

Министерство образования и науки Российской Федерации  
 Федеральное государственное бюджетное образовательное учреждение  
 высшего образования  
 «Владимирский государственный университет имени  
 Александра Григорьевича и Николая Григорьевича Столетовых»  
 (ВлГУ)



УТВЕРЖДАЮ  
 Проректор по учебно-методической  
 работе

А.А. Панфилов

« 17 » 03 2016 г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ  
Практикум по решению задач на ЭВМ

Направление подготовки 44.03.05 Педагогическое образование

Профиль подготовки «Информатика. Математика»

Уровень высшего образования бакалавриат

Форма обучения очная

Семестр	Трудоём- кость зач. ед, час.	Лекций, час.	Практ. зан., час.	Лаборат. работ, час.	СРС, час.	Форма промежуточного контроля (экз./зачет)
8	2 / 72	-	-	20	52	ЗАЧЕТ
Итого	2 / 72	-	-	20	52	ЗАЧЕТ

Владимир, 2016

# 1. ЦЕЛИ И ЗАДАЧИ ОСВОЕНИЯ ДИСЦИПЛИНЫ

## Цели:

1. Формирование у студентов навыков работы с современными технологиями в программировании для решения прикладных задач.
2. Развитие операционного мышления направленного на выбор оптимальных действий, на умение планировать свою деятельность и предвидеть ее результаты.
3. Формирование опыта работы в коллективе, в частности рефлексии.

## Задачи дисциплины:

- Сформировать навыки работы с программной платформой .NET Framework и реализации ООП парадигмы.
- 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.

# 2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОПОП ВО

Дисциплина «Практикум по решению задач на ЭВМ» относится к вариативной части учебного плана по направлению «Педагогическое образование» (Б1.В.ОД.14).

Для освоения дисциплины студенты используют знания и умения, сформированные в процессе изучения таких дисциплин, как «Современные ИТ», «Программирование», «Функциональное программирование», «Теория алгоритмов», «Теоретическая информатика».

Освоение данной дисциплины способствует подготовке студентов к итоговой государственной аттестации.

# 3. КОМПЕТЕНЦИИ ОБУЧАЮЩЕГОСЯ, ФОРМИРУЕМЫЕ В РЕЗУЛЬТАТЕ ОСВОЕНИЯ ДИСЦИПЛИНЫ (МОДУЛЯ)

В результате освоения дисциплины формируются следующие компетенции:

Шифр компетенции	Расшифровка компетенции
ОК-6	способностью к самоорганизации и самообразованию;
ПК-1	готовностью реализовывать образовательные программы по учебным предметам в соответствии с требованиями общеобразовательных стандартов;
ПК-12	способностью руководить учебно-исследовательской деятельностью обучающихся.

В результате освоения дисциплины обучающийся должен демонстрировать следующие результаты образования:

## Знать:

- теоретическую основу платформы .NET Framework 4.5 (ОК-6 / ПК-1 / ПК-12).

- синтаксис языка программирования C# и реализация принципов ООП средствами .NET Framework (ПК-1);
- методы и приемы реализации алгоритмов на базе компонентной модели проекта (ОК-6 / ПК-1).

**Уметь:**

- моделировать практические задачи средствами .NET Framework (ОК-6 / ПК-1);
- применять новые технологии на основе практических задач (ПК-1 / ПК-12).
- осуществлять согласованную работу в коллективе из нескольких человек в целях достижения поставленной учебной задачи (ПК-12).

**Владеть:**

- приемами исследования математических задач средствами .NET Framework (ПК-1);
- навыками работы со справочными системами по технологии NET Framework и языку программирования C# (ПК-1).

## 4. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)

Общая трудоёмкость дисциплины составляет 2 зачетные единицы, 72 часов.

№ п/п	Раздел (тема) 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 (по неделям семестра), форма промежуточной аттестации (по семестрам)
				Лекции	Практические	Лабораторные	Контрольные работы, коллоквиум	СРС	КП/КР		
1	Статические компоненты.	8	1			2		8		1/50%	
2	Структуры и перечисления	8	2			2		4		1/50%	
3	Строки и файлы	8	3-4			4		12		2/50%	Рейтинг-контроль №1
4	Обработка исключений	8	5			2		6			
5	Наследование и полиморфизм	8	6			2		8			Рейтинг-контроль №2
6	Интерфейсы. Делегаты.	8	7-8			4		8		1/25%	

	События									
7	Обобщения	8	9-10			4		6	1/25%	Рейтинг-контроль №3
	<b>Всего</b>					<b>20</b>		<b>52</b>	<b>6/30.0%</b>	<b>ЗАЧЕТ</b>

## Темы и содержание лабораторных занятий

### Тема 1. Статические компоненты

- Задачи, приводящие к необходимости введение статических компонентов.
- Модификатор `static`.
- Статичные поля класса.
- Статичные методы класса.
- Статичные классы.
- Примеры использования.

### Тема 2. Структуры и перечисления

- Использование структур в качестве параметров методов.
- Использование структур в иерархии типов данных.
- Перечислимый тип как механизм повышения качества логики построения приложения.
- Обработка данных перечислимого типа.
- Решение задач.

### Тема 3. Строки и файлы

- Методы класса `System.String`. Примеры обработки строк.
- Класс `StringBuilder`.
- Регулярные выражения.
- Пространство имен `System.IO`.
- Запись и чтение текстовых файлов.
- Работа с жестким диском.
- Решение задач.

### Тема 4. Обработка исключений

- Перехват исключений.
- Класс `Exception`.
- Обработка многочисленных исключений.
- Решение задач.

### Тема 5. Наследование и полиморфизм

- Основы наследования.
- Конструкторы класса в наследовании.
- Виртуальные методы.
- Абстрактные классы.
- Решение задач.

### Тема 6. Интерфейсы. Делегаты. События

- Интерфейсы и их роль в иерархии типов.
- 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.

#### **Тема 7. Обобщения**

- Назначение обобщений и их применение.
- Обобщённые классы.
- Стек.
- Решение практических задач.

## **5. ОБРАЗОВАТЕЛЬНЫЕ ТЕХНОЛОГИИ**

Изучение курса «ПРЗ на ЭВМ» предполагает сочетание лабораторных занятий и самостоятельной работы студентов.

На лабораторных занятиях, общий объем которых указан в тематическом плане, студенты изучают теоретический минимум, выполняют задания (индивидуально / попарно или в группах из нескольких человек), консультируются по самостоятельной работе с преподавателем.

Самостоятельная работа предполагает более детальное знакомство с теоретическим материалом и предварительную подготовку к новым лабораторным работам.

При изучении учебного материала данной дисциплины следующие технологии обучения: учебные групповые дискуссии: обсуждения задач (методы, приемы решения, выбор оптимального способа решения, количество возможных случаев для рассмотрения и т.п.), мозговой штурм, презентация микроисследований и их обсуждение, технология проблемного обучения.

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

### **6.1 ТЕКУЩИЙ КОНТРОЛЬ УСПЕВАЕМОСТИ**

#### **Примеры заданий для проведения рейтинг-контроля**

##### **Рейтинг-контроль №1**

1. Разработайте статический класс MuMath 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 отслеживать 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.

## 6.2 ПРОМЕЖУТОЧНАЯ АТТЕСТАЦИЯ ПО ИТОГАМ ОСВОЕНИЯ ДИСЦИПЛИНЫ

### Вопросы к зачету

1. Статические компоненты
2. Задачи, приводящие к необходимости введение статических компонентов.
3. Модификатор static. Статичные поля класса.
4. Статичные методы класса.
5. Статичные классы. Примеры использования.
6. Структуры и перечисления.
7. Использование структур в качестве параметров методов.
8. Использование структур в иерархии типов данных.
9. Перечислимый тип как механизм повышения качества логики построения приложения.
10. Обработка данных перечислимого типа.
11. Строки и файлы.
12. Методы класса System.String. Примеры обработки строк.
13. Класс StringBuilder.
14. Регулярные выражения.
15. Пространство имен System.IO.
16. Запись и чтение текстовых файлов.
17. Работа с жестким диском.
18. Обработка исключений и перехват исключений.
19. Класс Exception.
20. Обработка многочисленных исключений.
21. Наследование и полиморфизм
22. Конструкторы класса в наследовании.
23. Виртуальные методы.
24. Абстрактные классы.
25. Интерфейсы и их роль в иерархии типов.
26. Интерфейсные свойства.

27. Делегаты.
28. Групповой вызов и адресация делегируемых методов.
29. Делегаты Action<T> и Func<T>.
30. События как методы обратного вызова.
31. Связь событий и делегатов.
32. Обобщения.
33. Назначение обобщений и их применение.
34. Обобщённые классы.
35. Стек на обобщенном типе.

## **6.3 МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ САМОСТОЯТЕЛЬНОЙ РАБОТЫ СТУДЕНТОВ**

### **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.

### **Примеры заданий 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.

## **7. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)**

### **Основная литература:**

1. Златопольский Д.М. Программирование. Типовые задачи, алгоритмы, методы [Электронный ресурс]/ Златопольский Д.М.— Электрон. текстовые данные.— М.: БИНОМ. Лаборатория знаний, 2015.— 224 с. Режим доступа: <http://www.iprbookshop.ru/12264>
2. Объектно-ориентированное программирование с примерами на С#: 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. Режим доступа: <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. Режим доступа: <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: <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. Режим доступа: <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. Режим доступа: <http://www.iprbookshop.ru/8701>

### **Программное обеспечение и Интернет-ресурсы**

1. Портал: Компьютерные технологии, <http://ru.wikipedia.org/wiki>, 2016.
2. Официальный сайт поддержки компании Microsoft: <https://msdn.microsoft.com>, 2016.
3. <http://professorweb.ru/>, электронные материалы по технологии .NET, 2016.

## **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>

## **8. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)**

- 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.

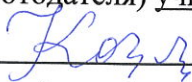
Рабочая программа дисциплины составлена в соответствии с требованиями ФГОС ВО по направлению 44.03.05 «Педагогическое образование», профиль «Информатика. Математика»

---

Рабочую программу составил асс. Якубович Д.А.  
(ФИО, подпись)



Рецензент (представитель работодателя) учитель высшей категории МБОУ СОШ №15  
г.Владимир Козлова С.А.  
(место работы, должность, ФИО, подпись)



Программа рассмотрена и одобрена на заседании кафедры

Протокол № 7а от 10.03.16 года

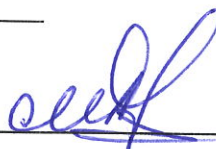
Заведующий кафедрой ИИТО, проф. Медведев Ю.А.  
(ФИО, подпись)



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

Протокол № 3 от 17.03.16 года

Председатель комиссии Артамонова М.В.  
(ФИО, подпись)



**ЛИСТ ПЕРЕУТВЕРЖДЕНИЯ  
РАБОЧЕЙ ПРОГРАММЫ ДИСЦИПЛИНЫ (МОДУЛЯ)**

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 \_\_\_\_\_

---