

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РФ  
Федеральное государственное бюджетное образовательное учреждение высшего  
образования  
«ДАГЕСТАНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»  
Факультет математики и компьютерных наук

**РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ**  
Методы организации параллельных вычислений

Кафедра дискретной математики и информатики факультета математики и  
компьютерных наук

Образовательная программа  
**01.04.02** – Прикладная математика и информатика

Профиль подготовки  
Математическое моделирование и вычислительная математика

Уровень высшего образования **магистратура**

Форма обучения **очная**

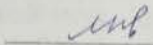
Статус дисциплины: вариативная по выбору

Махачкала, 2017

Рабочая программа дисциплины «Методы организации параллельных вычислений» составлена в 2017 году в соответствии с требованиями ФГОС ВО по направлению подготовки 01.04.02 – Прикладная математика и информатика (уровень магистратуры) от 28 августа 2015 г. № 911.  
Разработчик(и): кафедра дискретной математики и информатики,  
Ханикалов Х.Б. - старший преподаватель кафедры дискретной математики и информатики

Рабочая программа дисциплины одобрена:

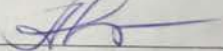
на заседании кафедры дискретной математики и информатики от «13» января 2017 г., протокол № 5.

Зав. кафедрой  Магомедов А.М.  
(подпись)

на заседании Методического совета факультета математики и компьютерных наук «17» января 2017 г., протокол № 5.

Председатель  Меджидов З.Г.  
(подпись)

Рабочая программа дисциплины согласована с учебно-методическим

управлением «    » \_\_\_\_\_ 20   г.   
(подпись)

## Аннотация рабочей программы дисциплины

Дисциплина «Методы организации параллельных вычислений» входит в вариативную часть образовательной программы магистратуры по направлению **01.04.02 – Прикладная математика и информатика** и является дисциплиной по выбору.

Дисциплина реализуется на факультете математики и компьютерных наук кафедрой дискретной математики и информатики.

Содержание дисциплины охватывает круг вопросов, связанных с ключевыми понятиями из области параллельных вычислений и формированием у студентов представлений о современных параллельных вычислительных архитектурах, моделях, методах и технологий их программирования, привить навыки работы с современными вычислительными системами.

Дисциплина способствует формированию следующих компетенций выпускника: профессиональных – ОК – 1, ПК – 1, ПК – 2

Преподавание дисциплины предусматривает проведение следующих видов учебных занятий: лекции, лабораторные занятия.

Рабочая программа дисциплины предусматривает проведение следующих видов контроля успеваемости: .

Объем дисциплины – 3 зачетные единицы, в том числе в академических часах по видам учебных занятий:

Семестр	Учебные занятия						СРС, в том числе экзамен	Форма промежуточной аттестации (зачет, дифференци- рованный зачет, экзамен
	в том числе							
	Контактная работа обучающихся с преподавателем							
	Всего	из них						
Лек- ции		Лабораторные занятия	Практические занятия	КСР	Консультации			
	108	2	30				76	

## 1. Цели изучения дисциплины

Цели изучения дисциплины:

- освоение базовых понятий из области параллельных вычислений,
- формирование представлений о современных параллельных вычислительных архитектурах, моделях, методах и технологий их программирования,
- выработать навыки работы с современными вычислительными системами,

Задачами изучения дисциплины является:

Обучение студентов методам параллельных вычислений с использованием мощных вычислительных систем с распределенной памятью, развитие у студентов доказательного, логического мышления; знакомство с различными языками, применяемыми на упомянутых системах, подготовка к самостоятельному решению различных алгоритмических задач с использованием этих систем.

## 2. Место дисциплины в структуре ООП магистратуры

Дисциплина «Методы организации параллельных вычислений» входит в вариативную часть и является дисциплиной по выбору образовательной программы магистратуры, по направлению 01.04.02 – «Фундаментальная информатика и информационные технологии».

Успешному изучению дисциплины способствуют знания, полученные по дисциплине «Языки программирования» и «Дискретная математика», а также при изучении фундаментальных и общематематических дисциплин.

## 3. Компетенции обучающегося, формируемые в результате освоения дисциплины (перечень планируемых результатов обучения)

Компетенции	Формулировка компетенции из ФГОС ВО	Планируемые результаты обучения (показатели достижения заданного уровня освоения компетенций)
ОК-1	Способностью к абстрактному мышлению, анализу, синтезу.	Знать: принципы самостоятельного поиска достоверных источников информации. Уметь: обрабатывать, анализировать и синтезировать информацию для выбора метода решения проблемы в стандартных условиях. Владеть: навыками решения проблемы с использованием выбранного метода.
ПК-1	Способностью проводить научные исследования и получать новые научные и прикладные результаты	Знать: ключевые понятия и принципы организации параллельных вычислений. Уметь: работать с базовым набором средств разработки параллельных программ для

	самостоятельно и в составе научного коллектива	вычислительных кластеров, построенных как на базе стандартной вычислительной архитектуры, так и с применением графических процессоров. Владеть: определением общих форм, закономерностей, инструментальных средств для анализа архитектуры параллельных систем.
ПК-2	Способностью использовать углубленные теоретические и практические знания в области информационных технологий и прикладной математики, фундаментальных концепций и системных методологий, международных и профессиональных стандартов в области информационных технологий	Знать: основные тенденции развития параллельных архитектур, факторы, влияющие на производительность, критерии выбора программно-аппаратной платформы для решения вычислительно-сложных задач заданного класса. Уметь: пользоваться средствами удаленного доступа к вычислительным ресурсам коллективного пользования и запуска параллельных программ на вычислительных кластерах. Владеть: общей методикой разработки параллельных программ, способами оценки эффективности параллельных алгоритмов и максимально достижимого параллелизма на целевой вычислительной архитектуре.

#### 4. Объем, структура и содержание дисциплины

4.1. Объем дисциплины составляет 3 зачетных единиц, 108 академических часа.

4.2. Структура дисциплины

№ п/п	Раздел дисциплины	Се м е ст р	Не де ля се ме ст ра	Виды учебной работы, включая самостоятельную работу студентов и трудоемкость (в часах)					Ф ор м ы те ку щ ег о ко н т ро л я
				Всего	Лек	Лаб.	Пр.	Сам.	
<b>Модуль 1. Введение в теорию параллельных вычислений</b>									
1.	Цели, задачи и проблемы параллельных вычислений.		1	12	2	2		8	
2.	Обзор параллельных вычислительных систем и их классификация. Способы организации параллельной обработки данных.		2	12		4		8	

3.	Архитектура высокопроизводительных ЭВМ.		3	12		4		8	
	<b>Итого по модулю 1</b>			<b>36</b>	<b>2</b>	<b>10</b>		<b>24</b>	коллоквиум
<b>Модуль 2. Средства разработки параллельных программ</b>									
4.	Базовые средства параллельного программирования систем с общей памятью.		4	16		4		14	
5.	Спецификации OpenMP для языков программирования.		5	16		4		14	
	<b>Итого по модулю 2</b>			<b>36</b>		<b>8</b>		<b>28</b>	коллоквиум
<b>Модуль 3. Принципы разработки параллельных алгоритмов и программ</b>									
6.	Использование распространенных языков программирования и коммуникационных библиотек и интерфейсов.		6	18		6		12	
7	Построение исполняемой программы для параллельной вычислительной системы.		7	18		6		12	
	<b>Итого по модулю 3</b>			<b>36</b>		<b>12</b>		<b>24</b>	коллоквиум
	<b>Итого за семестр</b>			<b>108</b>	<b>2</b>	<b>30</b>		<b>76</b>	<b>Экзамен</b>

#### 4.3. Подробное содержание дисциплины, структурированное по разделам

##### Модуль 1. Введение в теорию параллельных вычислений

**Тема 1. Цели, задачи и проблемы параллельных вычислений** Цели и задачи параллельной обработки данных. Необходимость и актуальность параллельных вычислений. Различия между многозадачным, параллельным и распределенным режимами выполнения программ.

**Тема 2. Обзор параллельных вычислительных систем и их классификация. Способы организации параллельной обработки данных.** Способы построения многопроцессорных вычислительных систем. Краткая история развития высокопроизводительных вычислений. Примеры параллельных вычислительных систем.

**Тема 3. Архитектура высокопроизводительных ЭВМ.** Введение в вычисления общего назначения с использованием GPU. Основные архитектурные отличия GPU от CPU.

Архитектура современных GPU. Технология CUDA. Модели и шаблоны программирования с использованием технологии CUDA. Модель памяти CUDA. Типы памяти. Оптимизация CUDA-приложений. Модель исполнения CUDA. Компиляция CUDA-приложений.

## **Модуль 2. Средства разработки параллельных программ.**

**Тема 1. Базовые средства параллельного программирования систем с общей памятью.** Базовая вычислительная система. Понятие о временной развертке. Достоинства и недостатки базовой вычислительной системы. О построении графа вычислительной системы.

**Тема 2. Спецификации OpenMP для языков программирования.** Общие сведения. Структура стандарта OpenMP. Достоинства технологии OpenMP. Модель параллелизма OpenMP. Модель памяти OpenMP. Директивы OpenMP. Типы директив.

## **Модуль 3. Принципы разработки параллельных алгоритмов и программ**

**Тема 1. Использование распространенных языков программирования и коммуникационных библиотек и интерфейсов.** Традиционные последовательные языки и распараллеливающие компиляторы, проблема выделения потенциального параллелизма последовательных программ. Параллельные языки программирования и расширения стандартных языков. Средства автоматического распараллеливания, параллельные компиляторы.

**Тема 2. Построение исполняемой программы для параллельной вычислительной системы.** Пример программы произведения матриц. Ускорение Matlab-расчетов на GPU.

### **Темы лабораторных занятий.**

Лабораторные занятия проводятся в специально оборудованных помещениях – компьютерных классах, где установлено необходимое программное обеспечение. Ниже приведены темы лабораторных занятий:

1. Программирование операций на параллельной системе.
2. Отображение задач на вычислительную систему.
3. Построение параллельных форм. Направленный граф.
4. Параллелизм на примере модельных задач нахождения частных сумм последовательности числовых значений и умножения матриц.
5. Выравнивание массивов. Параллельное выполнение циклов.
6. Модельный пример: умножение матрицы на вектор.

7. Операции передачи данных между двумя процессами.
8. Разработка, отладка и исполнение параллельной программы с использованием функций MPI.
9. Анализ параллельных программ с использованием MPI.
10. Параллельное программирование в среде OpenMP .
11. Технология программирования DVM.
12. Ускорение Matlab-расчетов на GPU.

## 5. Образовательные технологии

Процесс изложения учебного материала сопровождается систематическими компьютерными презентациями и демонстрацией решения задач в интерактивном режиме с использованием мультимедийного оборудования, предусмотрено широкое использование в учебном процессе активных и интерактивных форм проведения занятий. Предусмотрено общение и консультации с представителями российских и зарубежных компаний (из числа выпускников кафедры) как по электронной почте и скайпу, так и очные встречи.

## 6. Учебно-методическое обеспечение самостоятельной работы студентов

*Виды самостоятельной работы обучающегося, порядок их выполнения и контроля, учебно-методическое обеспечение (возможно в виде ссылок) самостоятельной работы по отдельным разделам дисциплины.*

### 6.1. Виды и порядок выполнения самостоятельной работы

1. Изучение конспектов лекций и рекомендованной литературы.
2. Выполнение лабораторных работ
3. Подготовка к текущему и промежуточному контролю
4. Поиск материала на интернет-форумах
5. Подготовка к экзамену.

### 6.2. Порядок контроля:

1. Опрос на лабораторных занятиях
2. отчеты по лабораторным
3. Коллоквиумы
4. Экзамен.

Раздел (модуль, тема)	Вид самостоятельной работы и практическое содержание	Контрольные сроки (в нед.) и вид контроля	Уч.-мет.обеспечение (указаны источники из списка основной литературы)
1.1	Программирование операций на параллельной системе.	1	[1]
1.2	Построение параллельных форм.	2	[1]



	Направленный граф.		
1.3	Параллелизм на примере модельных задач нахождения частных сумм последовательности числовых значений и умножения матриц.	3	[1]
2.1	Выравнивание массивов. Параллельное выполнение циклов.	4	[2]
2.2	Разработка, отладка и исполнение параллельной программы с использованием функций MPI.	5	[2]
3.1	Параллельное программирование в среде OpenMP .	6	[3]
3.2	Технология программирования DVM.	7	[3]
	Темы 1.1-3.2 (подготовка к экзамену)	Экзамен	[1]-[3], электронные материалы дисциплины

**Текущий контроль:**

1. Проверка программ на языке высокого уровня по заданиям;
2. Проверка изученного материала;
3. Промежуточная аттестация в форме письменной работы.

**Текущий контроль** включает, кроме еженедельного опроса и проверки знаний по текущему материалу, ведение электронного журнала посещаемости, проверку выполнения компьютерных программ. Подразумевается непрерывное общение по электронной почте (общение по скайпу не целесообразно, т.к. не позволяет осуществлять доскональную проверку заданий).

**Промежуточный контроль** проводится в виде письменной работы, рассчитанной на 20-30 минут.

**Итоговый контроль** проводится в виде письменной работы с обязательным устным собеседованием по результатам предварительной проверки.

**Критерии выставления оценок** «отлично», «хорошо», «удовлетворительно» определяются степенью владения материалом и достигнутым уровнем компетентности в решении задач дискретной математики. В исключительных случаях учитываются успехи на всероссийских олимпиадах и конкурсах по номинации данной дисциплины.

Для обеспечения самостоятельной работы используется разработанный на кафедре пакет заданий и методических указаний, издано учебное пособие с алгоритмами решения базовых заданий по дискретной математике и соответствующими программами на языке С#. Самостоятельная работа студентов складывается из проработки лекционного материала, материала учебника и соответствующих форумов интернет, решения всех заданий из индивидуальных заданий, решения рекомендуемых задач, подготовки к сдаче промежуточных форма контроля.

## **7. Фонд оценочных средств для проведения текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины**

7.1. Перечень компетенций с указанием этапов их формирования в процессе освоения образовательной программы.

Компетенция	Знания, умения, навыки	Процедура освоения
ОК-1	<p>Знать: о принципах разработки моделей параллельных вычислений.</p> <p>Уметь: выполнять постановки задач, связанных с параллельными вычислениями для решения в среде процессорных элементов.</p> <p>Владеть: навыками разработки и анализа алгоритмов, математических моделей, архитектур и структур аппаратно-программных комплексов</p>	Устный опрос, письменный опрос
ПК-1	<p>Знать: способах оценки эффективности использования компьютеров при управлении информацией при решении задач параллельных вычислений, времени выполнения параллельных программ на ЭВМ и оценки вычислительных ресурсов.</p> <p>Уметь: построения параллельных вычислительных алгоритмов, оценки время выполнения параллельных программ и вычислительные ресурсы.</p> <p>Владеть: навыками разработки компонентов информационных</p>	Устный опрос, письменный опрос, представление реферата

	систем и баз данных.	
ПК-2	<p>Знать: гибридные вычислительные системы с использованием технологии CUDA</p> <p>Уметь: оценивать коммуникационную трудоемкость параллельных алгоритмов, управлять группами, виртуальными топологиями в MPI.</p> <p>Владеть: навыками параллельного программирования на основе MPI.</p>	Проработка конспектов лекций и подготовка к контрольным работам. Подготовка и презентация реферата.

7.2. Описание показателей и критериев оценивания компетенций, описание шкал оценивания.

#### ОК-1

Схема оценки уровня формирования компетенции «способность строго доказывать утверждение, сформулировать результат, увидеть следствия полученного результата».

Уровень	Показатели (что обучающийся должен продемонстрировать)	Оценочная шкала		
		Удовлетворительно	Хорошо	Отлично
Пороговый	Знание параллельных вычислительных систем и их классификация. Способы организации параллельной обработки данных.	Для простых задач	Для задач средней трудности	Для всех задач из лекционного курса, относящихся к динамическому программированию

#### ПК-1

Схема оценки уровня формирования компетенции «способностью публично представлять собственные и известные научные результаты»

Уровень	Показатели (что обучающийся должен продемонстрировать)	Оценочная шкала		
		Удовлетворительно	Хорошо	Отлично

Пороговый	Способность применять базовые средства параллельного программирования систем с общей памятью.	Пересказ алгоритмов	Обоснование алгоритмов	Обоснование и умение программировать
-----------	---	---------------------	------------------------	--------------------------------------

ПК-2

Схема оценки уровня формирования компетенции «способность к организации учебной деятельности в конкретной предметной области (математика, физика, информатика)»

Уровень	Показатели (что обучающийся должен продемонстрировать)	Оценочная шкала		
		Удовлетворительно	Хорошо	Отлично
Пороговый	Умение применять распространенные языки программирования для параллельных вычислений.	Указать, в каких задачах параллельного вычисления применяются эти знания	Продемонстрировать на конкретном примере (например, при вычислении самой экономичной системы)	Выполнить исследование методами анализа

### 7.3. Типовые контрольные задания

*(Указываются темы эссе, рефератов, курсовых работ и др. Приводятся примерные тестовые задания, контрольные вопросы и задания для проведения текущего контроля и промежуточной аттестации по итогам освоения дисциплины.)*

#### 7.3.1. Темы рефератов:

1. Основные проблемы использования параллельной обработки данных.
2. Закон Мура о росте производительности последовательных компьютеров.
3. Модели многопроцессорных систем с общей и распределенной памятью. Модель конвейерной системы.
4. Модель параллельных вычислений в виде сети Петри.
5. Основные проблемы параллельных вычислений: синхронизация, взаимоисключение, блокировка (тупики).
6. Топология обменов. Декартовы топологии. Топологии произвольного графа
7. Основные возможности системы DVM. Мобильность и эффективность выполнения программ.
8. Перспективы развития MVC и параллельного программирования.
9. Алгоритмы маршрутизации. Методы передачи данных. Анализ трудоемкости основных операций передачи данных.
10. Эффективность использования вычислительной системы. Способы оценки показателей.

### **7.3.2. Перечень вопросов к промежуточному (по частям) и итоговому контролю**

1. О параллельных вычислительных системах (ВС).
2. Однопроцессорные, многопроцессорные ВС и трудности их использования. Идея конвейерных вычислений.
3. О классификации многопроцессорных ВС. Параллельная форма алгоритма.
4. Примеры распараллеливания.
5. Построение параллельных форм. Направленный граф.
6. Функциональные устройства (ФУ)
7. Определения: простое ФУ, конвейерное ФУ, длина конвейера, стоимость операции, стоимость работы.
8. Загруженность, асимптотическая загруженность.
9. Свойства простых и конвейерных ФУ. Номинальная (пиковая) производительность, реальная производительность.
10. О времени реализации алгоритма. Об ускорении при распараллеливании. Базовая вычислительная система. Понятие о временной развертке.
11. Достоинства и недостатки базовой вычислительной системы.
12. О построении графа вычислительной системы.
13. О реализации разветвлений на параллельной системе.
14. Программирование операций на параллельной системе.
15. Технология программирования DVM.
16. Основные принципы. Распределение массивов.
17. Выравнивание массивов. Параллельное выполнение циклов.
18. Отображение задач на вычислительную систему.
19. Удаленные данные. Редукционные данные.
20. Пересеченные (across) данные.
21. Пример программы. Отладка программы.
22. Заключительные замечания.
23. Программирование на mpC.
24. Базовые и узловые функции.
25. Автоматические сети.
26. Статические сети.
27. Синхронизация работы сети.

### **7.3.3. Вопросы к экзамену**

28. О постановке задачи распараллеливания .
29. Введение. О некоторых вычислительных задачах.
30. Численный эксперимент и его целесообразность.
31. Параллельная форма алгоритма.
32. Примеры распараллеливания.
33. О схеме сдваивания. О вычислении степени на параллельной системе.
34. Некоторые сведения о графах в связи с распараллеливанием.
35. О понятии графа. Ориентированный граф.
36. Топологическая сортировка. Примеры графов параллельных форм.
37. О времени реализации алгоритма. Об ускорении при распараллеливании.
38. Алгоритмы и вычислительные системы.

39. О соотношении графов алгоритма и вычислительной системы. О двух задачах реализации алгоритма.
40. Базовая вычислительная система. Понятие о временной развертке.
41. Достоинства и недостатки базовой вычислительной системы.
42. О построении графа вычислительной системы.
43. О реализации разветвлений на параллельной системе.
44. Программирование операций на параллельной системе.
45. Обзор основных процедур интерфейса MPI.
46. Загруженность системы при использовании интерфейса MPI.
47. Технология программирования DVM.
48. Основные принципы. Распределение массивов.
49. Выравнивание массивов. Параллельное выполнение циклов.
50. Отображение задач на вычислительную систему.
51. Удаленные данные. Редукционные данные.

### **Примерное содержание экзаменационного билета**

Билет № 1.

1. Достоинства и недостатки базовой вычислительной системы.
2. О понятии графа. Ориентированный граф.

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

Общий результат складывается из текущего контроля - 50% и промежуточного контроля - 50%.

Текущий контроль по дисциплине включает:

- Посещение занятий – 20%
- выполнение текущих лабораторных заданий – 40 %
- выполнение домашних (аудиторных) контрольных работ – 40 %.
- Промежуточный контроль по дисциплине включает:
- устный опрос – 100% .

### **8. Перечень основной и дополнительной учебной литературы, рекомендуемых для освоения дисциплины.**

#### **Основная литература:**

1. Гергель В.П. Современные языки и технологии параллельного программирования: учеб. пособие для студ. вузов, обучающихся по направлениям 010400 "Прикладная математика и информатика" и 010300 "Фундаментальная информатика и информационные технологии". М.: Московский университет, 2012. - 407 с.
2. Антонов А.С. Технологии параллельного программирования MPI и OpenMP учеб. пособие для студ. Вузов. МГУ им. М.В. Ломоносова: М. Московский университет: 2012. - 340с
3. Гергель В.П. Теория и практика параллельных вычислений: БИНОМ. Лаборатория знаний, Интернет университет информационных технологий – ИНТУИТ.ру, 2007

4. Немнюгин С., Стесик О. Параллельное программирование для многопроцессорных вычислительных систем. – СПб.: БХВ-Перербург, 2002.
5. Корнеев В.Д. Параллельное программирование в MPI. – Ижевск, 2004.
6. Воеводин В.В., Воеводин Вл.В. Параллельные вычисления. – СПб.:БХВ-Петербург, 2002.
7. Боресков А.В., Харламов А.А. Основы работы с технологией CUDA. – М.: Изд-во «ДМК Пресс», 2010. – 232 с.

#### **Дополнительная литература:**

8. Антонов А.С. Параллельное программирование с использованием технологии OpenMP: Учебное пособие. – М.: Изд-во МГУ, 2009. – 77 с.
9. Старченко А.В., Есаулов А.О. Параллельные вычисления на многопроцессорных вычислительных системах. – Томск: ТГУ, 2002
10. Богачев К.Ю. Основы параллельного программирования. – М.: БИНОМ. Лаборатория знаний, 2003.

#### **9. Перечень рекомендуемых ресурсов информационно-телекоммуникационной сети «Интернет»**

1. Гергель В.П. Теория и практика параллельных вычислений. (<http://marc.bsu.edu.ru/Katalog/AllFields.asp?db=MarcBSU&id=134621>)
2. Немнюгин С.А. Средства программирования для многопроцессорных вычислительных систем. – СПб: СПбГУ, 2012 (<http://marc.bsu.edu.ru/Katalog/AllFields.asp?db=MarcBSU&id=134587>)
3. Foster I. Designed and Building Parallel Programs. – Addison Wesley, 1994. (<http://www.mcs.anl.gov/dvpp.html>)
4. Introduction to parallel Computing (Teaching Course). (<http://www.ece.northwestern.edu/course/358.html>)

#### **10. Методические указания для обучающихся по освоению дисциплины**

- 1) Выполнение заданий по дискретной математике требует (дополнительно к изучению теоретического материала и получению математического решения) реализации алгоритмов с применением языка высокого уровня. Рекомендуемые языки: Delphi, C#.
- 2) Выбор структур для представления исходных данных особенно важен в тех случаях, когда в задании имеются требования к оценке сложности алгоритма.
- 3) Рекомендуется принимать участие в интернет-олимпиадах. Большинство заданий по программированию подразумевает уверенное владение базовыми алгоритмами: полный перебор и элиминация полного перебора, рекурсия и рекуррентные формулы, различные методы поиска в графах (кратчайшие пути, поиск вширь и в глубину), потоковые и комбинаторные методы и др.

**11. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень программного обеспечения и информационных справочных систем**

Пакет видеолекций и видео-презентации.

Электронные учебные пособия (Магомедов А.М.).

Системы компьютерной математики (Mathematica, MathCad, MathLab, Maple), предпочтение отдается Mathematica.

10 прикладных программ, разработанных на кафедре дискретной математики и информатики и зарегистрированных в гос.реестре Роспатента.

**12. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине.**

Лабораторные занятия проводятся в компьютерных классах с современным аппаратным и программным обеспечением. При выполнении лабораторных заданий студенту предоставляется право выбора одного из двух языков программирования из поддерживаемых MS VisualStudio. На сайте кафедры размещаются учебные пособия и презентации к лекции.