

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

Педагогический институт

УТВЕРЖДАЮ:
Директор института
М.В. Артамонова
31 08 2021 г.



РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ
СОВРЕМЕННЫЕ ЯЗЫКИ ПРОГРАММИРОВАНИЯ

направление подготовки / специальность
44.03.05 – Педагогическое образование (с двумя профилями подготовки)

направленность (профиль) подготовки
Математика. Информатика

г. Владимир

2021

1. ЦЕЛИ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Целью освоения дисциплины «Современные языки программирования» является формирование у студентов опыта работы с современными языками и платформами программирования.

Задачи дисциплины:

- Формирование знаний в области современных языков и платформ программирования.
- Формирование навыков работы с промышленными языками программирования и языками веб-программирования.
- Изучение программной платформы .NET Framework и языка C#.
- Изучение языка программирования JavaScript.
- Развитие опыта разработки алгоритмов и решения практических задач.
- Изучение и использование различных методов программирования.
- Освоение понятий и принципов ООП на примере C# и JavaScript.
- Изучение возможностей современных языков программирования в образовательной деятельности.

2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОПОП ВО

Дисциплина «Современные языки программирования» относится к обязательной части учебного плана по направлению «Педагогическое образование».

3. ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ

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

Формируемые компетенции (код, содержание компетенции)	Планируемые результаты обучения по дисциплине, в соответствии с индикатором достижения компетенции		Наименование оценочного средства
	Индикатор достижения компетенции (код, содержание индикатора)	Результаты обучения по дисциплине	
ПК-1 Способен успешно взаимодействовать в различных ситуациях педагогического общения	ПК.1.1. Демонстрирует знания особенностей педагогического общения и профессиональной рефлексии	Знать: <ul style="list-style-type: none">• основы и базовые концепции объектно-ориентированного программирования;• особенности и возможности современных ООП в обучении школьников программированию. Уметь: <ul style="list-style-type: none">• проецировать полученные знания для реализации педагогических задач в процессе обучения основам алгоритмизации и программирования и веб-технологий;• применять изученные ЯП в обучении школьников программированию. Владеть: <ul style="list-style-type: none">• навыками рефлексии;• приемами организации групповой работы.	Тесты Ситуационные задачи
	ПК.1.2. Применяет успешные технологии взаимодействия в профессиональном коллективе и решения педагогических конфликтов		
	ПК.1.3. Целесообразно выбирает и использует методы педагогического общения с обучающимися и их родителями		

<p>ПК-3 Способен реализовывать образовательные программы различных уровней в соответствии с современными методиками и технологиями, в том числе информационными, для обеспечения качества учебно-воспитательного процесса</p>	<p>ПК.3.1. Разрабатывает и реализует основные и дополнительные образовательные программы по своей дисциплине с учетом современных методов и технологий</p>	<p>Знать:</p> <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; теоретическую основу языка JavaScript и особенности взаимодействия с HTML и CSS. возможности абстрагирования в ООП-модели. <p>Уметь:</p> <ul style="list-style-type: none"> анализировать возможности языков промышленного программирования и веб-разработки; моделировать научные и практические задачи средствами .NET Framework и JavaScript. <p>Владеть:</p> <ul style="list-style-type: none"> приемами программирования средствами .NET Framework; приемами веб-программирования. приемами исследования математических задач средствами .NET Framework и JavaScript. 	<p>Тесты Ситуационные задачи Выполнение учебных проектов</p>
	<p>ПК.3.2. Применяет современные информационные технологии в урочной и внеурочной деятельности сопровождения образовательного процесса</p>		
	<p>ПК.3.3. Применяет современные методики в организации воспитательного процесса</p>		

4. ОБЪЕМ И СТРУКТУРА ДИСЦИПЛИНЫ

Трудоемкость дисциплины составляет 3 зачетных единиц, 108 часов.

Тематический план Форма обучения – очная

№ п/п	Наименование тем и/или разделов/тем дисциплины	Семестр	Неделя семестра	Контактная работа обучающихся с педагогическим работником				Самостоятельная работа	Формы текущего контроля успеваемости, форма промежуточной аттестации (по семестрам)
				Лекции	Практические занятия	Лабораторные работы	В форме практической подготовки		
1	Современные языки и платформы программирования.	5	1-2	2		4		6	Рейтинг-контроль №1
2	Платформа .NET и язык программирования C#.	5	3-12	8		16		24	Рейтинг-контроль №2
3	Веб-программирование. Язык JavaScript.	5	13-18	8		16		24	Рейтинг-контроль №3
Всего за 5 семестр				18		36		54	Зачет с оценкой
Наличие в дисциплине КП/КР					+				КР
Итого по дисциплине				18		36		54	Зачет с оценкой, КР

Содержание лекционных занятий по дисциплине

Раздел 1. Современные языки и платформы программирования

- 1.1 Основные концепции программирования.
- 1.2 Объектно-ориентированное программирование. Принципы ООП.
- 1.3 Современные языки и платформы программирования.
- 1.4 Интегрированная среда разработки: возможности, структура.
- 1.5 Программирование настольных приложений. Промышленное программирование.
- 1.6 Веб-программирование. Фронтенд и бэкенд.

Раздел 2. Платформа .NET и язык программирования C#

- 2.1 Платформа .NET Framework и ее возможности.
- 2.2 Язык C#. Среда разработки Visual Studio.
- 2.3 Типы данных. Арифметические и логические операторы.
- 2.4 Условные и циклические конструкции C#.
- 2.5 Массивы.
- 2.6 Функции.
- 2.7 Примеры решения задач.
- 2.8 Классы. Компоненты класса: поля, методы, конструктор, свойства.
- 2.9 Обработка строк. Классы System.String и StringBuilder.
- 2.10 Обработка текстовых файлов Пространство имен System.IO.
- 2.11 Исключения. Механизм обработки исключений.
- 2.12 Основы наследования. Виртуальные методы. Абстрактные классы.
- 2.13 Полиморфизм. Перегрузка методов.
- 2.14 Инкапсуляция данных. Свойства. Пространства имен.
- 2.15 Решение задач с классами.
- 2.16 События.
- 2.17 Введение в визуальное программирование. Технология Windows Forms.

Раздел 3. Веб-программирование. Язык JavaScript

- 3.1 Технологии HTML5 и CSS3. Обзор (повтор) тегов и свойств.
- 3.2 Язык JavaScript: история. ECMAScript как стандарт фундаментального синтаксиса JavaScript.
- 3.3 Способы подключения скриптов в веб-странице.
- 3.4 Типы данных. Арифметические операции, операции отношения и логические операторы.
- 3.5 Операторы условного выбора и циклов.
- 3.6 Массивы.
- 3.7 Функции.
- 3.8 Создание объектов и основные принципы ООП в JS.
- 3.9 Основы DOM.
- 3.10 Формы.
- 3.11 Решение задач.

Содержание лабораторных занятий по дисциплине

Раздел 1. Платформа .NET и язык программирования C#

- 1.1 Язык C#. Среда разработки Visual Studio. Создание консольного приложения. Отладка ошибок.

- 1.2 Типы данных. Арифметические и логические операторы.
- 1.3 Условные и циклические конструкции C#.
- 1.4 Массивы: одномерные, двумерные, ступенчатые.
- 1.5 Классы. Компоненты класса: поля, методы, конструктор, свойства.
- 1.6 Обработка строк. Классы System.String и StringBuilder.
- 1.7 Обработка текстовых файлов Пространство имен System.IO.
- 1.8 Исключения. Механизм обработки исключений.
- 1.9 Основы наследования. Виртуальные методы. Абстрактные классы.
- 1.10 Полиморфизм. Перегрузка методов.
- 1.11 Интерфейсы.
- 1.12 Инкапсуляция данных. Свойства. Пространства имен.
- 1.13 События.
- 1.14 Введение в визуальное программирование. Технология Windows Forms. Формы, кнопки, переключатели, списки., текстовые поля, компоненты Image.

Раздел 2. Веб-программирование. Язык JavaScript

- 2.1 Язык разметки HTML5. Каскадные таблицы стилей CSS3. Обзор (повтор) тегов и свойств. Работа с заготовленными шаблонами.
- 2.2 Типы данных. Арифметические операции, операции отношения и логические операторы. Объект Math.
- 2.3 Механизмы ввода и вывода. Модальные окна. Работа с инструментами разработчика в браузере (Google Chrome, Mozilla FireFox, Opera).
- 2.4 Операторы условного выбора и циклов.
- 2.5 Массивы. Методы массивов. Объект Array.
- 2.6 Функции. Функции как объекты. Вариации синтаксиса.
- 2.7 Элементы форм. Обращение к элементам формы и их функциональная обработка.
- 2.8 Создание объектов.
- 2.9 Основы DOM. Обращение к элементам HTML. DHTML.
- 2.10 Введение в jQuery. Селекторы. Методы работы с DOM и CSS.

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

5.1. Текущий контроль успеваемости

Рейтинг-контроль 1

1. Напишите следующие функции:
 - a. позволяет вычислять модуль числа;
 - b. находить ближайшее целое;
 - c. вычислять факториал (если число целое, в противном случае возврат -1).
- Осуществить тестирование функций.
2. Задан массив чисел. Осуществить сортировку по убыванию (используя встроенные методы) и отобразить среди них простые числа. Вывести отдельно первоначальный и итоговый массивы.

Рейтинг-контроль 2

1. Создайте класс Parabola, описывающий квадратный трехчлен. Класс содержит три поля (свойства): коэффициенты. Конструктор заполняет поля, либо задает $a = 1, b = 0, c = 0$ в случае не параметризованного конструктора.
2. Реализовать методы сложения, разности, умножения на число, нахождения вершины параболы и корней уравнения. Тип данных методов соответствует классу Parabola.

Рейтинг-контроль 3

С помощью метода prompt запрашивается число в пределах от 1 до 30. Далее необходимо построить таблицу умножения (от 1 до этого числа); квадраты чисел должны быть выделены (см. рис. ниже). Воспользуйтесь кодом:

```
<html>
  <head>
    <title></title>
    <meta charset="utf-8">
    <style type="text/css">
      body { font: 14px Tahoma; }

      /* класс оформляет диагональную ячейку */
      .quad {
        background: #B0F9B0;
        font-weight: bold;
      }

      table, tr ,td { border: 2px solid #777; }

      td {
        width: 40px;
        text-align: center;
      }
    </style>
  </head>

  <body>
    <h2>Таблица умножения</h2>
    <script>
      var number = prompt("Введите число", "1");

      // В переменную text построчно "накапливается" таблица умножения
      // Не забывайте закрывать парные теги!
      var text = "<table>";

      // Здесь реализовать код для умножения и вывода ячеек

      text += "</table>";

      document.write(text); // Встраиваем результат в DOM
    </script>
  </body>
</html>
```

Ожидаемый результат изображен на следующем изображении:

task_1.4.html

file:///D:/Мои%20документы/Университет/Предметы/Программирование/JavaScript

Сервисы Добавьте на эту панель закладки, к которым хотите иметь быстрый доступ. [Импортировать закладки...](#)

Таблица умножения

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
2	4	6	8	10	12	14	16	18	20	22	24	26	28	30
3	6	9	12	15	18	21	24	27	30	33	36	39	42	45
4	8	12	16	20	24	28	32	36	40	44	48	52	56	60
5	10	15	20	25	30	35	40	45	50	55	60	65	70	75
6	12	18	24	30	36	42	48	54	60	66	72	78	84	90
7	14	21	28	35	42	49	56	63	70	77	84	91	98	105
8	16	24	32	40	48	56	64	72	80	88	96	104	112	120
9	18	27	36	45	54	63	72	81	90	99	108	117	126	135
10	20	30	40	50	60	70	80	90	100	110	120	130	140	150
11	22	33	44	55	66	77	88	99	110	121	132	143	154	165
12	24	36	48	60	72	84	96	108	120	132	144	156	168	180
13	26	39	52	65	78	91	104	117	130	143	156	169	182	195
14	28	42	56	70	84	98	112	126	140	154	168	182	196	210
15	30	45	60	75	90	105	120	135	150	165	180	195	210	225

5.2. Промежуточная аттестация (экзамен)

1. Принципы ООП. Описание каждого принципа.
2. Платформа .NET Framework и язык C#. Возможности.
3. Типы данных C#. Основные операторы. Классы Math и Random.
4. Инструкции условного выбора в C#.
5. Циклические инструкции в C#.
6. Массивы в C#.
7. Классы и их компоненты в C#.
8. Обработка строк и работа с текстовыми файлами в .NET.
9. Обработка исключений и перехват исключений в C#.
10. Наследование и полиморфизм в C#.
11. Виртуальные методы и абстрактные классы в C#.
12. Интерфейсы и их роль в иерархии типов в C#.
13. Понятие перегрузки операторов в C#.
14. Приложения с визуальным интерфейсом.
15. Компоненты формы. Основные свойства и события.
16. Разработка GUI с помощью Visual Studio, SharpDevelop.
17. Изучение C# как первого профессионального языка программирования в школьном курсе информатики.
18. ECMAScript и JavaScript.
19. Простейшие инструменты ввода/вывода/отладки в JavaScript.
20. Операторы условного выбора if и switch в JavaScript.
21. Операторы цикла в JavaScript.
22. Описание и обработка массивов в JavaScript.
23. Технология DOM.

24. Основные функции обработки разметки средствами DOM.
25. jQuery. Методы обработки DOM.
26. jQuery. Методы обработки CSS.
27. Знакомство с JavaScript как языком веб-разработки в школьном курсе информатики.

5.3. Самостоятельная работа обучающегося

Вопросы для самостоятельной работы

1. Статичные классы в пространстве имен System.
2. Строковые операции. Регулярны выражения. Преимущества и недостатки.
3. Абстрактные классы. Реализация математических моделей.
4. Наследование и полиморфизм. Целесообразность наследования при реализации собственных классов и сборок.
5. Интерфейсы и абстрактные классы.
6. Обобщения как средство расширения возможностей работы с компонентами.
7. Стандарт ECMAScript.
8. Функции. Function Declaration и Function Expression.
9. Обработка данных браузера средствами DOM.
10. Функции для обработки разметки в DOM.
11. HTML-формы. Основные элементы форм и их обработка.
12. Понятие прототипа и класса в JavaScript.

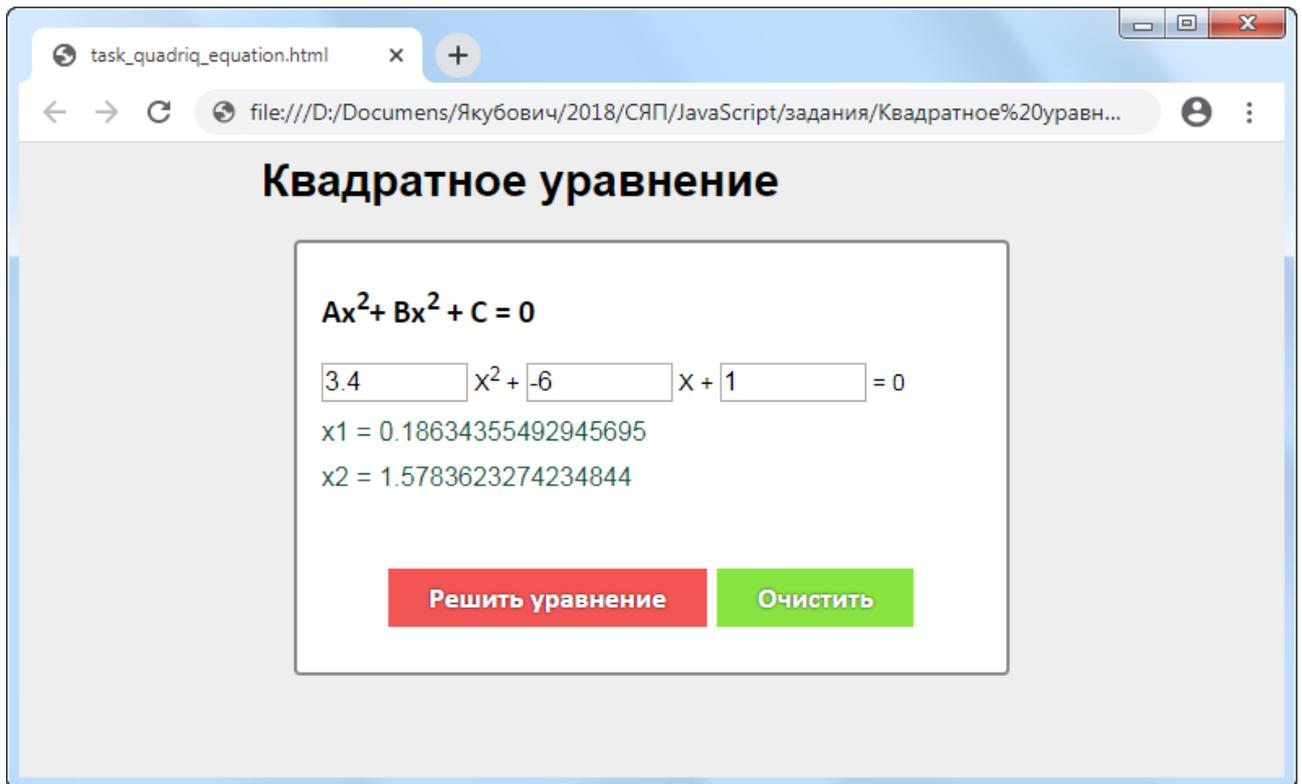
Примеры заданий для проектной деятельности

Для языка C#

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.

Для языка JavaScript

Разработать html-страницу с формой, позволяющей решать квадратное уравнение по заданным коэффициентам.



Обязательные требования:

- код с разметкой подключает стиль оформления и JS-код как внешние файлы;
- оформление элементов должно соответствовать изображению;
- JS-код должен использовать библиотеку jQuery (например, при обращении к элементам DOM);
- предусмотреть дружелюбный интерфейс:
 - a. поля автоматически заполняются значениями 1, 0, 0, если они пусты;
 - b. в случае ввода в качестве коэффициента нечислового выражения значение исправляется на значение по умолчанию.

Занятие 1.8 Классы Console, Convert, Math и Random

1.8.1 Класс Console и Convert

Для организации ввода и вывода информации в консоли используется класс Console, точнее его методы. В качестве аргументов им можно передавать строковые и числовые данные напрямую, либо через переменные:

```
Console.Write(n);  
Console.Write("Это текст");
```

Можно склеивать несколько значений в одну строку

```
Console.Write("На улице " + 21 +"й век!");
```

Значения можно вставлять в определенное место строки. Для этого используется форматный вывод. Чтобы подставить некоторое значение в указанное место строки, укажите это место фигурными скобками и индексом элемента:

```
Console.Write("Студент: {0} Номер: {1}", "Петров", 4);
```

Так вместо {0} в строку будет подставлено «Петров», а вместо {1} – число 4. Этот прием можно использовать для вывода вещественных чисел:

```
double num = 2.3456;  
Console.Write("Число: {0:#.###}", num); // 2.346
```

Знак # указывает на один разряд. Соответственно после запятой будет указано три знака вместо четырех, плюс округление младшего разряда при необходимости.

Для ввода данных с клавиатуры используется метод ReadLine.

C# воспринимает вводимую информацию как поток символов. Соответственно, для ввода числа необходимо конвертировать информацию в число.

Для преобразования используются методы класса Convert. Например,

```
string str = Console.ReadLine(); // конвертировать не нужно  
int num = Convert.ToInt32(Console.ReadLine()); // конвертируем введенное  
double x = Convert.ToDouble(Console.ReadLine()); // конвертируем введенное
```

1.8.2 Класс Math

Для математических вычислений часто требуется использование различных функций. Их предоставляет класс Math.

Рассмотрим примеры использования:

```
double a = Math.Sqrt(100); // корень из 100
double Pi = Math.PI; // постоянное число PI
double b = Math.Cos(2 * Pi); // cos(2PI)
int max = Math.Max(33, 67); // максимальное из чисел 33 и 67
```

1.8.3 Класс Random

Часто требуется генерировать случайные числа. Например, нам лень заполнять массив с клавиатуры, либо необходимо моделировать случайные процессы. Для генерации случайных чисел используется класс Random. Рассмотрим методы Next и NextDouble.

Метод Next имеет три различные вариации.

Первая генерирует целое число на всем допустимом диапазоне целых чисел:

```
Random X = new Random(); // создаем экземпляр класса Random
Console.Write(X.Next()); // генерируем случайное число из диапазона int
```

Вторая генерирует целые числа от нуля до указанного:

```
Console.Write(X.Next(10));
```

Третья генерирует числа в указанном диапазоне:

```
Console.Write(X.Next(3,10));
```

Метод NextDouble генерирует вещественное число от 0 до 1:

```
Random X = new Random();
Console.WriteLine(X.NextDouble());
```

Занятие 1.9 Подпрограмма

1.9.1 Подпрограмма

В .NET понятие **процедур** и **функций** обобщено в понятие **метода**. Метод не является подпрограммой сам по себе, он является частью класса.

Например, требуется написать приложение, которое вычисляет число комбинаций m объектов из n различных объектов. Это комбинаторная задача сочетаний без повторов, поэтому число комбинаций высчитывается по формуле:

$$C_n^m = \frac{n!}{m!(n-m)!}$$

Очевидно, что в программе будет полезно создать функцию, возвращающую факториал числа.

Примерные темы курсовой работы

1. Сравнительный анализ современных объектно-ориентированных языков программирования.

2. Классы и особенности наследования в языке C#.
3. LINQ как язык запросов.
4. Интерфейсы в языке C# как инструмент полиморфизма.
5. Объектно-ориентированное программирование на языке Java.
6. Классы и структуры в языке C++.
7. Основы программирования приложений с использованием кроссплатформенного фреймворка Qt.
8. Создание приложений с графическим интерфейсом средствами языка C#.
9. Языки программирования JavaScript и PHP и их использование в веб-разработке.
10. Работа с DOM средствами языка JavaScript и библиотеки jQuery.
11. Использование языка Python в обучении школьников программированию.
12. Основы программирования на языке Ruby.

Фонд оценочных материалов (ФОМ) для проведения аттестации уровня сформированности компетенций обучающихся по дисциплине оформляется отдельным документом.

6. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

6.1. Книгообеспеченность

Наименование литературы: автор, название, вид издания, издательство	Год издания	КНИГООБЕСПЕЧЕННОСТЬ
		Наличие в электронном каталоге ЭБС
Основная литература*		
1. Гуриков, С. Р. 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	http://znanium.com/catalog.php?bookinfo=404441
2. Хорев, П. Б. 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	http://znanium.com/bookread2.php?book=529350
3. Шашкин, В. Н. 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	http://znanium.com/bookread2.php?book=502047
4. Динамические сайты на HTML, CSS, Javascript и Bootstrap. Практика, практика и только практика / А. В. Кириченко, Е. В. Дубовик. — СПб: Наука и Техника, 2018. — 272 с. ISBN 978-5-94387-763-6.	2018	http://www.iprbookshop.ru/77578.html
Дополнительная литература		
1. Практикум на ЭВМ. Часть 1 [Электронный ресурс]: учебное пособие/ — Электрон. текстовые данные. — М.: Евразийский открытый институт, 2012. — 263 с. ISBN 978-5-374-00600-1.	2012	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.	2014	http://znanium.com/bookread2.php?book=429576

3. Никольский, А. П. JavaScript на примерах. Практика, практика и только практика / А. П. Никольский. — СПб.: Наука и Техника, 2018. — 272 с. ISBN 978-5-94387-762-9.	2018	http://www.iprbookshop.ru/78103.html
4. Прасти, Н. Введение в ECMAScript 6 / пер. с англ. Р.Н. Рагимова ; Нараян Прасти. – М.: ДМК Пресс, 2016. - 176 с. ISBN 978-5-97060-392-5.	2016	https://znanium.com/catalog/product/102808 2

6.2. 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>

6.3. 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.

7. 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;
- Среда Visual Studio 2012;
- браузер Mozilla Firefox: лицензия Mozilla Public License;
- лицензия на антивирусное ПО: Kaspersky Endpoint Security Standart 1356-161220-101943-827-71.

Рабочую программу составил _____

 / ст. преп. кафедр ФМОиИТ

(ФИО, должность, подпись)

Рецензент

(представитель работодателя) МАОУ «СОШ №25 г. Владимира»

заместитель директора Шавлинская Т.Ю.


(подпись)

Программа рассмотрена и одобрена на заседании кафедры ФМОиИТ

Протокол № 11 от 30.08.2021 года

Заведующий кафедрой Ю.Ю. Евсеева



(подпись)

Рабочая программа рассмотрена и одобрена

на заседании учебно-методической комиссии направления 44.03.05 – Педагогическое образование (с двумя профилями подготовки)

Протокол № 1 от 31.08.2021 года

Председатель комиссии

 М. В. Артамонова
(ФИО, должность, подпись)

ЛИСТ ПЕРЕУТВЕРЖДЕНИЯ РАБОЧЕЙ ПРОГРАММЫ ДИСЦИПЛИНЫ

Рабочая программа одобрена на 20____ / 20____ учебный года

Протокол заседания кафедры № _____ от _____ года

Заведующий кафедрой _____

Рабочая программа одобрена на 20____ / 20____ учебный года

Протокол заседания кафедры № _____ от _____ года

Заведующий кафедрой _____

Рабочая программа одобрена на 20____ / 20____ учебный года

Протокол заседания кафедры № _____ от _____ года

Заведующий кафедрой _____