекта в качестве основной компоненты программы и децентрализация управления, реализуемое представлением программы как описания взаимодействия объектов. Структура концепции ООП.
2.	Тема 2. Основные модели ООП.	Объект как совокупность данных и набора операций. Семантика объекта: данные как отражение состояния объекта и методы как средства обеспечения взаимодействия объектов. Представление данных. Классификация методов: конструкторы, деструкторы, селекторы и модификаторы. Примеры. Объекты ООП и объекты реального мира. Наглядное представление объектов. Классы объектов: назначение и семантика. Класс как абстракция совокупности объектов. Классы и абстрактные типы данных. Инкапсуляция: назначение, предмет, прагматика. Инкапсуляция и области видимости. Защита на уровне объекта и на уровне класса. Наглядное представление классов. Объекты как экземпляры классов. Необходимость определения состояния. Основные действия с объектами: создание, инициализация, использование, разрушение (уничтожение). Отношение наследования для классов. Реализация спецификации и обобщения свойств объектов. Простое и множественное наследование. Сценарии наследования. Иерархия классов. Наглядное представление классов, связанных отношением наследования. Модель вычислений: управление объектами с помощью передачи сообщений. Программа как описание взаимодействия объектов. Полиморфизм: назначение и семантика. Виды и реализации полиморфизма: чистый, включения, параметрический, специализации. Формулировка концепции ООП. Место и роль ООП в теории и практике разработки программных систем.
3.	Тема 3. Реализация концепции ООП в языках программирования	Основные свойства языка ООП. Классификация и номенклатура языков, сферы применения. Место языка VB во внедрении и развитии средств ООП. Метафора объектно-ориентированного программирования, реализуемая языком VB. Значение языка VB для профессиональной подготовки программистов.
4.	Тема 4. Классы объектов.	Синтаксическая структура определения класса. Члены-данные и члены-функции. Клиенты класса. Способы задания доступа к членам класса. Члены-данные: назначение и основные свойства. Синтаксис и семантика определений членов-данных. Использование

		<p>предопределенных и пользовательских типов, массивов и указателей. Примеры. Члены-функции: назначение и основные свойства. Доступ членов-функций к членам класса. Функции-селекторы и функции-модификаторы. Примеры. Определение и объявление членов-функций в классе. Параметры членов-функций. Способы передачи значений в тело и из тела функции. Встроенные функции: определение, семантика, использование. Определение членов-функций вне класса: синтаксис и семантика. Рекомендации по использованию. Инкапсуляция данных и методов в классах языка VB. Способы задание класса в программе. Проект класса: назначение и использование. Файловые структуры задания класса. Тестирование класса.</p>
5.	<p>Тема 5. Члены класса, общие для всех экземпляров.</p>	<p>Назначение членов класса, общих для всех объектов класса. Примеры. Реализация с помощью статических членов-данных и статических членов-функций. Рекомендации по использованию в программах. Примеры.</p>
6.	<p>Тема 6. Создание объектов класса.</p>	<p>Конструкторы класса как средства создания объектов. Возможность наличия нескольких конструкторов у одного класса. Создание объектов на основе различной априорной информации: конструкторы со значениями параметров, задаваемыми по умолчанию, конструкторы копирования, конструкторы преобразования. Конструкторы, создаваемые компилятором по умолчанию: виды, способы доступа и назначение. Явное описание (определение или объявление) конструктора. Параметры конструктора, их спецификация и возможности использования. Программирование алгоритма конструктора. Размещение описания в определении класса и влияние размещения на способы создания объектов. Конструкторы со значениями параметров, задаваемыми по умолчанию. Конструкторы копирования: назначение, номенклатура, способы задания. Конструкторы преобразования: назначение, номенклатура, способы задания. Вызов конструктора: назначение, способы реализации, задание аргументов. Явный и автоматический вызовы: ситуации использования, рекомендации по применению. Возможности конструкторов для реализации моделей ООП. Использование конструкторов для инициализации объектов и преобразования значений объектов различных классов. Отладка и тестирование конструкторов класса.</p>
7.	<p>Тема 7. Проектирование конструкторов.</p>	<p>Ситуации, в которых необходимо явное задание конструкторов. Информация, требуемая для разработки конструктора: структура объекта, условия его создания и использования в программе. Необходимость проектирования согласованной системы конструкторов. Методика и пример проектирования. Отладка и тестирование системы конструкторов.</p>
8.	<p>Тема 8. Разрушение (уничтожение) объектов.</p>	<p>Деструктор как средство разрушения и уничтожения объекта: основные свойства. Деструктор, создаваемый компилятором по умолчанию: семантика и способ доступа. Явное задание деструктора. Программирование тела: использование членов-данных и членов функций класса. Размещение описания в определении класса и влияние размещения на способы уничтожения объектов. Размещение описания вне класса. Вызов деструктора: ситуации автоматического вызова. Исполнение тела: вызов деструкторов членов-данных. Отладка и тестирование деструкторов.</p>

9.	<p>Тема 9. Конструирование класса объектов.</p>	<p>Необходимость конструирования класса. Принципы построения класса и их реализация. Примеры. Этапы построения класса. Определение состава членов класса и их свойств, способов доступа к членам класса. Проектирование класса по контракту. Контракт: назначение, содержание, прагматика. Инварианты класса, пред- и постусловия для членов-функций. Определение инвариантов и формулирование пред- и постусловий. Проектирование согласованной системы конструкторов и деструкторов класса: ситуации, требующие проектирования, определение состава системы. Реализация системы. Использование в классе переопределений операторов и функций. Определение структуры набора файлов, задающих определение класса. Методика тестирования класса на основе контракта.</p>
10.	<p>Тема 10. Классы, связанные отношением наследования.</p>	<p>Отношение наследования: способ задания в языке VB. Базовый и производный классы. Вложение моделей классов и областей видимости. Реализация сценариев наследования, предусмотренных моделями ООП. Соотношение наследования класса и включения в производный класс объекта этого же класса. Пример. Клиенты базового класса. Управление доступом к членам базового класса: влияние спецификаторов доступа, использованных при установлении отношения наследования. Построение производного класса специализацией базового. Построение базового класса обобщением свойств набора классов. Иерархия классов, определяемая отношением наследования. Конструирование иерархий классов. Принципы построения, рекомендации, примеры. Конструкторы и деструкторы производного и базового классов: определение, вызов. Виртуальные функции: назначение, определение, семантика. Вызов виртуальных функций. Программирование виртуальных функций. Рекомендации по использованию. Абстрактные классы: назначение, построение, использование. Виртуальные базовые классы: назначение, определение, семантика, применение. Множественное наследование как способ представления объекта с нескольких точек зрения. Определение отношения множественного наследования: синтаксис и семантика. Практическое использование множественного наследования. Виртуальные функции при множественном наследовании. Тестирование систем классов, связанных отношением наследования.</p>
11.	<p>Тема 11. Разработка объектно- ориентированных программ.</p>	<p>Структура объектно-ориентированной программы: разделение описаний (проектов) классов и определений членов-функций. Области видимости и время существования переменных. Локальные, статические и глобальные переменные. Организация файловой структуры программы для эффективной разработки, отладки и модификации. Методика тестирования объектно-ориентированных программ. Преимущества и недостатки объектно-ориентированного программирования, выявленные практикой разработки программных систем. Основные тенденции и направления развития объектно-ориентированных технологий создания программ в условиях распределения процесса разработки в пространстве и во времени и необходимости обеспечения качества программных продуктов.</p>

Рекомендуемая тематика учебных занятий в форме контактной работы:

Рекомендуемая тематика учебных занятий лекционного типа (предусматривающих преимущественную передачу учебной информации преподавателями):

Тема 1. Основные идеи ООП.

Тема 2. Основные модели ООП.

Тема 3. Реализация концепции ООП в языках программирования.

Тема 4. Классы объектов.

Тема 5. Члены класса, общие для всех экземпляров.

Тема 6. Создание объектов класса.

Тема 7. Проектирование конструкторов.

Тема 8. Разрушение (уничтожение) объектов.

Тема 9. Конструирование класса объектов.

Тема 10. Классы, связанные отношением наследования.

Тема 11. Разработка объектно-ориентированных программ.

Рекомендуемая тематика учебных занятий семинарского типа

(семинары, практические занятия, коллоквиумы и иные аналогичные занятия)

Тема 1: Основные идеи ООП.

Вопросы для обсуждения:

1. Причины появления и актуальность объектно-ориентированного программирования (ООП) как направления современной теории и практики создания программных систем.
2. Основные области применения ООП: разработка программ, создание баз данных, построение прикладных систем.
3. Основные преимущества применения ООП при создании программных систем.

Тема 2: Основные модели ООП.

Вопросы для обсуждения:

1. Объект как совокупность данных и набора операций.
2. Семантика объекта: данные как отражение состояния объекта и методы как средства обеспечения взаимодействия объектов.
3. Классификация методов: конструкторы, деструкторы, селекторы и модификаторы.
4. Объекты ООП и объекты реального мира.
5. Наглядное представление объектов.
6. Классы объектов: назначение и семантика.
7. Инкапсуляция и области видимости.

Тема 3: Реализация концепции ООП в языках программирования.

Вопросы для обсуждения:

1. Основные свойства языка ООП.
2. Классификация и номенклатура языков, сферы применения.
3. Место языка VB во внедрении и развитии средств ООП.
4. Метафора объектно-ориентированного программирования, реализуемая языком VB.
5. Значение языка VB для профессиональной подготовки программистов.

Тема 4: Классы объектов.

Вопросы для обсуждения:

1. Синтаксическая структура определения класса.
2. Члены-данные и члены-функции.
3. Клиенты класса.
4. Синтаксис и семантика определений членов-данных.
5. Использование предопределенных и пользовательских типов, массивов и указателей.

Тема 5: Члены класса, общие для всех экземпляров.

Вопросы для обсуждения:

1. Назначение членов класса, общих для всех объектов класса.
2. Примеры членов класса.
3. Реализация с помощью статических членов-данных и статических членов- функций.

Тема 6: Создание объектов класса.

Вопросы для обсуждения:

1. Конструкторы класса как средства создания объектов.
2. Создание объектов на основе различной априорной информации: конструкторы со значениями параметров, задаваемыми по умолчанию, конструкторы копирования, конструкторы преобразования.
3. Конструкторы, создаваемые компилятором по умолчанию: виды, способы доступа и назначение.
4. Параметры конструктора, их спецификация и возможности использования.
5. Программирование алгоритма конструктора.

Тема 7: Проектирование конструкторов.

Вопросы для обсуждения:

1. Ситуации, в которых необходимо явное задание конструкторов.
2. Информация, требуемая для разработки конструктора: структура объекта, условия его создания и использования в программе.
3. Необходимость проектирования согласованной системы конструкторов.

Тема 8: Разрушение (уничтожение) объектов.

Вопросы для обсуждения:

1. Деструктор как средство разрушения и уничтожения объекта: основные свойства.
2. Деструктор, создаваемый компилятором по умолчанию: семантика и способ доступа.
3. Явное задание деструктора

Тема 9: Конструирование класса объектов.

Вопросы для обсуждения:

1. Принципы построения класса и их реализация.
2. Этапы построения класса.
3. Определение состава членов класса и их свойств, способов доступа к членам класса.
4. Проектирование класса по контракту. Контракт: назначение, содержание, прагматика.
5. Инварианты класса, пред- и постусловия для членов-функций.
6. Определение инвариантов и формулирование пред- и постусловий.

Тема 10: Классы, связанные отношением наследования.

Вопросы для обсуждения:

1. Отношение наследования: способ задания в языке VB.
2. Базовый и производный классы.
3. Вложение моделей классов и областей видимости.
4. Реализация сценариев наследования, предусмотренных моделями ООП.
5. Соотношение наследования класса и включения в производный класс объекта этого же класса.

Тема 11: Разработка объектно- ориентированных программ.

Вопросы для обсуждения:

1. Структура объектно-ориентированной программы: разделение описаний (проектов) классов и определений членов-функций.
2. Области видимости и время существования переменных. Локальные, статические и глобальные переменные.
3. Организация файловой структуры программы для эффективной разработки, отладки и модификации.
4. Методика тестирования объектно- ориентированных программ.
5. Преимущества и недостатки объектно-ориентированного программирования.

Требования к самостоятельной работе студентов по освоению дисциплины

Самостоятельная работа студентов при изучении дисциплины «Объективно-ориентированное программирование» направлена на решение

следующих задач:

- формирование представлений об общей методологии и средствах технологии объектно-ориентированного программирования;
- углубленная подготовка студентов в области применения технологии объектно-ориентированного программирования.

Результаты самостоятельной работы контролируются преподавателем и учитываются при текущей аттестации студента. При этом проводятся: тестирование, экспресс-опрос на практических занятиях, заслушивание докладов, проверка письменных работ и т.д. Несомненно, умение анализировать юридические источники, работать с литературой, навыки поиска, обработки и оформления необходимой информации, способность обосновывать собственную позицию помогут студенту в дальнейшей самостоятельной учебной и научной работе.

Исходя из рамок, установленных учебным планом по трудоемкости и видам учебной работы по дисциплине, преподаватель самостоятельно выбирает тематику занятий по формам и количеству часов проведения контактной работы: лекции и иные учебные занятия, предусматривающие преимущественную передачу учебной информации преподавателем и (или) занятия семинарского типа (семинары, практические занятия, практикумы, лабораторные работы, коллоквиумы и иные аналогичные занятия), и (или) групповые консультации, и (или) индивидуальную работу обучающихся с преподавателем, в том числе индивидуальные консультации (по курсовым работам/проектам – при наличии курсовой работы/проекта по данной дисциплине в учебном плане).

Рекомендуемая тематика занятий максимально полно реализуется в контактной работе со студентами очной формы обучения. В случае реализации образовательной программы в заочной / очно-заочной форме трудоемкость дисциплины сохраняется, однако объем учебного материала в значительной части осваивается студентами в форме самостоятельной работы. При этом требования к ожидаемым образовательным результатам студентов по данной дисциплине не зависят от формы реализации образовательной программы.

Примерная тематика рефератов для самостоятельных работ

1. Моделирование размещения массивов.
2. Моделирование динамического распределения памяти.
3. Реализация разреженных матриц и работа с ними.
4. Методы внутренней сортировки.
5. Методы внешней сортировки.
6. Методы кодирования информации с целью устранения избыточности текста.
7. Методы кодирования информации с целью повышения надежности передаваемых данных.

8. Методы кодирования информации с целью повышения секретности передаваемых данных.
9. Бинарные деревья. Операции над ними.
10. Исследование алгоритма балансировки АВЛ - дерева.
11. Реализация Б-деревьев.
12. Создание объектно-ориентированной библиотеки для работы с бинарными деревьями.
13. Реализация графов и операции над ними

Руководствуясь положениями статьи 47 и статьи 48 Федерального закона от 29 декабря 2012 г. N 273-ФЗ "Об образовании в Российской Федерации" научно-педагогические работники и иные лица, привлекаемые университетом к реализации данной образовательной программы, пользуются предоставленными академическими правами и свободами в части свободы преподавания, свободы от вмешательства в профессиональную деятельность; свободы выбора и использования педагогически обоснованных форм, средств, методов обучения и воспитания; права на творческую инициативу, разработку и применение авторских программ и методов обучения и воспитания в пределах реализуемой образовательной программы и отдельной дисциплины.

Исходя из рамок, установленных учебным планом по трудоемкости и видам учебной работы по дисциплине, преподаватель самостоятельно выбирает тематику занятий по формам и количеству часов проведения контактной работы: лекции и иные учебные занятия, предусматривающие преимущественную передачу учебной информации преподавателем и (или) занятия семинарского типа (семинары, практические занятия, практикумы, лабораторные работы, коллоквиумы и иные аналогичные занятия), и (или) групповые консультации, и (или) индивидуальную работу обучающихся с преподавателем, в том числе индивидуальные консультации (по курсовым работам/проектам – при наличии курсовой работы/проекта по данной дисциплине в учебном плане).

Рекомендуемая тематика занятий максимально полно реализуется в контактной работе со студентами очной формы обучения. В случае реализации образовательной программы в заочной / очно-заочной форме трудоемкость дисциплины сохраняется, однако объем учебного материала в значительной части осваивается студентами в форме самостоятельной работы. При этом требования к ожидаемым образовательным результатам студентов по данной дисциплине не зависят от формы реализации образовательной программы.

6. Виды учебной работы по дисциплине зафиксированы учебным планом основной профессиональной образовательной программы по

указанному направлению и профилю, выражаются в академических часах.

Часы контактной работы и самостоятельной работы студента и часы, отводимые на процедуры контроля, могут различаться в учебных планах ОПОП по формам обучения.

Объем контактной работы включает часы контактной аудиторной работы (лекции/практические занятия/ лабораторные работы), контактной внеаудиторной работы (контроль самостоятельной работы), часы контактной работы в период аттестации. Контактная работа, в том числе может проводиться посредством электронной информационно- образовательной среды института с использованием ресурсов сети Интернет и дистанционных технологий.

7. Учебно-методическое обеспечение дисциплины:

основная литература:

1. Объектно-ориентированное программирование : учебное пособие : в 3 частях : [16+] / П. П. Степанов, А. А. Кабанов, В. А. Никонов, Т. С. Павлюченко ; ред. К. В. Обухова ; Омский государственный технический университет. – Омск : Омский государственный технический университет (ОмГТУ), 2021. – Часть 1. – 112 с. : ил. – Режим доступа: по подписке. – URL: <https://biblioclub.ru/index.php?page=book&id=700657> – ISBN 978-5-8149-3301-0 (ч. 1). – ISBN 978-5-8149-3300-3. – Текст : электронный.

2. Зюзев, А. М. Объектно ориентированное программирование : учебно-методическое пособие / А. М. Зюзев, К. Е. Нестеров ; науч. ред. В. П. Метельков ; Уральский федеральный университет им. первого Президента России Б. Н. Ельцина. – Екатеринбург : Издательство Уральского университета, 2019. – 119 с. : ил. – Режим доступа: по подписке. – URL: <https://biblioclub.ru/index.php?page=book&id=697656> – Библиогр. в кн. – ISBN 978-5-7996-2634-1. – Текст : электронный.

3. Литвиненко, В. А. Основы объектно-ориентированного программирования задач на графах : учебное пособие : [16+] / В. А. Литвиненко ; Южный федеральный университет. – Ростов-на-Дону ; Таганрог : Южный федеральный университет, 2019. – 133 с. : ил., табл., схем. – Режим доступа: по подписке. – URL: <https://biblioclub.ru/index.php?page=book&id=612227> – Библиогр.: с. 99. – ISBN 978-5-9275-3472-2. – Текст : электронный.

дополнительная литература:

1. Окулов, С.М. Основы программирования: учебник / С. М. Окулов. - 4-е изд. - М. : Бинوم. Лаборатория знаний, 2008. - 440 с.

2. Гарбер, Г.З. Основы программирования на Visual Basic и VBA в Excel 2007 [Текст] : учебное пособие / Г. З. Гарбер. - М. : Солон-Пресс, 2008. - 192 с.

3. Объектно-ориентированное программирование в научных исследованиях : практикум : учебное пособие : [16+] / авт.-сост. В. П. Герасимов, В. Д. Ковалев. – Ставрополь : Северо-Кавказский Федеральный университет (СКФУ), 2018. – 119 с. : ил. – Режим доступа: по подписке. – URL: <https://biblioclub.ru/index.php?page=book&id=563230> – Библиогр.: с. 108. – Текст : электронный.

4. Зайцев, М. Г. Объектно-ориентированный анализ и программирование : учебное пособие : [16+] / М. Г. Зайцев ; Новосибирский государственный технический университет. – Новосибирск : Новосибирский государственный технический университет, 2017. – 84 с. : ил., табл. – Режим доступа: по подписке. – URL: <https://biblioclub.ru/index.php?page=book&id=576800> – Библиогр. в кн. – ISBN 978-5-7782-3308-9. – Текст : электронный.

5. Лебеденко, Л. Ф. Основы программирования на C++ : учебное пособие : [16+] / Л. Ф. Лебеденко, О. И. Моренкова. – 2-е изд., перераб. и доп. – Новосибирск : Сибирский государственный университет телекоммуникаций и информатики, 2021. – 200 с. : ил., табл., схем. – Режим доступа: по подписке. – URL: <https://biblioclub.ru/index.php?page=book&id=694769> – Библиогр. в кн. – Текст : электронный.

8. Перечень программного обеспечения, информационных справочных систем и интернет - ресурсов, необходимых для освоения дисциплины

Операционная система Windows XP Professional Service Pack 3: инв. №931, 932, 934, 936, 938, 940, 941, 942, 953: (Договор б\н от 29.03.2008) инв. №21747-217450, 21798, 21808: Лицензии № 42302228

Microsoft Office Professional Plus 2007 Лицензии № 42302228

Star Board Software (Договор Б/Н от 20.11.2008)

Антивирус: Kaspersky Endpoint Security 10 (Договор № 5337-ПАО/2015 от 30.09.2015 г.)

Доступ в интернет: Договор № РК 10091-08 от 31.12.2013

Справочная правовая система Консультант Плюс (Договор №3/4 от 01.02.2012 г.)

9. Требования к материально-техническому и учебно-методическому обеспечению дисциплины:

Для проведения занятий лекционного типа используются специальные помещения (учебные аудитории), оборудованные техническими средствами обучения – мультимедийной проекционной техникой.

Лекционные и практические занятия проводятся в аудиториях, оснащенных презентационным оборудованием (компьютер, имеющий выход в Интернет, мультимедийный проектор, экран, акустические системы), доской, рабочими учебными столами и стульями.

При необходимости занятия проводятся в компьютерных классах, оснащенных доской, экраном, рабочими учебными столами и стульями, персональными компьютерами, объединенными в локальные сети с выходом в Интернет, с установленным лицензионным программным обеспечением, с подключенным к ним периферийным устройством и оборудованием (мультимедийный проектор, акустическая система и пр.).

Для проведения занятий семинарского типа, курсового проектирования (выполнения курсовых работ), консультаций, текущего контроля и промежуточной аттестации достаточно специальных помещений (учебных аудиторий), оборудованных специализированной мебелью (для обучающихся) меловой / маркерной доской.

Для организации самостоятельной работы обучающимся предоставляются помещения, оснащенные компьютерной техникой с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду института.

10. Образовательные технологии

В процессе освоения дисциплины используются следующие образовательные технологии:

1. Стандартные методы обучения:

- лекции;
- практические занятия, на которых обсуждаются основные проблемы, освещенные в лекциях и сформулированные в домашних заданиях;
- письменные или устные домашние задания;
- самостоятельная работа студентов, в которую входит освоение теоретического материала, подготовка к практическим занятиям, выполнение указанных выше письменных/устных заданий, работа с литературой.

2. Методы обучения с применением интерактивных форм образовательных технологий:

- интерактивные лекции;
- компьютерные симуляции;

- анализ деловых ситуаций на основе кейс-метода и имитационных моделей;
- деловые и ролевые игры;
- круглые столы;
- групповые дискуссии и проекты

11. Требования к промежуточной аттестации и оценочные материалы для ее проведения

Промежуточная аттестация выполняется в форме экзамена, контрольной работы.

Оценочные материалы текущего контроля и промежуточной аттестации представлены в форме вопросов для зачета с оценкой. Примерные вопросы, задания, темы рефератов для проведения промежуточной аттестации по дисциплине и критерии оценивания представлены на сайте (<https://ufabist.ru/sveden/education/eduop/>)

В полном объеме оценочные материалы хранятся на кафедре, реализующей данную дисциплину.

Разработчик:

Преподаватель кафедры экономики и информационных технологий
Р.Ф. Ганеева