

МИНОБРНАУКИ РОССИИ  
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ  
ВЫСШЕГО ОБРАЗОВАНИЯ  
«ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»  
(ФГБОУ ВО «ВГУ»)

**УТВЕРЖДАЮ**  
Заведующий кафедрой  
функционального анализа  
и операторных уравнений

 Каменский М.И.

26.06.2018 г.

## РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ

### Б1.В.ДВ.02.01. Технологии программирования

- 1. Код и наименование направления подготовки/специальности:** 01.03.04 Прикладная математика
- 2. Профиль подготовки/специализация:** Применение математических методов к решению инженерных и экономических задач
- 3. Квалификация (степень) выпускника:** бакалавр
- 4. Форма обучения:** очная
- 5. Кафедра, отвечающая за реализацию дисциплины:** функционального анализа и операторных уравнений
- 6. Составители программы:** Пономарев Сергей Сергеевич, преподаватель кафедры функционального анализа и операторных уравнений.
- 7. Рекомендована:** НМС математического факультета, протокол №0500-07 от 03.07.2018
- 8. Учебный год:** 2018-2019 **Семестр:** 2

**9. Цели и задачи учебной дисциплины:** Основной целью освоения дисциплины (модуля) "Технологии программирования" является овладение базовыми навыками построения и анализа алгоритмов, необходимых для разработки эффективных прикладных программ. Целями курса так же является подготовка к работе с различными структурами данных.

В результате усвоения дисциплины студенты должны приобрести практический опыт реализации алгоритмов, выбора подходящей структуры данных для конкретной задачи. Проверять корректность работы заданного алгоритма.

**10. Место учебной дисциплины в структуре ООП:**

Дисциплина входит в вариативную часть цикла естественно-научных дисциплин. И является дисциплиной по выбору. Для изучения и освоения дисциплины нужны первоначальные знания из курсов математического анализа, алгебры, аналитической геометрии. Знания и умения, приобретенные студентами в результате изучения дисциплины, будут использоваться при изучении курсов численных методов, при выполнении курсовых и дипломных работ, связанных с математическим моделированием и обработкой наборов данных.

Дисциплина "Технологии программирования" является предшествующей для изучения следующей дисциплины: «Программирование для ЭВМ».

**11. Планируемые результаты обучения по дисциплине/модулю (знания, умения, навыки), соотнесенные с планируемыми результатами освоения образовательной программы (компетенциями выпускников):**

Компетенция		Планируемые результаты обучения
Код	Название	
ОПК-1	готовность к самостоятельной работе	<p>знать: основные понятия теории алгоритмов</p> <p>уметь: использовать специализированную литературу, описывающую современные методы построения алгоритмов и структур данных</p> <p>владеть: навыками чтения специализированной литературы по алгоритмическим конструкциям</p>
ОПК-2	способность использовать современные математические методы и современные прикладные программные средства и осваивать современные технологии программирования	<p>знать: принципы работы в интегрированной среде разработки программ (например, Microsoft Visual Studio 2015)</p> <p>уметь: реализовывать алгоритм в виде программы на языке C++ в различных интегрированных средах разработки программ (например, Microsoft Visual Studio 2015);</p> <p>владеть: основными инструментами отладки, компиляции в интегрированной среде разработки программ (например, Microsoft Visual Studio 2015)</p>
ПК-9	способность выявить естественнонаучную сущность проблем, возникающих в ходе профессиональной деятельности, готовностью использовать для их решения соответствующий естественнонаучный аппарат	<p>знать: методы оценки эффективности алгоритмов</p> <p>уметь: реализовывать алгоритмы с использованием языка C++</p> <p>владеть: навыками создания эффективных прикладных программ на C++</p>
ПК-11	готовность применять знания и навыки	<p>знать: алгоритмы сортировки, поиска, алгоритмы работы со строками, алгоритмы на графах.</p>

	управления информацией	уметь: применять перечисленные в графе знать понятия для создания программ владеть: навыками поиска информации по особенностям конкретной поставленной задачи
ПК-10	готовность применять математический аппарат для решения поставленных задач, способностью применить соответствующую процессу математическую модель и проверить ее адекватность, провести анализ результатов моделирования, принять решение на основе полученных результатов	Знать: как применять математический аппарат для решения поставленных задач и для разработки программ. Уметь: применять математический аппарат для решения поставленных задач Владеть: способностью применить соответствующую процессу математическую модель и проверить ее адекватность, провести анализ результатов моделирования, принять решение на основе полученных результатов
ПК-12	способность самостоятельно изучать новые разделы фундаментальных наук	Знать: как самостоятельно изучать новые разделы фундаментальных наук. Уметь: самостоятельно изучать новые разделы фундаментальных наук. Владеть: навыками самостоятельного изучения новых разделов фундаментальных наук и уметь применять знания в программировании.

**12. Объем дисциплины в зачетных единицах/час — 4/144.**

**Форма промежуточной аттестации экзамен.**

### 13. Виды учебной работы

Вид учебной работы	Трудоемкость	
	Всего	По семестрам
		№ семестра
		2
Аудиторные занятия	50	50
в том числе: лекции	16	16
Практические	-	-
Лабораторные	34	34
Самостоятельная работа	58	58
Контроль	36	36
Форма промежуточной аттестации		2 контрольные работы, экзамен
Итого:	144	144

#### 13.1. Содержание дисциплины

п/п	Наименование раздела дисциплины	Содержание раздела дисциплины
<b>1. Лекции</b>		
1.1	Основы анализа алгоритмов	Общая характеристика Классов входных данных. Основные

		понятия сложности. Выбор подходящего языка программирования.
1.2	Необходимые математические сведения	Логарифмы, бинарные деревья, O-символика. Классификация скоростей роста
1.3	Алгоритмы поиска и выборки	Последовательный поиск, двоичный поиск, выборка
1.4	Алгоритмы сортировки	Сортировка вставками, пузырьковая сортировка, сортировка Шелла, корневая сортировка, сортировка слиянием.
1.5	Численные алгоритмы	Вычисление значений многочленов, Схема Горнера, умножение матриц, решение линейных уравнений, метод Гаусса-Жордана
1.6	Алгоритмы сравнения с образцом	Сравнение строк, Конечные автоматы, алгоритм Кнута-Морриса-Пратта, алгоритм Бойера-Мура, приблизительное сравнение строк
1.7	Алгоритмы на графах	Основные понятия теории графов, структуры данных для представления графов. Определение Матрицы примыканий и Списка примыканий. Алгоритмы обхода в глубину и по уровням. Алгоритм поиска минимального остовного дерева. Алгоритм поиска кратчайшего пути.
1.8	Параллельные алгоритмы	Введение в параллелизм. Категории компьютерных систем, примеры параллельных архитектур, принципы анализа параллельных алгоритмов. Простые параллельные операции. Параллельный поиск. Параллельная сортировка.
1.9	Недетерминированные алгоритмы	Определение NP. Сведение данной задачи к другой. NP-полные задачи. Раскраска графа. Раскладка по ящикам. Задача о суммах элементов подмножеств. Проверка возможных решений.
1.10	Жадные приближенные алгоритмы	Приближения в задаче о коммивояжере. Приближения в задаче о раскладке по ящикам. Приближения в задаче об упаковке рюкзака. Приближения в задаче о раскраске графа.
<b>2. Практические занятия</b>		
<b>3. Лабораторные работы</b>		
3.1	Основы анализа алгоритмов	Алгоритмы «грубой силы»
3.2	Необходимые математические сведения	Рекурсия. Логарифмы, бинарные деревья, O-символика.
3.3	Алгоритмы поиска и выборки	Линейный поиск. Бинарный поиск. Интерполяционный поиск. Асимптотическая сложность алгоритма
3.4	Алгоритмы сортировки	Базовый случай и рекурсивный случай. Стек. Стек вызовов. Стек вызовов с рекурсией. «Разделяй и властвуй»
3.5	Численные алгоритмы	Рандомизация данных. Генерирование случайных величин. Рандомизация массивов. Работа с простыми числами: нахождение простых множителей, нахождение простых элементов, проверка на простоту.
3.6	Алгоритмы сравнения с образцом	Сравнение строк. Конечные автоматы. Приблизительное сравнение строк
3.7	Алгоритмы на графах	Обход в глубину. Обход по уровням. Анализ алгоритмов обхода. Алгоритм поиска минимального остовного дерева. Алгоритм поиска кратчайшего пути
3.8	Параллельные алгоритмы	Сортировка на линейных сетях. Четно-нечетная сортировка перестановками. Параллельный алгоритм поиска кратчайшего пути.
3.9	Недетерминированные	Задача планирования работ. Задача о суммах элементов

	алгоритмы	подмножеств. Упаковка рюкзака.
3.10	Жадные приближенные алгоритмы	Преобразование решения динамического программирования в жадное решение. Оптимальная подструктура задачи о выборе процессов

### 13.2. Темы (разделы) дисциплины и виды занятий

№ п/п	Наименование темы (раздела) дисциплины	Виды занятий (часов)				Всего
		Лекции	Контроль	Лабораторные	Самостоятельная работа	
1.1	Основы анализа алгоритмов	1	4	2	3	10
1.2	Необходимые математические сведения	2	2	2	4	10
1.3	Алгоритмы поиска и выборки	2	4	4	8	18
1.4	Алгоритмы сортировки	2	4	6	8	20
1.5	Численные алгоритмы	1	6	4	6	17
1.6	Алгоритмы сравнения с образцом	2	4	2	6	14
1.7	Алгоритмы на графах	1	2	4	6	13
1.8	Параллельные алгоритмы	2	4	4	6	16
1.9	Недетерминированные алгоритмы	2	4	4	6	16
1.10	Жадные приближенные алгоритмы	1	2	2	5	10
	Итого	16	36	34	58	144

### 14. Методические указания для обучающихся по освоению дисциплины

Аудиторные занятия, лекции и лабораторные занятия, предполагают самостоятельную работу студентов по данному курсу. Ряд тем выносятся для самостоятельного изучения, предлагаются темы для создания докладов с презентациями. Предусмотрены домашние задания и оформление отчетов выполнения лабораторных заданий, а также дополнительные задания для сильных студентов.

### 15. Перечень основной и дополнительной литературы, ресурсов интернет, необходимых для освоения дисциплины (список литературы оформляется в соответствии с требованиями ГОСТ и используется общая сквозная нумерация для всех видов источников)

а) основная литература:

1	Левитин, Ананий. Алгоритмы: введение в разработку и анализ. / Ананий Левитин ; [пер. с англ. Издательский дом «Вильямс»] — 3-е изд. — Москва ; Санкт-Петербург ; Киев : Вильямс, 2006 .— 576 с.
2	Кормен, Томас. Алгоритмы: построение и анализ. / Т. Кормен, Ч. Лейзерсон, Р. Ривест, — М: МЦНМО, 2001. — 960 с

б) дополнительная литература:

№ п/п	Источник
3	Кнут, Дональд. Искусство программирования, том 1. Основные алгоритмы. = The Art of Computer Programming. Volume 1. Fundamental Algorithms / под ред. С. Г. Тригуб (гл. 1), Ю. Г. Гордиенко (гл. 2) и И. В. Красикова (разд. 2.5 и 2.6). — 3. — Москва: Вильямс, 2002. — Т. 1. — 720 с.

в) информационные электронно-образовательные ресурсы:

№ п/п	Источник
4	lib.vsu.ru

\* Вначале указываются ЭБС, с которыми имеются договора у ВГУ, затем открытые электронно-образовательные ресурсы

### 16. Перечень учебно-методического обеспечения для самостоятельной работы

В течение семестра студенты разбирают и решают задачи, указанные преподавателем к каждому лабораторному занятию. Каждый студент получает свое индивидуальное задание (см. пример индивидуального задания). Предусмотрены следующие самостоятельные работы: 1 семестр – 5.

### 17. Информационные технологии, используемые для реализации учебной дисциплины, включая программное обеспечение и информационно-справочные системы (при необходимости)

#### Microsoft Visual Studio 2015 Community

### 18. Материально-техническое обеспечение дисциплины:

(при использовании лабораторного оборудования указывать полный перечень, при большом количестве оборудования можно вынести данный раздел в приложение к рабочей программе)

Лекционная аудитория (доска, мел, маркеры), компьютерные классы для проведения лабораторных работ, мультимедийный проектор.

### 19. Фонд оценочных средств:

#### 19.1. Перечень компетенций с указанием этапов формирования и планируемых результатов обучения

Код и содержание компетенции (или ее части)	Планируемые результаты обучения (показатели достижения заданного уровня освоения компетенции посредством формирования знаний, умений, навыков)	Этапы формирования компетенции (разделы (темы) дисциплины или модуля и их наименование)	ФОС* (средства оценивания)
ОПК-1 готовность к самостоятельной работе	знать: основные понятия теории алгоритмов	1.1-1.10	19.3.1 19.3.2 19.3.4
	уметь: реализовывать алгоритм в виде программы на языке С++ в различных интегрированных средах разработки программ (например, Microsoft Visual Studio 2015);	1.1-1.10	19.3.3
	2015); владеть: основными инструментами отладки, компиляции в интегрированной среде разработки программ (например, Microsoft Visual Studio 2015)	1.1-1.10	19.3.3
ОПК-2 способность использовать современные математические методы и современные прикладные программные средства и осваивать	знать: принципы работы в интегрированной среде разработки программ (например, Microsoft Visual Studio 2015)	1.1-1.10	19.3.2
	уметь: реализовывать алгоритм в виде программы на	1.1-1.10	19.3.2

современные технологии программирования	языке C++ в различных интегрированных средах разработки программ (например, Microsoft Visual Studio 2015);		
	владеть: основными инструментами отладки, компиляции в интегрированной среде разработки программ (например, Microsoft Visual Studio 2015)	1.1-1.10	19.3.2
ПК-9 способность выявить естественнонаучную сущность проблем, возникающих в ходе профессиональной деятельности, готовностью использовать для их решения соответствующий естественнонаучный аппарат	знать: особенности синтаксиса языка C++	1.1-1.10	19.3.1 19.3.2 19.3.4
	уметь: реализовывать алгоритмы с использованием языка C++	1.1-1.10	19.3.1 19.3.2 19.3.4
	владеть: навыками создания прикладных программ на C++	1.1-1.10	19.3.1 19.3.2 19.3.4
ПК-11 готовность применять знания и навыки управления информацией	знать: методы оценки эффективности алгоритмов	1.1-1.10	19.3.2 19.3.4
	уметь: реализовывать алгоритмы с использованием языка C++	1.1-1.10	19.3.2 19.3.4
	владеть: навыками создания эффективных прикладных программ на C++	1.1-1.10	19.3.2 19.3.4
ПК-12 способность самостоятельно изучать новые разделы фундаментальных наук	Знать: как самостоятельно изучать новые разделы фундаментальных наук.	1.1-1.10	19.3.1 19.3.2 19.3.4
	Уметь: самостоятельно изучать новые разделы фундаментальных наук.	1.1-1.10	19.3.1 19.3.2 19.3.4
	Владеть: навыками самостоятельного изучения новых разделов фундаментальных наук и уметь применять знания в программировании.	1.1-1.10	19.3.2 19.3.4
ПК-10 готовность применять математический аппарат для решения поставленных задач, способностью применить соответствующую процессу математическую модель и проверить ее адекватность, провести анализ результатов моделирования, принять решение на основе полученных результатов	Знать: как применять математический аппарат для решения поставленных задач и для разработки программ.	1.1-1.10	19.3.1 19.3.2 19.3.4
	Уметь: применять математический аппарат для решения поставленных задач	1.1-1.10	19.3.2 19.3.4
	Владеть: способностью	1.1-1.10	19.3.1

	применить соответствующую процессу математическую модель и проверить ее адекватность, провести анализ результатов моделирования, принять решение на основе полученных результатов		19.3.2 19.3.4
Промежуточная аттестация			КИМ

## 19.2 Описание критериев и шкалы оценивания компетенций (результатов обучения) при промежуточной аттестации

владение понятийным аппаратом данной области науки (теоретическими основами дисциплины), способность иллюстрировать ответ примерами, фактами, применять теоретические знания для решения практических задач в области технологии программирования

Для оценивания результатов обучения на экзамене используется 4-балльная шкала: «отлично», «хорошо», «удовлетворительно», «неудовлетворительно».

Критерии оценивания компетенций	Уровень сформированности компетенций	Шкала оценок
<i>Обучающийся в полной мере владеет понятийным аппаратом теории алгоритмов. Способен иллюстрировать ответ примерами, применять теоретические знания для создания алгоритмов и реализации в виде программ на языке C++</i>	<i>Повышенный уровень</i>	<i>Отлично</i>
<i>Обучающийся владеет понятийным аппаратом теории алгоритмов. Способен иллюстрировать ответ примерами допускает ошибки при создании алгоритмов и реализации в виде программ на языке C++ или способен применять теоретические знания для создания алгоритмов и реализации в виде программ на языке C++, но допускает неточности при применении понятийного аппарата, но отвечает на дополнительные вопросы</i>	<i>Базовый уровень</i>	<i>Хорошо</i>
<i>Обучающийся владеет частично теоретическими основами дисциплины, фрагментарно способен иллюстрировать ответ примерами, фактами, не отвечает на дополнительные вопросы. Не умеет применять теоретические знания для создания алгоритмов и реализации в виде программ на языке C++</i>	<i>Пороговый уровень</i>	<i>Удовлетворительно</i>
<i>Ответ на контрольно-измерительный материал не соответствует любым трем(четырем) из перечисленных показателей. Обучающийся демонстрирует отрывочные, фрагментарные знания, допускает грубые ошибки.</i>	–	<i>Неудовлетворительно</i>

**19.3 Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующие этапы формирования компетенций в процессе освоения образовательной программы**

**19.3.1**

**Перечень вопросов к экзамену:**

1. Свойства алгоритма. Асимптотическая сложность алгоритма. Обычные функции рабочего цикла. Визуализация функций.
2. Рандомизация данных. Генерирование неравномерных распределений. Возведение в степень. Нахождение наибольшего общего делителя.
3. Численное интегрирование. Формула прямоугольников. Формула трапеций. Адаптивная квадратура.
4. Связные списки. Однонаправленные связные списки. Передвижение по спискам. Нахождение ячеек. Использование ограничителей. Добавление ячеек в начало списка. Добавление ячеек в конец списка.
5. Двухнаправленные связные списки. Копирование. Сортировка вставкой. Сортировка методом выбора
6. Многопоточковые связные списки. Связные списки с циклами. Маркировка ячеек. Использование хеш-таблиц.
7. Одномерные массивы. Нахождение элементов. Нахождение минимальной, максимальной и средней величин. Вставка элементов. Удаление элементов.
8. Треугольные массивы. Массивы с разрывом. Нахождение строки и столбца. Получение значения. Установка значения. Удаление значения.
9. Стеки связных списков. Стеки массивов. Двойные стеки.
10. Алгоритмы с использованием стеков
11. Очереди связных списков. Очереди массивов. Специализированные очереди.
12. Алгоритмы с использованием очередей.
13. Сортировка вставкой в массивах. Сортировка выбором в массивах. Пузырьковая сортировка.
14. Пирамидальная сортировка. Быстрая сортировка. Сортировка слиянием.
15. Сортировка подсчетом. Блочная сортировка.
16. Линейный поиск. Бинарный поиск. Интерполяционный поиск.
17. Рекурсия, базовые алгоритмы. Факториал. Числа Фибоначчи. Ханойская башня.
18. Алгоритмы с возвратом. Сочетания и размещения. Сочетания с циклами. Сочетания с повторениями.
19. Свойства бинарного дерева. Общие правила построения деревьев. Построение завершенных деревьев.
20. Обход в прямом порядке. Симметричный обход. Обход в обратном порядке. Обход в ширину.

**19.3.2 Перечень практических заданий**

**Перечень индивидуальных заданий**

1. Разработайте алгоритм для нахождения наибольшего элемента в несортированном однонаправленном списке с ячейками, содержащими целые числа.
2. Создайте алгоритм для добавления элемента в конец двунаправленного списка.
3. Напишите алгоритм, где значение каждого элемента прямоугольного массива — это расстояние до ближайшего края массива.
4. Напишите программу, которая реализует пузырьковую сортировку.
5. Напишите программу, которая реализует пирамидальную сортировку.
6. Напишите псевдокод, выполняющий обратный симметричный обход в связанном сортированном дереве.

#### 19.3.4 Перечень заданий для контрольных работ

### Форма контрольно-измерительного материала № 1

УТВЕРЖДАЮ  
Заведующий кафедрой функционального анализа и операторных уравнений

\_\_\_\_\_  
*подпись, расшифровка подписи*

\_\_ . \_\_ . 20\_\_

Направление подготовки / специальность 01.03.04 Прикладная математика  
*шифр, наименование*

Дисциплина Б1.В.ДВ.2.1. Технологии программирования

Форма обучения очная  
*очное, очно-заочное, заочное*

Вид контроля экзамен  
*экзамен, зачет;*

Вид аттестации промежуточная  
*текущая, промежуточная*

#### Контрольно-измерительный материал № 1

1. Напишите программу, которая разбирает и вычисляет арифметические выражения, содержащие вещественные числа и операторы +, -, \* и /.
2. Создайте программу, применяющую интерполяционный поиск.

Преподаватель \_\_\_\_\_ С.С. Пономарев  
*подпись расшифровка подписи*

## Форма контрольно-измерительного материала № 2

УТВЕРЖДАЮ  
Заведующий кафедрой функционального анализа и операторных уравнений

\_\_\_\_\_  
подпись, расшифровка подписи

\_\_ . \_\_ . 20\_\_

Направление подготовки / специальность 01.03.04 Прикладная математика

*шифр, наименование*

Дисциплина Б1.В.ДВ.2.1. Технологии программирования

Форма обучения очная

*очное, очно-заочное, заочное*

Вид контроля экзамен

*экзамен, зачет;*

Вид аттестации промежуточная

*текущая, промежуточная*

### Контрольно-измерительный материал № 2

1. Реализуйте алгоритм линейного поиска в отсортированных связных списках.
2. Реализуйте программу рекурсивного интерполяционного поиска.

Преподаватель \_\_\_\_\_ С.С. Пономарев  
*подпись расшифровка подписи*

# Комплект заданий для контрольной работы № 1

по дисциплине *Технологии программирования*  
(наименование дисциплины)

Темы:.....Алгоритмы сортировки. Сортировка вставкой в массивах

**Задание.** Составьте алгоритм и напишите программу для решения ниже приведенной задачи. Ввод данных и вывод результатов оформляйте с выводом на экран поясняющих сообщений.

Создайте программу, которая реализует сортировку выбором.

## Критерии оценки:

- оценка «отлично» выставляется студенту, если правильно и рационально составлен алгоритм, написана работающая программа, соответствующая составленному алгоритму;

- оценка «хорошо» выставляется студенту, если правильно, но нерационально составлен алгоритм, написана работающая программа, соответствующая составленному алгоритму;

- оценка «удовлетворительно» выставляется студенту, если составлен алгоритм с негрубыми ошибками и написана работающая программа, соответствующая составленному алгоритму;

- оценка «неудовлетворительно» выставляется студенту, если алгоритм не составлен или составлен алгоритм с грубыми ошибками; не написана программа или написана неработающая программа.

-

- оценка «зачтено» выставляется студенту, если .....

- оценка «не зачтено» .....

Составитель \_\_\_\_\_ С.С. Пономарев.  
(подпись)

19.06.2017 г.

## Комплект заданий для контрольной работы № 2

по дисциплине *Технологии программирования*  
(наименование дисциплины)

**Тема** Алгоритмы на графах

**Задание.** Напишите псевдокод, выполняющий обратный симметричный обход в связном сортированном дереве.

- оценка «отлично» выставляется студенту, если правильно и рационально составлен алгоритм, написана работающая программа, соответствующая составленному алгоритму;

- оценка «хорошо» выставляется студенту, если правильно, но нерационально составлен алгоритм, написана работающая программа, соответствующая составленному алгоритму;

- оценка «удовлетворительно» выставляется студенту, если составлен алгоритм с негрубыми ошибками и написана работающая программа, соответствующая составленному алгоритму;

- оценка «неудовлетворительно» выставляется студенту, если алгоритм не составлен или составлен алгоритм с грубыми ошибками; не написана программа или написана неработающая программа.

-

- оценка «зачтено» выставляется студенту, если .....

- оценка «не зачтено» .....

Составитель \_\_\_\_\_ С.С. Пономарев  
(подпись)

19.06.2017 г.

#### **19.4. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций**

Оценка знаний, умений и навыков, характеризующая этапы формирования компетенций в рамках изучения дисциплины осуществляется в ходе текущей и промежуточной аттестаций.

Текущая аттестация проводится в соответствии с Положением о текущей аттестации обучающихся по программам высшего образования Воронежского государственного университета. Текущая аттестация проводится в формах: *устного опроса (индивидуальный опрос,); письменных работ (контрольные) и практических заданий в виде создания программ*. Критерии оценивания приведены выше.

Промежуточная аттестация проводится в соответствии с Положением о промежуточной аттестации обучающихся по программам высшего образования.

Контрольно-измерительные материалы промежуточной аттестации (экзамен) включают в себя 1 теоретический вопрос, позволяющий оценить уровень полученных знаний и практическое задание, позволяющее оценить степень сформированности умений и навыков.

Контрольно-измерительные материалы промежуточной аттестации (экзамен) включают в себя 2 теоретических вопроса, позволяющих оценить уровень полученных знаний и практическое задание, позволяющее оценить степень сформированности умений и навыков.

При оценивании используются качественные шкалы оценок. Критерии оценивания приведены выше.