

МИНИСТЕРСТВО НАУКИ И ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение
высшего образования «Кабардино-Балкарский государственный университет им. Х.М.
Бербекова» (КБГУ)

ИНСТИТУТ ИНФОРМАТИКИ, ЭЛЕКТРОНИКИ И РОБОТОТЕХНИКИ
КАФЕДРА ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ

СОГЛАСОВАНО

УТВЕРЖДАЮ

Руководитель образовательной программы
_____ А.С. Ксенофонтов

Директор ИИЭР
_____ Н.В. Черкесова

«__» _____ 2020 г.

«__» _____ 2020 г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ
«Объектно-ориентированное программирование»

Направление подготовки
10.03.01 – Информационная безопасность
(код и наименование направления подготовки)

Профиль подготовки
«Организация и технология защиты информации »

Квалификация (степень) выпускника
бакалавр

Форма обучения
очная

Нальчик 2020

Рабочая программа дисциплины «Объектно-ориентированное программирование» /сост. Е.А. Акбашева – Нальчик: ФГБОУ КБГУ, 2020. – 26 с.

Рабочая программа предназначена для преподавания дисциплины по выбору студентам очной формы обучения по направлению подготовки 10.03.01 Информационная безопасность профиль «Организация и технология защиты информации » 6 семестра, 3 курса.

Рабочая программа составлена с учетом Федерального государственного образовательного стандарта высшего образования по направлению подготовки 10.03.01 Информационная безопасность (уровень бакалавриата), утвержденного приказом Министерства образования и науки Российской Федерации от «01» декабря 2016 г. № 1515.

Составитель _____ **Е.А. Акбашева**

Содержание

1. ЦЕЛЬ И ЗАДАЧИ ОСВОЕНИЯ ДИСЦИПЛИНЫ	4
2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОПОП ВО	4
3. ТРЕБОВАНИЯ К РЕЗУЛЬТАТАМ ОСВОЕНИЯ ДИСЦИПЛИНЫ.....	4
4. СОДЕРЖАНИЕ И СТРУКТУРА ДИСЦИПЛИНЫ	5
5. ОЦЕНОЧНЫЕ МАТЕРИАЛЫ ДЛЯ ТЕКУЩЕГО И РУБЕЖНОГО КОНТРОЛЯ УСПЕВАЕМОСТИ И ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ	12
6. МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ, ОПРЕДЕЛЯЮЩИЕ ПРОЦЕДУРЫ ОЦЕНИВАНИЯ ЗНАНИЙ, УМЕНИЙ, НАВЫКОВ И (ИЛИ) ОПЫТА ДЕЯТЕЛЬНОСТИ	20
7. УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ	22
8. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ.....	28

1. ЦЕЛЬ И ЗАДАЧИ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Целью изучения дисциплины «Объектно-ориентированное программирование» является теоретическая и практическая подготовка студентов в области информационных технологий в такой степени, чтобы они могли выбирать необходимые технические, алгоритмические, программные и технологические решения, уметь объяснить принципы их функционирования и правильно их использовать.

Задачи:

- формирование у студентов знаний по дисциплине, достаточных для самостоятельного освоения новых систем и языков программирования;
- выработка практических навыков написания программ на языках высокого уровня в современных средах программирования.

2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОПОП ВО

Дисциплина относится к факультативным дисциплинам.

Для изучения данной учебной дисциплины необходимы знания, умения и навыки, формируемые предшествующими дисциплинами:

- Языки и методы программирования;
- Практикум на ЭВМ;
- Практикум по программированию;
- Технологии программирования;
- Информатика;
- Базы данных.

3. ТРЕБОВАНИЯ К РЕЗУЛЬТАТАМ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Процесс изучения дисциплины направлен на формирование элементов следующих компетенций в соответствии с ФГОС ВО по данному направлению подготовки:

способностью применять программные средства системного, прикладного и специального назначения, инструментальные средства, языки и системы программирования для решения профессиональных задач (ПК-2);

способностью принимать участие в организации и проведении контрольных проверок работоспособности и эффективности применяемых программных, программно-аппаратных и технических средств защиты информации (ПК-6).

В результате освоения дисциплины обучающийся должен:

знать:

- основные теоретические положения структурного и объектно-ориентированного метода программирования и его свойства;
- состав, структуру и основные характеристики сред программирования, основанных на объектно-ориентированном методе;

уметь:

- выбирать наиболее оптимальную систему программирования исходя из постановки задачи, анализа аппаратной и программной платформы, а также финансовых ресурсов;

- иметь навыки по установке и настройке программного обеспечения одного из вариантов объектно-ориентированных сред программирования (например, Visual Studio).
- уверенно работать в визуальных средах программирования и разрабатывать с их помощью компьютерные программы для решения широкого класса задач;

владеть:

- Способностью овладеть основными методами, способами и средствами получения, хранения, переработки информации, иметь навыки работы с компьютером как средством управления информацией;
- Способностью использовать в познавательной и профессиональной деятельности навыки работы с информацией из различных источников;
- Способностью использовать в познавательной и профессиональной деятельности базовые знания в области информатики и современных информационных технологий, навыки использования программных средств и навыков работы в компьютерных сетях; умением создавать базы данных использовать ресурсы Интернет;
- способностью использовать базовые теоретические знания для решения профессиональных задач;
- способностью применять на практике базовые профессиональные навыки;

4. СОДЕРЖАНИЕ И СТРУКТУРА ДИСЦИПЛИНЫ

Таблица 1

Содержание дисциплины «Объектно-ориентированное программирование»

№ раздела	Наименование раздела	Содержание раздела	Код контролируемой компетенции (или ее части)	Форма текущего контроля
1	Теоретические основы объектно-ориентированного программирования	Основные подходы к разработке программного обеспечения. Понятие объектно-ориентированного программирования (ООП). Понятие объекта и класса. Основные принципы ООП. Абстрагирование. Инкапсуляция. Агрегирование. Наследование. Объектная модель программы. Универсальный язык моделирования UML.	ПК-2	ТК, К, Т
2	Основные понятия платформы Microsoft .Net	Основные идеи и компоненты платформы .NET Framework. Новый тип приложений – сборка (assembly). Метаданные. Промежуточный код	ПК-2	ТК, К, Т, ЛР

		(Intermediate Language). Единая среда выполнения (Common Language Runtime). Пространства имен (namespaces). Единая библиотека типов (классов, интерфейсов, структур) платформы - Microsoft Framework Library, основные пространства имен.		
3	Основные понятия языка программирования C#	Типы приложений. Общая структура программы. Пример простой программы. Использование командной строки для компиляции программы. Оператор using. Точка входа программы функция Main(). Статические методы. Базовые классы для консольных приложений. Класс Console. Класс Math. Класс Convert. Переменные методов. Типы данных. Стек. Куча. Система типов языка C#. Встроенные типы и преобразование типов. Константы.	ПК-2	ТК, К, Т, ЛР
4	Операции и операторы	Понятие операции. Приоритеты операций. Операция присваивания. Специальные варианты присваивания. Арифметические операции. Вычисление выражений. Операции отношения. Логические операции. Условная операция. Понятие оператора. Оператор присваивания. Операторы выбора (if, switch). Операторы перехода (goto, break, continue). Операторы цикла (for, while, foreach). Обработка исключений.	ПК-2	ТК, К, Т, ЛР
5	Работа с массивами	Типы массивов. Одномерные массивы. Многомерные массивы. Ступенчатые массивы. Массивы как коллекции.	ПК-2	ТК, К, Т, ЛР

		Методы класса Array. Создание и использование массивов.		
6	Коллекции и словари	Понятие коллекции – класса, предназначенного для хранения и доступа к упорядоченному списку объектов произвольного вида. Свойства и методы коллекций (Add, Remove, Item, Count). Создание коллекций объектов. Работа с объектами в коллекции - перечисление элементов коллекции (foreach). Нетипизированные коллекции. Типизированные коллекции.	ПК-2	ТК, К, Т, ЛР
7	Встроенный язык поисковых запросов LINQ	Язык Language INtegrated Query. Архитектура LINQ. Разновидности LINQ. Преимущества LINQ. Технология LINQ to Objects. Интерфейс IEnumerable<T>. Общий вид LINQ запроса. Операции LINQ запроса. Преобразование выборки в коллекции. Анонимные типы. LINQ и обобщенные коллекции.	ПК-6	ТК, К, Т, ЛР
8	Описание классов	Основные элементы классов. Режимы доступа. Поля класса. Методы класса. Тело метода. Вызов метода. Перегрузка методов. Конструкторы класса. Свойства класса. Автоматически реализуемые свойства. Инициализация объектов класса. Индексаторы. Статические поля и методы класса. Переопределение операций класса. Определение преобразования типов.	ПК-6	ТК, К, Т, ЛР
9	Делегаты и события классов	Новый тип данных – делегат (delegate). Объявление делегатов и создание экземпляров.	ПК-6	ТК, К, Т, ЛР

		Использование экземпляров делегатов. Взаимодействия объекта с вызывающей его программой с помощью событий. Примеры событий. Описание событий в классе. Объявление в программе объектов с событиями. Создание обработчиков событий (event handler).		
10	Описание отношений между классами	Отношение вложенности. Отношение наследования. Описание производных классов. Конструкторы производного класса. Добавление методов в производном классе. Абстрактные классы. Присвоение ссылок с учетом наследования. Полиморфизм.	ПК-6	ТК, К, Т, ЛР
11	Дополнительные пользовательские типы	Структуры. Перечисления. Интерфейсы. Две способа реализации интерфейса. Приведение к типу интерфейса. Встроенные интерфейсы.	ПК-6	ТК, К, Т, ЛР
12	Создание форм и работа с ними	Пространство имен System.Windows.Forms. Основной класс окон – Form. Отображение и закрытие форм. Элементы управления – controls. Работа с элементами управления. Основные элементы управления, Виды меню. Основные классы по работе с меню. Основное и контекстное меню. Работа с диалоговыми окнами. Понятие диалогового окна. Типы диалоговых окон. Классы общих диалогов. Использование общих диалогов. Создание собственных диалогов. Рисование в окне программы. Пространство имен System.Windows.Drawing.	ПК-6	ТК, К, Т, ЛР

		Класс Graphics. Основные классы для рисования – Pen, Brush, Color, Font. Внешние устройства – клавиатура, мышь, таймер. События внешних устройств. Обработка событий формы от внешних устройств.		
13	Работа с файлами	Файловая систем. Структура файла. Классы для работы с каталогами Directory и DirectoryInfo. Классы для работы с файлами File и FileInfo. Классы для работы с содержанием файлов FileStream, StreamReader и StreamWriter, BynaryReader и BynaryWriter.	ПК-6	ТК, К, Т, ЛР
14	Работа с базами данными	Назначение технологии ADO.Net. Реляционная модель данных. Язык работы с данными SQL. Основные операторы языка SQL (Select, Update, Delete). Понятие источника данных и его создание. Технологии ODBC и OLEDB. Присоединенная работа с базами данных. Провайдеры ADO.Net. Объектная модель провайдеров ADO.Net. Основные классы провайдеров. Класс Connection. Строка соединения с БД. Класс Command. Выполнение команд к базе данных с использованием объектов Command. Класс DataReader. Методы класса DataReader. Чтение, корректировка и добавление новых данных. Отсоединенная работа с БД. Класс DataAdapter. Класс DataSet. Заполнение данных в DataSet из базы данных. Классы DataTable и DataRow. Работа с данными объекта DataTable.	ПК-6	ТК, К, Т, ЛР

		Сохранение изменений DataSet в базе данных. Создание типизированных классов DataSet. Использование объектов типизированного класса DataSet. Выполнение LINQ запросов к объектам DataSet.		
--	--	--	--	--

Таблица 2

Структура дисциплины

Вид работы	Трудоемкость, часы
	6 семестр
Общая трудоемкость (в зачетных единицах)	3
Контактная работа (в часах):	30
Лекции (Л)	-
Практические занятия (ПЗ)	30
Семинарские занятия (СЗ)	—
Лабораторные работы (ЛР)	-
Самостоятельная работа (в часах):	78
Курсовой проект (КП), курсовая работа (КР)	—
Расчетно-графическое задание (РГЗ)	—
Реферат (Р)	—
Эссе (Э)	—
Самостоятельное изучение разделов	76
Контрольная работа (К)	—
Подготовка и прохождение промежуточной аттестации	—
Вид промежуточной аттестации	зачет

Лекции не предусмотрены

Практические занятия

Таблица 3

№ раз-дела	Наименование разделов
1	Теоретические основы объектно-ориентированного программирования
2	Основные понятия платформы Microsoft .Net
3	Основные понятия языка программирования C#
4	Операции и операторы
5	Работа с массивами
6	Коллекции и словари
7	Встроенный язык поисковых запросов LINQ
8	Описание классов
9	Делегаты и события классов
10	Описание отношений между классами

11	Дополнительные пользовательские типы
12	Создание форм и работа с ними
13	Работа с файлами
14	Работа с базами данными

Таблица 4

Самостоятельное изучение разделов дисциплины

№ раздела	Вопросы, выносимые на самостоятельное изучение
1	Обобщенные интерфейсы в C#.
2	Делегаты в C#.
3	События в C#.
4	Анонимные методы в C#.
5	Лямбда-выражения в C#.
6	Коллекции в C#.
7	Необобщенные коллекции в C#.
8	Обобщенные коллекции в C#.

5. ОЦЕНОЧНЫЕ МАТЕРИАЛЫ ДЛЯ ТЕКУЩЕГО И РУБЕЖНОГО КОНТРОЛЯ УСПЕВАЕМОСТИ И ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ

Конечными результатами освоения программы дисциплины являются сформированные когнитивные дескрипторы «знать», «уметь», «владеть», расписанные по отдельным компетенциям. Формирование этих дескрипторов происходит в течение всего семестра по этапам в рамках различного вида занятий и самостоятельной работы.

В ходе изучения дисциплины предусматриваются текущий, рубежный контроль и промежуточная аттестация.

5.1. Оценочные материалы для текущего контроля

Цель текущего контроля – оценка результатов работы в семестре и обеспечение своевременной обратной связи, для коррекции обучения, активизации самостоятельной работы обучающегося. Объектом текущего контроля являются конкретизированные результаты обучения (учебные достижения) по дисциплине.

Текущий контроль успеваемости обеспечивает оценивание хода освоения дисциплины «Объектно-ориентированное программирование» и включает: отчет по результатам выполнения лабораторных работ, самостоятельное выполнение заданий с отчетом (защитой) в установленный срок.

Оценка качества подготовки на основании выполненных заданий ведется преподавателем (с обсуждением результатов), баллы начисляются в зависимости от сложности задания.

5.1.1. Оценочные материалы для самостоятельной работы обучающегося

Перечень типовых заданий для самостоятельной работы сформирован в соответствии с тематикой лабораторных занятий по дисциплине «Объектно-ориентированное программирование».

Образцы заданий для самостоятельной работы (контролируемые компетенции ПК-2, ПК-6)

Задание 1

Составить на любом языке программирования консольное приложение, которое содержит описание класса **Time** (время), который должен содержать:

Класс должен включать:

- Закрытые свойства для хранения часов и минут
- Методы доступа к закрытым свойствам
- Конструктор или несколько конструкторов, для создания экземпляров класса
- Метод отображения на экране времени в формате (чч:мм)

Программа должна делать следующее:

1. В функции main() нужно объявить и создать массив из 3 объектов описанного класса
2. Задать им следующие значения (2ч 30м, 5ч 15м, 3ч 45м)
3. Вывести на экран время, хранящееся во всех объектах.
4. Рассчитать разницу в днях между 1 и 2 объектами и вывести ее на экран.

Задание 2

Составить на любом языке программирования консольную программу, которая содержит описание класса

Date - дата (год, месяц, день)

Класс должен включать:

- Закрытые свойства для хранения год, месяц, день.
 - Методы доступа к закрытым свойствам.
 - Конструктор или несколько конструкторов, для создания объектов класса.
 - Метод - показать на экране время в формате (дд/мм/гг)
 - Метод - рассчитать количество дней с начала года до даты
- public int Days()

Программа должна делать следующее:

1. В функции main() нужно объявить и создать массив из 3 объектов описанного класса
2. Задать им следующие значения (1.5.2001 5.2.2002 13.7.2001)
3. Вывести на экран даты, хранящиеся во всех объектах.
4. Рассчитать разницу в днях между 1 и 3 объектами и вывести ее на экран.

Критерии формирования оценок по заданиям для самостоятельной работы студента

«Отлично» (5 баллов) – обучающийся показал глубокие знания материала по поставленным вопросам, грамотно, логично его излагает, структурировал и детализировал информацию, избегая простого повторения информации из текста, информация представлена в переработанном виде. Свободно использует необходимые формулы при решении задач;

«Хорошо» (4 балла) – обучающийся твердо знает материал, грамотно его излагает, не допускает существенных неточностей в процессе решения задач;

«Удовлетворительно» (3 балла) – обучающийся имеет знания основного материала по поставленным вопросам, но не усвоил его деталей, допускает отдельные неточности при решении задач;

«Неудовлетворительно» (менее 3 баллов) – обучающийся допускает грубые ошибки в ответе на поставленные вопросы и при решении задач

5.2. Оценочные материалы для рубежного контроля

Рубежный контроль осуществляется по более или менее самостоятельным разделам – учебным модулям курса и проводится по окончании изучения материала модуля в заранее установленное время. Рубежный контроль проводится с целью определения качества усвоения материала учебного модуля в целом. В течение семестра проводится три таких контрольных мероприятия по графику.

В качестве форм рубежного контроля используется тестирование (письменное или компьютерное), проведение коллоквиума.

5.2.1. Оценочные материалы для проведения коллоквиума (контролируемые компетенции ПК-2, ПК-6)

Перечень вопросов, выносимых на рейтинговый контроль первой точки:

1. Обзор .NET. Основные понятия.
2. Принципы объектно-ориентированного программирования. Язык C# и платформа .NET
3. JIT-компиляция.
4. Компиляция в командной строке.
5. Типы данных и переменные в C#.
6. Объявление переменных в C#.
7. Неявная типизация в C#.
8. Область видимости (контекст) переменных в C#.
9. Преобразования базовых типов данных в C#.
10. Явные и неявные преобразования в C#.
11. Операции языка C#.
12. Преобразования базовых типов данных в C#.
13. Явные и неявные преобразования в C#.
14. Массивы в C#.
15. Многомерные массивы в C#.
16. Массив массивов в C#.
17. Методы и свойства массивов в C#.
18. Условные конструкции в C#.
19. Конструкция switch/case Тернарная операция в C#.
20. Циклы в C#.
21. Цикл foreach в C#.
22. Операторы continue и break в C#.

Перечень вопросов, выносимых на рейтинговый контроль второй точки:

1. Методы в C#.
2. Функции в C#.
3. Параметры методов в C#.
4. Передача параметров по ссылке и модификатор ref в C#.
5. Необязательные параметры методов в C#.
6. Именованные параметры методов в C#.

7. Массив параметров и ключевое слово params в C#.
8. Рекурсивные функции в C#.
9. Перечисления enum в C#.
10. Структуры в C#.
11. Конструкторы в структурах в C#.
12. Обработка исключений в C#.
13. Обработка исключений и условные конструкции в C#.
14. Фильтры исключений в C#.
15. Работа с консолью и класс Console в C#.
16. Типы значений и ссылочные типы в C#.
17. Копирование значений в C#.
18. Ссылочные типы внутри типов значений в C#.
19. Объекты классов как параметры методов в C#.
20. Классы и объекты в C#.
21. Инициализаторы объектов в C#.
22. Частичные классы в C#.

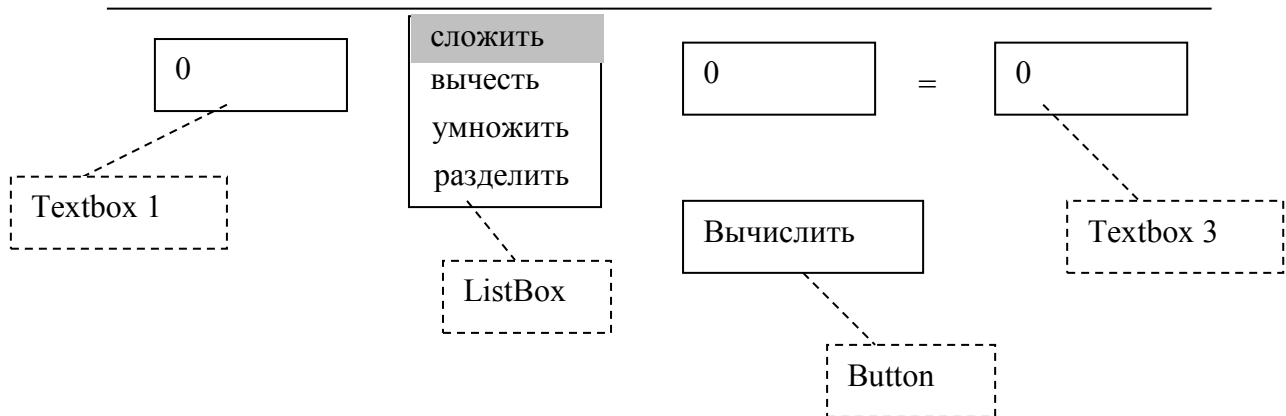
Перечень вопросов, выносимых на рейтинговый контроль третьей точки:

1. Модификаторы доступа в C#.
2. Константы и поля для чтения в C#.
3. Инкапсуляция в C#.
4. Автоматические свойства в C#.
5. Перегрузка методов и операторов в C#.
6. Перегрузка операторов в C#.
7. Статические члены класса в C#.
8. Статический конструктор в C#.
9. Статические классы в C#.
10. Наследование в C#.
11. Доступ к членам базового класса из класса-наследника в C#.
12. Ключевое слово base в C#.
13. Конструкторы в производных классах в C#.
14. Полиморфизм и переопределение методов в C#.
15. Абстрактные классы в C#.
16. Класс System.Object и его методы в C#.
17. Преобразование типов в C#.
18. Обобщенные типы в C#.
19. Анонимные типы в C#.
20. Методы расширения в C#.
21. Пространства имен в C#.
22. Псевдонимы в C#.
23. Интерфейсы в C#.

Образцы заданий рубежного контроля

Задание 1

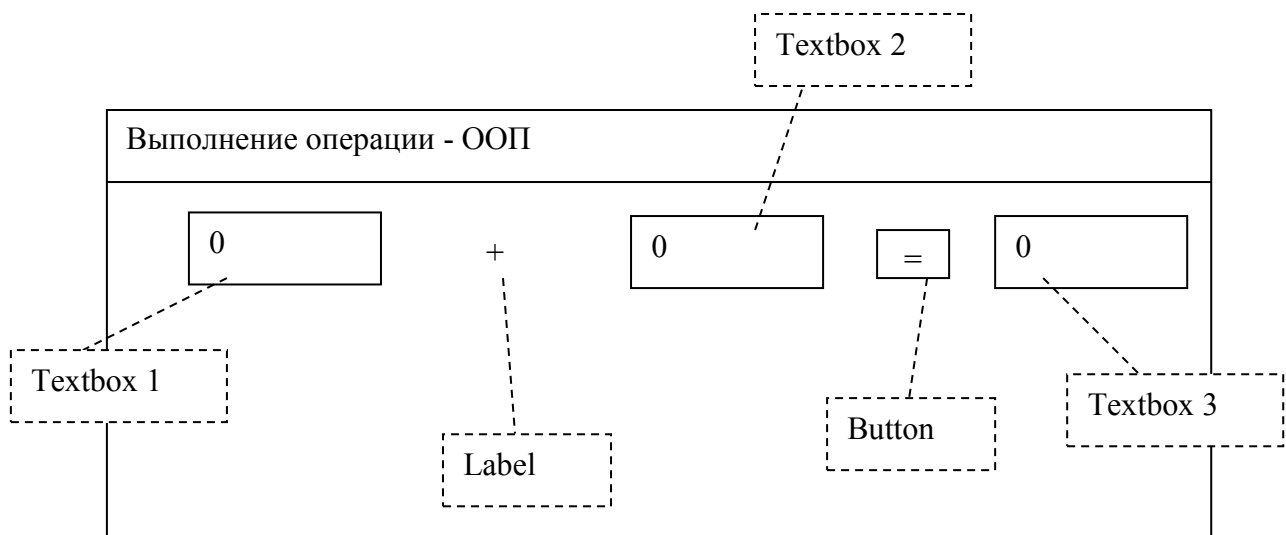
1. Написать программу, которая, используя объект класса производного от класса Form, выводит следующее окно:



При нажатии кнопки Button, текстовое поле Textbox3 должно показать результат выбранной в ListBox операции над числами, введенными в Textbox1 и Textbox2

Задание 2

1. Написать программу, которая, используя объект класса производного от класса Form, выводит следующее окно:



При нажатии кнопки Button, текстовое поле Textbox3 должно показать результат выбранной в ListBox операции над числами, введенными в Textbox1 и Textbox2

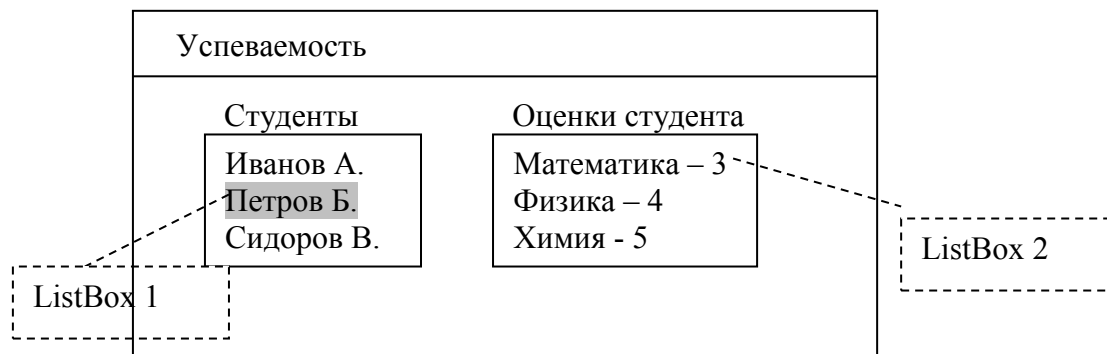
При выборе студента в ListBox1 в ListBox2 должны показываться его оценки.

Задание 3

1. Написать программу, которая, используя объект класса производного от класса Form, позволяет просматривать информацию об успеваемости студентов по годам:

ФИО студента	Математика	Физика	Химия
Иванов А.	4	5	4
Петров Б.	3	4	5
Сидоров В.	5	5	5

Форма окна программы:



При выборе студента в ListBox1 в ListBox2 должны показываться его оценки.

Критерии формирования оценок по контрольным точкам (коллоквиум)

4 балла ставится за работу, выполненную полностью без ошибок и недочетов; обучающийся демонстрирует знание теоретического и практического материала по теме практической работы, решено 100% задач;

3 балла ставится за работу, выполненную полностью, но при наличии в ней не более одной негрубой ошибки и одного недочета, не более трех недочетов. Обучающийся демонстрирует знание теоретического и практического материала по теме практической работы, допуская незначительные неточности при решении задач, решено 70% задач;

2 балла ставится за работу, если обучающийся правильно выполнил не менее 2/3 всей работы или допустил не более одной грубой ошибки и двух недочетов, не более одной грубой и одной негрубой ошибки, не более трех негрубых ошибок, одной негрубой. Обучающийся затрудняется с правильной оценкой предложенной задачи, дает неполный ответ, решено 55% задач

1 балл ставится за работу, если число ошибок и недочетов превысило норму или правильно выполнено менее 2/3 всей работы. Обучающийся дает неверную оценку ситуации, решено менее 50 % задач.

5.2.2. Оценочные материалы: тестирование (контролируемые компетенции ПК-2, ПК-6)

Тестирование обучающихся проводится в онлайн-режиме согласно расписанию в ЭИОС open.kbsu.

Критерии формирования оценок по тестовым заданиям

Выполнение тестирования оценивается согласно проценту правильных ответов. Максимально возможное количество баллов за тестирование – 5.

5.3. Оценочные материалы для промежуточной аттестации

Целью промежуточных аттестаций по дисциплине является оценка качества освоения дисциплины обучающимися.

Промежуточная аттестация предназначена для объективного подтверждения и оценивания достигнутых результатов обучения после завершения изучения дисциплины. Осуществляется в конце семестра и представляет собой итоговую оценку знаний по дисциплине «Объектно-ориентированное программирование» в виде проведения зачета.

Промежуточная аттестация может проводиться в устной, письменной форме, и в форме тестирования. На промежуточную аттестацию отводится до 30 баллов.

Вопросы итоговой аттестации по дисциплине (контролируемые компетенции ПК-2, ПК-6)

1. Обзор .NET. Основные понятия.
2. Принципы объектно-ориентированного программирования. Язык C# и платформа .NET
3. JIT-компиляция.
4. Компиляция в командной строке.
5. Типы данных и переменные в C#.
6. Объявление переменных в C#.
7. Неявная типизация в C#.
8. Область видимости (контекст) переменных в C#.
9. Преобразования базовых типов данных в C#.
10. Явные и неявные преобразования в C#.
11. Операции языка C#.
12. Преобразования базовых типов данных в C#.
13. Явные и неявные преобразования в C#.
14. Массивы в C#.
15. Многомерные массивы в C#.
16. Массив массивов в C#.
17. Методы и свойства массивов в C#.
18. Условные конструкции в C#.
19. Конструкция switch/case Тернарная операция в C#.
20. Циклы в C#.
21. Цикл foreach в C#.
22. Операторы continue и break в C#.
23. Методы в C#.
24. Функции в C#.
25. Параметры методов в C#.
26. Передача параметров по ссылке и модификатор ref в C#.
27. Необязательные параметры методов в C#.
28. Именованные параметры методов в C#.
29. Массив параметров и ключевое слово params в C#.
30. Рекурсивные функции в C#.
31. Перечисления enum в C#.
32. Структуры в C#.
33. Конструкторы в структурах в C#.
34. Обработка исключений в C#.
35. Обработка исключений и условные конструкции в C#.
36. Фильтры исключений в C#.
37. Работа с консолью и класс Console в C#.
38. Типы значений и ссылочные типы в C#.
39. Копирование значений в C#.
40. Ссылочные типы внутри типов значений в C#.
41. Объекты классов как параметры методов в C#.

42. Классы и объекты в C#.
43. Инициализаторы объектов в C#.
44. Частичные классы в C#.
45. Модификаторы доступа в C#.
46. Константы и поля для чтения в C#.
47. Инкапсуляция в C#.
48. Автоматические свойства в C#.
49. Перегрузка методов и операторов в C#.
50. Перегрузка операторов в C#.
51. Статические члены класса в C#.
52. Статический конструктор в C#.
53. Статические классы в C#.
54. Наследование в C#.
55. Доступ к членам базового класса из класса-наследника в C#.
56. Ключевое слово `base` в C#.
57. Конструкторы в производных классах в C#.
58. Полиморфизм и переопределение методов в C#.
59. Абстрактные классы в C#.
60. Класс `System.Object` и его методы в C#.
61. Преобразование типов в C#.
62. Обобщенные типы в C#.
63. Анонимные типы в C#.
64. Методы расширения в C#.
65. Пространства имен в C#.
66. Псевдонимы в C#.
67. Интерфейсы в C#.

Критерии формирования оценок по промежуточной аттестации

«Зачтено» получают обучающиеся, которые

- свободно ориентируются в материале и отвечают без затруднений. Обучающийся способен к выполнению сложных заданий, постановке целей и выборе путей их реализации. Работа выполнена полностью без ошибок, решено 100% задач;
- относительно полно ориентируются в материале, отвечают без затруднений, допускают незначительное количество ошибок. Обучающийся способен к выполнению сложных заданий. Работа выполнена полностью, но имеются не более одной негрубой ошибки и одного недочета, не более трех недочетов. Допускаются незначительные неточности при решении задач, решено 70% задач;
- недостаточно высок уровень владения материалом. В процессе ответа на зачете допускаются ошибки и затруднения при изложении материала. Обучающийся правильно выполнил не менее 2/3 всей работы или допустил не более одной грубой ошибки и двух недочетов, не более одной грубой и одной негрубой ошибки, не более трех негрубых ошибок, одной негрубой. Обучающийся затрудняется с правильной оценкой предложенной задачи, дает неполный ответ, решено 55% задач;

«Не зачтено» получают обучающиеся, которые допускают значительные ошибки. Обучающийся имеет лишь начальную степень ориентации в материале. В работе число ошибок и недочетов превысило норму для оценки 3 или правильно выполнено менее 2/3 всей работы. Обучающийся дает неверную оценку ситуации, решено менее 50% задач.

6. МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ, ОПРЕДЕЛЯЮЩИЕ ПРОЦЕДУРЫ ОЦЕНИВАНИЯ ЗНАНИЙ, УМЕНИЙ, НАВЫКОВ И (ИЛИ) ОПЫТА ДЕЯТЕЛЬНОСТИ

Максимальная сумма (61 балл), набираемая студентом по дисциплине включает две составляющие:

- первая составляющая – оценка регулярности, своевременности и качества выполнения студентом учебной работы по изучению дисциплины в течение периода изучения дисциплины (семестра, или нескольких семестров) (сумма – не более 70 баллов). Баллы, характеризующие успеваемость студента по дисциплине, набираются им в течение всего периода обучения за изучение отдельных тем и выполнение отдельных видов работ.
- вторая составляющая – оценка знаний студента по результатам промежуточной аттестации (до 61 балла).

Критерием оценки уровня сформированности компетенций в рамках учебной дисциплины «Объектно-ориентированное программирование» является зачет.

Общий балл текущего и рубежного контроля складывается из следующих составляющих (таблица 6).

Таблица 6

Распределение баллов текущего и рубежного контроля

№п/п	Вид контроля	Сумма баллов			
		Общая сумма	1-я точка	2-я точка	3-я точка
1	Посещение занятий	до 10 б.	до 3 б.	до 3б.	до 4б.
2	Текущий контроль:	до 28 б.	до 9 б.	до 9 б.	до 10 б.
	Выполнение лабораторных работ	до 18 б.	до 6 б.	до 6 б.	до 6 б.
	Выполнение самостоятельных заданий	от 0 до 10 б.	от 0 до 3 б.	от 0 до 3 б.	от 0 до 4 б.
3	Рубежный контроль	до 27 баллов	до 9 б.	до 9 б.	до 9 б.
	тестирование	от 0 до 15 б.	от 0 до 5 б.	от 0 до 5 б.	от 0 до 5 б.
	коллоквиум	от 0 до 12 б.	от 0 до 4 б.	от 0 до 4 б.	от 0 до 4 б.
4	Итого сумма текущего и рубежного контроля	до 70 б.	до 23 б.	до 23 б.	до 24 б.

Целью промежуточных аттестаций по дисциплине является оценка качества освоения дисциплины обучающимися.

Критерии оценки качества освоения дисциплины

«Зачтено» – 61 балл:

- теоретическое содержание курса освоено полностью, без пробелов, необходимые практические навыки работы с освоенным материалом сформированы. Все предусмотренные программой обучения учебные задания выполнены, качество их выполнения оценено числом баллов, близким к максимальному. На экзамене студент демонстрирует глубокие знания предусмотренного программой материала, умеет четко, лаконично и логически последовательно отвечать на поставленные вопросы.

- теоретическое содержание курса освоено, необходимые практические навыки работы сформированы, выполненные учебные задания содержат незначительные ошибки. На экзамене студент демонстрирует твердые знания основного (программного) материала, умеет четко, грамотно, без существенных неточностей отвечать на поставленные вопросы.
- теоретическое содержание курса освоено не полностью, необходимые практические навыки работы сформированы частично, выполненные учебные задания содержат грубые ошибки. На экзамене студент демонстрирует знание только основного материала, ответы содержат неточности, слабо аргументированы, нарушена последовательность изложения материала
«Не зачтено» от 36 до 60 баллов:
- теоретическое содержание курса не освоено, необходимые практические навыки работы не сформированы, выполненные учебные задания содержат грубые ошибки, дополнительная самостоятельная работа над материалом курса не приведет к существенному повышению качества выполнения учебных заданий. На экзамене студент демонстрирует незнание значительной части программного материала, существенные ошибки в ответах на вопросы, неумение ориентироваться в материале, незнание основных понятий дисциплины.

Таблица 7

Результаты освоения учебной дисциплины, подлежащие проверке

Результаты обучения (компетенции)	Основные показатели оценки результатов обучения	Вид оценочного материала, обеспечивающего формирование компетенций
<p>ПК-2 – способность применять программные средства системного, прикладного и специального назначения, инструментальные средства, языки и системы программирования для решения профессиональных задач</p>	<p>Знать:</p> <ul style="list-style-type: none"> – Технологию разработки алгоритмов и программ, методы отладки и решения задач на ЭВМ в различных режимах. – Основные критерии качества ПО. – Фазы жизненного цикла программного обеспечения. – Возможности интегрированной среды при написании программы, ее тестировании и отладке. – Виды программных средств для использования в научных исследованиях, проектно-конструкторской деятельности, управлении технологическими, экономическими, социальными системами и в гуманитарных областях деятельности человека; <p>Общие принципы работы программных средств под управлением современных операционных систем.</p> <p>Уметь:</p> <ul style="list-style-type: none"> – Ставить задачу и разрабатывать алгоритм ее решения. – Использовать прикладные системы программирования. – Разрабатывать основные программные документы. – Анализировать методы решения задачи и обосновывать выбранный метод. – Приводить решение задачи к решению подзадач. – Обобщать данные при написании кода программы. – Оценивать степень соответствия 	<p>Выполнение лабораторных работ Коллоквиум Тестирование Зачет</p>

	<p>разработанной программы требованиям, приведенным в спецификациях.</p> <ul style="list-style-type: none"> – Разрабатывать спецификаций на программу. – Применять возможности интегрированной среды при написании программы, ее тестировании и отладке. – Выбирать и применять программные средства для эффективного решения практических задач; – Самостоятельно осваивать новые программные средства; <p>Работать с современными системами программирования, включая объектно-ориентированные.</p> <p>Владеть:</p> <ul style="list-style-type: none"> – Навыками разработки и отладки программ на одном из алгоритмических процедурных языков программирования высокого уровня. – Навыками составления отчетов по методикам исследования и их реализации в виде ПО, анализа результатов обработки. – Навыками работы с типовыми и специализированными программными продуктами. 	
<p>ПК-6 – способность принимать участие в организации и проведении контрольных проверок работоспособности и эффективности применяемых программных, программно-аппаратных и технических средств защиты информации</p>	<p>Владеть:</p> <ul style="list-style-type: none"> – Методами манипулирования данными (в том числе хранение, добавление, редактирование и удаление данных), навигация по набору данных. – Владеет сортировкой, поиском и фильтрацией (выборка) данных. – Методами разработки моделей изучаемых объектов. – Приемами рекурсивного программирования, реализации рекурсивных структур данных в языках программирования; – Методиками представления задач в пространстве состояний и оптимизации поиска решений; <p>Уметь:</p> <ul style="list-style-type: none"> – Проводить исследование предметной области; – Проводить анализ пользователей и их требований; – Определять структуру системы; <p>Знать:</p> <ul style="list-style-type: none"> – Формальные модели основных вычислительных процессов, методы управления процессами и их синхронизации, протоколы взаимодействия объектов, методы анализа вычислительных процессов. – Этапы трансляции программы. – Современные технические средства взаимодействия с ЭВМ 	<p>Выполнение лабораторных работ Коллоквиум Тестирование Зачет</p>

7. УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

7.1. Основная литература

1. Казанский А.А. Объектно-ориентированное программирование на языке Microsoft Visual C# в среде разработки Microsoft Visual Studio 2008 и .NET Framework. 4.3 [Электронный ресурс] : учебное пособие и практикум / А.А. Казанский. — Электрон. текстовые данные. — М. : Московский государственный строительный университет, ЭБС АСВ, 2011. — 180 с. — 2227-8397. — Режим доступа: <http://www.iprbookshop.ru/19258.html>
2. Мейер Б. Объектно-ориентированное программирование и программная инженерия [Электронный ресурс] / Б. Мейер. — Электрон. текстовые данные. — М. : Интернет-Университет Информационных Технологий (ИНТУИТ), 2016. — 285 с. — 2227-8397. — Режим доступа: <http://www.iprbookshop.ru/39552.html>
3. Лисицин Д.В. Объектно-ориентированное программирование [Электронный ресурс] : конспект лекций / Д.В. Лисицин. — Электрон. текстовые данные. — Новосибирск: Новосибирский государственный технический университет, 2010. — 88 с. — 978-5-7782-1454-5. — Режим доступа: <http://www.iprbookshop.ru/44970.html>
4. Николаев Е.И. Объектно-ориентированное программирование [Электронный ресурс] : учебное пособие / Е.И. Николаев. — Электрон. текстовые данные. — Ставрополь: Северо-Кавказский федеральный университет, 2015. — 225 с. — 2227-8397. — Режим доступа: <http://www.iprbookshop.ru/62967.html>
5. Букунов С.В. Основы объектно-ориентированного программирования [Электронный ресурс] : учебное пособие / С.В. Букунов, О.В. Букунова. — Электрон. текстовые данные. — СПб. : Санкт-Петербургский государственный архитектурно-строительный университет, ЭБС АСВ, 2017. — 196 с. — 978-5-9227-0713-8. — Режим доступа: <http://www.iprbookshop.ru/74339.html>
6. Биллиг В.А. Основы объектного программирования на C# (C# 3.0, Visual Studio 2008) [Электронный ресурс] : учебное пособие / В.А. Биллиг. — Электрон. текстовые данные. — Москва, Саратов: Интернет-Университет Информационных Технологий (ИНТУИТ), Вузовское образование, 2017. — 583 с. — 978-5-4487-0145-0. — Режим доступа: <http://www.iprbookshop.ru/72339.html>
7. 10. Марченко А.Л. Основы программирования на C# 2.0 [Электронный ресурс] : учебное пособие / А.Л. Марченко. — Электрон. текстовые данные. — Москва, Саратов: Интернет-Университет Информационных Технологий (ИНТУИТ), Вузовское образование, 2017. — 552 с. — 978-5-4487-0084-2. — Режим доступа: <http://www.iprbookshop.ru/67382.html>

7.2. Дополнительная литература

8. Фленов М. Е. Библия C#, 2-е издание. СПб: БХВ-Петербург, 2011, 560с.
9. Шилд Г. C# 4.0: Полное руководство. М: Вильямс, 2011, 1056 с.
10. Адриянова А.А., Исмагилов Л. Н., Мухтарова Т. М. Объектно-ориентированное программирование на C#: Учебное пособие. Казань: Казанский (Приволжский) Федеральный Университет, 2012, 134 с.
11. Давыдов В.Г. Программирование и основы алгоритмизации: учеб. пособие 2-е изд. - М.: Высшая школа, 2005, 447 с.
12. Фаулер М. Рефакторинг: улучшение существующего кода. – СПб.: «Символ-Плюс», 2002 г.
13. Рамбо Дж., Якобсон А., Буч Г. Унифицированный процесс разработки программного обеспечения. – СПб.: Питер, 2002 г.

14. Скотт. Унифицированный процесс. Основные концепции. – М.: «Вильямс», 2002 г.
15. Ларман. Применение UML и шаблонов проектирования. – М.: «Вильямс», 2002 г.
16. Мандел Т. Разработка пользовательского интерфейса. ДМК Пресс, 2001, 408 с.
17. Гаврилова А. В. Разработка приложений: учебное пособие. Изд. Флинта, 2012, 242с.
18. Ипатова Э. Р., Ипатов Ю.В. Методологии и технологии системного проектирования. Изд. Флинта, МПСИ, 2008, 256 с.
19. Ахо, Сети, Ульман 2001 - Ахо А., Сети Р., Ульман Дж. Компиляторы: принципы, технологии и инструменты. - М.: Издательский дом «Вильямс», 2001.
20. Эрих Гамма, Ричард Хелм, Ральф Джонсон, Джон Влиссидес. Приемы объектно-ориентированного проектирования. Паттерны проектирования. Питер, 2016.
21. Мейер Б. Основы программирования. ИНТУИТ, 2011.
22. Мэтт Вайсфельд. Объектно-ориентированное мышление. Питер, 2014.
23. Эдвард Йордон, Карл Аргила. Объектно-ориентированный анализ и проектирование систем. Лори, 2014.

7.3. Периодические издания

7.4. Интернет-ресурсы

1. С.В. Зыков. Технологии и средства разработки корпоративных систем. Интуит.ру, <http://www.intuit.ru/department/itmngt/techcorpsys/>
2. Т.С. Васючкова, М.А. Держо, Н.А. Иванчева, Т.П. Пухначева. Управление проектами с использованием Microsoft Project. Интуит.ру, <http://www.intuit.ru/department/itmngt/pmusemspr/>
3. Д.В. Андреев. Организация процессов разработки программного обеспечения с использованием Team Foundation Server 2010. Интуит.ру, <http://www.intuit.ru/department/se/vdevtfds2010/>
4. А.Э. Кузьмичёв. Программирование для Windows Phone для начинающих. Интуит.ру, <http://www.intuit.ru/department/se/devwphonebeg/>
5. А.В. Марчуков, А.О. Савельев. Работа в Microsoft Visual Studio. Интуит.ру, <http://www.intuit.ru/department/itmngt/workinmsvistudio/>
6. В.А. Петрухин, Е.М. Лаврищева. Методы и средства инженерии программного обеспечения. Интуит.ру, <http://www.intuit.ru/department/se/swebok/>
7. В.А. Биллиг. Основы программирования на C# 3.0: ядро языка. Интуит.ру, <http://www.intuit.ru/department/pl/tincsharp3/>
8. В.П. Котляров. Основы тестирования программного обеспечения. Интуит.ру, <http://www.intuit.ru/department/se/testing/>

-профессиональные базы:

1. PCI Security Standards Council – <http://www.pcisecuritystandards.org>.
2. Стандарты информационной безопасности в кредитно-финансовой сфере. Стандарты Банка России – <http://www.abiss.ru/doc>

3. **Threatpost** <https://threatpost> Сайт об информационной безопасности от Kaspersky Lab. Авторитетный источник, на который ссылаются ведущие новостные агентства, такие как The New York Times и The Wall Street Journal.
4. **Security Lab** <http://www.securitylab.ru/> Проект компании Positive Technologies. Помимо новостей, экспертных статей, софта, форума, на сайте есть раздел, где оперативно публикуется информация об уязвимостях, а также даются конкретные рекомендации по их устранению.
5. **Anti-Malware** <https://www.anti-malware.ru/> Информационно-аналитический центр, посвященный информационной безопасности. Anti-Malware проводит сравнительные тесты антивирусов, публикует аналитические статьи, эксперты принимают участие в дискуссиях на форуме.
6. **ISO27000.RU** <http://www.iso27000.ru/> Интернет-портал ISO27000.RU – это площадка для общения специалистов по ИБ. Есть тематический каталог ссылок на ресурсы по информационной безопасности и защите информации.
7. **Naked Security** <https://nakedsecurity.sophos.com/> Сайт компании Sophos, цитируемый крупными изданиями. Освещается широкий круг вопросов: последние события в мире информационной безопасности, новые угрозы, обзор самых важных новостей недели.
8. **Dark Reading** <http://www.darkreading.com/> Сообщество профессионалов, где обсуждаются кибер-угрозы, уязвимости и методы защиты от атак, а также ключевые технологии и методы, которые могут помочь защитить данные в будущем.

- общие информационные, справочные и поисковые:

1. **Scopus** <http://scopus.com> Scopus – крупнейшая единая база данных, содержащая аннотации и информацию о цитируемости рецензируемой научной литературы, со встроенными инструментами отслеживания, анализа и визуализации данных. В базе содержится 23700 изданий от 5000 международных издателей, в области естественных, общественных и гуманитарных наук, техники, медицины и искусства.
2. **Web of Science** <http://apps.webofknowledge.com> Наукометрическая реферативная база данных журналов и конференций. С платформой Web of Science вы можете получить доступ к непревзойденному объему исследовательской литературы мирового класса, связанной с тщательно отобранным списком журналов, и открыть для себя новую информацию при помощи скрупулезно записанных метаданных и ссылок.
3. **zbMATH** <http://zbmath.org> самая полная математическая база данных, охватывающая материалы с конца 19 века. zbMath содержит около 4 000 000 документов, из более 3 000 журналов и 170 000 книг по математике, статистике, информатике, а также машиностроению, физике, естественным наукам и др.
4. Справочная правовая система «Гарант». URL: <http://www.garant.ru>.
5. Справочная правовая система «КонсультантПлюс». URL: <http://www.consultant.ru>
6. Полнотекстовая база данных ScienceDirect: URL: <http://www.sciencedirect.com>.
7. Реферативная база данных зарубежных изданий по экономике EconLit: URL: <http://www.ebscohost.com>
8. Economics online <http://www.econline.h1.ru> - целью данного проекта является создание коллекции ссылок на ресурсы WWW, предоставляющие экономическую и финансовую информацию бесплатно в режиме онлайн. На сайте вы найдете каталог ссылок на лучшие экономические ресурсы, новости, информацию по экономической теории, финансам, статистике, архивы научных работ по экономике и т. д.
9. Электронная библиотека по бизнесу и финансам <http://www.finbook.biz/> - сайт предоставляет бесплатный доступ к электронным книгам по бизнесу, финансам, экономике.
10. Служба тематических толковых словарей <http://glossary.ru/>
11. Защита от компьютерных вирусов. Антивирусные программы [Электронный ресурс] = www.lessons-tva.info/edu/e-inf1/e-inf1-4-1-3.html
12. Антивирусная защита информации: способы и средства-

<https://www.google.ru/webhpsourceid=chrome-instant&ion=1&espv>

7.5. Методические указания по проведению различных учебных занятий и самостоятельной работы

Методические указания к практическим занятиям

Выполнение каждой лабораторной работы складывается из следующих этапов.

Самостоятельная подготовка студентов к работе. Перед началом работы студенты должны четко представлять себе цель работы, знать схему, метод измерения, физическую сущность ожидаемых результатов. Должен быть подготовлен отчет, содержащий о порядке выполнения лабораторной работы. Студенты, не подготовившиеся к работе в соответствии с этими требованиями, к выполнению работы не допускаются.

Проведение эксперимента. Этот этап осуществляется в соответствии с методическими указаниями, которые содержатся в описании к каждой работе. Приступать к работам на стенде студент может начать только после ознакомления с теоретической частью и описания хода выполнения работы. Любые изменения в схеме проводятся при тщательной проверке схемы, для исключения короткого замыкания. Результаты выполнения проверяются преподавателем.

Составление отчета о проделанной работе. Отчёт должен содержать исчерпывающие данные, как о цели работы, так и о результатах в следующей последовательности: задание; схема установки и описание хода выполнения; результаты выполнения работы, включая рисунки, схемы, таблицы; общие выводы и заключение.

Текст отчета должен быть написан аккуратно и разборчиво от руки или представлен в виде распечатки, после компьютерной верстки. В обоих случаях текст должен представлять собой логическое изложение существа вопроса. Недопустимо приведение формул, таблиц без разъяснений всех обозначений и сокращений. Полученные зависимости должны сопровождаться теоретическим обоснованным объяснением причин, влияющих на их ход, для чего в процессе составления отчета студент обязан по литературным источникам ознакомиться с материалом, который был объектом его исследования в лаборатории. Без такого ознакомления с испытуемым методом студент не будет в состоянии дать правильный анализ процессов, происходящих в материале при эксперименте.

Защита результатов работы с представлением отчета. При сдаче отчета студенты должны показать понимание сущности проведенных исследований, объяснить полученные результаты и сделать выводы. При работе в лаборатории необходимо строго выполнять все правила техники безопасности и указания преподавателя.

Методические рекомендации по работе с литературой

Всю литературу можно разделить на учебники и учебные пособия, оригинальные научные монографические источники, научные публикации в периодической печати. Из них можно выделить литературу основную (рекомендуемую), дополнительную и литературу для углубленного изучения дисциплины.

Изучение дисциплины следует начинать с учебника, поскольку учебник – это книга, в которой изложены основы научных знаний по определенному предмету в соответствии с целями и задачами обучения, установленными программой.

При работе с литературой необходимо учитывать, что имеются различные виды чтения, и каждый из них используется на определенных этапах освоения материала.

Предварительное чтение направлено на выявление в тексте незнакомых терминов и поиск их значения в справочной литературе. В частности, при чтении указанной литературы необходимо подробнейшим образом анализировать понятия.

Сквозное чтение предполагает прочтение материала от начала до конца. Сквозное чтение литературы из приведенного списка дает возможность студенту сформировать свод основных понятий из изучаемой области и свободно владеть ими.

Выборочное – наоборот, имеет целью поиск и отбор материала. В рамках данного курса выборочное чтение, как способ освоения содержания курса, должно использоваться при подготовке к практическим занятиям по соответствующим разделам.

Аналитическое чтение – это критический разбор текста с последующим его конспектированием. Освоение указанных понятий будет наиболее эффективным в том случае, если при чтении текстов студент будет задавать к этим текстам вопросы. Часть из этих вопросов сформулирована в ФОС в перечне вопросов для собеседования. Перечень этих вопросов ограничен, поэтому важно не только содержание вопросов, но сам принцип освоения литературы с помощью вопросов к текстам.

Целью изучающего чтения является глубокое и всестороннее понимание учебной информации. Есть несколько приемов изучающего чтения:

1. Чтение по алгоритму предполагает разбиение информации на блоки: название; автор; источник; основная идея текста; фактический материал; анализ текста путем сопоставления имеющихся точек зрения по рассматриваемым вопросам; новизна.
2. Прием постановки вопросов к тексту имеет следующий алгоритм:
 - медленно прочитать текст, стараясь понять смысл изложенного;
 - выделить ключевые слова в тексте;
 - постараться понять основные идеи, подтекст и общий замысел автора.
3. Прием тезирования заключается в формулировании тезисов в виде положений, утверждений, выводов.

К этому можно добавить и иные приемы: прием реферирования, прием комментирования.

Важной составляющей любого солидного научного издания является список литературы, на которую ссылается автор. При возникновении интереса к какой-то обсуждаемой в тексте проблеме всегда есть возможность обратиться к списку относящейся к ней литературы. В этом случае вся проблема как бы разбивается на составляющие части, каждая из которых может изучаться отдельно от других. При этом важно не терять из вида общий контекст и не погружаться чрезмерно в детали, потому что таким образом можно не увидеть главного.

Подготовка к экзамену должна проводиться на основе лекционного материала, материала практических занятий с обязательным обращением к основным учебникам по курсу. Это позволит исключить ошибки в понимании материала, облегчит его осмысление, прокомментирует материал многочисленными примерами.

Методические рекомендации для подготовки к зачету

Зачет является формой итогового контроля знаний и умений обучающихся по данной дисциплине, полученных на лекциях, лабораторных занятиях и в процессе самостоятельной работы. Основой для определения оценки служит уровень усвоения обучающимися материала, предусмотренного данной рабочей программой. К зачету допускаются студенты, набравшие 36 и более баллов по итогам текущего и промежуточного контроля. На зачете студент может набрать от 0 до 25 баллов.

В период подготовки к зачету обучающиеся вновь обращаются к учебно-методическому материалу и закрепляют промежуточные знания.

Подготовка обучающегося к зачету включает три этапа:

- самостоятельная работа в течение семестра;
- непосредственная подготовка в дни, предшествующие зачету по темам курса;
- подготовка к ответу на вопросы к зачету.

При подготовке к зачету обучающимся целесообразно использовать материалы лекций, учебно-методические комплексы, нормативные документы, основную и дополнительную литературу.

На зачет выносятся материал в объеме, предусмотренном рабочей программой учебной дисциплины за семестр. зачет проводится в письменной / устной форме.

При проведении зачета в письменной (устной) форме, ведущий преподаватель составляет билеты к зачету, которые включают в себя: тестовые задания; теоретические задания; задачи или ситуации. Формулировка теоретических заданий совпадает с формулировкой перечня вопросов на зачет, доведенных до сведения. Содержание вопросов одного билета относится к различным разделам программы с тем, чтобы более полно охватить материал учебной дисциплины.

В аудитории, где проводится устный зачет, должно одновременно находиться не более шести студентов на одного преподавателя, принимающего зачет. На подготовку ответа на билет на зачете отводится 40 минут.

При проведении письменного зачета на работу отводится 60 минут.

Результат устного (письменного) зачета выражается:

«Зачтено»:

- теоретическое содержание курса освоено полностью, без пробелов, необходимые практические навыки работы с освоенным материалом сформированы. Все предусмотренные программой обучения учебные задания выполнены, качество их выполнения оценено числом баллов, близким к максимальному. На экзамене студент демонстрирует глубокие знания предусмотренного программой материала, умеет четко, лаконично и логически последовательно отвечать на поставленные вопросы.
- теоретическое содержание курса освоено, необходимые практические навыки работы сформированы, выполненные учебные задания содержат незначительные ошибки. На экзамене студент демонстрирует твердые знания основного (программного) материала, умеет четко, грамотно, без существенных неточностей отвечать на поставленные вопросы.
- теоретическое содержание курса освоено не полностью, необходимые практические навыки работы сформированы частично, выполненные учебные задания содержат грубые ошибки. На экзамене студент демонстрирует знание только основного материала, ответы содержат неточности, слабо аргументированы, нарушена последовательность изложения материала.

«Не зачтено»:

- теоретическое содержание курса не освоено, необходимые практические навыки работы не сформированы, выполненные учебные задания содержат грубые ошибки, дополнительная самостоятельная работа над материалом курса не приведет к существенному повышению качества выполнения учебных заданий. На экзамене студент демонстрирует незнание значительной части программного материала, существенные ошибки в ответах на вопросы, неумение ориентироваться в материале, незнание основных понятий дисциплины.

8. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

Стандартно оборудованные лекционные аудитории для проведения лекций: ПЭВМ, видеопроектор, экран настенный.

Лабораторный практикум проводится в компьютерных классах, оснащенных ПК со стандартным ПО.

Для студентов с ограниченными возможностями здоровья созданы специальные условия для получения образования. В целях доступности получения высшего образования по образовательным программам инвалидами и лицами с ограниченными возможностями здоровья университетом обеспечивается:

1. Альтернативной версией официального сайта в сети «Интернет» для слабовидящих.
2. Присутствие ассистента, оказывающего обучающемуся необходимую помощь.
3. Для инвалидов и лиц с ограниченными возможностями здоровья по слуху – дублирование вслух справочной информации о расписании учебных занятий; обеспечение надлежащими звуковыми средствами воспроизведения информации.
4. Для инвалидов и лиц с ограниченными возможностями здоровья, имеющих нарушения опорно-двигательного аппарата, созданы материально-технические условия,

обеспечивающие возможность беспрепятственного доступа обучающихся в учебные помещения, объекты питания, туалетные и другие помещения университета, а также пребывания в указанных помещениях (наличие расширенных дверных проемов, поручней и других приспособлений).

Лист переутверждения рабочей программы дисциплины

Рабочая программа:

одобрена на 2020/2021 учебный год. Протокол № ____ заседания кафедры от «__» __ 2020 г.

Разработчик программы _____

Зав. кафедрой _____