

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ**

Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Воронежский государственный технический университет»

УТВЕРЖДАЮ

И.о. декана факультета информационных  
технологий и компьютерной безопасности



/ А.В. Бредихин /

19.03.2024

**РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ**

**«Программирование на платформе .NET Framework»**

**Направление подготовки 09.03.02 Информационные системы и технологии**

**Профиль Технологии искусственного интеллекта**

**Квалификация выпускника бакалавр**

**Нормативный период обучения 4 года**

**Форма обучения очная**

**Год начала подготовки 2024**

Автор программы

Д.В. Иванов

И.о. заведующего кафедрой  
систем

автоматизированного  
проектирования и  
информационных систем

П.Ю. Гусев

Руководитель ОПОП

Д.В. Иванов

Воронеж 2024

## 1. ЦЕЛИ И ЗАДАЧИ ДИСЦИПЛИНЫ

### 1.1. Цели дисциплины

Обеспечение знаний по основам разработки визуального программного обеспечения на платформе .NET Framework; изучение дисциплины должно способствовать формированию у студентов основ научного мышления, в том числе: пониманию основных сведений о принципах построения визуального программного обеспечения, особенностях организации процесса программирования на платформе .NET Framework; умению оценивать степень правильности формирования программного кода и визуального отображения информации.

### 1.2. Задачи освоения дисциплины

- получение основных сведений о технологиях построения программ на платформе .NET Framework;
- изучение видов, назначения и принципов работы визуальных компонентов программ и их возможностях;
- приобретение навыков визуального программирования.

## 2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОПОП

Дисциплина «Программирование на платформе .NET Framework» относится к дисциплинам обязательной части блока Б1.

## 3. ПЕРЕЧЕНЬ ПЛАНИРУЕМЫХ РЕЗУЛЬТАТОВ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ

Процесс изучения дисциплины «Программирование на платформе .NET Framework» направлен на формирование следующих компетенций:

ОПК-2 - Способен понимать принципы работы современных информационных технологий и программных средств, в том числе отечественного производства, и использовать их при решении задач профессиональной деятельности;

ОПК-4 - Способен участвовать в разработке технической документации, связанной с профессиональной деятельностью с использованием стандартов, норм и правил;

ОПК-6 - Способен разрабатывать алгоритмы и программы, пригодные для практического применения в области информационных систем и технологий;

Компетенция	Результаты обучения, характеризующие сформированность компетенции
ОПК-2	знать технические основы программирования на платформе .NET Framework;
	уметь осуществлять разработку визуальных программ;
	владеть методами визуального программирования;
ОПК-4	знать основы интеграционных решений приложений на платформе .NET Framework;
	уметь создавать функциональные диаграммы ПО;

	владеть навыками документирования архитектуры ПО;
ОПК-6	знать особенности организации процесса визуального программирования;
	уметь осуществлять сборку проекта визуального программного средства;
	владеть навыками визуального программирования на платформе .NET Framework.

#### 4. ОБЪЕМ ДИСЦИПЛИНЫ

Общая трудоемкость дисциплины «Программирование на платформе .NET Framework» составляет 5 з.е.

Распределение трудоемкости дисциплины по видам занятий  
**очная форма обучения**

Виды учебной работы	Всего часов	Семестры
		3
<b>Аудиторные занятия (всего)</b>	90	90
В том числе:		
Лекции	36	36
Лабораторные работы (ЛР)	54	54
<b>Самостоятельная работа</b>	54	54
<b>Курсовой проект</b>	+	+
Часы на контроль	36	36
Виды промежуточной аттестации - экзамен	+	+
Общая трудоемкость:		
академические часы	180	180
зач.ед.	5	5

#### 5. СОДЕРЖАНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)

**5.1 Содержание разделов дисциплины и распределение трудоемкости по видам занятий**  
**очная форма обучения**

№ п/п	Наименование темы	Содержание раздела	Лекц	Лаб. зан.	СРС	Всего, час
1	Основы платформы .NET Framework	Архитектура платформы .NET Framework. Модель выполнения кода в среде CLR. Двухэтапная компиляция. Компоновка, упаковка, развертывание и администрирование приложений и типов. Совместно используемые сборки. Работа CLR и домены приложений. Загрузка сборок и отражение. Сериализация.	6	10	12	28
2	Реализация основных конструкций языка C#	Базовые конструкции языка C#. Символы, строки и обработка текста. Перечислимые типы и битовые флаги. Массивы. Делегаты. Настраиваемые атрибуты. Null-совместимые значимые типы.	8	16	18	42
3	Проектирование типов	Основы типов. Примитивные, ссылочные и значимые типы. Основные сведения о членах и типах. Классы. Члены классов. Константы и поля. Методы. Параметры. Свойства. События. Обобщения. Абстрактные классы. Интерфейсы.	8	16	24	48
4	Ключевые механизмы	Исключения и управление состоянием.	8	8	24	40

		Автоматическое управление памятью (сборка мусора). Полиморфизм. Реализация взаимодействия с интерфейсами. Реализация отношений между классами.				
5	Принципы проектирования	Основные принципы проектирования ООП. Принципы проектирования SOLID. Основные принципы паттерн-программирования.	6	4	12	22
<b>Итого</b>			<b>36</b>	<b>54</b>	<b>90</b>	<b>180</b>

## 5.2 Перечень лабораторных работ

1. Разработка пользовательских интерфейсов для программирования ввода-вывода на платформе .NET Framework.

2. Разработка пользовательских интерфейсов для программирования основных операторов языка C# на платформе .NET Framework.

3. Разработка пользовательских интерфейсов для программирования конструкций условного оператора и оператора выбора на платформе .NET Framework.

4. Разработка пользовательских интерфейсов для программирования конструкций циклов на платформе .NET Framework.

5. Разработка пользовательских интерфейсов для программирования массивов на платформе .NET Framework.

6. Разработка пользовательских интерфейсов для программирования строк на платформе .NET Framework.

7. Разработка пользовательских интерфейсов для программирования подпрограмм на платформе .NET Framework.

8. Разработка интерфейса приложения для работы с файловой системой на платформе .NET Framework.

9. Разработка интерфейса приложения для работы с двоичными (бинарными) файлами на платформе .NET Framework.

10. Разработка интерфейса приложения, реализующего принципы инкапсуляции на платформе .NET Framework.

11. Разработка многооконного пользовательского интерфейса на платформе .NET Framework.

12. Разработка визуального приложения, реализующего принципы полиморфизма на платформе .NET Framework.

## 6. ПРИМЕРНАЯ ТЕМАТИКА КУРСОВЫХ ПРОЕКТОВ (РАБОТ) И КОНТРОЛЬНЫХ РАБОТ

В соответствии с учебным планом освоение дисциплины предусматривает выполнение курсового проекта в 3 семестре для очной формы обучения.

Примерная тематика курсового проекта: «Разработка программного обеспечения на платформе .NET Framework»

Задачи, решаемые при выполнении курсового проекта:

- Анализ требований к программному обеспечению.
- Анализ платформ и прикладных средств реализации.
- Требования к спецификации программного обеспечения.
- Проектирование архитектуры программного обеспечения.

- Проектирование архитектуры классов.
- Модульная структура приложения.
- Выбор средств реализации.
- Разработка алгоритмов и методов.
- Описание диалога с пользователем.
- Минимальные системные требования.

Курсовой проект включает в себя приложение на платформе .NET Framework и расчетно-пояснительную записку.

Примерная тематика вариантов курсовых проектов:

1. Программная реализация кодирования и декодирования деревьев с помощью последовательностей Прюфера.
2. Программная реализация метода Магу определения хроматического числа графа.
3. Программная реализация топологической сортировки методом Форда.
4. Программная реализация алгоритма Краскала для построения кратчайшего остова.
5. Программная реализация разработки программы построения фундаментальных разрезов и циклов неориентированного графа.
6. Программная реализация алгоритмов Робертса и Флореса для построения гамильтоновых цепей и циклов.
7. Программная реализация нахождения эйлеровых циклов и цепей в неориентированных графах.
8. Программная реализация алгоритма Форда-Фалкерсона.
9. Программная реализация алгоритма Беллмана — Форда.
10. Программная реализация решения задачи коммивояжера.

Курсовой проект включает в себя графическую часть и расчетно-пояснительную записку.

Учебным планом по дисциплине «Программирование на платформе .NET Framework» не предусмотрено выполнение контрольной работы (контрольных работ) в 3 семестре.

## **7. ОЦЕНОЧНЫЕ МАТЕРИАЛЫ ДЛЯ ПРОВЕДЕНИЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ОБУЧАЮЩИХСЯ ПО ДИСЦИПЛИНЕ**

**7.1. Описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкал оценивания**

### **7.1.1 Этап текущего контроля**

Результаты текущего контроля знаний и межсессионной аттестации оцениваются по следующей системе:

«аттестован»;

«не аттестован».

Компетенция	Результаты обучения, характеризующие сформированность компетенции	Критерии оценивания	Аттестован	Не аттестован
-------------	---	---------------------	------------	---------------

ОПК-2	знать технические основы программирования на платформе .NET Framework;	Знание теоретического материала для лабораторных работ № 1-4	Выполнение работ в срок, предусмотренный в рабочих программах	Невыполнение работ в срок, предусмотренный в рабочих программах
	уметь осуществлять разработку визуальных программ;	Выполнение лабораторных работ № 1-4	Выполнение работ в срок, предусмотренный в рабочих программах	Невыполнение работ в срок, предусмотренный в рабочих программах
	владеть методами визуального программирования;	Выполнение лабораторных работ № 1-4	Выполнение работ в срок, предусмотренный в рабочих программах	Невыполнение работ в срок, предусмотренный в рабочих программах
ОПК-4	знать основы интеграционных решений приложений на платформе .NET Framework;	Знание теоретического материала для лабораторных работ № 9-12	Выполнение работ в срок, предусмотренный в рабочих программах	Невыполнение работ в срок, предусмотренный в рабочих программах
	уметь создавать функциональные диаграммы ПО;	Выполнение лабораторных работ № 9-12	Выполнение работ в срок, предусмотренный в рабочих программах	Невыполнение работ в срок, предусмотренный в рабочих программах
	владеть навыками документирования архитектуры ПО;	Выполнение лабораторных работ № 9-12	Выполнение работ в срок, предусмотренный в рабочих программах	Невыполнение работ в срок, предусмотренный в рабочих программах
ОПК-6	знать особенности организации процесса визуального программирования;	Знание теоретического материала для лабораторных работ № 5-8	Выполнение работ в срок, предусмотренный в рабочих программах	Невыполнение работ в срок, предусмотренный в рабочих программах
	уметь осуществлять сборку проекта визуального программного средства;	Выполнение лабораторных работ № 5-8	Выполнение работ в срок, предусмотренный в рабочих программах	Невыполнение работ в срок, предусмотренный в рабочих программах
	владеть навыками визуального программирования на платформе .NET Framework.	Выполнение лабораторных работ № 5-8	Выполнение работ в срок, предусмотренный в рабочих программах	Невыполнение работ в срок, предусмотренный в рабочих программах

### 7.1.2 Этап промежуточного контроля знаний

Результаты промежуточного контроля знаний оцениваются в 3 семестре для очной формы обучения по четырехбалльной системе:

«отлично»;

«хорошо»;

«удовлетворительно»;

«неудовлетворительно».

Компетенция	Результаты обучения, характеризующие сформированность компетенции	Критерии оценивания	Отлично	Хорошо	Удовл.	Неудовл.
ОПК-2	знать технические основы программирования на платформе .NET	Вопросы к экзамену	Выполнение теста на 90-100%	Выполнение теста на 80-90%	Выполнение теста на 70-80%	В тесте менее 70% правильных ответов

	Framework; уметь осуществлять разработку визуальных программ;	Решение стандартных практических задач (тест)	Задачи решены в полном объеме и получены верные ответы	Продемонстрирован верный ход решения всех, но не получен верный ответ во всех задачах	Продемонстрирован верный ход решения в большинстве задач	Задачи не решены
	владеть методами визуального программирования;	Решение прикладных задач в конкретной предметной области (тест)	Задачи решены в полном объеме и получены верные ответы	Продемонстрирован верный ход решения всех, но не получен верный ответ во всех задачах	Продемонстрирован верный ход решения в большинстве задач	Задачи не решены
ОПК-4	знать основы интеграционных решений приложений на платформе .NET Framework;	Вопросы к экзамену	Выполнение теста на 90-100%	Выполнение теста на 80-90%	Выполнение теста на 70-80%	В тесте менее 70% правильных ответов
	уметь создавать функциональные диаграммы ПО;	Решение стандартных практических задач (тест)	Задачи решены в полном объеме и получены верные ответы	Продемонстрирован верный ход решения всех, но не получен верный ответ во всех задачах	Продемонстрирован верный ход решения в большинстве задач	Задачи не решены
	владеть навыками документирования архитектуры ПО;	Решение прикладных задач в конкретной предметной области (тест)	Задачи решены в полном объеме и получены верные ответы	Продемонстрирован верный ход решения всех, но не получен верный ответ во всех задачах	Продемонстрирован верный ход решения в большинстве задач	Задачи не решены
ОПК-6	знать особенности организации процесса визуального программирования;	Вопросы к экзамену	Выполнение теста на 90-100%	Выполнение теста на 80-90%	Выполнение теста на 70-80%	В тесте менее 70% правильных ответов
	уметь осуществлять сборку проекта программного средства;	Решение стандартных практических задач (тест)	Задачи решены в полном объеме и получены верные ответы	Продемонстрирован верный ход решения всех, но не получен верный ответ во всех задачах	Продемонстрирован верный ход решения в большинстве задач	Задачи не решены
	владеть навыками визуального программирования на платформе .NET Framework.	Решение прикладных задач в конкретной предметной области (тест)	Задачи решены в полном объеме и получены верные ответы	Продемонстрирован верный ход решения всех, но не получен верный ответ во всех задачах	Продемонстрирован верный ход решения в большинстве задач	Задачи не решены

## 7.2 Примерный перечень оценочных средств (типовые

**контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности)**

**7.2.1 Примерный перечень заданий для подготовки к тестированию**

Дополнительное тестирование не предусмотрено

**7.2.2 Примерный перечень заданий для решения стандартных задач**

Вопрос 1: Что такое ENUM?

- (A) Используется для инициализации переменных
- (B) Используется для определения констант
- (C) Он используется для определения переменных
- (D) Нет верного ответа

Вопрос 2: Что делает метод Dispose?

- (A) Закрывает соединение
- (B) Временно избавиться от соединения
- (C) Удаляет его из памяти
- (D) Все вышеперечисленное

Вопрос 3: Особенность локальной переменной

- (A) Она должна быть объявлена в методе
- (B) представляет объект класса
- (C) Может использоваться в любом месте программы
- (D) Она должна использоваться в классе

Вопрос 4: В чем разница между Convert.ToInt32 и Int.Parse?

- (A) Оба одинаковы
- (B) Convert.ToInt32 Не может обработать нулевые значения, он выдаст ошибку rgumentNullException.
- (C) Int. Анализ не может обработать нулевые значения, он выдает ошибку ArgumentException.
- (D) Оба могут обрабатывать нулевые значения

Вопрос 5: C# не поддерживает:

- (A) абстракция
- (B) полиморфизм
- (C) множественное наследование
- (D) наследство

Вопрос 6: Какое из следующих утверждений верно в отношении управляемого кода?

- (A) Управляемый код, который выполняется поверх Windows.
- (B) Управляемый код CLR.
- (C) Управляемый код - это то место, где код является сборщиком мусора.

(D) Управляемый код для компиляторов.

Вопрос 7: Можем ли мы переопределить частные виртуальные методы?

- (A) да
- (B) Нет

Вопрос 8: Какие из следующих сборок можно сохранить в Global Assembly Cache?

- (A) private
- (B) shared
- (C) protected
- (D) friend

Вопрос 9: Сборка - это

- (A) декларативный синтаксис Java
- (B) свойства C#
- (C) события C#
- (D) Коллекция файлов, которые представляются программисту в виде одной DLL или EXE.

Вопрос 10: Какие из следующих утверждений правильно определяют .NET Framework?

- (A) Это среда для разработки, создания, развертывания и выполнения настольных приложений, веб-приложений и веб-служб.
- (B) Это среда для разработки, создания, развертывания и выполнения распределенных приложений.
- (C) Это среда для разработки и исполнения приложений Windows.
- (D) Это среда для разработки, создания, развертывания и выполнения только веб-приложений.

Вопрос 11: Какое из следующих утверждений ИСТИННО в отношении .NET CLR?

1. Он обеспечивает нейтральную для языка среду разработки и исполнения.
  2. Это гарантирует, что он не авторизован для доступа.
  3. Предоставляет сервисы для запуска «управляемых» приложений.
  4. Ресурсы мусора.
  5. Предоставляет сервисы для запуска «неуправляемых» приложений.
- (A) только 1 и 2
  - (B) только 1, 2 и 4
  - (C) 1, 2, 3, 4
  - (D) только 4 и 5

Вопросы 12: Что можно использовать в считывателе данных до чтения?

- (A) getvalue

- (B) getstring
- (C) getnumber
- (D) None

Вопрос 13: Может ли интерфейс быть создан непосредственно?

- (A) да
- (B) нет

Вопрос 14: Какие из перечисленных ниже являются действительными счетчиками производительности .NET CLR JIT?

1. Общая память, используемая для компиляции JIT
2. Средняя память, используемая для компиляции JIT
3. Количество методов, которые не удалось скомпилировать со стандартным JIT

4. Процент JIT-компиляции процессорного времени
5. Процент памяти, выделенной для компиляции JIT

- (A) 1, 2
- (B) 1, 5
- (C) 3, 4
- (D) 4, 5

Вопрос 15: Автоматическая страничная подкачка возможна в

- (A) datareader
- (B) dataset
- (C) datatable
- (D) все вариант

Вопрос 16: Какой класс унаследован от класса?

- (A) Sealed
- (B) Gather
- (C) Static
- (D) Constru

Вопрос 17: Особенности переменных только для чтения

- (A) Декларация и инициализация разделены
- (B) Распределяется во время компиляции
- (C) Распределяется во время выполнения
- (D) все вышеперечисленное

Вопрос 18: Событие имеет \_\_\_\_\_ в качестве типа возврата по умолчанию

- (A) Нет типа возврата для событий
- (B) String
- (C) Double
- (D) Integer

Вопрос 19: Код, предназначенный для общезыковой среды выполнения, известен как

- (A) Наследие
- (B) Управляемый код
- (C) Неуправляемый
- (D) родной код

Вопрос 20: Какой метод вы вызываете для объекта SqlDataAdapter для загрузки вашего набора данных с данными?

- (A) Обновить
- (B) Заполнить
- (C) Собрать
- (D) Загрузить

### 7.2.3 Примерный перечень заданий для решения прикладных задач

Вопрос 1: Могут ли частные переменные уровня класса наследоваться?

- (A) Да, и мы можем получить к ним доступ
- (B) Нет, и мы не можем получить к ним доступ
- (C) Да, но мы не можем получить к ним доступ
- (D) Все вышеперечисленное неверно

Вопросы 2: Есть ли ошибки в этом конструкторе EmployeeMgmt: Public  
int EmployeeMgmt {emp\_id = 100; }

- (A) Тип возврата
- (B) Формальные параметры
- (C) ошибок нет
- (D) Имя

Вопрос 3: Что верно в отношении интерфейса и абстрактных методов?  
(A) Мы можем написать только один абстрактный метод внутри интерфейса.

- (B) Ни один метод не является абстрактным внутри интерфейса
- (C) В абстрактном методе.
- (D) Ничего из вышеперечисленного

Вопрос 4: Какой объект вы можете использовать?

- (A) Объект сеанса
- (B) Объект сервера
- (C) Объект ответа
- (D) Объект приложения

Вопрос 5: Два метода с одинаковым именем, но с разными параметрами.

- (A) перегрузка

- (B) мультиплексирование
- (C) дуплекс
- (D) Загрузка

Вопрос 6: В чем разница между методом Convert.ToString (str) и str.ToString ()?

- (A) Функция Convert.ToString (str) обрабатывает NULL, а функция str.ToString () - нет. Будет выброшено исключение NULL.
- (B) функция str.ToString () обрабатывает NULL, а Convert.ToString (str) - нет. Будет выброшено исключение NULL.
- (C) Оба могут обрабатывать NULL
- (D) Никто не может обращаться с NULL

Вопрос 7: Какое из этих определений строк предотвратит обратную косую черту в C#?

- (A) строка s = #.n Тестовая строка .;
- (B) строка s = @.n Тестовая строка .;
- (C) строка s = ..n тестовая строка .;
- (D) строка s = .n тестовая строка .;

Вопрос 8: Разница между Convert.ToString () и ToString ()

- (A) Convert.ToString () обрабатывает нулевые значения, но ToString () нет
- (B) Convert.ToString () обрабатывает только нулевые значения
- (C) ToString () обрабатывает нулевые значения, но Convert.ToString () нет
- (D) Вывод ToString () в соответствии с предоставленным форматом

Вопрос 9: Как мы можем отсортировать элементы массива в порядке убывания?

- (A) вызывая метод Sortdescending()
- (B) вызывая метод Sort()
- (C) Вызывая методы Sort() и затем Reverse()
- (D) вызывая метод Sortreverse()

Вопрос 10: Что из следующего составляет .NET Framework?

1. Приложения ASP.NET
  2. CLR
  3. Framework Class Library
  4. Приложения WinForm
  5. Службы Windows
- (A) 2, 5
  - (B) 2, 1
  - (C) 2, 3
  - (D) 3, 4

#### **7.2.4 Примерный перечень вопросов для подготовки к зачету**

Не предусмотрено учебным планом

#### **7.2.5 Примерный перечень заданий для подготовки к экзамену**

1. Архитектура платформы .NET Framework.
2. Библиотека классов FCL.
3. Общезыковая исполнительная среда CLR.
4. Двухэтапная компиляция.
5. Общие спецификации и совместимые модули.
6. Классификация типов данных в C#. Применение типов данных.
7. Явные и неявные преобразования типов. Работа с типами данных.
8. Ссылочные и значимые типы данных. Стек и куча.
9. Небезопасный код. Указатели в языке C#.
10. Автоматическое управление памятью (сборка мусора).
11. Диаграмма классов языка UML. Отношения между классами.
12. Инкапсуляция. Класс. Объект. Структура класса.
13. Члены класса. Применение свойств без параметров.
14. Члены класса. Применение индексаторов.
15. Члены класса. Применение конструкторов и деструкторов.
16. Наследование. Применение модификаторов доступа.
17. Полиморфизм. Перегрузка методов.
18. Полиморфизм. Переопределение виртуальных методов.
19. Абстрактные классы. Абстрактные члены класса. Применение абстрактных классов.
20. Интерфейсы. Применение интерфейсов.
21. Обработка исключительных ситуаций.
22. Базовые принципы проектирования.
23. Принципы проектирования SOLID.
24. Структура консольного проекта и проекта Windows Forms на языке C# в среде Visual Studio (.NET Framework).
25. Работа с визуальными компонентами среды Visual Studio.
26. Базовые конструкции языка C#.
27. Двоичные операции. Постфиксная и префиксная формы.
28. Массивы. Работа с массивами в C#.
29. Строки. Работа со строками в C#.
30. Коллекции. Работа с коллекциями в C#.
31. Именованные и неименованные параметры методов. Параметры методов по умолчанию.
32. Передача параметров по ссылке и значению: модификаторы ref, out, in.
33. Текстовые типы файлов. Работа с текстовыми файлами в C#.
34. Бинарные типы файлов. Работа с бинарными файлами в C#.

#### **7.2.6. Методика выставления оценки при проведении промежуточной аттестации**

Экзамен проводится по билетам, каждый из которых содержит 10 тестовых вопросов и 2 теоретических вопроса для устного ответа. Каждый

правильный ответ на вопрос в тесте оценивается 1 баллом; ответ на теоретический вопрос оценивается до 5 баллов. Максимальное количество набранных баллов – 20.

Оценка ответа на теоретический вопрос.

5 баллов – студент продемонстрировал глубокое, всестороннее знание вопроса. Информация изложена последовательно, системно и логично. Вывода правильны и обоснованы. Приведены различные точки зрения и мнения по заданному вопросу. Изложенные фактические и иные данные подкреплены ссылками на актуальные нормативные акты и другие источники. Обозначено собственное оригинальное мнение по вопросу.

4 балла – студент продемонстрировал глубокое, всестороннее знание вопроса. Материал изложен логично и последовательно. В целом ответ верен. Однако студент не смог продемонстрировать собственной позиции по вопросу, не привел разнообразных мнений и точек зрения по нему или не смог обосновать ответ ссылками на нормативные акты и иные источники. Допускаются некоторые незначительные неточности по вопросу.

3 балла – студент продемонстрировал неполные, не достаточно глубокие знания по вопросу либо допустил достаточные фактические ошибки. Отсутствует самостоятельный анализ и выводы по вопросу. Материал изложен непоследовательно или бессистемно.

2 балла – студент продемонстрировал слабые знания по вопросу, однако относительно удачно попытался подменить их собственными оригинальными выводами близкими к правильным.

1 балл – студент продемонстрировал крайне слабые знания по вопросу, однако пытался самостоятельно прийти к выводам по заданной теме, попытки анализа первоисточников достигли результатов близких к правильным.

0 баллов – студент продемонстрировал полное незнание материала и не предпринял попыток добиться верных результатов.

Методика выставления итоговой оценки по экзамену.

1. Оценка «Неудовлетворительно» ставится в случае, если студент набрал менее 12 баллов.

2. Оценка «Удовлетворительно» ставится в случае, если студент набрал от 12 до 14 баллов.

3. Оценка «Хорошо» ставится в случае, если студент набрал от 15 до 17 баллов.

4. Оценка «Отлично» ставится, если студент набрал от 18 баллов.

### 7.2.7 Паспорт оценочных материалов

№ п/п	Контролируемые разделы (темы) дисциплины	Код контролируемой компетенции	Наименование оценочного средства
-------	--	--------------------------------	----------------------------------

1	Основы CLR	ОПК-2	Вопросы к экзамену, защита лабораторных работы, решение стандартных задач (тест), решение прикладных задач (тест)
2	Проектирование типов	ОПК-2	Вопросы к экзамену, защита лабораторных работы, решение стандартных задач (тест), выполнение предпроектной части КП
3	Основные типы данных	ОПК-6	Вопросы к экзамену, защита лабораторных работы, решение стандартных задач (тест), решение прикладных задач (тест)
4	Ключевые механизмы	ОПК-4, ОПК-6	Вопросы к экзамену, защита лабораторных работы, решение стандартных задач (тест), реализация основных видов проектирования в КП
5	Принципы проектирования	ОПК-6	Вопросы к экзамену, защита лабораторных работы, решение стандартных задач (тест), реализация интерфейса ПО в КП

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

Экзамен проводится в письменной форме. Время на теоретические вопросы 20 мин. Осуществляется проверка теста экзаменатором и выставляется оценка согласно методике выставления оценки при проведении промежуточной аттестации.

Решение стандартных задач осуществляется, либо при помощи компьютерной системы тестирования, либо с использованием выданных задач на бумажном носителе. Время решения задач 10 мин. Затем осуществляется проверка решения задач экзаменатором и выставляется оценка, согласно методике выставления оценки при проведении промежуточной аттестации.

Решение прикладных задач осуществляется, либо при помощи компьютерной системы тестирования, либо с использованием выданных задач на бумажном носителе. Время решения задач 10 мин. Затем осуществляется проверка решения задач экзаменатором и выставляется оценка, согласно методике выставления оценки при проведении промежуточной аттестации.

Защита курсового проекта осуществляется согласно требованиям, предъявляемым к работе, описанным в методических материалах. Примерное время защиты на одного студента составляет 20 мин.

## **8 УЧЕБНО МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ)**

### **8.1 Перечень учебной литературы, необходимой для освоения дисциплины**

1. Кариев Ч.А. Разработка Windows-приложений на основе Visual C# [Электронный ресурс]: учебное пособие/ Кариев Ч.А.— Электрон. текстовые данные.— Москва, Саратов: Интернет-Университет Информационных Технологий (ИНТУИТ), Вузовское образование, 2017.— 768 с.— Режим доступа: <http://www.iprbookshop.ru/72340.html>.— ЭБС «IPRbooks».

2. Биллиг В.А. Основы объектного программирования на C# (C# 3.0, Visual Studio 2008) [Электронный ресурс]: учебное пособие/ Биллиг В.А.— Электрон. текстовые данные.— Москва, Саратов: Интернет-Университет Информационных Технологий (ИНТУИТ), Вузовское образование, 2017.— 583 с.— Режим доступа: <http://www.iprbookshop.ru/72339.html>.— ЭБС «IPRbooks».

3. Разработка Windows-приложений в среде программирования Visual Studio.Net [Электронный ресурс]: учебно-методическое пособие по дисциплине Информатика и программирование/ — Электрон. текстовые данные.— М.: Московский технический университет связи и информатики, 2016.— 20 с.— Режим доступа: <http://www.iprbookshop.ru/61536.html>.— ЭБС «IPRbooks».

4. Иванов Д.В., Тишуков Б.Н. Программирование на языках высокого уровня [Электронный ресурс] / Учебно-методическое пособие. - Электрон. текстовые, граф. дан. ( 1,03 Мб ). - Воронеж : ФГБОУ ВО "Воронежский государственный технический университет", 2016. - Режим доступа: <https://cloud.cchgeu.ru/index.php/s/AT9Z9scBaDN5obw>

5. Ганцева Е.А. Программирование на C# : Учеб. пособие. - Воронеж : ФГБОУ ВПО "Воронежский государственный технический университет", 2012. - 135с.

6. Сергеев М.Ю. Программирование и основы алгоритмизации : Учеб. пособие. - Воронеж : ГОУВПО "Воронежский государственный технический университет", 2010. - 231 с. - 76-53; 250 экз.

**8.2 Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень лицензионного программного обеспечения, ресурсов информационно-телекоммуникационной сети «Интернет», современных профессиональных баз данных и информационных справочных систем:**

Перечень ПО, включая перечень лицензионного программного обеспечения:

Microsoft Windows 64-bit

Microsoft Visual Studio Community  
Google Chrome  
LibreOffice

Ресурсы информационно-телекоммуникационной сети «Интернет»:

<http://window.edu.ru>

<http://www.edu.ru/>

Образовательный портал ВГТУ

Современные профессиональные базы данных и информационно-справочные системы:

<https://proglib.io/>

<https://habr.com/ru/>

<https://docs.microsoft.com/ru-RU/dotnet/csharp/>

## **9 МАТЕРИАЛЬНО-ТЕХНИЧЕСКАЯ БАЗА, НЕОБХОДИМАЯ ДЛЯ ОСУЩЕСТВЛЕНИЯ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА**

Для проведения обучения по дисциплине используется компьютерный класс. Аудитория:

Компьютерный класс

Учебная аудитория для проведения учебных занятий

Комплект учебной мебели:

– рабочее место преподавателя (стол, стул);

– рабочие места обучающихся (столы, стулья);

– персональные компьютеры с установленным ПО, подключенные к сети Интернет (12 шт.);

– принтер;

– доска магнитно-маркерная поворотная

Помещение для самостоятельной работы. Читальный зал с возможностью подключения к сети "Интернет" и обеспечением доступа в электронную информационно-образовательную среду.

## **10. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ (МОДУЛЯ)**

По дисциплине «Программирование на платформе .NET Framework» читаются лекции, проводятся лабораторные работы, выполняется курсовой проект.

Основой изучения дисциплины являются лекции, на которых излагаются наиболее существенные и трудные вопросы, а также вопросы, не нашедшие отражения в учебной литературе.

Лабораторные работы выполняются на лабораторном оборудовании в соответствии с методиками, приведенными в указаниях к выполнению работ.

Методика выполнения курсового проекта изложена в учебно-методическом пособии. Выполнять этапы курсового проекта должны своевременно и в установленные сроки.

Контроль усвоения материала дисциплины производится проверкой

курсового проекта, защитой курсового проекта.

Вид учебных занятий	Деятельность студента
Лекция	Написание конспекта лекций: кратко, схематично, последовательно фиксировать основные положения, выводы, формулировки, обобщения; помечать важные мысли, выделять ключевые слова, термины. Проверка терминов, понятий с помощью энциклопедий, словарей, справочников с выписыванием толкований в тетрадь. Обозначение вопросов, терминов, материала, которые вызывают трудности, поиск ответов в рекомендуемой литературе. Если самостоятельно не удастся разобраться в материале, необходимо сформулировать вопрос и задать преподавателю на лекции или на практическом занятии.
Лабораторная работа	Лабораторные работы позволяют научиться применять теоретические знания, полученные на лекции при решении конкретных задач. Чтобы наиболее рационально и полно использовать все возможности лабораторных для подготовки к ним необходимо: следует разобрать лекцию по соответствующей теме, ознакомиться с соответствующим разделом учебника, проработать дополнительную литературу и источники, решить задачи и выполнить другие письменные задания.
Самостоятельная работа	Самостоятельная работа студентов способствует глубокому усвоению учебного материала и развитию навыков самообразования. Самостоятельная работа предполагает следующие составляющие: <ul style="list-style-type: none"><li>- работа с текстами: учебниками, справочниками, дополнительной литературой, а также проработка конспектов лекций;</li><li>- выполнение домашних заданий и расчетов;</li><li>- работа над темами для самостоятельного изучения;</li><li>- участие в работе студенческих научных конференций, олимпиад;</li><li>- подготовка к промежуточной аттестации.</li></ul>
Подготовка к промежуточной аттестации	Готовиться к промежуточной аттестации следует систематически, в течение всего семестра. Интенсивная подготовка должна начаться не позднее, чем за месяц-полтора до промежуточной аттестации. Данные перед экзаменом три дня эффективнее всего использовать для повторения и систематизации материала.

## ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ

№ п/п	Перечень вносимых изменений	Дата внесения изменений	Подпись заведующего кафедрой, ответственной за реализацию ОПОП
----------	-----------------------------	----------------------------	--