

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 УМР

А.А.Панфилов

« 17 » марта 2016 г.

**РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ
СОВРЕМЕННЫЕ ЯЗЫКИ ПРОГРАММИРОВАНИЯ**

Направление подготовки 44.03.05 – Педагогическое образование

Профили подготовки Математика. Информатика

Уровень высшего образования бакалавриат

Форма обучения очная

Семестр	Трудоемкость зач. ед./ час.	Лекции, час.	Практич. занятия, час.	Лаборат. работы, час.	СРС, час.	Форма промежуточного контроля (экз./зачет)
4	3 / 108	18		36	54	зачет (с оценкой)
Итого	3 / 108	18		36	54	зачет (с оценкой)

Владимир 2016

1. ЦЕЛИ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Целями освоения дисциплины «Современные языки программирования» являются:

- изучение основных концепций структурного и объектно-ориентированного программирования;
- развитие логического и алгоритмического мышления у студентов;
- формирование и развитие умений планирования действий и структурирования информации, необходимых для решения поставленной задачи, при помощи фиксированного набора средств;
- формирование способности отображать реальные объекты и явления в компьютерные информационные структуры;
- использование языка программирования как средства автоматизации вычислений, необходимых в учебной деятельности;
- расширение знаний о языках программирования;
- формирование умений разработки визуальных приложений.

В качестве объектно-ориентированного языка программирования рассматривается язык C#, который в настоящий момент является одним из основных языков разработки программного обеспечения. В школьном курсе информатики данный язык программирования не всегда рассматривается, однако на уровне консольных приложений алгоритмические конструкции языка C# идентичны конструкциям языка C++, который включен в перечень языков, используемых в Едином государственном экзамене по Информатике и ИКТ, а также на Всероссийской олимпиаде школьников по информатике. Таким образом, знание этих языков является необходимым для современного учителя информатики.

2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОПОП ВО

Дисциплина «Современные языки программирования» является курсом по выбору вариативной части дисциплин.

В ходе изучения дисциплины студенты применяют знания и умения, сформированные при изучении дисциплины «Программирование», учатся переводить решения задач с одного языка программирования на другой, тем самым прослеживая сходства и различия между языками программирования и отслеживая тенденции в развитии языков программирования.

Для успешного освоения содержания дисциплины студент должен знать:

- один из языков программирования;
- основные управляющие алгоритмические конструкции;
- основы объектно-ориентированного программирования.

Знания, умения и практические навыки, приобретенные студентами в результате изучения дисциплины «Современные языки программирования», будут использоваться при освоении дисциплин: «Компьютерное моделирование», «Частные методики», «Методика обучения информатике», а также при прохождении педагогической практики.

3. КОМПЕТЕНЦИИ ОБУЧАЮЩЕГОСЯ, ФОРМИРУЕМЫЕ В РЕЗУЛЬТАТЕ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Дисциплина «Современные языки программирования» нацелена на формирование у выпускника следующих компетенций:

- способность к самоорганизации и к самообразованию (ОК-6);
- готовность реализовывать образовательные программы по учебным предметам в соответствии с требованиями профессиональных стандартов (ПК-1);
- готовность использовать систематизированные теоретические и практические знания для постановки и решения исследовательских задач в области образования (ПК-11).

В результате освоения дисциплины обучающийся должен демонстрировать следующие результаты образования:

Знать:

- основные концепции объектно-ориентированного программирования (ПК-1);
- структуру программы и основные операторы языков программирования C# (C++) (ПК-1, ПК-11);
- основные типы данных и управляющие конструкции языка C# (C++) (ПК-1);
- основные компоненты, используемые при разработке визуальных приложений (ОК-6, ПК-11).

Уметь:

- создавать собственные объекты и классы объектов в C# (ПК-1, ПК-11);
- разрабатывать внешний интерфейс приложения для решения задачи (ПК-11).

Владеть:

- основными методами проектирования и отладки алгоритмов (ОК-6, ПК-1);
- навыками проектирования программ на языке программирования высокого уровня (ОК-6, ПК-1).

4. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

Общая трудоемкость дисциплины составляет 3 зачетных единиц, 108 часов.

№	Раздел (тема) дисциплины	Семестр	Неделя семестра	Виды учебной работы, включая самостоятельную работу студентов и трудоемкость (в часах)						Объем УР с применением интерактивных методов	Формы ТКУ, форма ПА
				Лекции	Практические занятия	Лабораторные работы	Контрольные работы	СРС	КП / КР		
Семестр		4									
1.	Основы C++. Ввод-вывод информации. Описание переменных		1-2	2		4		6		2 (33%)	
2.	Основные управляющие конструкции языка C# (ветвление, выбор, циклы)		3-4	2		4		6		2 (33%)	
3.	Массивы. Линейные и двумерные массивы		5-6	2		4		6		2 (33%)	Рейтинг-контроль №1
4.	Строки		7-8	2		4		6		2 (33%)	
5.	Функции пользователя		9-10	2		4		6		2 (33%)	
6.	Понятия класса и объекта. Создание классов		11-12	2		4		6		2 (33%)	Рейтинг-контроль №2
7.	Визуальное программирование. Компоненты Button, Label, TextBox, CheckBox, RadioButton		13-14	2		4		6		2 (33%)	
8.	Создание меню приложения. Компоненты MenuStrip, ContextMenuStrip		15-16	2		4		6		2 (33%)	
9.	Графические возможности среды. Компоненты Chart, PictureBox. Анимация в языке программирования C#		17-18	2		4		6		2 (33%)	Рейтинг-контроль №3
Итого в 4 семестре				18	0	36		54		38	зачет (с оценкой)

5. ОБРАЗОВАТЕЛЬНЫЕ ТЕХНОЛОГИИ

Процесс изложения учебного материала на лекции сопровождается демонстрацией компьютерной презентации по рассматриваемой теме, а также выполнением интерактивных заданий при наличии требуемого технического обеспечения (интерактивной доски). Для изучения новых разделов проводятся проблемные лекции.

На практических занятиях проводятся групповые консультации, мозговые штурмы, используются разыгрывание ролей (выступление студента в роли школьного учителя). Решение задач выполняется в двух режимах: с использованием компьютера и последующей отладкой программы и проведением эксперимента по работоспособности программ при различных исходных данных, и без использования компьютера с дальнейшей имитацией работы компьютера при наличии созданной программы.

На лабораторных занятиях могут проводиться тренировки на машинных моделях с использованием компьютерных моделей различной степени сложности.

В рамках изучения дисциплины осуществляется поддержка студентов с использованием электронных образовательных технологий (размещение учебных материалов на сайте кафедры), организуются консультации с использованием дистанционных образовательных технологий

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

Текущий контроль успеваемости проводится в форме рейтинг-контроля в соответствии с планом работы три раза в семестр. В рейтинг-контроль включаются следующие аспекты работы студента:

- выполнение и защита лабораторных работ;
- участие в устных опросах и коллоквиумах;
- выполнение письменных работ;
- выполнение дополнительных заданий в рамках аудиторной или самостоятельной работы.

По результатам изучения дисциплины в 3 и 4 семестрах выставляется зачет с оценкой, в 5 семестре – экзамен.

По завершении изучения объектно-ориентированного программирования студенты создают и защищают проекты – самостоятельно разработанные приложения в соответствии с поставленными задачами.

Примерные задания рейтинг-контролей

Рейтинг-контроль №1

1. Записать синтаксис команды ветвления в языке C#
2. Определить значение переменных a, b, c в результате выполнения фрагмента алгоритма

```
int a=8, b=9;
double c;
if (a>b) && (a%2==0)
{
    a-=4;
    b=b+a/2;
    c=a+Math.Sqrt(b-2);
}
else
{
    b+=5;
```

```
a=b-a/2;  
c=a+Math.Pow(b, 2); }
```

3. Заменить команду ветвления тернарным оператором

```
if (x!=y)  
    z=Math.Abs(x-y);  
else  
    z=0;
```

4. Приведите пример цикла с параметром, который не выполнится ни одного раза.

5. Приведите пример бесконечного цикла с предусловием.

6. Дан фрагмент программы

```
int a, s=0;  
for (int i=5; i<=15; i++)  
{  
    a=int.Parse(Console.ReadLine());  
    s+=a;  
}  
Console.WriteLine(s);
```

Перепишите алгоритм, заменив цикл for

а) циклом с предусловием

б) циклом с постусловием

так, чтобы результат работы алгоритма не изменился.

7. Определите значение переменных s и t в конце работы алгоритма

```
int s, t, n;  
s=5; t=1; n=2;  
while (n<=4)  
{ s=s+(n-1); t=t+s/2; n++}
```

Рейтинг-контроль №2

1. Приведите пример цикла с параметром, который не выполнится ни одного раза.

2. Приведите пример бесконечного цикла с предусловием.

3. Дан фрагмент программы

```
int a, s=0;  
for (int i=5; i<=15; i++)  
{  
    a=int.Parse(Console.ReadLine());  
    s+=a;  
}  
Console.WriteLine(s);
```

Перепишите алгоритм, заменив цикл for

а) циклом с предусловием

б) циклом с постусловием

так, чтобы результат работы алгоритма не изменился.

4. Определите значение переменных s и t в конце работы алгоритма

```
int s, t, n;  
s=5; t=1; n=2;  
while (n<=4)  
{ s=s+(n-1); t=t+s/2; n++}
```

5. Дан алгоритм, который, получив на вход число x, печатает числа a и b. Укажите наименьшее положительное пятизначное число x, при вводе которого алгоритм печатает сначала 6, а потом 3.

```
int x, y, a, b;  
a = 0;
```

```

b = 10;
x=int.Parse(Console.ReadLine());
while (x > 0)
{
    y = x % 10;    x = x / 10;
    if (y > a)    a = y;
    if (y < b)    b = y;
}
Console.WriteLine(a+" "+b);

```

Рейтинг-контроль №3

1. На форме расположены три текстовых поля и кнопка. В первое и третье поля вводятся числа, во второе поле – один из знаков арифметических действий (+, -, *, /, ^). По нажатию кнопки происходит вычисление значения полученного арифметического выражения.
2. Доработать задачу 1 так, чтобы вычисление арифметического выражения происходило при изменении любого текстового поля. Значения по умолчанию – в числовых полях 0, в поле арифметической операции – «+».
3. На форме расположены два текстовых поля и две кнопки. В поля вводятся числа a и b . Первая кнопка осуществляет вывод на форму значений функции $y = x^2$, вторая – значений функции $y = \sqrt{x}$.
4. На форме расположены два текстовых поля и надпись. В поля вводятся числа a и b (в момент начала работы программы в полях находятся соответственно значения 1 и 0). Программа должна находить решения уравнения $a|x| = b$ и выводить найденное решение в надпись.
5. Составить программу, которая рисует ломаную линию по заданным точкам. Координаты вершин ломаной задаются щелчком кнопки мыши по области рисования.
6. Написать программу для построения графика функции $y = 2x^3 + 3x - 4$ на промежутке $[-4,8]$
7. В массиве хранятся N чисел (вводятся с клавиатуры или генерируются случайным образом). На основе имеющихся в массиве данных построить столбчатую или круговую диаграмму. Выбор типа диаграммы осуществляется с помощью переключателей RadioButton.

Примерные задания для самостоятельной работы студентов

1. Создать главное меню с командами Треугольник и Программа. Меню Треугольник содержит пункты: Ввод, Рисование, Вычисление.
 Пункт Ввод обеспечивает отображение на экране 6 текстовых полей для ввода координат вершин треугольника.
 Пункт Рисование обеспечивает отображение картинки, содержащей чертеж построенного по введенным координатам треугольника.
 Пункт Вычисление обеспечивает появление на форме двух флажков. При отмеченном первом флажке на форму выводится значение периметра треугольника, при отмеченном втором флажке – площадь треугольника.
 Пункт меню Программа содержит две команды: Выход (которая закрывает форму) и О разработчике (которая выводит информацию об авторе проекта).
2. Создать главное меню с командами Прямоугольник и Программа. Меню Прямоугольник содержит пункты: Размер, Рисование, Оформление.
 Пункт Размер обеспечивает отображение на экране 2 текстовых полей для ввода ширины и высоты прямоугольника.

Пункт Рисование обеспечивает отображение картинке, содержащей прямоугольник с указанными размерами, расположенный в центре графического поля.

Пункт Оформление обеспечивает появление на форме двух флажков, содержащей возможность изменять стиль прямоугольника: закрашенный прямоугольник и (или) контур прямоугольника.

Пункт меню Программа содержит две команды: Выход (которая закрывает форму) и О разработчике (которая выводит информацию об авторе проекта).

3. Создать главное меню с командами График и Программа. Меню График содержит пункты: Диапазон, Функция, Таблица.

Пункт Диапазон обеспечивает отображение на экране 2 текстовых полей для ввода диапазона значений аргумента функции и построение графика функции для указанного диапазона значений.

Пункт Функция обеспечивает отображение двух переключателей, позволяющих выбрать одну из двух доступных для построения графика функций: $y = 2x^3 - 4x^2 + x - 2$ или $y = -2x^2 + 5x - 2$.

Пункт Таблица обеспечивает вывод таблицы значений аргумента и функции во введенном диапазоне.

Пункт меню Программа содержит две команды: Выход (которая закрывает форму) и О разработчике (которая выводит информацию об авторе проекта).

Вопросы к зачету с оценкой по дисциплине

1. Основы C++. Структура программы
2. Ввод-вывод информации.
3. Описание переменных
4. Основные управляющие конструкции языка C# (ветвление)
5. Основные управляющие конструкции языка C# (выбор)
6. Основные управляющие конструкции языка C# (циклы с условием)
7. Основные управляющие конструкции языка C# (цикл с параметром)
8. Линейные массивы в C#
9. Двумерные массивы в C#
10. Строковые величины и функции для работы со строками
11. Функции пользователя
12. Понятия класса и объекта. Создание классов
13. Визуальное программирование. Компоненты Button, Label, TextBox.
14. Визуальное программирование. Компоненты CheckBox, RadioButton
15. Создание меню приложения. Компоненты MenuStrip, ContextMenuStrip
16. Графические возможности среды. Компонент Chart.
17. Графические возможности среды. Компонент PictureBox.
18. Анимация в языке программирования C#

7. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

а) основная литература

1. Подбельский, В.В. Язык C#. Базовый курс: учеб. пособие / В.В. Подбельский. – М.: Финансы и статистика, 2015. – 408 с.

- <http://www.studentlibrary.ru/book/ISBN9785279035342.html>
2. Подбельский, В.В. Язык Си#. Решение задач [Электронный ресурс]: учеб. пособие / В.В. Подбельский. – М.: Финансы и статистика, 2014. – 296 с.
<http://www.studentlibrary.ru/book/ISBN9785279035533.html>
 3. Гуриков, С.Р. Введение в программирование на языке Visual C#: Учебное пособие / С.Р. Гуриков. – М.: Форум: НИЦ ИНФРА-М, 2013. – 448 с.
<http://znanium.com/catalog.php?bookinfo=404441>
 4. Агапов, В.П. Основы программирования на языке C# [Электронный ресурс]: учебное пособие/ В.П. Агапов – Электрон. текстовые данные. – М.: Московский государственный строительный университет, ЭБС АСВ, 2012. – 128 с.
<http://www.iprbookshop.ru/16366>

б) дополнительная литература:

1. Фарафонов А.С. Программирование на языке высокого уровня [Электронный ресурс]: методические указания к проведению лабораторных работ по курсу «Программирование»/ Фарафонов А.С. – Электрон. текстовые данные. – Липецк: Липецкий государственный технический университет, ЭБС АСВ, 2013. – 32 с.
<http://www.iprbookshop.ru/22912>
2. Биллиг, В.А. Основы объектного программирования на C# (C# 3.0, Visual Studio 2008) [Электронный ресурс] / В.А. Биллиг. – Электрон. текстовые данные. – М.: БИНОМ. Лаборатория знаний, Интернет-Университет Информационных Технологий (ИНТУИТ), 2010. – 582 с.
<http://www.iprbookshop.ru/16092>
3. Давыдова Н.А. Программирование [Электронный ресурс]: учебное пособие/ Давыдова Н.А., Боровская Е.В. – Электрон. текстовые данные. – М.: БИНОМ. Лаборатория знаний, 2015. – 239 с.
<http://www.iprbookshop.ru/6485>
4. Степанов, В.П. Лабораторный практикум по программированию на языке Си: учеб.-метод. пособие по курсу "Информатика" [Электронный ресурс] / В.П. Степанов. – М.: Издательство МГТУ им. Н.Э. Баумана, 2007. – 48 с.
http://www.studentlibrary.ru/book/bauman_0404.html
5. Бабушкина, И.А. Практикум по объектно-ориентированному программированию [Электронный ресурс]/ И.А. Бабушкина, С.М. Окулов – Электрон. текстовые данные. – М.: БИНОМ. Лаборатория знаний, 2015. – 367 с.
<http://www.iprbookshop.ru/12254>
6. Окулов, С.М. Программирование в алгоритмах [Электронный ресурс] / С.М. Окулов. – М.: БИНОМ, 2014. – 383 с.: ил. – (Развитие интеллекта школьников).
<http://www.studentlibrary.ru/book/ISBN9785996323111.html>

в) интернет-ресурсы:

1. msdn.microsoft.com
2. <https://projecteuler.net/>


8. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

Проведение практических и лекционных занятий осуществляется с использованием мультимедийного комплекса (компьютер +проектор) или интерактивной доски.

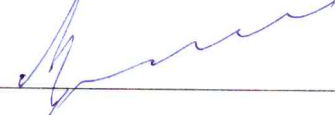
Лабораторные занятия проводятся в компьютерном классе с возможностью индивидуальной работы каждого студента за компьютером. Необходимое программное обеспечение – Microsoft Visual Studio 2008 (и выше).

Рабочая программа дисциплины составлена в соответствии с требованиями ФГОС ВО по направлению 44.03.05 Педагогическое образование.

Рабочую программу составил
ст. преподаватель кафедры информатики и ИТО _____  А.А. Мартынова

Рецензент
Заместитель директора ГБПОУ Владимирской области
«Владимирский педагогический колледж» _____  Н.И. Коршунова

Программа рассмотрена и одобрена на заседании кафедры информатики и ИТО
протокол № 7а от 10.03.2016 года

Заведующий кафедрой
информатики и ИТО _____  Ю.А. Медведев

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

Протокол № 3 от 17.03.2016 года

Председатель комиссии _____ 