

МИНИСТЕРСТВО НАУКИ И ОБРАЗОВАНИЯ РФ
Федеральное государственное бюджетное образовательное
учреждение высшего образования
«ДАГЕСТАНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Объектно-ориентированные языки и системы программирования

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: базовый

Махачкала, 2018

Рабочая программа дисциплины «Объектно-ориентированные языки и системы программирования» составлена в 2018 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. – ст. преподаватель кафедры дискретной математики и информатики.

Рабочая программа дисциплины одобрена:
на заседании кафедры дискретной математики и информатики от 27.04.2018, протокол № 8

Зав. кафедрой *Ab* Магомедов А.М.

и
на заседании Методической комиссии факультета математики и компьютерных наук от
27.06.2018, протокол №6

Председатель *Вен* Бейбалаев В.Д.

Рабочая программа дисциплины согласована с учебно-методическим
управлением «*об*» *об* 2018 г. *Ab*

(подпись)

Аннотация рабочей программы дисциплины

Дисциплина «Объектно-ориентированные языки и системы программирования» входит в базовую часть образовательной программы магистратуры по направлению 01.04.02 – Прикладная математика и информатика и является обязательной дисциплиной.

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

Содержание дисциплины охватывает круг базовых вопросов, связанных с современными знаниями в области объектно-ориентированного программирования.

Дисциплина способствует формированию следующих компетенций выпускника: общепрофессиональных - ОПК-3, ОПК-4, профессиональных - ПК-3.

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

Рабочая программа дисциплины предусматривает проведение следующих видов контроля успеваемости в форме контрольной работы и итоговый контроль в форме экзамена.

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

Семес тр	Учебные занятия						СРС, в том числе экза мен	Форма промежуточной аттестации (зачет, дифференциро ванный зачет, экзамен
	в том числе							
	Контактная работа обучающихся с преподавателем							
	Все го	из них						
		Лекц ии	Лаборатор ные занятия	Практич еские занятия	КСР	консульт ации		
А	108	4	16				88	экзамен

1. Цели изучения дисциплины

Цели изучения дисциплины:

- Формирование представлений об объектно-ориентированном программировании, о методах проектирования объектно-ориентированных программ, об объектно-ориентированных языках программирования и визуальном программировании.
- Изучение современных методов объектно-ориентированного программирования при кодировании программных систем разного уровня сложности.

Задачами изучения дисциплины является:

Изучение объектно-ориентированного программирования, принципов построения классов, критерий проверки правильности построения классов, основных тенденций в области развития технологий объектно-ориентированного программирования.

2. Место дисциплины в структуре ОПОП магистратуры

Дисциплина «Объектно-ориентированные языки и системы программирования» входит в базовую часть образовательной программы магистратуры, по направлению 01.04.02 – Прикладная математика и информатика и изучается в соответствии с графиком учебного процесса во втором семестре

Дисциплина «Объектно-ориентированные языки и системы программирования» входит в базовую часть и является дисциплиной по выбору образовательной программы магистратуры, по направлению 01.04.02 – Прикладная математика и информатика. Изучение предмета производится в течение одного семестра и заканчивается экзаменом.

Успешное овладение дисциплиной предполагает предварительные знания операционных систем, сетей ЭВМ и телекоммуникаций, предусмотренном специальностью «Прикладная математика и информатика», а также навыки программирования на языках высокого уровня. Знания, полученные при изучении данной дисциплины, будут в дальнейшем использоваться в научно-исследовательской работе и при изучении современных компьютерных технологий.

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)
ОПК-3	Способность самостоятельно приобретать с помощью информационных технологий и	Знает: языки программирования; Умеет: разрабатывать ясные и надежные программы

	использовать в практической деятельности новые знания и умения, в том числе, в новых областях знаний, непосредственно не связанных со сферой деятельности, расширять и углублять своё научное мировоззрение	Владеет: навыками работы в современных средах программирования
ОПК-4	Способность использовать и применять углубленные знания в области прикладной математики и информатики	Знает: основы объектно-ориентированных языков, алгоритмические и программные решения в области системного и прикладного программирования; Умеет: использовать типы, операции, управляющие структуры и визуальные компоненты, классы и пакеты в разработке приложений, информационных и имитационных моделей, созданию информационных ресурсов глобальных сетей; Владеет: навыками разработки программ с применением методов объектно- ориентированного, визуального и событийно-управляемого программирования, навыками работы в различных программных средах.
ПК-3	Способность разрабатывать и применять математические методы, системное и прикладное программное обеспечение для решения задач научной и проектно-технологической деятельности;	Знает: подходы использования современных методов для решения научных и практических задач; Умеет: использовать современные методы прикладной математики для решения научно-исследовательских и прикладных задач; Владеет: определением общих форм, закономерностей, инструментальных средств для анализа архитектуры параллельных систем.

4. Объем, структура и содержание дисциплины

4. Объем, структура и содержание дисциплины

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

4.2. Структура дисциплины

№ п/п	Разделы и темы дисциплин	Семестр	Неделя семестра	Виды учебной работы, включая самостоятельную работу студентов и трудоемкость (в часах)					Формы текущего контроля успеваемости (по неделям семестра) Форма промежуточной аттестации (по семестрам)
				Всего	Лекции	Лаб.занят.	Самост. раб.	Контроль за сам.работой	
Модуль 1. Основы объектно-ориентированных языков. Классы и объекты									
1	Основы объектно-ориентированного программирования. Классы и объекты.	2	1	8	2		6		Устный опрос
2	Работа со строками	2	2-3	14		4	10		
3	Обработка исключений	2	4-5	14		4	10		Прием лабораторных работ
Итого по модулю 1				36	2	8	26		
Модуль 2. Введение в язык Java									
4	Виртуальная машина Java.	2	6	8	2		6		
5	Алгоритмические средства языка Java.	2	7-8	14		4	10		
6	Средства объектно-ориентированного программирования	2	8-9	14		4	10		коллоквиум
Итого по модулю 2				36	2	8	26		
Модуль 3. Подготовка к экзамену									
7	Подготовка и сдача экзамена	2						36	
Итого по модулю 3				36				36	экзамен
Итого за семестр				108	4	16	52	36	

4.3. Содержание дисциплины, структурированное по темам (разделам)

4.3.1 Содержание лекционных занятий по дисциплине

Модуль 1. Основы объектно-ориентированных языков. Классы и объекты.

Тема 1. Основы объектно-ориентированного программирования. Классы и объекты. Основные понятия. Наследование. Классы, интерфейсы, абстрактные классы. Реализация классов. Использование полиморфизма. Создание объекта, оператор new. Правила доступа к данным и методам объекта. Переопределение методов. Метод конструктор.

Модуль 2. Введение в язык Java.

Тема 4. Виртуальная машина Java.

История и предпосылки появления Java. Понятие виртуальной машины. Среда исполнения и байт-код. Взаимодействие виртуальной машины с операционной системой. Пространства классов. Структура приложений на Java. Загрузка классов и инициализация объектов. Сферы применения Java в современном информационном мире. Версии Java машины и их эволюция. Средства ООП, непосредственно поддерживаемые в Java. Простейшее приложение на Java.

4.3.2 Содержание лабораторно-практических занятий по дисциплине

Лабораторные занятия предусмотрены по всем темам модулей и их содержание совпадает с содержанием тем модулей.

Модуль 1. Основы объектно-ориентированных языков. Классы и объекты.

Тема 1. Основы объектно-ориентированного программирования. Классы и объекты. Основные понятия. Наследование. Классы, интерфейсы, абстрактные классы. Реализация классов. Использование полиморфизма. Создание объекта, оператор new. Правила доступа к данным и методам объекта. Переопределение методов. Метод конструктор.

Тема 2. Работа со строками.

Классы String, StringBuffer. Доступ к символам. Равенство строк. Сравнение. Упорядочение. Методы работы со строками. Методы indexOf, lastIndexOf.

Тема 3. Обработка исключений на Java.

Типы исключений. Вложенные операторы try. Блок finally. Операторы throw, catch. Подклассы Exception.

Модуль 2. Введение в язык Java

Тема 4. Виртуальная машина Java.

История и предпосылки появления Java. Понятие виртуальной машины. Среда исполнения и байт-код. Взаимодействие виртуальной машины с операционной системой. Пространства классов. Структура приложений на Java. Загрузка классов и инициализация объектов. Сферы применения Java в современном информационном мире. Версии Java машины и их эволюция. Средства ООП, непосредственно поддерживаемые в Java. Простейшее приложение на Java.

Тема 5. Алгоритмические средства языка Java.

Строгая типизация Java. Базовые типы языка. Строки и литералы. Преобразования типов в выражениях. Особенности инициализации массивов. Операторы управления памятью. Краткий обзор операторов. Использование break.

Тема 6. Средства объектно-ориентированного программирования языка Java.

Представление объектов и классов. Структура объявления класса. Доступ к членам класса.

. Спецификаторы доступа для обеспечения инкапсуляции. Знакомство с final. Средства управления жизнью объекта. Конструкторы и метод finalize(). Работа с массивами объектов. Статические поля и методы классов. Классы – утилиты. Блок статической инициализации.

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

6. Учебно-методическое обеспечение самостоятельной работы студентов

Самостоятельная работа студентов складывается из:

- проработки лекционного материала (настоятельно рекомендуется самостоятельное практическое решение всех разобранных на лекциях упражнений);
- изучения рекомендованной литературы и материалов соответствующих форумов интернет;
- подготовки к сдаче промежуточных форм контроля (контрольных работ)
- подготовка к сдаче экзамена.

№	Вид самостоятельной работы	Вид контроля	Учебно-методическое обеспечение
1.	Проработка лекционного материала	Контрольный фронтальный опрос	См. разделы 7.3, 8, 9 данного документа
2.	Изучение рекомендованной литературы и материалов соответствующих форумов интернет	Контрольный фронтальный опрос, прием и представление рефератов.	См. разделы 7.3, 8, 9 данного документа
3.	Подготовка к отчетам по практическим работам.	Проверка выполнения работ, опрос по теме работы.	См. разделы 7.3, 8, 9 данного документа
4.	Подготовка к сдаче промежуточных	Контрольные работы по каждому модулю.	См. разделы 7.3, 8, 9 данного документа

форм 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.

Код и наименование компетенции из ФГОС ВО	Код и наименование компетенции индикатора достижения компетенций	Процедура освоения
ОПК-3	Знает: языки программирования; Умеет: разрабатывать ясные и надежные программы Владеет: навыками работы в современных средах программирования	Устный опрос,
ОПК-4	Знает: основы объектно-ориентированных языков, алгоритмические и программные решения в области системного и прикладного программирования; Умеет: использовать типы, операции, управляющие структуры и визуальные компоненты, классы и пакеты в разработке приложений, информационных и имитационных моделей, созданию информационных ресурсов глобальных сетей; Владеет: навыками разработки программ с применением методов объектно- ориентированного, визуального и событийно-управляемого программирования, навыками работы в различных программных средах.	Письменный опрос
ПК-3	Знает: подходы использования современных методов для решения научных и практических задач; Умеет: использовать современные методы прикладной математики для решения научно-исследовательских и прикладных задач; Владеет: определением общих форм, закономерностей, инструментальных средств для анализа архитектуры параллельных систем.	Устный опрос, письменный опрос

7.2. Типовые контрольные задания

7.2.1. Темы рефератов:

1. История развития языков высокого уровня.
2. Система объектно-ориентированного программирования .
3. Этапы разработки проектов на языке.
4. Структура проекта.
5. Управление проектом.
6. Структура программы
7. Арифметические выражения, логические выражения.
8. Массивы. Разработка программ с использованием массивов, алгоритмы сортировки.
9. Двумерные массивы.
10. Строки. Строковая константа, ввод строк с клавиатуры, некоторые стандартные функции для строк. Массивы строк, массив указателей, ввод массива строк из файла. Некоторые задачи со строками.
11. Проектирование главного меню.
12. Проектирование локального меню.
13. Отображение картинок.
14. Абстрактные типы данных: инкапсуляция, классы и объекты, параметризация, спецификация, реализация.
15. Круг задач, решаемых с помощью рекурсии.
16. Назначение диалоговых окон и управляющих элементов.
17. Группировка управляющих элементов.
18. Создание файла ресурсов.
19. Алгоритмы, предназначенные для работы со списками, стеком, очередью, деревьями

7.3.2. Варианты заданий коллоквиума

Вариант 1

Задание 1. Дан фрагмент программы

```
_____ s =120; System.out.println(s-20);
```

Какой тип данных нужно подставить вместо прочерка, чтобы программа работала корректно, переменная *s* занимала наименьший объём памяти, а в результате исполнения на экран вывелось натуральное число?

Задание 2. Что выведется на экран в результате работы следующего фрагмента программы?

```
float w =2.5F; int z =10/(int) w;  
System.out.println(z+w);
```

Задание 3. Что выведется на экран в результате работы следующего фрагмента программы?

```
float y =2F; System.out.println((int)y+"2"+y*1 .5);
```

Вариант 1

Задание 1. Дан фрагмент программы

_____ s =3; System.out.println(10/s);

Какой тип данных нужно подставить вместо прочерка, чтобы программа работала корректно, переменная s занимала наименьший объём памяти, а в результате исполнения на экран вывелось вещественное число?

Задание 2. Что выведется на экран в результате работы следующего фрагмента программы?

```
short w =9; double z = w *1.5; System.out.println(w-(int)z);
```

Задание 3. Что выведется на экран в результате работы следующего фрагмента программы? double y =3.5; System.out.println("7"+y*3);

Вариант 3

Задание 1. Дан фрагмент программы:

```
_____ s =990; System.out.println(s*2);
```

Какой тип данных нужно подставить вместо прочерка, чтобы программа работала корректно, переменная s занимала наименьший объём памяти, а в результате исполнения на экран вывелось натуральное число?

Задание 2. Что выведется на экран в результате работы следующего фрагмента программы?

```
int w =10; double z =4; System.out.println(1+w/z);
```

Задание 3. Что выведется на экран в результате работы следующего фрагмента программы?

```
double y =3;  
System.out.println((int)y+6+"y"+6 );
```

Вариант 4

Задание 1. Дан фрагмент программы:

```
_____ s =3; System.out.println(7.5/s);
```

Какой тип данных нужно подставить вместо прочерка, чтобы программа работала корректно, переменная s занимала наименьший объём памяти, а в результате исполнения на экран вывелось вещественное число?

Задание 2. Что выведется на экран в результате работы следующего фрагмента программы?

```
double w =1.75; int z =(int)(2*w); System.out.println(w*2+z);
```

Задание 3. Что выведется на экран в результате работы следующего фрагмента программы?

```
byte y =14;  
System.out.println(2+y+"2"+y);
```

Вариант 5

Задание 1. Дан исходный код программы:

```
classExample{
    publicstaticvoid
    main(String[]args){    _____ b =5.0;
        b = b*2;
        if(b >10|| b <-10){
            System.out.println("12"+ 0+4);
        }else{
            System.out.println(0+4+" 12");    }    } }
```

Какой тип данных из набора boolean, int, double нужно подставить вместо прочерка, чтобы программа компилировалась корректно?

Задание 2.

Что выведется на экран в результате работы представленной выше программы?

Задание 3.

В программе имеется объявленная переменная s типа int. Напишите выражение, в результате которого переменной s будет присвоено случайное значение из отрезка [-6;2].

Вариант 6

Задание 1. Дан исходный код программы:

```
classExample{    publicstaticvoid main(String[]args){
    _____ a =-3.0;    a =2+ a;
    if(3>2||3>2&&false){    System.out.println("4"+a);
        }else{    System.out.println(a+"4");    }    } }
```

Какой тип данных из набора boolean, int, double нужно подставить вместо прочерка, чтобы программа компилировалась корректно?

Задание 2. **Что выведется на экран в результате работы представленной выше программы?**

Задание 3. **В программе имеется объявленная переменная s типа int. Напишите выражение, в результате которого переменной s будет присвоено случайное значение из отрезка [-6;8].**

Вариант 7

Задание 1. Дан исходный код программы:

```
classExample{    publicstaticvoid main(String[]args){    _____ a =0==8%2;
        if(a ||(a &&false)){
            System.out.println((1+1.5)+"2");    }else{
            System.out.println(3+"a"+"2"+a);    }    } }
```

Какой тип данных из набора boolean, int, double нужно подставить вместо прочерка, чтобы программа компилировалась корректно?

Задание 2. Что выведется на экран в результате работы представленной выше программы?

Задание 3. В программе имеется объявленная переменная `s` типа `int`. Напишите выражение, в результате которого переменной `s` будет присвоено случайное значение из отрезка `[-9;0]`.

Вариант 8

Задание 1. Дан исходный код программы:

```
classExample{ publicstaticvoid main(String[]args){  
    _____ a =5>3;   if(a ||(5+3)*2<16){  
        System.out.println(4+"12"+2);   }else{  
        System.out.println(2+12*2);   } } }
```

Какой тип данных из набора `boolean`, `int`, `double` нужно подставить вместо прочерка чтобы программа компилировалась корректно?

Задание 2. Что выведется на экран в результате работы представленной выше программы?

Задание 3. В программе имеется объявленная переменная `s` типа `int`. Напишите выражение, в результате которого переменной `s` будет присвоено случайное значение из отрезка `[4;12]`.

7.2.2. Перечень вопросов к промежуточному (по частям) и итоговому контролю

1. Выполнение Java-программы. Средства в составе пакета JDK. Кроссплатформенность.
2. Понятие байт кода. Типы Java-программ. Различия между приложениями и апплетами.
3. Идентификаторы. Комментарий. Управляющие символы.
4. Типы данных. Примитивные типы.
5. Типы данных. Ссылочные типы. Массивы.
6. Типы данных. Ссылочные типы. Классы. Интерфейсы.
7. Логические операции. Преобразование типа. Переменные.
8. Операторы. Логические, присваивания. Приоритеты.
9. Операторы. Условные, выбора.
10. Операторы цикла. Три инструкции перехода.
11. Свойства ООП.
12. Классы. Передача параметров по значению и по ссылке. Ссылка `this`.
13. Модификаторы доступа. Типы методов: перегруженные, переопределенные. Ключевое слово `super`.
14. Пакеты. Включение пакета, класса пакета. Пакет `AbstractWindowToolkit`, `GUI`.
15. Контейнеры: фрейм, панель.
16. Компоненты GUI: `Label`, `Text Field`, `TextArea`.
17. Компоненты GUI: `Button`, `Checkbox`, радиокнопки, списки.
18. Типы макетов.
19. Обработка событий от кнопки, от мыши. Реализация интерфейсов.
20. Апплеты - создание, жизненный цикл, вывод апплета.
21. Апплеты – вывод изображений, передача параметров.

22. Классы Graphics, Colors, Fonts.
23. Обработка исключений. Классы Error и Exception.
24. Ключевые слова: try, catch, throw, throws, finally.
25. Потoki, 2 способа создания потоков.
26. Состояния потоков, два типа потоков.
27. Пакеты, управление доступом

7.3. 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 - 30 баллов,
- выполнение лабораторных заданий – 20 баллов,
- выполнение самостоятельных работ - 50 баллов.

Промежуточный контроль по дисциплине включает:

- устный опрос - 30 баллов,
- письменная контрольная работа - 50 баллов,
- тестирование - 20 баллов.

Текущий контроль по дисциплине включает:

- Посещение занятий – 25%
- 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.

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

а) основная литература:

1. Санников Е.В. Курс практического программирования в Delphi. Объектно-ориентированное программирование [Электронный ресурс]/ Санников Е.В.— Электрон. текстовые данные. — М: СОЮЗ ПРЕСС, 2013. — 188 с.— Режим доступа: <http://www.iprbookshop.ru/26921.html>. — ЭБС «IPRbooks»
2. Мейер Б. Объектно-ориентированное программирование и программная инженерия [Электронный ресурс]/ Мейер Б.— Электрон. текстовые данные. — М: Интернет-Университет Информационных

Технологий (ИНГУИП), Ай Ги Эр Медиа, 2019.— 285 с.— Режим доступа: <http://www.iprbookshop.ru/79706.html>.— ЭБС «IPRbooks»

3. Новиков П.В. Объектно-ориентированное программирование [Электронный ресурс]: учебно-методическое пособие к лабораторным работам/ Новиков П.В.— Электрон. текстовые данные.— Саратов: Вузовское образование, 2017.— 124 с.— Режим доступа: <http://www.iprbookshop.ru/64650.html>.— ЭБС «IPRbooks»

5. Патрик Ноутон, Герберт Шилдт. Java™2. Наиболее полное руководство. Санкт-Петербург. “БХВ-Петербург” 2013, 1050 с.

6. Казанский АА Объектно-ориентированное программирование на языке Microsoft Visual C# в среде разработки Microsoft Visual Studio 2008 и .NET Framework. 4.3 [Электронный ресурс]: учебное пособие и практикум/ Казанский АА— Электрон. текстовые данные.— М: Московский государственный строительный университет, ЭБС АСВ, 2011.— 180 с.— Режим доступа: <http://www.iprbookshop.ru/19258.html>.— ЭБС «IPRbooks»

7. Вебер Д. Технология Java в подлиннике. С.Пб: BHV-Санкт-Петербург, 2012, 1104 с.

8. Мейнджер Д. JAVA: Основы программирования. С.Пб: BHV-Санкт-Петербург, 2012, 320 с.

б) дополнительная литература:

1. Гномарева ЛА Лабораторные работы по курсу «Информатика и программирование» для подготовки специалистов в области управления. Часть 1 [Электронный ресурс]: практикум/ Гномарева ЛА— Электрон. текстовые данные. — М: Московский городской педагогический университет, 2012.— 120 с.— Режим доступа: <http://www.iprbookshop.ru/26514.html>.— ЭБС «IPRbooks»

2. Николаев Е.И. Объектно-ориентированное программирование. Часть 2 [Электронный ресурс]: лабораторный практикум/ Николаев ЕИ— Электрон. текстовые данные. — Ставрополь: Северо-Кавказский федеральный университет, 2015. — 156 с.— Режим доступа: <http://www.iprbookshop.ru/63218.html>. — ЭБС «IPRbooks»

3. Николаев ЕИ Объектно-ориентированное программирование. Часть 1 [Электронный ресурс]: лабораторный практикум/ Николаев ЕИ— Электрон. текстовые данные. — Ставрополь: Северо-Кавказский федеральный университет, 2015. — 183 с.— Режим доступа: <http://www.iprbookshop.ru/62966.html>. — ЭБС «IPRbooks»

4. Практикум по дисциплине Логическое и функциональное программирование [Электронный ресурс]/ — Электрон. текстовые данные. — М: Московский технический университет связи и информатики, 2016. — 16 с.— Режим доступа: <http://www.iprbookshop.ru/61489.html>. — ЭБС «IPRbooks»

5. Комлев НЮ Объектно-ориентированное программирование. Хорошая книга для Хороших Людей [Электронный ресурс]/ Комлев НЮ— Электрон. текстовые данные. — М: СОЛОН-ПРЕСС, 2014.— 298 с.— Режим доступа: <http://www.iprbookshop.ru/26923.html>.— ЭБС «IPRbooks»

6. Фултон Хэл Программирование на языке Ruby [Электронный ресурс]/ Фултон Хэл— Электрон. текстовые данные. — Саратов: Профобразование, 2017. — 685 с.— Режим доступа: <http://www.iprbookshop.ru/63952.html>. — ЭБС «IPRbooks»

7. Практикум по дисциплине Логическое и функциональное программирование [Электронный ресурс]/ — Электрон. текстовые данные. — М: Московский технический университет связи и информатики, 2016. — 16 с.— Режим доступа: <http://www.iprbookshop.ru/61489.html>.— ЭБС «IPRbooks»

9. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины.

1. <http://www.intuit.ru/>

2. <https://ru.wikipedia.org/wiki/Java>
3. <http://citforum.ru/>
4. <http://www.compdoc.ru/>
5. <http://www.emanual.ru/>

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#, Java.
- 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 видео лекций 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 Visual Studio. На сайте кафедры размещаются учебные пособия и презентации к лекции.

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

Основная литература:

1. Биллиг В.А. Параллельные вычисления и многопоточное программирование [Электронный ресурс] / В.А. Биллиг. — 2-е изд. — Электрон. текстовые данные. — М.: Интернет Университет Информационных Технологий (ИНТУИТ), 2016. — 310 с. — 2227-8397. — Режим доступа: <http://www.iprbookshop.ru/73705.html> (15.04.2018) 2. 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. 3. Антонов А.С. Технологии параллельного программирования MPI и OpenMP учеб. пособие для студ. Вузов. МГУ им. М.В. Ломоносова: М. Московский университет: 2012. – 340 с. 4. Боресков А.В. и др. Параллельные вычисления на GPU. Архитектура и программная модель CUDA. Учеб. пособие. 2-е изд. М.: Издательство Московского университета, 2015. – 336с

Дополнительная литература:

1. Барский А.Б. Параллельные информационные технологии [Электронный ресурс] : учебное пособие / А.Б. Барский. — Электрон. текстовые данные. — Москва, Саратов: Интернет-Университет Информационных Технологий (ИНТУИТ), Вузовское образование, 2017. — 503 с. — 978-5-4487-0087-3. — Режим доступа: <http://www.iprbookshop.ru/67379.html> 2. Левин М.П. Параллельное программирование с использованием OpenMP [Электронный ресурс] / М.П. Левин. — Электрон. текстовые данные. — М.: Интернет Университет Информационных Технологий (ИНТУИТ), 2016. — 133 с. — 978-5-94774-857-4. — Режим доступа: <http://www.iprbookshop.ru/52216.html> 3. Косяков М.С. Введение в распределенные вычисления. – СПб: НИУ ИТМО, 2014. – 155 с.

9. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», полезных для освоения дисциплины

1. Поток выполнения [Электронный ресурс] // URL = <https://ru.wikipedia.org/wiki/> / (дата обращения: 21.03.2018) 2. Операционная платформа. Определения и классификация [Электронный ресурс] // URL = <http://gigabaza.ru/doc/33138.html> / время доступа: 21.04.2018. 3. Процессы и потоки in-depth. Обзор различных потоковых моделей [Электронный ресурс] // URL = <http://habrahabr.ru/post/40227/> время доступа: 22.03.2018. 4. Автоматическое распараллеливание программ для распределенных систем. Статическое построение расширенного графа управления [Электронный ресурс] // URL = <http://xreferat.com/33/4673-1/> время доступа: 20.02.2018. 5. <http://www.intuit.ru> [Электронный ресурс] 6. <http://www.parallel.ru> Материалы на сайте Лаборатории параллельных информационных технологий МГУ.

10. Методические указания для обучающихся по освоению дисциплины.

1. При решении лабораторных заданий программистский подход непременно должен присутствовать (без него решение не будет полноценным), однако, он не должен заслонять сугубо математические (доказательство и др.) и алгоритмические (построение, оптимизация, верификация и др.) аспекты.
2. Необходимо обратить внимание на распознавание ситуаций, когда распараллеливание: а) допустимо, б) целесообразно, в) необходимо. Нельзя игнорировать «накладные расходы» ресурсов, выделяемых собственно распараллеливанию вычислений.
3. Важно различать архитектурные и теоретические проблемы распараллеливания.
4. При решении проблемы автоматического распараллеливания особое внимание следует уделить созданию внутреннего представления программы, органично соответствующего проблематике решаемой задачи.

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

При осуществлении образовательного процесса студентами и профессорско-преподавательским составом используются следующее программное обеспечение:

Microsoft Visual Studio Express, Microsoft Windows, Ubuntu Linux, Skype.

Также студентам

предоставляется доступ к российским и международным электронным библиотекам через

компьютеры университета. На кафедре имеются также прикладные программы,

разработанные на кафедре дискретной математики и информатики и зарегистрированные

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. На сайте кафедры размещаются учебные пособия и презентации к лекциям.