

2015

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

УТВЕРЖДАЮ

Проректор  
по учебно-методической работе

А.А.Панфилов

« 29 » 09 2015 г.



**РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ**  
**ОСНОВЫ ПРОГРАММИРОВАНИЯ**  
(наименование дисциплины)

Направление подготовки 02.03.01 Математика и компьютерные науки  
Профиль/программа подготовки Математические методы в экономике и финансах  
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 (экс./зачет)
2	5/180	36	-	36	72	Экзамен(36)
Итого	5/180	36	-	36	72	Экзамен(36)

Владимир 2015

02

## 1. ЦЕЛИ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Целями освоения дисциплины Основы программирования являются: ознакомление с основными методологиями и парадигмами программирования, семантикой и синтаксисом императивного языка программирования и средой разработки приложений.

Задачами дисциплины являются:

- понимание отличий различных парадигм программирования;
- изучение принципов разработки алгоритмов;
- изучение стандартов общего представления алгоритмов на языках программирования;
- изучение семантики и синтаксиса языка программирования;
- знакомство с особенностями сред разработки программного обеспечения;
- приобретение навыков разработки алгоритмов, для решения поставленной задачи, и навыков работы в команде.

## 2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОПОП ВО

Данная дисциплина относится к базовой части ОПОП раздел Б1. Логически и содержательно методологически данная дисциплина связана с дисциплиной «Информатика». Для освоения данной дисциплины требуется наличие у студентов фундаментальных знаний в области информатики, а также навыков построения простейших алгоритмов, знаний понятий алгоритмизации. Знания, полученные в рамках изучения данной дисциплины, могут быть применены при изучении дисциплин «Фундаментальная и компьютерная алгебра», «Алгоритмы и анализ сложности» а также дисциплин, в рамках которых требуется описывать модели на языке программирования и выполнять автоматизированные эксперименты.

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

В результате изучения дисциплины обучающийся должен освоить следующие компетенции:

- способностью находить, анализировать, реализовывать программно и использовать на практике математические алгоритмы, в том числе с применением современных вычислительных систем (ОПК-4).

В результате освоения дисциплины обучающийся должен демонстрировать следующие результаты образования:

- 1) Знать: основные методы и способы программной реализации математических алгоритмов, в том числе с применением современных вычислительных систем (ОПК-4).

- 2) Уметь: формализовать задачу, строить алгоритм её решения (ОПК-4)
- 3) Владеть: навыками анализа, программной реализации математических алгоритмов с применением современных вычислительных (ОПК-4).

#### 4. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)

Общая трудоемкость дисциплины составляет 5 зачетных единиц, 180 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 занятия	Лабораторные работы	Контрольные работы	СРС	КП/КР		
1	Введение в программирование. Среда разработки программного обеспечения	2	1-2	6		4	-	12	-	5 / 50%	Рейтинг-контроль №1
2	Базовые сведения о C++	2	3-6	12		12	-	20	-	12 / 50%	
3	Пользовательские типы данных языка C++	2	7-12	12		12	-	18	-	12 / 50%	Рейтинг-контроль №2
4	Работа с функциями и файлами	2	13-18	6		8	-	22	-	7 / 50%	Рейтинг-контроль №3
<b>Всего</b>		<b>2</b>	<b>18</b>	<b>36</b>		<b>36</b>	<b>-</b>	<b>72</b>		<b>36 / 50%</b>	<b>экзамен (36 час.)</b>

#### СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

##### Лекции

**Раздел 1. Введение в программирование. Среда разработки программного обеспечения.**

1. Развитие языков программирования. Парадигмы программирования
2. Основы алгоритмизации
3. Среда разработки программного обеспечения

**Раздел 2. Базовые сведения о C++.**

1. Общие сведения, структура программы, базовые типы данных, операции на них, встроенные в C++. Основы ввода/вывода.
2. Пространство имен в C++. Понятие области видимости и времени жизни переменных. Классы памяти
3. Операторы языка.

**Раздел 3. Пользовательские типы данных языка C++.**

1. Массивы
2. Строки и операции со строками
3. Указатели и ссылки. Работа с динамической памятью

4. Структуры
5. Другие составные типы данных: перечисления, объединения

#### **Раздел 4. Работа с функциями и файлами.**

1. Работа с функциями: прототип, определение, способы обмена данными, перегружаемые, операторные, подставляемые функции. Шаблоны функций. Параметры, передаваемые через командную строку. Рекурсия
2. Файлы. Базовые функции работы с потоками
3. Работа с линейными списками и двоичными деревьями

#### **Тематика лабораторных занятий**

1. Изучение линейных алгоритмов на языке C++.(4ч.)
2. Изучение разветвлённой алгоритмической структуры на языке C++.(4ч.)
3. Изучение циклических структур алгоритмов.(6ч.)
4. Изучение модификаций циклических алгоритмов.(8ч.)
5. Изучение одномерных массивов.(4ч.)
6. Изучение двумерных массивов и операций над ними.(10ч)

### **5. ОБРАЗОВАТЕЛЬНЫЕ ТЕХНОЛОГИИ**

В данной дисциплине используются следующие методы обучения:

- обучение в малых группах (выполнение лабораторных работ в группах из двух или трёх человек);
- мастер-классы (демонстрация на лабораторных занятиях принципов работы со средой разработки программного обеспечения, особенностей работы с языком программирования и компилятором);
- применение мультимедиа технологий (проведение лекционных занятий с применением компьютерных презентаций и демонстрационных роликов с помощью проектора или ЭВМ, использование сайта поддержки дисциплины для проведения лабораторных занятий);
- технология развития критического мышления (прививание студентам навыков критической оценки разработанных ими программ);
- информационно-коммуникационные технологии (применение информационных технологий для мониторинга текущей успеваемости студентов и контроля знаний).
- метод (case-study) студенты получают «проблемные» задания по тематике изучаемого раздела.

#### **Рейтинговая система обучения**

Рейтинг-контроль проводится три раза за семестр. Он предполагает оценку суммарных баллов по следующим составляющим: баллы на контрольных занятиях; качество выполнения домашних типовых заданий, рассматриваемых на практических

занятиях. Распределение баллов по контрольным мероприятиям определяется лектором, ведущим дисциплину.

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

Текущий контроль успеваемости проводится по всем видам занятий с использованием рейтинговой системы.

### А. Вопросы для рейтинг-контроля.

#### Вопросы для рейтинг-контроля №1.

1. Дайте определение понятию алгоритм.

2. Выберите из списка правильный вариант утверждения: «Линейный алгоритм – это..

А) вычислительная схема, которая содержит не одну, а несколько возможных ветвей решения;

Б) вычислительная структура, при которой все предписания выполняются в строго линейной последовательности друг за другом;

В) схема разветвлённой структуры, в которой одна ветвь операции сравнения является обратной связью.

3. Исправьте ошибку(и) в программе и объясните причину их возникновения.

```
#include {stdafx.h}
```

```
#include <conio.h>
```

```
#include <math.h>
```

```
#define <math.h>
```

```
int main()
```

```
{
```

```
int a
```

```
double x;
```

```
int y;
```

```
y=pow(a,2)+(x/4);
```

```
printf(“y=”,y);
```

```
}
```

4. Выберите правильный вариант записи команды в программе.

А.  $Y = \sqrt{x^2 + \text{abs}(3) - 2} / \text{pow}((1-x), 2)$

Б.  $y = \sqrt{x^2 + (\text{abs}(3) - 2)} / \text{pow}(1-x, 2)$

$$B. Y = \sqrt{\text{POW}(x, 2) + \text{ABS}(3 - 2)}} / \text{POW}(1 - x, 2))$$

Г. Другой вариант

6. Как работает префиксная операция ++? Какое значение будет записано в память и какое использоваться в выражении? Привести пример с пояснениями.

7. Вывести на экран значение целочисленной переменной int x.

### Вопросы для рейтинг-контроля №2.

1. Дайте определение понятию массив.

2. Выберите из списка правильный вариант утверждения: «Указатели – это..

А) переменные, предназначенные для хранения в них адресов каких-либо объектов программы;

Б) тип данных, указывающий на определенную ячейку памяти;

В) переменные, указывающие на местонахождение конкретного параметра в программе.

3. Составьте программу на языке C++, позволяющую отсортировать целочисленный одномерный массив, состоящий из 14 элементов по четным и нечетным позициям и вывести на экран элементы, имеющие нечетную позицию. (В ответе должна быть построена блок-схема).

4. Выберите правильный вариант записи команды в программе.

А. `int mas{12,5,8,9}`

Б `int mas[12], int mas[4]`

В. `int mas[8]`

Г. Другой вариант

5. Составьте программу на языке C++, позволяющую находить сумму элементов матрицы (5\*5), лежащих ниже главной диагонали. (В ответе должна быть построена блок-схема).

6. Дайте определение понятию «утечка памяти».

7. Дать определение массива с точки зрения хранения.

### Вопросы для рейтинг-контроля №3.

1. Дайте определение понятию массив.

2. Выберите из списка правильный вариант утверждения: «Указатели – это..

А) переменные, предназначенные для хранения в них адресов каких-либо объектов программы;

Б) тип данных, указывающий на определенную ячейку памяти;

В) переменные, указывающие на местонахождение конкретного параметра в программе.

3. Составьте программу на языке C++, позволяющую отсортировать целочисленный одномерный массив, состоящий из 14 элементов по четным и нечетным позициям и вывести на экран элементы, имеющие нечетную позицию. (В ответе должна быть построена блок-схема).

4. Выберите правильный вариант записи команды в программе.

А. `int mas{12,5,8,9}`

Б `int mas[12], int mas[4]`

В. `int mas[8]`

Г. Другой вариант

5. Составьте программу на языке C++, позволяющую находить сумму элементов матрицы (5\*5), лежащих ниже главной диагонали. (В ответе должна быть построена блок-схема).

6. Дайте определение понятию «утечка памяти»

7. Выберите из списка правильный вариант утверждения: «Циклический алгоритм – это..

А) вычислительная схема, которая содержит не одну, а несколько возможных ветвей решения;

Б) вычислительная структура, при которой все предписания выполняются в строго линейной последовательности друг за другом;

В) схема разветвлённой структуры, в которой одна ветвь операции сравнения является обратной связью.

8. Составьте программу, которая создает два текстовых файла: в первом файле должна содержаться количество студентов в группе(22), а второй должен содержать отредактированную информацию (20)

## **Б. Вопросы к экзамену.**

1. Понятие алгоритма. Алгоритмизация. Основные этапы разработки алгоритма.
2. Способы записи алгоритма. Блок-схема. Тестирование и отладка программы.
3. Элементы блок-схемы алгоритма: оператор «процесс» и его функции при разработке блок-схемы.
4. Элементы блок-схемы алгоритма: оператор «модификация» и его функции при составлении блок-схемы алгоритма.
5. Элементы блок-схемы алгоритма: оператор «условие» и его применение при составлении блок-схемы.
6. Алгоритмические структуры: линейный алгоритм, основные требования к записи данного вида алгоритма.

7. Алгоритмические структуры: разветвляющийся алгоритм, описание алгоритма, ветвление алгоритма.
8. Алгоритмические структуры: цикл с предусловием, описание алгоритма, форма записи, задание параметров цикла.
9. Алгоритмические структуры: цикл с постусловием, описание алгоритма, форма записи.
10. Алгоритмические структуры: вложенные циклы, правила определения порядка вложения циклов в алгоритме, понятие внешнего и внутреннего циклов.
11. Операторы условия в языке C++: конструкция if...else. Форма записи. Применение при разработке программы.
12. Операторы переключатели в языке C++: оператор switch...case. Форма записи. Применение при разработке программы.
13. Побитовые операции: сложение, вычитание, умножение и т.д.
14. Логические операции.
15. Оператор sizeof. Функции в программе.
16. Операторы ввода-вывода данных: printf(), scanf().
17. Поточковый ввод-вывод данных: cin>>, cout>>.
18. Структура программы на языке C++. Основные элементы структуры.
19. Библиотека <stdafx.h>. Параметры применения и функции.
20. Библиотека <stdio.h>. Форма записи и функции.
21. Библиотека <conio.h>. Форма записи и функции.
22. Библиотека <math.h>. Форма записи и функции.
23. Библиотека <iostream>. Форма записи и функции.
24. Массив: понятие, форма записи, инициализация массивов.
25. Одномерный массив: понятие, форма записи, способ записи.
26. Двумерный массив: понятие, форма записи, способ записи.
27. Сортировка массива. Методы сортировки: бинарный поиск, «пузырек».
28. Указатели: понятие, форма записи, вызов в программе и функции.
29. Статический массив. Форма записи.
30. Динамический массив. Форма записи.
31. Понятие «утечка памяти».
32. Понятие «индексация массива». Применение в программировании.
33. Операция упорядочивание массива по убыванию. Принцип действия в программе.
34. Операция упорядочивание массива по возрастанию.



35. Вспомогательный алгоритм: понятие, форма записи, структура.
36. Функция: понятие, форма записи, структура.
37. Фактические параметры функции.
38. Формальные параметры функции.
39. Фактические аргументы.
40. Прототипы функции: понятие, применение в программировании.
41. Способы передачи параметров в функцию. Ссылки.
42. Вызов функции в программе. Правила оформления функции в программе.
43. Описание функции.
44. Объявление функции.
45. Файлы: определение, классификация, сфера применения в программировании.
46. Форма записи файла в программе.
47. Текстовые файлы.
48. Двоичные файлы.
49. Основные функции для работы с файлами.
50. Подключение стандартных библиотек для ввода-вывода файлов.

## **В. Самостоятельная работа**

### **Вопросы для контроля самостоятельной работы:**

1. Можно ли использовать ключевые слова для именованя объектов программы?
2. Перечислите основные элементы программы.
3. В чем отличия определения и объявления объектов программы?
4. Перечислите базовые типы данных языка и их характеристики.
5. Поясните принцип работы префиксной и постфиксной операций увеличения.
6. Каким образом можно не используя операцию умножения, умножить число на 4.
7. Объясните в каком порядке будут исполнены операции
 
$$a = b + c * d \ll 2 \& 8$$

$$a \& 077 != 3$$

$$a == b \parallel a == c \&\& c < 5$$

$$c = x != 0$$

$$a = -1 ++ b -- -5$$

$$a = b == c ++$$

$$a = b = c = 0$$

$$a-b, c=d$$
8. Перечислите циклические операторы: синтаксис и семантику.

9. Назначение блочного оператора.
10. Поясните принцип работы операторов break и continue.
11. В операторе-переключателе является ли обязательной метка default?
12. Следующий цикл for перепишите с помощью оператора while:  
for (i=0; i<max\_length; i++)  
if (input\_line[i] == '?') quest\_count++;  
Запишите цикл, используя в качестве его управляющей переменной указатель так, чтобы условие имело вид \*p=='?'.
13. Дайте определение массива с точки зрения хранения.
14. Приведите общий принцип обращения к элементам массива.
15. С какого числа начинается нумерация элементов массива? Можно ли нумерацию изменить?
16. Каким образом определить размер массива?
17. Объявить трехмерный массив и изобразить его с точки зрения языка C++.
18. Объявить двумерный массив вещественных элементов размером 5 строк и 4 столбца. Далее:
  - a. Проинициализировать при объявлении;
  - b. Организовать заполнение массива с клавиатуры;
  - c. Организовать заполнение случайными числами.
19. Посчитать сумму элементов массива.
20. Может ли элементом массива быть массив?
21. Что такое строка с точки зрения языка C++?
22. В чем отличие строки от массива символов?
23. Можно ли со строкой работать поэлементно?
24. Написать программу, вычисляющую сумму цифр в строке вида "1ab3c405". Ввод строки организовать с клавиатуры.
25. Написать программу, удаляющую все цифры из символьной строки.
26. Написать фрагмент кода, осуществляющий смену двух строк str1 и str2, если они одинаковой длины.
27. Что такое указатель?
28. Перечислите проблемы, которые могут возникнуть при работе с неинициализированными указателями.
29. Назначение нетипизированного указателя? Каким образом можно объявить нетипизированный указатель?
30. Будет ли корректно работать следующий код:

```
int a = 5;
int *pf = &a;
float *p;
p = pf;
```

Если код работает некорректно, то внести исправления.

31. Объявить массив из трех указателей на вещественные переменные. Задать значения переменных через указатели.

32. Разместить в динамической памяти одномерный массив, двумерный массив.

33. Поясните, что объявлено, проинициализировать все объявленные переменные и нарисовать картинки в памяти и с точки зрения языка Си.

```
int (*pM)[3];
```

```
int *(*pMM)[2];
```

```
int m[2][3];
```

34. Написать фрагмент программы, используя оператор выделения динамической памяти `new`. В программе должен выполняться захват памяти для пяти символов, ввод строки с клавиатуры и освобождение захваченной памяти.

35. Что такое ссылка

36. Объявить ссылку на константу.

37. Назовите отличия структуры от массива

38. Назовите операции доступа к полям структуры по указателю и через объект.

39. Назовите назначение битовых полей и особенности работы с ними.

40. Что такое функция

41. Для чего применяются прототипы функций

42. В чем отличия объявления и определения функции

43. Что означает передача аргумента в функцию по ссылке Приведите пример.

44. Написать функцию ввода с клавиатуры переменной структурного типа (структура Студент содержит поля: ФИО, дата рождения, успеваемость по 5 дисциплинам). Сформированные структуры из функции получать следующими способами:

a. В качестве параметра функции.

b. В качестве возвращаемого значения.

Продемонстрировать работу функции.

45. Вернуть из функции массив указателей на целочисленные переменные.

Продемонстрировать работу функции.

46. Написать шаблонную функцию для нахождения суммы элементов числового массива, массив передавать в функцию в качестве параметра. Привести пример использования функции для нескольких числовых типов.
47. Объявить указатель на функцию, имеющую следующий прототип:  
`int func(char *, int);`
48. Каким образом можно передавать одномерные массивы в функции. Приведите примеры.
49. Поясните принцип передачи многомерных массивов в функции
50. Что такое операторные функции? Назначение перегрузки операторов
51. Дайте определение перегруженной функции.
52. За счет какого механизма возможна перегрузка функций в C++
53. Объявить `inline` функцию, например, вычисления квадрата числа. Объявить указатель на эту функцию. Вызвать `inline` функцию через указатель. Будет ли функция в этом случае трактоваться как встраиваемая
54. Что называется файлом
55. Приведите отличия логического и физического файлов.
56. Перечислите типы файлов
57. Приведите классификацию файлов по способам доступа к информации.
58. Какие действия необходимо сделать для работы с файлом
59. Какая функция используется для открытия файла. Опишите параметры функции.
60. Каким образом можно определить достигнут ли конец файла
61. Написать программу, которая считывает из текстового файла три предложения и выводит их в обратном порядке.
62. Если требуется осуществить быстрое копирование файлов неизвестной структуры, какого типа файл нужно использовать.
63. Написать программу, которая считывает текст из файла и выводит на экран, после каждого предложения добавляя, сколько раз встречалось в нем введенное с клавиатуры слово.

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

### а) основная литература:

1. Основы алгоритмизации и программирования: Учебное пособие / В.Д. Колдаев; Под ред. Л.Г. Гагариной. - М.: ИД ФОРУМ: ИНФРА-М, 2015. - 416 с.: ил.; 60x90 1/16. - (Профессиональное образование). (переплет) ISBN 978-5-8199-0279-0

2. Программирование на языке Си/А.В.Кузин, Е.В.Чумакова - М.: Форум, НИЦ ИНФРА-М, 2015. - 144 с.: 70x100 1/16. - (Высшее образование) (Обложка) ISBN 978-5-00091-066-5
3. Царев, Р. Ю. Программирование на языке Си [Электронный ресурс] : учеб. пособие / Р. Ю. Царев. – Красноярск : Сиб. федер. ун-т, 2014. – 108 с. - ISBN 978-5-7638-3006-4

**б) дополнительная литература:**

1. Программирование на языке высокого уровня. Программирование на языке Object Pascal: Учебное пособие / Т.И. Немцова; Под ред. Л.Г. Гагариной. - М.: ФОРУМ: ИНФРА-М, 2015. - 496 с.: ил.; 60x90 1/16. - (ПО). (п, cd rom) ISBN 978-5-8199-0372-8
2. Введение в среду визуального программирования Turbo Delphi. Ч.3 [Электронный ресурс] : Учеб. пособие / С. В. Борисов, С. С. Комалов, О. Б. Пащенко, И. Л. Серебрякова. - М. : Издательство МГТУ им. Н. Э. Баумана, 2013
3. Задачи по программированию [Электронный ресурс] / С. М. Окулов [и др.] ; под ред. С. М. Окулова. - 2-е изд., испр. (эл.).-М. : БИНОМ. Лаборатория знаний, 2014. - 823 с. : ил. - ISBN 978-5-9963-2372-2

**в) периодические издания**

1. Программист. Архив номеров.2012,2013г. Режим доступа:// <http://procoder.info/>
2. LINUX-FORMAT. Архив номеров. Режим доступа:// <http://www.linuxformat.ru/>

**в) интернет-ресурсы**

1. Основ программирования на C++. Режим доступа:// <https://code-live.ru/tag/cpp-manual/>
2. Основы программирования на языках Си и C++ для начинающих. Режим доступа:// <http://cppstudio.com/>
3. Программирование на С и C++. Онлайн справочник программиста. Режим доступа:// <http://www.c-cpp.ru/books/osnovy-s/>
4. Основы программирования на C++ для начинающих. Режим доступа// <http://purecodecpp.com/>

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

В качестве материально-технического обеспечения дисциплины используются следующие средства: проектор, наборы слайдов по учебной тематике, компьютерные классы с установленным ПО: VS 2012, 2013, 2015, мультимедийные аудитории.

Рабочая программа дисциплины составлена в соответствии с требованиями ФГОС ВО по направлению 02.03.01 Математика и компьютерные науки

Рабочую программу составил Касьянов А.А., доцент кафедры ФиПМ  
(ФИО, подпись)

Рецензент

(представитель работодателя)

Красов Д.С. Руководитель ООО "МС Сервис"  
(место работы, должность, ФИО, подпись)

Программа рассмотрена и одобрена на заседании кафедры ФиПМ

Протокол № 2/12 от 29.01.15 года

Заведующий кафедрой \_\_\_\_\_

Аракелян С.М.

(ФИО, подпись)

Рабочая программа рассмотрена и одобрена на заседании учебно-методической комиссии направления 02.03.01 Математика и компьютерные науки

Протокол № 5/11 от 29.01.15 года

Председатель комиссии \_\_\_\_\_

Давыдов А.А.

(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 \_\_\_\_\_