

1. ЦЕЛИ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Целью освоения дисциплины «Практикум по решению задач на ЭВМ» является развитие у студентов компетенций по работе с современными технологиями в программировании для решения прикладных задач.

Задачи дисциплины:

- Формирование навыков работы с программной платформой .NET Framework.
- Развитие принципов разработки алгоритмов и программ, их оптимизации.
- Изучение и использование различных методов программирования.
- Формирование опыта разработки алгоритмов и решения задач.
- Освоение понятий и принципов ООП в рамках платформы .NET Framework.
- Изучение возможностей платформы .NET в образовательной деятельности.

2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОПОП ВО

Дисциплина «Практикум по решению задач на ЭВМ» относится к вариативной части учебного плана по направлению 44.03.05 «Педагогическое образование».

Пререквизиты дисциплины: «Современные ИТ», «Программирование», «Теория алгоритмов», «Теоретическая информатика», «Методика обучения информатике».

3. ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ

Планируемые результаты обучения по дисциплине, соотнесенные с планируемыми результатами освоения ОПОП

Код формируемых компетенций	Уровень освоения компетенции	Планируемые результаты обучения по дисциплине характеризующие этапы формирования компетенций (показатели освоения компетенции)
1	2	3
ОК-6	Частичное	<i>Знать:</i> <ul style="list-style-type: none">• основные подходы к научной постановке и решению практических задач. <i>Уметь:</i> <ul style="list-style-type: none">• использовать современные компьютерные технологии при решении задач;• осуществлять согласованную работу в коллективе из нескольких человек в целях достижения поставленной учебной задачи. <i>Владеть:</i> <ul style="list-style-type: none">• навыками работы с справочными системами.
ПК-1	Частичное	<i>Знать:</i> <ul style="list-style-type: none">• теоретическую основу, важнейшие особенности и возможности программной платформы .NET Framework;• синтаксис языка программирования C# и реализацию принципов ООП средствами .NET Framework;• методы и приемы реализации основных алгоритмов на базе объектно-ориентированного языка программирования;• приемы разработки программных продуктов на базе компонентной модели. <i>Уметь:</i> <ul style="list-style-type: none">• просеивать полученные знания для реализации педагогических задач в процессе обучения основам алгоритмизации и программирования. <i>Владеть:</i> <ul style="list-style-type: none">• приемами современного программирования и проектирования приложений;• навыками работы со справочными системами по технологии NET Framework и языку программирования C#.

ПК-11	Частичное	<p><i>Знать:</i></p> <ul style="list-style-type: none"> • возможности реализации математических абстракций на базе ООП-модели; • приемы разработки программных продуктов на базе компонентной модели. <p><i>Уметь:</i></p> <ul style="list-style-type: none"> • моделировать научные и практические задачи средствами .NET Framework; • использовать современное инструментальное ПО для создания программных продуктов и учебных проектов; • организовывать учебную работу с технологией .NET. <p><i>Владеть:</i></p> <ul style="list-style-type: none"> • приемами исследования математических задач средствами .NET Framework; • приемами современного программирования и проектирования приложений.
-------	-----------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

4. ОБЪЕМ И СТРУКТУРА ДИСЦИПЛИНЫ

Трудоемкость дисциплины составляет 4 зачетных единицы, 144 часов

№ п/п	Наименование тем и/или разделов/тем дисциплины	Семестр	Неделя семестра	Виды учебной работы, включая самостоятельную работу студентов и трудоемкость (в часах)				Объем учебной работы, с применением интерактивных методов (в часах /%)	Формы текущего контроля успеваемости, форма промежуточной аттестации (по семестрам)
				Лекции	Практические занятия	Лабораторные работы	СРС		
1	Платформа .NET и язык программирования C#. Основы синтаксиса C#.	9	7-8			4	12	1/25%	
2	Классы. Структуры и перечисления.	9	9-10			6	18	2/33.3%	Рейтинг-контроль №1
3	Строки и файлы.	9	11-12			4	14	1/25%	
4	Обработка исключений.	9	13-14			2	12	1/50%	Рейтинг-контроль №2
5	Наследование и полиморфизм.	9	15-16			4	14	1/25%	
6.	Интерфейсы. Делегаты. События.	9	17-18			4	14	1/25%	Рейтинг-контроль №3
Всего за 9 семестр:						24	84	7/29%	зачет
1.	Перегрузка операторов. Методы расширения.	10	1-4			2	2	1/25%	
2.	Разработка проектов с визуальным интерфейсом на базе технологии WindowsForms.	10	5-8			8	8	2/25%	Рейтинг-контроль №1
3.	Программирование практических задач.	10	9-12			8	4	4/50%	Рейтинг-контроль №2
4.	Язык C# в качестве инструмента обучения программированию.	10	13-18			2	2	1/50%	Рейтинг-контроль №3
Всего за 10 семестр:						20	16	8/40%	Зачет с оценкой
Наличие в дисциплине КИ-КР						+			
Итого по дисциплине						44	100	17/35,4%	Зачет, зачет с оценкой, КР

9 семестр

Содержание лабораторных занятий по дисциплине

Раздел 1. Платформа .NET и язык программирования C#. Основы синтаксиса C#

Тема 1 Платформа .NET Framework и ее возможности.

Тема 2 Язык C#. Среда разработки Visual Studio Code.

Типы данных. Арифметические и логические операторы. Условные и циклические конструкции C#. Массивы. Функции. Примеры использования.

Раздел 2. Классы. Структуры и перечисления.

Тема 1 Классы как тип данных.

Поля класса. Методы класса. Конструкторы класса. Модификатор static. Использование структур в качестве параметров методов.

Тема 2 Перечислимый тип как механизм повышения качества логики построения приложения.

Раздел 3. Строки и файлы

Тема 1 Методы класса System.String. Примеры обработки строк.

Тема 2 Класс StringBuilder.

Тема 3 Введение в регулярные выражения.

Пространство имен System.IO. Запись и чтение текстовых файлов. Работа с жестким диском. Решение задач.

Раздел 4. Обработка исключений

Тема 1 Перехват исключений. Класс Exception. Обработка многочисленных исключений.

Раздел 5. Наследование и полиморфизм

Тема 1 Основы наследования.

Конструкторы класса в наследовании. Виртуальные методы. Абстрактные классы. Решение задач.

Раздел 6. Интерфейсы. Делегаты. События

Тема 1 Интерфейсы и их роль в иерархии типов.

Тема 2 Интерфейсные свойства.

Делегаты. Групповой вызов и адресация делегируемых методов. Делегаты Action<T> и Func<T>. События как методы обратного вызова. Связь событий и делегатов. Решение практических задач.

10 семестр

Содержание лабораторных занятий по дисциплине

Раздел 1. Перегрузка операторов. Методы расширения

Тема 1 Понятие перегрузки, ее применение в реальных задачах. Синтаксис перегрузок.

Тема 2 Изучение возможности перегрузки операторов на базе реализации модуля класса комплексных чисел.

Роль методов расширения, дополнительный функционал. Методика использования методов расширения.

Раздел 2. Разработка проектов с визуальным интерфейсом на базе WindowsForms

Тема 1 Приложение с визуальным интерфейсом.

Тема 2 Редактор форм. Создание и настройка свойств форм.

Тема 3 Элементы управления формы. Основные события и свойства элементов класса Control.

Раздел 3. Программирование практических задач

Тема 1 Правила стандартизации и документирования кода.

Тема 2 Приемы оптимизации фрагментов кода.

Тема 3 Разработки библиотек для работы с комплексными числами и векторами.

Раздел 4. Язык C# в качестве инструмента обучения программированию

Тема 1 Сравнение языка C# с языками Python, C++.

Тема 2 Особенности C#.

Тема 3 Работа с открытыми средами разработки (SharpDevelop, MonoDevelop).

5. ОБРАЗОВАТЕЛЬНЫЕ ТЕХНОЛОГИИ

В преподавании дисциплины «ПРЗ на ЭВМ» используются разнообразные образовательные технологии – как традиционные, так и с применением активных и интерактивных методов обучения.

В работе используются следующие активные и интерактивные методы обучения:

- Интерактивное занятие (9 семестр, разделы 1, 2);
- Анализ ситуаций (9 семестр, раздел 3);
- Разбор конкретных ситуаций (9 семестр, раздел 2, 3, 5);
- Групповая дискуссия (10 семестр, разделы 2, 4);
- Тренинг (10 семестр, раздел 3);
- Работа над проектными заданиями (10 семестр, раздел 2-3).

6. ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ ТЕКУЩЕГО КОНТРОЛЯ УСПЕВАЕМОСТИ, ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ПО ИТОГАМ ОСВОЕНИЯ ДИСЦИПЛИНЫ И УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ САМОСТОЯТЕЛЬНОЙ РАБОТЫ СТУДЕНТОВ

9 семестр

Текущий контроль успеваемости

Рейтинг-контроль №1

1. Разработайте статический класс MyMath с методами, позволяющими вычислять модуль числа, находить целую и дробную часть указанного числа, вычислять факториал.
2. Разработать класс Figure, описывающий геометрические фигуры на плоскости. Класс обладает свойствами «тип» и «площадь». Типы описать перечислениями и ограничить треугольником, прямоугольником и окружностью. Продемонстрировать применение экземпляров этого класса.

Рейтинг-контроль №2

1. Создайте интерфейс Supercar, наследующий интерфейс Luxury (роскошь) с дополненным методом moreInfo(), выводящим небольшое сообщение о суперкарах в целом.
2. Цель: Создать проект, реализующий абстрактные классы и методы для некоторых элементарных геометрических фигур в двумерном и трехмерном пространстве.

Ход работы: Проект содержит два абстрактных класса Figure2 и Figure3, задающих абстрактные методы:

```
abstract class Figure2
{
    public abstract double Square();
    public abstract double Perimetr();
    public abstract void Type();
}
abstract class Figure3
{
    public abstract double Volume(); // объем
    public abstract double Surface(); // площадь поверхности
    public abstract void Type();
}
```

Первые два метода определяют числовые характеристики, последний – тип фигуры в своем классе.

Описать несколько классов, наследующих абстрактные. Каждый класс должен содержать конструктор, получающий в качестве параметров длины сторон (ребер), радиусов и т.п. в зависимости от типа фигуры. Например:

```
class Triangle: Figure2
{
    // реализация
}
// ... ..
Triangle Tr1 = new Triangle(2,1,2);
Console.WriteLine("Тип треугольника: ");
Tr1.Type();
Console.WriteLine("P={0} S={1}",Tr1.Perimetr(),Tr1.Square());
// При выводе на консоль получим сообщение:
// Тип треугольника:
// равнобедренный
// P=5 S=0,96...
```

В качестве классов-фигур рассмотреть треугольник, трапецию (обобщает в т.ч. параллелограмм, ромб, прямоугольник, квадрат), круг; прямоугольный параллелепипед, цилиндр, шар.

Рейтинг-контроль №3

1. Создайте приложение WindowsForms. Поместите текстовое поле на форму. Требуется, чтобы во время работы приложения в заголовке окошка отображался текущий размер формы, а в текстовом поле – ее положение. Любое изменение размера и положения должно автоматически отслеживаться соответствующими событиями.
2. Создайте приложение WindowsForms. Поместите на форму кнопку. При наведении курсора на область кнопки она должна изменить положение случайным образом. Если пользователю удастся нажать на нее, то выводится сообщение о победе в игре.

Промежуточная аттестация по итогам освоения дисциплины (зачет).

1. Платформа .NET Framework. Возможности.
2. Язык программирования C#.
3. Типы данных C#. Основные операторы.
4. Инструкции условного выбора.
5. Циклические инструкции.
6. Массивы.
7. Понятие функции.
8. Классы и их компоненты.
9. Статические компоненты класса. Статические классы.
10. Структуры и перечисления.
11. Использование структур в качестве параметров методов.
12. Перечислимый тип как механизм повышения качества логики построения приложения.
Обработка данных перечислимого типа.
13. Строки и файлы.
14. Обработка строк. Методы класса System.String.
15. Запись и чтение текстовых файлов.
16. Работа с жестким диском.
17. Обработка исключений и перехват исключений.
18. Наследование и полиморфизм
19. Виртуальные методы и абстрактные классы.
20. Интерфейсы и их роль в иерархии типов.

21. Интерфейсные свойства.
22. Делегаты.
23. Обобщения.
24. Назначение обобщений и их применение.

Темы курсовых работ

1. Верстка документов в издательской системе LaTeX.
2. Применение табличного процессора MS Excel в задачах моделирования.
3. Графические возможности языка C#.
4. Верстка веб-страниц на базе современных технологий HTML5 и CSS3.
5. Основы программирования приложений с использованием кроссплатформенного фреймворка Qt.
6. Современные веб-технологии в работе учителя информатики (математики).
7. Современные мультимедийные технологии в дистанционном образовании.
8. Решение уравнений в системе MathCad.
9. Электронные учебные курсы как инструмент дистанционного обучения.
10. Экспертные системы и их использование в сфере образования.
11. Изучение языка программирования Python в школьном курсе информатики и ИКТ.
12. Развитие творческих способностей старшеклассников в процессе изучения содержательной линии «Формализация и моделирование».
13. Развитие творческих способностей старшеклассников в процессе изучения содержательной линии «Алгоритмизация и программирование».
14. Разработка лабораторного учебного практикума по работе с PowerPoint.
15. Использование систем компьютерной алгебры для решения прикладных задач по математике и информатике в рамках школьного обучения.
16. Тема «Сортировка массива» и методика ее изучения в профильном курсе «Информатика и ИКТ».
17. Методика изучения циклов в школьном курсе информатики по содержательной линии «Алгоритмизация и программирование».
18. Организация деловых игр на уроках информатики.
19. Разработка учебно-методического материала по теме «Базы данных. СУБД» в рамках школьного курса информатики.
20. Метод проектов в реализации творческого потенциала учащихся на уроках информатики.
21. Обучение школьников решению математических задач с использованием прикладного ПО.
22. Технология проблемного обучения при организации уроков информатики и ИКТ.
23. Методика изучения архитектуры современных компьютеров в школьном курсе информатики и ИКТ.

Вопросы для самостоятельной работы

1. Статичные классы в пространстве имен System.
2. Строковые операции. Исследование задач, требующих работы со строками.
3. Регулярны выражения. Преимущества и недостатки.
4. Абстрактные классы. Реализация математических моделей.
5. Наследование и полиморфизм. Целесообразность наследования при реализации собственных классов и сборок.
6. Делегаты. Принципы построения. Связь с указателями на методы в языке C++. Вопросы безопасности.
7. Интерфейсы и абстрактные классы.
8. Обобщения как средство расширения возможностей работы с компонентами.

Задания для проектной деятельности

1. Реализуйте класс Complex, позволяющий работать с комплексными числами в удобной форме.

Класс позволяет:

- создавать комплексные числа (с помощью параметризованного конструктора);
- складывать, вычитать, умножать и делить числа (с помощью статических методов);
- вычислять модуль и аргумент числа;
- выводить результат в удобной форме.

2. Реализуйте класс Matrix, позволяющий работать с квадратными матрицами. Класс позволяет:

- создавать матрицы по указанной размерности или массиву;
- складывать, вычитать, умножать, матрицы, домножать матрицу на число.
- выводить результат в удобной форме.

Пример дополнительного материала для организации самостоятельной работы

Обработка поведения элементов перечисления

Перечисления задают логику построения, но не определяют поведение элементов объекта в каждом конкретном значении. Эта работа должна быть проделана отдельно. Обычно все элементы перечисления равнозначны в плане реализации. Т.е. алгоритм их обработки одинаков, а разница лишь в итоговом значении. Так, в нашей задаче определено два перечисления, отвечающие за тип фигуры и ее цвет, то однотипная обработка очевидна. Если мы хотим на выводе получать названия цветов и типов фигур на родном языке, то необходимо добавить функцию, которая для каждого значения задаст корректное соответствие.

Перечисления

```
using System;
enum FigureType { Triangle, Rectangle, Circle };
enum FigureColor { Red, Green, Blue, Black };
class Figure
{
    public FigureType Type { get; set; }
    public FigureColor Color { get; set; }
    public Figure(FigureType type, FigureColor color)
    {
        Type = type;
        Color = color;
    }
    // метод выводит общие данные по объекту
    public void Info()
    {
        Console.WriteLine("Тип: " + GetFigureType(Type));
    }
    // метод возвращает название указанного цвета на русском языке
    private string GetFigureType(FigureType type)
    {
        switch (type)
        {
            case FigureType.Triangle:
                return "Треугольник";
            case FigureType.Rectangle:
                return "Квадрат";
            default:
                return "Круг";
        }
    }
}
class Program
{
    static void Main()
    {
        // создание объекта и инициализация его свойств
```



```

    Figure fig = new Figure(
        FigureType.Rectangle,
        FigureColor.Black
    );
    fig.Info();
    Console.ReadKey();
}
}

```

Обратите внимание, что методу `GetFigureType` определен закрытый доступ. Таким образом мы скрываем его работу внутри класса и запрещаем внешний вызов метода (что и нежелательно). Разумеется, аналогичный механизм необходимо проделать и для элементов перечисления `FigureColor`.

Инкапсуляция по смыслу

Последнее, что необходимо отметить в нашей задаче – достаточно ограниченное распространение типов `FigureType` и `ColorType`. Класс `Figure` весьма узкопрофилирован, и нет смысла распространять перечисления на другие классы. Разумная идея – внедрить перечисления `FigureType` и `ColorType` внутри класса `Figure`.

Перечисления

```

using System;
class Figure
{
    public enum FigureType { Triangle, Rectangle, Circle };
    public enum FigureColor { Red, Green, Blue, Black };
    // свойства, конструктор, вывод и т.д.
    // ...
}
class Program
{
    static void Main()
    {
        // создание объекта и инициализация его свойств
        Figure fig = new Figure(
            Figure.FigureType.Rectangle,
            Figure.FigureColor.Black
        );
        fig.Color = Figure.FigureColor.Green;
        fig.Info();
        Console.ReadKey();
    }
}

```

Перечисления в примере являются уже элементами класса, поэтому его необходимо будет указывать при каждом обращении к значению перечисления.

10 семестр

Текущий контроль успеваемости

Рейтинг-контроль №1

1. разработать класс, реализующий операции над векторами в трехмерном евклидовом пространстве. С помощью перегрузки операторов определить следующие операции над векторами:
 - сложение / разность;
 - умножение на скаляр;
 - скалярное умножение векторов;
 - метод, вычисляющий модуль вектора.
2. Объясните, какую роль играет ссылка `this` в контексте создания следующего метода расширения:

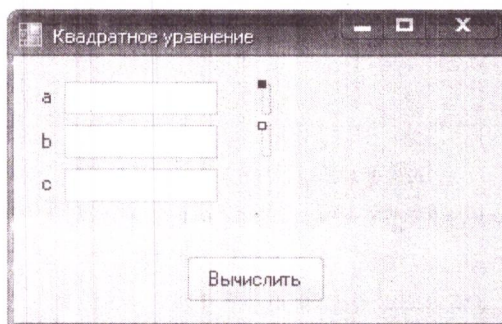

```
public static bool Contains_Sim(this String str, char sim)
```

Рейтинг-контроль №2

1. Создайте библиотеку динамической компоновки, реализующую класс SortArray с методами, осуществляющими пузырьковую сортировку и сортировку простым выбором. Во внешней программе продемонстрировать подключение и использование возможностей библиотеки.
2. Разработать библиотеку динамической компоновки, реализующую рациональные числа, а также операции сложения и разности (с помощью перегрузки операторов). Во внешней программе продемонстрировать подключение и использование возможностей библиотеки.

Рейтинг-контроль №3

1. Разработать GUI приложение, решающее квадратное уравнение. Заявленный визуальный интерфейс:



2. Дополнительно, обработать ситуацию, при которой в случае ввода в текстовые поля нечисловых значений они сбрасываются, выводится сообщение о недопустимости значения, и выставляются некоторые значения по умолчанию.

Промежуточная аттестация по итогам освоения дисциплины (зачет с оценкой).

1. Понятие перегрузки операторов. Математические задачи, реализуемые с помощью перегрузки.
2. Перегрузка операторов на базе реализации модуля класса комплексных чисел.
3. Перегрузка операторов на базе реализации модуля класса матриц.
4. Методы расширения. Роль методов расширения, дополнительный функционал.
5. Разработка проектов WindowsForms.
6. Приложения с визуальным интерфейсом.
7. Компоненты формы. Основные свойства и события.
8. Разработка GUI с помощью Visual Studio, SharpDevelop.
9. Реализация меню.
10. Особенности преподавания языка C# в школьном курсе информатики.

Вопросы для самостоятельной работы

1. Реализация абстрактной модели векторных пространств и основных операций над векторами средствами C#.
2. Модель рационального числа. Реализация операций над рациональными числами
3. Проблемы использования DLL.
4. Использование библиотеки .NET Framework для получения информации о компьютере.
5. Основные классы для работы с сетью Интернет. Примеры подключения и обработки запросов.
6. Работа с графическими примитивами.
7. Программирование GUI в рамках школьного курса. Проблемы и трудности в реализации подхода.

Задания для проектной деятельности

Разработать класс, реализующий основные операции над квадратными матрицами. С помощью перегрузки операторов определить следующие операции:

- сложения;

- разности;
- умножения матрицы на матрицу;
- умножение матрицы на число;
- изменение знака на противоположный.

Перегрузить оператор равенства, проверяющий матрицы на равенство.

Предполагается, что операции совершаются над матрицами одинаковой размерности. В противном случае возвращается ложный результат.

В классе определить два конструктора: первому передается двумерный массив, отвечающий за матрицу, а второму передается число (размерность матрицы) и создается матрица из нулей.

Пример дополнительного материала для организации самостоятельной работы

Класс ImageBox

Элемент управления PictureBox применяется для отображения графических изображений. Изображение может быть в формате BMP, JPEG, GIF, PNG, метафайла или пиктограммы.

Свойство Image задает ссылку на изображение:

```
public Image Image { get; set; }
```

Из одноименного класса Image отдельно выделим статический метод FromFile, загружающий изображение по указанному пути.

Свойство SizeMode определяет способ отображения внутри рамки:

```
public PictureBoxSizeMode SizeMode { get; set; }
```

где перечисление PictureBoxSizeMode задает следующие варианты:

```
public enum PictureBoxSizeMode = {
    AutoSize, CenterImage, Normal, StretchImage, Zoom
};
```

Form1.cs

Следующее приложение позволяет просматривать изображения в текущей директории.



```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;

namespace WF_2
{
    public partial class Form1 : Form
```

```
{
    public Form1()
    {
        InitializeComponent();
    }
    private string[] files;        // массив путей к файлам изображений
    private int index = 0;        // индекс текущего изображения
    private void Form1_Load(object sender, EventArgs e)
    {
        // директория с изображениями
        string path =
            @"D:\Мои документы\Visual Studio 2010\Projects\WF_2\WF_2\Pictures";
        // получить массив из имен файлов с расширением ".jpg"
        files = System.IO.Directory.GetFiles(path, "*.jpg");
        // загрузить первое изображение
        this.pictureBox1.Image = Image.FromFile(files[0]);
    }
    private void button1_Click(object sender, EventArgs e)
    {
        if (index > 0) index--; else index = files.Length - 1;
        this.pictureBox1.Image = Image.FromFile(files[index]);
    }
    private void button2_Click(object sender, EventArgs e)
    {
        if (index < files.Length - 1) index++; else index = 0;
        this.pictureBox1.Image = Image.FromFile(files[index]);
    }
}
}
```

При настройке свойств формы объекту PictureBox рекомендуется задать якорь со всех сторон, чтобы при изменении размера формы изображение масштабировалось автоматически.

Класс необходимо расширить двумя полями. Первое хранит пути к изображениям: их можно получить с помощью метода GetFiles() класса Directory. Второе поле – индекс текущего изображения; он будет увеличиваться либо уменьшаться при прокрутке. На этом уровне переменные будут общедоступны остальным управляющим элементам формы, в частности кнопкам.

Фонд оценочных средств для проведения аттестации уровня сформированности компетенций обучающихся по дисциплине оформляется отдельным документом.

7. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

7.1. Книгообеспеченность

Наименование литературы: автор, на звание, вид издания, издательство	Год издания	КНИГООБЕСПЕЧЕННОСТЬ	
		Количество экземпляров изданий в библиотеке ВлГУ в соответствии с ФГОС ВО	Наличие в электронной библиотеке ВлГУ
1	2	3	4
Основная литература*			
1. Введение в программирование на языке Visual C#: Учебное пособие / С.Р. Гуриков. - М.: Форум: НИЦ ИНФРА-М, 2013. - 448 с.: 70x100 1/16. - (Высшее образование: Бакалавриат). (переплет) ISBN 978-5-91134-738-3, 500 экз.	2013	ЭБС	http://znanium.com/catalog.php?bookinfo=404441
2. Объектно-ориентированное программирование с примерами на C#: Учебное пособие / Хорев П.Б. - М.: Форум, НИЦ ИНФРА-М; 2016. - 200 с.: 70x100 1/16. - (Высшее образование: Бакалавриат)(Обложка) ISBN 978-5-00091-144-0.	2016	ЭБС	http://znanium.com/bookread2.php?book=529350
3. Базовые средства программирования на Visual Basic в среде VisualStudio Net. Практикум: Учебное пособие / Шакин В.Н. - М.: Форум, НИЦ ИНФРА-М, 2015. - 288 с.: 70x100 1/16. - (Высшее образование: Бакалавриат) (Переплёт 7БЦ) ISBN 978-5-00091-054-2.	2015	ЭБС	http://znanium.com/bookread2.php?book=502047
Дополнительная литература			
1. Практикум на ЭВМ. Часть 1 [Электронный ресурс]: учебное пособие/ — Электрон. текстовые данные.— М.: Евразийский открытый институт, 2012.— 263 с. Режим доступа:	2012	ЭБС	http://www.iprbookshop.ru/14644
2. Алгоритмизация и программирование : Учебное пособие / С.А. Канцедал. - М.: ИД ФОРУМ: НИЦ ИНФРА-М, 2014. - 352 с.: ил.; 60x90 1/16. - (Профессиональное образование). (переплет) ISBN 978-5-8199-0355-1, 500 экз.	2014	ЭБС	http://znanium.com/bookread2.php?book=429576
3. Туркин О.В. VBA. Практическое программирование [Электронный ресурс]/ Туркин О.В.— Электрон. текстовые данные.— М.: СОЛОН-ПРЕСС, 2010.— 128 с.			http://www.iprbookshop.ru/8701

7.2. Периодические издания

1. Журнал «Информатика и образование»: <http://infojournal.ru/>
2. Журнал «Информационные технологии»: <http://novtex.ru/IT/>
3. Журнал «Информационное общество»: <http://www.infosoc.iis.ru/index.html>

7.3. Интернет-ресурсы

1. Портал: Компьютерные технологии: <http://ru.wikipedia.org/wiki>, 2018.
2. Официальный сайт поддержки компании Microsoft: <https://msdn.microsoft.com>, 2018.
3. Электронные материалы по технологии.NET: <http://professorweb.ru/>, 2018.

8. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

Для реализации данной дисциплины имеются специальные помещения для проведения занятий лекционного типа, занятий практического типа, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации, а также помещения для самостоятельной работы. Лабораторные работы проводятся в компьютерных аудиториях 241-7, 242-7.

Оснащенность компьютерных аудиторий:

- Компьютерный класс на основе ЭВМ ПК IntelCore с доступом в сеть Интернет, маркерная и интерактивная доски, переносной ноутбук, наушники, колонки.

- Мультимедийный комплекс в составе: Ноутбук с выходом в сеть Интернет, мультимедиа проектор, экран белый матовый, доска маркерная.

Перечень используемого лицензионного программного обеспечения:

- MS Office 2013: лицензии на Microsoft Windows/Office: Microsoft Open License 61248656/62857078/63848368/64196124;
- Notepad++: лицензия GNU GPL;
- браузер Mozilla Firefox: лицензия Mozilla Public License;
- лицензия на антивирусное ПО: Kaspersky Endpoint Security Standart 1356-161220-101943-827-71.

Рабочую программу составил ст. пр. Якубович Д.А.

Рецензент
(представитель работодателя) МАОУ «СОШ № 25 г. Владимира»,
заместитель директора Шавлинская Т.Ю.

Программа рассмотрена и одобрена на заседании кафедры МОиИТ

Протокол № 10 от 29.06.2018 года

Заведующий кафедрой к. ф.-м. н., доц. Евсеева Ю.Ю.

Рабочая программа рассмотрена и одобрена на заседании учебно-методической комиссии
направления 44.03.05 «Педагогическое образование»

Протокол № 1 от 28.08.2018 года

Председатель комиссии к. филол. н., доц. Аргамонова М.В.



[Signature]

[Signature]

[Signature]

[Signature]

**ЛИСТ ПЕРЕУТВЕРЖДЕНИЯ
РАБОЧЕЙ ПРОГРАММЫ ДИСЦИПЛИНЫ**

Рабочая программа одобрена на _____ учебный год

Протокол заседания кафедры № _____ от _____ года

Заведующий кафедрой _____

Рабочая программа одобрена на _____ учебный год

Протокол заседания кафедры № _____ от _____ года

Заведующий кафедрой _____

Рабочая программа одобрена на _____ учебный год

Протокол заседания кафедры № _____ от _____ года

Заведующий кафедрой _____

ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ

в рабочую программу дисциплины

«Практикум по решению задач на ЭВМ»

образовательной программы направления подготовки 44.03.05 – Педагогическое образование,

направленность: *Математика. Информатика (бакалавриат)*

Номер изменения	Внесены изменения в части/разделы рабочей программы	Исполнитель ФИО	Основание (номер и дата протокола заседания кафедры)
1			
2			

Зав. кафедрой МОиИТ _____ / _____
Подпись / ФИО