

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«ДАГЕСТАНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»  
*ФАКУЛЬТЕТ МАТЕМАТИКИ И КОМПЬЮТЕРНЫХ НАУК*

**РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ**  
**Технологии программирования и работа на ЭВМ**

Кафедра дискретной математики и информатики  
факультета математики и компьютерных наук

Образовательная программа  
**02.03.01 – Математика и компьютерные науки**

Направленность (профиль) программы  
**Математический анализ и приложения**

Уровень высшего образования  
**бакалавриат**

Форма обучения  
**очная**

Статус дисциплины: входит в часть ОПОП, формируемую  
участниками образовательных отношений

Махачкала, 2019 г.

Рабочая программа дисциплины «Технологии программирования и работа на ЭВМ» составлена в 2019 году в соответствии с требованиями ФГОС ВО по направлению подготовки 02.03.01 – Математика и компьютерные науки (уровень бакалавриата)

Приказ Минобрнауки России от «23» августа 2017г. №807.


Разработчик(и): кафедра дискретной математики и информатики, ст. преподаватель Мирзабеков Я.М.


Рабочая программа дисциплины одобрена:

на заседании кафедры дискретной математики и информатики от «26» июня 2019г., протокол №10

Зав. кафедрой  Магомедов А.М.  
(подпись)

на заседании Методического совета факультета математики и компьютерных наук от «27» июня 2019г., протокол №6.

Председатель  Бейбалаев В.Д.  
(подпись)

Рабочая программа дисциплины согласована с учебно-методическим управлением « 30 » 08 2019г.   
(подпись)

**Аннотация рабочей программы дисциплины** Дисциплина “Технологии программирования и работа на ЭВМ” входит в часть, формируемую участниками образовательных отношений программы бакалавриата по направлению 02.03.01 – Математика и компьютерные науки.

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

Содержание дисциплины охватывает круг вопросов, связанных с формированием алгоритмического мышления у студентов, объектно-ориентированным программированием, созданием консольных и графических приложений.

Дисциплина нацелена на формирование следующих компетенций выпускника: общепрофессиональных – ОПК-4, профессиональных – ПК-1, ПК-7.

Преподавание дисциплины предусматривает проведение следующих видов учебных занятий: лекции, практические занятия, лабораторные занятия и самостоятельную работу студентов.

Рабочая программа дисциплины предусматривает проведение следующих видов контроля успеваемости: 1 семестр – зачет, 2 семестр – экзамен, 3 семестр – зачет, 4 семестр – экзамен.

Объем дисциплины – 16 зачетных единиц, в том числе в академических часах по видам учебных занятий:

Семестр	Учебные занятия							Форма промежуточной аттестации (зачет, дифференцированный зачет, экзамен)	
	в том числе:								
	всего	Контактная работа обучающихся с преподавателем							СРС, в том числе экзамен
		всего	из них						
		Лекции	Лабораторные занятия	Практические занятия	КСР	консультации			
1	144	80	32	32	16			64	Зачет
2	144	60	30	30				84	Экзамен
3	108	56	28	28				52	Зачет
4	180	76	30	30	16			104	Экзамен
<b>Итого</b>	<b>576</b>	<b>272</b>	<b>120</b>	<b>120</b>	<b>32</b>			<b>304</b>	

## 1. Цели освоения дисциплины

Целями освоения дисциплины Технологии программирования и работа на ЭВМ являются овладение знаниями в области технологии программирования; подготовка к осознанному использованию как языков программирования, так и методов программирования. Формирование у студентов научного, творческого подхода к освоению технологий, методов и средств производства программного обеспечения. Получение необходимых знаний, умений и навыков в области применения современной вычислительной техники для решения практических задач обработки данных, математического моделирования, информатики, получение высшего профессионального (на уровне бакалавра) образования, позволяющего выпускнику успешно работать в избранной сфере деятельности с применением современных компьютерных технологий.

Основные задачи дисциплины: овладение методами структурного и объектно-ориентированного программирования; закрепление навыков алгоритмизации и программирования, полученных в предыдущих семестрах; создание практической базы для написания качественной выпускной квалификационной работы.

## 2. Место дисциплины в структуре ОПОП бакалавриата

Дисциплина входит в вариативную по выбору часть образовательной программы бакалавриата по направлению 02.03.01 – Математика и компьютерные науки и изучается в соответствии с графиком учебного процесса в 1, 2, 3, 4 семестре.

Для изучения и освоения дисциплины нужны первоначальные знания алгоритмизации из школьного курса. Знания, умения и навыки, полученные при изучении данной дисциплины, используются, закрепляются и развиваются при проведении учебной практики, курсов численных методов, вычислительного практикума, выполнении курсовых работ и выпускной квалификационной работы.

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

Код и наименование компетенции из ФГОС ВО	Код и наименование индикатора достижения компетенций	Планируемые результаты обучения
ОПК-4. Способен находить, анализировать, реализовывать программно и использовать на практике математические алгоритмы, в том числе с применением	ОПК-4.1. Знает базовые основы современного математического аппарата, связанного с проектированием, разработкой, реализацией и оценкой качества программных продуктов и программных комплексов в различных областях человеческой деятельности.	<i>Знает:</i> базовые основы современного математического аппарата, связанного с проектированием программных продуктов и программных комплексов. <i>Умеет:</i> использовать математический аппарат в профессиональной деятельности. <i>Владеет:</i> практическим опытом применения современного математического аппарата, связанного с проектированием и разработкой

<p>современных вычислительных систем</p>	<p>ОПК-4.2. Умеет использовать этот математический аппарат в профессиональной деятельности. ОПК-4.3. Имеет практический опыт применения современного математического аппарата, связанного с проектированием, разработкой, реализацией и оценкой качества программных продуктов и программных комплексов в различных областях человеческой деятельности.</p>	<p>программных продуктов и программных комплексов.</p>
<p>ПК-1. Способен демонстрировать базовые знания математических и естественных наук, основ программирования и информационных технологий.</p>	<p>ПК-1.1. Обладает базовыми знаниями, полученными в области математических и (или) естественных наук, программирования и информационных технологий. ПК-1.2. Умеет находить, формулировать и решать стандартные задачи в собственной научно-исследовательской деятельности в математике и информатике. ПК-1.3. Имеет практический опыт научно-исследовательской деятельности в математике и информатике.</p>	<p><i>Знает:</i> базовые понятия в области математических наук и программирования. <i>Умеет:</i> находить, формулировать и решать стандартные задачи в собственной научно-исследовательской деятельности. <i>Владеет:</i> практическим опытом научно-исследовательской деятельности в математике и информатике.</p>

<p>ПК-7. Способен создавать и исследовать новые атематические модели в естественных науках, промышленности и бизнесе, с учетом возможностей современных информационных технологий и программирования и компьютерной техники</p>	<p>ПК-7.1. Знает основные методы проектирования и производства программного продукта, принципы построения, структуры и приемы работы с инструментальными средствами, поддерживающими создание программных продуктов и программных комплексов, их сопровождения, администрирования и развития (эволюции).</p> <p>ПК-7.2. Умеет использовать методы проектирования и производства программного продукта, принципы построения, структуры и приемы работы с инструментальными средствами, поддерживающими создание программного продукта.</p> <p>ПК-7.3. Имеет практический опыт применения указанных выше методов и технологий.</p>	<p><i>Знает:</i> основные методы проектирования и производства программного продукта.</p> <p><i>Умеет:</i> использовать методы проектирования и производства программного продукта.</p> <p><i>Владеет:</i> практическим опытом применения указанных выше методов и технологий.</p>
---	--	--

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

**4.1. Объем дисциплины** составляет 16 зачетных единиц, 576 академических часов: 120 ч. лекций, 120 ч. лаб., 32 ч. практ., 232 – СРС, 72 – контроль в виде экзаменов во 2 и 4 семестрах.

#### 4.2. Структура дисциплины

Структура и содержание дисциплины «Технологии программирования и работа на ЭВМ»

№ п/п	Разделы и темы дисциплины	Семестр	Неделя семестра	Виды учебной работы, включая самостоятельную работу студентов и трудоемкость (в часах)				Формы текущего контроля успеваемости ( <i>по неделям семестра</i> ) Форма промежуточной аттестации ( <i>по семестрам</i> )
				Лекции	Практические занятия	Лабораторные занятия	Самостоятельная работа	
<b>Модуль 1. Основы программирования.</b>								
1	Алгоритмизация задачи. Понятие алгоритма. Свойства алгоритма	1	1	2		2	6	
2	Базовые конструкции языка Дельфи, простые типы данных, структура программы. Операторы языка.	1	2-6	6	2	6	12	
	<i>Итого по модулю 1:</i>	36	1-6	8	2	8	18	Устный опрос
<b>Модуль 2. Структурированные типы данных.</b>								
3	Структурированные типы данных. Работа с массивами. Алгоритмы сортировки.	1	7,8	4	2	4	16	
4	Строковый тип.	1	9	2		2	6	

	<i>Итого по модулю 2:</i>	36	7-9	6	2	6	22	Контрольная работа №1
<b>Модуль 3. Работа с памятью.</b>								
5	Множества. Записи.	1	10	2	2	2	2	
6	Подпрограммы	1	11	2	2	2	4	
7	Работа с файлами	1	12-14	6	2	6	4	
	<i>Итого по модулю</i>	36	10-14	10	6	10	10	Устный опрос
<b>Модуль 4. Работа с памятью.</b>								
8	Процедуры и функции модуля Graph	1	15, 16	4	2	4	4	
9	Динамическая память и указатели	1	17	2	2	2	4	
10	Динамические структуры данных. Списки	1	18	2	2	2	6	
	<i>Итого по модулю 4:</i>	36	15-18	8	6	8	14	Контрольная работа №2
	<b>ИТОГО (1 сем.):</b>	<b>144</b>		<b>32</b>	<b>16</b>	<b>32</b>	64	<b>зачет</b>
<b>Модуль 5. Основы объектно-ориентированного программирования.</b>								
11	Объектно-ориентированный подход в программировании. Принципы ООП. Классы. Основные понятия.	2	1-2	2		2	4	
12	Типы данных. Динамические массивы. Создание консольных приложений в Delphi.	2	3	2		2		
13	Класс TApplication. Сообщения Windows. События от мыши и клавиатуры.	2	4-5	4		4	2	



14	Класс TStringList. Списки. Классификация компонентов. Массив Components.	2	6	2		4		
15	Формы. Компоненты Standard.	2	7-8	2		2	4	
	<i>Итого по модулю</i>	36	1-8	12		14	10	Контрольная работа №3
<b>Модуль 6. Работа с графикой.</b>								
16	Класс TCanvas, TGraphic, TPicture.	2	9	2		2	8	
17	Анимация в Delphi	2	10	2		2	8	
18	Работа с файлами	2	11	2		2	8	
	<i>Итого по модулю 6:</i>		9-11	6		6	24	Устный опрос
<b>Модуль 7. Работа с динамическими структурами.</b>								
19	Динамическое создание компонентов. Создание собственных компонентов.	2	12- 13	4		4	3	
20	Динамически подключаемые библиотеки.	2	14	2		2	3	
21	Интерфейсы.	2	15	2		2	4	
22	Процессы и потоки	2	16- 17	4		2	4	
	<i>Итого по модулю 7:</i>	36	12- 17	12		10	14	Контрольная работа №4
<b>Модуль 8. Подготовка к экзамену.</b>								
	<b>Подготовка и сдача экзамена</b>	2					36	экзамен
	<b>ИТОГО (2 сем.):</b>	<b>144</b>		<b>30</b>		<b>30</b>	<b>48+36</b>	<b>Экзамен</b>
<b>Модуль 9. Программирование на C#.</b>								

23	Основы .Net Framework. Введение в С#	3	1	2		2	4	
24	Типы данных С#.	3	2	2		2	4	
25	Базовые конструкции языка.	3	3	2		2	6	
26	Строки и массивы	3	4	2		2	6	
	<i>Итого по модулю 8:</i>	36	1-4	8		8	20	Устный опрос
<b>Модуль 10. Работа с классами на С#.</b>								
27	Классы и сборка мусора	3	5	2		2	4	
28	Обработка исключений в С#. Disposable.	3	6	2		2	4	
29	Наследование в С#.	3	7	2		2	6	
30	Интерфейсы	3	8	2		2	6	
	<i>Итого по модулю 9:</i>	36	5-8	8		8	20	Контрольная работа №5
<b>Модуль 11. Обработка событий и работа с графикой на С#.</b>								
31	Классы-коллекции	3	9	2		2	4	
32	Обобщенные классы	3	10	2		2	4	
33	Делегаты. События.	3	11	2		2	4	
34	Сериализация.	3	12	2		2	4	
35	Работа с DLL	3	13	2		2	8	
36	Обработка графики в С#	3	14	2		2	8	
	<i>Итого по модулю 11:</i>	36	9-14	12		12	32	Контрольная работа №6
	<b>ИТОГО (3 сем.):</b>	<b>108</b>		<b>28</b>		<b>28</b>	<b>52</b>	<b>зачет</b>
<b>Модуль 12. Технологии Java-программирования.</b>								

37	Введение в программирование на Java. Инструментальные средства разработки Java-программ.	4	1	2		2	2	
38	Базовые элементы языка. Типы, операторы.	4	2-3	4	2	4	4	
39	Основы объектно-ориентированного программирования для Java. Классы и объекты.	4	4-5	4	2	4	6	
	<i>Итого по модулю 12:</i>	36	1-5	10	4	10	12	Устный опрос
<b>Модуль 13. Обработка исключений на Java.</b>								
40	Работа со строками.	4	6-7	4	2	4	4	
41	Обработка исключений на Java.	4	8	2	2	2	8	
42	Создание Java-приложения. Компиляция и выполнение.	4	9	2		2	4	
	<i>Итого по модулю 13:</i>	36	6-9	8	4	8	16	Контрольная работа №7
<b>Модуль 14. GUI-программирование на Java.</b>								
43	Средства пользовательского интерфейса. Компоненты и контейнеры.	4	10	2		2	6	
44	События. Обработка событий от компонент.	4	11	2	2	2	6	
46	Создание, выполнение и синхронизация потоков. Многопоточность. Приоритеты потоков.	4	12	2	2	2	8	

	<i>Итого по модулю 14:</i>	36	10-12	6	4	6	20	Устный опрос
<b>Модуль 15. GUI-программирование на Java.</b>								
47	Создание потоков связанных с файлами. Файловый ввод/вывод.	4	13-14	4	2	4	10	
48	Программирование меню на Java.	4	15	2	2	2	10	
	<i>Итого по модулю 15:</i>	36	13-15	6	4	6	20	Контрольная работа №8
<b>Модуль 16. Подготовка к экзамену.</b>								
	<b>Подготовка и сдача экзамена</b>	4					36	экзамен
	<b>ИТОГО 4 сем.:</b>	<b>180</b>		<b>30</b>	<b>16</b>	<b>30</b>	<b>68+36</b>	<b>Экзамен</b>
	<b>ИТОГО:</b>	<b>576</b>		<b>120</b>	<b>32</b>	<b>120</b>	<b>304</b>	<b>1,3 сем. - зачеты, 2,4 сем. - экзамены</b>

### 4.3. Содержание дисциплины, структурированное по темам

#### Семестр 1. Программирование на Delphi

##### Модуль 1. Основы программирования.

Тема 1. Алгоритмизация задачи. Понятие алгоритма. Свойства алгоритма

Тема 2. Базовые конструкции языка, простые типы данных, структура программы. Операторы языка.

##### Модуль 2. Структурированные типы данных.

Тема 3. Структурные типы данных. Работа с массивами. Алгоритмы сортировки.

Тема 4. Строковый тип.

##### Модуль 3. Работа с памятью.

Тема 5. Множества. Записи.

Тема 6. Подпрограммы

Тема 7. Работа с файлами

##### Модуль 4. Работа с графикой.

Тема 8. Процедуры и функции модуля Graph

Тема 9. Динамическая память и указатели

Тема10. Динамические структуры данных.

Списки

## **Семестр 2. Среда визуального программирования Delphi.**

### **Модуль 5. Основы объектно-ориентированного программирования.**

Тема11. Объектно-ориентированный подход в программировании. Принципы ООП.

Классы. Основные понятия.

Тема12. Типы данных. Динамические массивы. Создание консольных приложений в Delphi.

Тема 13. Класс TApplication. Сообщения Windows. События от мыши и клавиатуры.

Тема 14. Класс TStringList. Списки. Классификация компонентов. Массив Components.

Тема 15. Формы. Компоненты Standard.

### **Модуль 6. Работа с графикой.**

Тема 16. Класс TCanvas, TGraphic, TPicture.

Тема 17. Анимация в Delphi

Тема 18. Работа с файлами

### **Модуль 7. Работа с динамическими структурами.**

Тема 19. Динамическое создание компонентов. Создание собственных компонентов.

Тема 20. Динамически подключаемые библиотеки.

Тема 21. Интерфейсы.

Тема 22. Процессы и потоки

## **Семестр 3.**

### **Модуль 9. Программирование на C#.**

Тема 23. Основы .Net Framework. Введение в C#

Тема 24. Типы данных C#.

Тема 25. Базовые конструкции языка.

Тема 26. Строки и массивы

### **Модуль 10. Работа с классами на C#.**

Тема 27. Классы и сборка мусора

Тема 28. Обработка исключений в C#. Disposable.

Тема 29. Наследование в C#.

Тема 30. Интерфейсы

### **Модуль 11. Обработка событий и работа с графикой на C#.**

Тема 31. Классы-коллекции

Тема 32. Обобщенные классы

Тема 33. Делегаты. События.  
Тема 34. Сериализация.  
Тема 35. Работа с DLL  
Тема 36. Обработка графики в C#

#### **Семестр 4.**

##### **Модуль 12. Технологии Java-программирования.**

Тема 37. Введение в программирование на Java. Инструментальные средства разработки Java-программ.

Тема 38. Базовые элементы языка. Типы, операторы.

Тема 39. Основы объектно-ориентированного программирования для Java. Классы и объекты.

##### **Модуль 13. Обработка исключений на Java.**

Тема 40. Работа со строками.

Тема 41. Обработка исключений на Java.

Тема 42. Создание Java-приложения. Компиляция и выполнение апплета.

Передача параметров в апплет.

##### **Модуль 14. GUI-программирование на Java.**

Тема 43. Средства пользовательского интерфейса. Компоненты и контейнеры.

Тема 44. События. Обработка событий от компонент

Тема 45. Растровые изображения и анимация в апплетах. Звук в апплетах Java.

Тема 46. Создание, выполнение и синхронизация потоков. Многопоточность.

Приоритеты потоков.

Тема 47. Создание потоков, связанных с файлами. Файловый ввод/вывод.

##### **Модуль 15. GUI-программирование на Java.**

Тема 48. Программирование меню на Java.

Тема 49. Понятие сервлета. Архитектура, жизненный цикл, размещение сервлетов

#### **Темы лабораторных занятий**

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

1. Базовые конструкции языка Паскаль, простые типы данных, структура программы. Операторы языка.
2. Структурные типы данных. Работа с массивами. Алгоритмы сортировки.
3. Строковый тип. Множества. Записи.
4. Подпрограммы
5. Работа с файлами
6. Процедуры и функции модуля Graph

7. Динамическая память и указатели. Динамические структуры данных. Списки.
8. Типы данных. Динамические массивы. Создание консольных приложений в Delphi.
9. Класс TApplication. Сообщения Windows. События от мыши и клавиатуры.
10. Класс TStringList. Списки. Классификация компонентов. Массив Components.
11. Формы. Компоненты Standard. Класс TCanvas, TGraphic, TPicture.
12. Анимация в Delphi
13. Работа с файлами
14. Динамическое создание компонентов. Создание собственных компонентов.
15. Динамически подключаемые библиотеки.
16. Интерфейсы. Процессы и потоки
17. Основы .Net Framework.
18. Типы данных C#. Базовые конструкции языка.
19. Строки и массивы.
20. Обработка исключений в C#. Disposable.
21. Наследование в C#.
22. Обобщенные классы. Делегаты. События.
23. Обработка графики в C#
24. Базовые элементы языка Java. Типы, операторы.
25. Основы объектно-ориентированного программирования для Java. Классы и объекты.
26. Работа со строками.
27. Обработка исключений на Java.
28. Создание Java-приложения. Компиляция и выполнение апплета. Передача параметров в апплет.
29. Средства пользовательского интерфейса. Компоненты и контейнеры.
30. События. Обработка событий от компонент
31. Растровые изображения и анимация в апплетах. Звук в апплетах Java.
32. Создание, выполнение и синхронизация потоков. Многопоточность. Приоритеты потоков.
33. Создание потоков связанных с файлами. Файловый ввод/вывод.
34. Программирование меню на Java.
35. Понятие сервлета. Архитектура, жизненный цикл, размещение сервлетов

## **5. Образовательные технологии**

Процесс изложения учебного материала сопровождается систематическими (на каждом занятии) 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.

Для эффективной реализации целей и задач ФГОС, для претворения компетентностного подхода в преподавании дисциплины «Технологии программирования и работы на ЭВМ», используются следующие образовательные технологии и методы обучения:

Вид занятия	Технология	Цель	Формы и методы обучения
1	2	3	4
Лекции	Технология проблемного обучения	Усвоение теоретических знаний, развитие мышления, формирование профессионального интереса к будущей деятельности	Мультимедийные лекция-объяснение, лекция-визуализация, с привлечением формы тематической дискуссии, беседы, анализа конкретных ситуаций
Практические занятия	Технология проблемного, модульного, дифференцированного и активного обучения, деловая игра	Развитие творческой и познавательной самостоятельности, обеспечение индивидуального подхода с учетом базовой подготовки.	Индивидуальный темп обучения. Инновационные интерактивные методы в обучении: использование Web-ресурсов для подготовки компьютерных презентаций, использование offline (электронная почта) для обмена информацией, консультаций с преподавателем, работа с электронными пособиями, возможность само тестирования.
Лабораторные занятия	Технология проблемного, модульного, дифференцированного и активного обучения, деловая игра	Организация активности студентов, обеспечение лично стно деятельного характера усвоения знаний, приобретения	Инновационные интерактивные методы в обучении. Постановка проблемных познавательных задач. Методы активного обучения: «круглый стол», игровое производственное



		навыков, умений.	
Самостоятельная работа	Технологии концентрированного, модульного, дифференцированного обучения	Развитие познавательной самостоятельности, обеспечение гибкости обучения, развитие навыков работы с различными источниками информации, развитие умений, творческих навыков	Индивидуальные, групповые, интерактивные (в режимах online и offline).

## 6. Учебно-методическое обеспечение самостоятельной работы студентов

### 6.1. Виды и порядок выполнения самостоятельной работы

1. Изучение конспектов лекций и рекомендованной литературы.
2. Подготовка к опросу на практических занятиях
3. Решение задач и упражнений
4. Подготовка к коллоквиуму и контрольным работам
5. Поиск материала на интернет-форумах
6. Подготовка к экзамену

Примерное распределение времени самостоятельной работы студентов

Вид самостоятельной работы	Примерная трудоёмкость, а.ч.		
	Очная	Очно-заочная	заочная
<b>Текущая СРС</b>			
Изучение конспектов лекций и рекомендованной литературы	<b>30</b>		
Подготовка к опросу на практических занятиях	<b>40</b>		
Решение задач и упражнений	<b>45</b>		
Подготовка к коллоквиуму и контрольным работам	<b>45</b>		
Поиск материала на интернет-форумах	<b>22</b>		
Подготовка к экзамену	<b>50</b>		
<b>Итого СРС:</b>	<b>232</b>		

## 6.2. Порядок контроля:

1. Опрос на лабораторном занятии
2. Проверка выполнения домашних заданий и контрольных работ
3. Коллоквиум
4. Зачет.

Раздел (модуль, тема)	Вид самостоятельной работы - практическое содержание	Контрольные сроки (в нед.) и вид контроля	Уч.-мет. обеспечение (указаны источники из списка основной литературы)
1	2	3	4
<b>1 семестр.</b>  Модуль 1. Основы программирования. Модуль 2. Структурированные типы данных. Модуль 3. Работа с памятью. Модуль 4. Работа с графикой.	<b>Программирование на Дельфи.</b>  Алгоритмизация задачи. Понятие алгоритма. Свойства алгоритма. Базовые конструкции языка, простые типы данных, структура программы. Операторы языка. Структурные типы данных. Работа с массивами. Алгоритмы сортировки. Строковый тип. Множества. Записи. Подпрограммы. Работа с файлами. Процедуры и функции модуля Graph. Динамическая память и указатели. Динамические структуры данных. Списки.	6 и 14 недели обучения – устный опрос. Проверка теоретических знаний на устном опросе и коллоквиуме.  9 и 18 недели – контрольные работы. Проверка решенных задач.	[1] – [5];  материалы сайтов:  <a href="http://citforum.ru/">http://citforum.ru/</a>
<b>2 семестр.</b>  Модуль 5. Основы объектно-ориентированного программирования.	<b>Объектно-ориентированный подход в программировании.</b>  Принципы ООП. Система программирования Delphi. Классы. Основные понятия. Типы данных. Динамические массивы. Создание консольных приложений в Delphi. Класс TApplication. Сообщения Windows. События от мыши и	8 и 17 недели обучения – контрольные работы. Проверка решенных задач.  Проверка выполнения	[3], [4];  материалы сайтов:  <a href="http://www.emanual.ru/">http://www.emanual.ru/</a>

	клавиатуры.Класс TStringList.		
Модуль 6. Работа с графикой. Модуль 7. Работа с динамическими структурами.	Списки. Классификация компонентов. Массив Components. Формы. Компоненты Standard. Класс TCanvas, TGraphic, TPicture. Анимация в Delphi. Работа с файлами. Динамическое создание компонентов. Создание собственных компонентов. Динамически подключаемые библиотеки. Интерфейсы. Процессы и потоки	компьютерных программ	
<b>3 семестр.</b>  Модуль 9. Программирование на С# Модуль 10. Работа с классами на С#. Модуль 11. Обработка событий и работа с графикой на С#.	<b>Программирование на С#.</b>  Основы .Net Framework. Введение в С#. Типы данных С#. Базовые конструкции языка. Строки и массивы. Классы и сборка мусора. Обработка исключений в С#. Disposable. Наследование в С#. Интерфейсы. Классыколлекции. Обобщенные классы. Делегаты. События. Сериализация. Работа с DLL. Обработка графики в С#	5 и 14 недели обучения – устный опрос. Проверка теоретических знаний на устном опросе и коллоквиуме.  6 и 18 недели – контрольные работы. Проверка решенных задач.	[2], [6] ;  материалы сайтов: <a href="http://www.emanual.ru/">http://www.emanual.ru/</a>

<p><b>4 семестр</b></p> <p>Модуль 12. Технологии Java программирования.</p> <p>Модуль 13. Обработка исключений на Java.</p> <p>Модуль 14. GUI программирование на Java.</p> <p>Модуль 15. GUI программирование на Java.</p>	<p><b>Введение в программирование на Java.</b> Инструментальные средства разработки Java-программ. Базовые элементы языка. Типы, операторы. Основы объектно-ориентированного программирования для Java. Классы и объекты. Работа со строками. Обработка исключений на Java. Создание Java-приложения. Компиляция и выполнение апплета. Передача параметров в апплет. Средства пользовательского интерфейса. Компоненты и контейнеры. События. Обработка событий от компонент. Растровые изображения и анимация в апплетах. Звук в апплетах Java. Создание, выполнение и синхронизация потоков. Многопоточность. Приоритеты потоков. Создание потоков связанных с файлами. Файловый ввод/вывод. Программирование меню на Java. Понятие сервлета. Архитектура, жизненный цикл, размещение сервлетов</p>	<p>4 неделя обучения – устный опрос. Проверка теоретических знаний на устном опросе и коллоквиуме.</p> <p>8 и 17 недели – контрольные работы. Проверка решенных задач.</p>	<p>[7] – [10]; материалы сайтов:</p> <p><a href="https://ru.wikipedia.org/wiki/Java">https://ru.wikipedia.org/wiki/Java</a></p> <p><a href="http://citforum.ru/">http://citforum.ru/</a></p> <p><a href="https://ru.wikipedia.org/wiki/Java">https://ru.wikipedia.org/wiki/Java</a></p>
---	---	--	---

Для обеспечения самостоятельной работы используется разработанный на кафедре пакет заданий и методических указаний. Самостоятельная работа студентов складывается из проработки лекционного материала и материалов форумов программирования, материала учебника, решения всех заданий из индивидуальных вариантов, решения рекомендуемых задач.

### **Примеры индивидуальных вариантов задач для самостоятельного выполнения:**

#### **Вариант 1**

1. Выведите на экран все положительные делители натурального числа, введенного пользователем с клавиатуры.
2. Создайте два массива из 10 целых случайных чисел из отрезка [1;9] и третий массив из 10 действительных чисел. Каждый элемент с *i*-ым индексом третьего массива должен равняться отношению элемента из первого массива с *i*-ым индексом к элементу из второго массива с *i*-ым индексом. Вывести все три массива

на экран (каждый на отдельной строке), затем вывести количество целых элементов в третьем массиве.

3. Создайте класс прямоугольников, описав в нём все необходимые свойства, подобрав им понятные имена и правильные типы данных.

Опишите в классе конструктор, позволяющий при создании нового объекта явно задать все его свойства. Если это необходимо, то проверьте допустимость их значений в конструкторе (например, в классе обыкновенных дробей нельзя создавать дробь с нулевым знаменателем).

Создайте в классе метод, проверяющий равны ли два прямоугольника по площади.

С использованием построенного класса создайте один прямоугольник со сторонами 3 и 8 и второй прямоугольник со сторонами 6 и 4. Проверьте с помощью созданного метода равны ли прямоугольники по площади и если да, то выведите соответствующее сообщение на экран. **Вариант 2**

1. Выведите на экран все двузначные члены последовательности  $2a_{n-1}+50$ , где  $a_1 = -26$ .
2. Создайте массив из 11 случайных целых чисел из отрезка  $[-1;1]$ , выведите массив на экран в строку. Определите какой элемент встречается в массиве чаще всего и выведите об этом сообщение на экран. Если два каких-то элемента встречаются одинаковое количество раз, то не выводите ничего.
3. Создайте класс углов отложенных против часовой стрелки от положительного направления оси абсцисс, описав в нём все необходимые свойства, подобрав им понятные имена и правильные типы данных.

Опишите в классе конструктор, позволяющий при создании нового объекта явно задать все его свойства. Если это необходимо, то проверьте допустимость их значений в конструкторе (например, в классе обыкновенных дробей нельзя создавать дробь с нулевым знаменателем).

Создайте в классе метод, проверяющий задают ли углы перпендикулярные прямые.

С использованием построенного класса создайте угол в  $10^\circ$  и второй угол в  $280^\circ$ . Проверьте с помощью созданного метода задают ли углы перпендикулярные прямые и если да, то выведите соответствующее сообщение на экран.

### Вариант 3

1. Создать программу, которая будет проверять попало ли случайно выбранное из отрезка  $[20;160]$  целое число в интервал  $(55;120)$  и сообщать результат на экран. 2.

Пользователь вводит с клавиатуры натуральное число большее 3, которое сохраняется в переменную n. Если пользователь ввёл не подходящее число, то программа должна просить пользователя повторить ввод. Создать массив из n случайных целых чисел из отрезка [0;n] и вывести его на экран. Создать второй массив только из чётных элементов первого массива, если они там есть, и вывести его на экран. 3. Создайте класс прямоугольных треугольников, описав в нём все необходимые свойства, подобрав им понятные имена и правильные типы данных.

Опишите в классе конструктор, позволяющий при создании нового объекта явно задать все его свойства. Если это необходимо, то проверьте допустимость их значений в конструкторе (например, в классе обыкновенных дробей нельзя создавать дробь с нулевым знаменателем).

Создайте в классе метод, вычисляющий длину высоты, опущенной на гипотенузу.

С использованием построенного класса создайте треугольник с катетами 3 и 4. Вычислите с помощью метода и выведите на экран длину высоты опущенной на гипотенузу.

#### **Вариант 4**

1. Создайте программу, выводящую на экран первые 20 элементов последовательности 2 4 8 16 32 64 128 ....
2. Создать двумерный массив из 8 строк по 5 столбцов в каждой из случайных целых чисел из отрезка [10;99]. Вывести массив на экран.
3. Создайте класс комплексных чисел, описав в нём все необходимые свойства, подобрав им понятные имена и правильные типы данных.

Опишите в классе конструктор, позволяющий при создании нового объекта явно задать все его свойства. Если это необходимо, то проверьте допустимость их значений в конструкторе (например, в классе обыкновенных дробей нельзя создавать дробь с нулевым знаменателем).

Создайте в классе метод, проверяющий являются ли два комплексных числа сопряженными.

С использованием построенного класса создайте два комплексных числа:  $3i+1$  и  $2i-1$ .

Проверьте с помощью созданного метода являются ли числа сопряженными и если да, то выведите соответствующее сообщение на экран.

### **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 индикатора достижения компетенций (в соответствии с ПООП (при наличии))	Планируемые результаты обучения	Процедура освоения
<p>ОПК-4. Способен находить, анализировать, реализовывать программно и использовать на практике математические алгоритмы, в том числе с применением современных вычислительных систем</p>	<p>ОПК-4.1. Знает базовые основы современного математического аппарата, связанного с проектированием, разработкой, реализацией и оценкой качества программных продуктов и программных комплексов в различных областях человеческой деятельности.  ОПК-4.2. Умеет использовать этот математический аппарат в профессиональной деятельности.  ОПК-4.3. Имеет практический опыт применения современного математического аппарата, связанного с проектированием, разработкой, реализацией и оценкой качества программных продуктов и программных комплексов в различных областях человеческой деятельности.</p>	<p><i>Знает:</i> базовые основы современного математического аппарата, связанного с проектированием программных продуктов и программных комплексов.  <i>Умеет:</i> использовать математический аппарат в профессиональной деятельности.  <i>Владеет:</i> практическим опытом применения современного математического аппарата, связанного с проектированием и разработкой программных продуктов и программных комплексов.</p>	<p>Компетенции формируются на лекционных, практических и лабораторных занятиях. Выполнение тестовых заданий на каждом лабораторном занятии для проверки усвоения самостоятельных заданий. Контроль усвоения компетенций осуществляется на контрольных работах, устных опросах, промежуточном контроле.</p>

<p>ПК-1. Способен демонстрировать базовые знания математических и естественных наук, основ программирования и информационных технологий.</p>	<p>ПК-1.1. Обладает базовыми знаниями, полученными в области математических и (или) естественных наук, программирования и информационных технологий.</p> <p>ПК-1.2. Умеет находить, формулировать и решать стандартные задачи в собственной научно-исследовательской деятельности в математике и информатике.</p> <p>ПК-1.3. Имеет практический опыт научно-исследовательской деятельности в математике и информатике.</p>	<p><i>Знает:</i> базовые понятия в области математических наук и программирования.</p> <p><i>Умеет:</i> находить, формулировать и решать стандартные задачи в собственной научно-исследовательской деятельности.</p> <p><i>Владеет:</i> практическим опытом научно-исследовательской деятельности в математике и информатике.</p>	<p>Компетенции формируются на лекционных, практических и лабораторных занятиях.</p> <p>Выполнение тестовых заданий на каждом лабораторном занятии для проверки усвоения самостоятельных заданий. Контроль усвоения компетенций осуществляется на контрольных работах, устных опросах, промежуточном контроле.</p>
<p>ПК-7. Способен создавать и исследовать новые математические модели в естественных науках, промышленности и бизнесе, с учетом возможностей современных информационных технологий и программирования и компьютерной техники</p>	<p>ПК-7.1. Знает основные методы проектирования и производства программного продукта, принципы построения, структуры и приемы работы с инструментальными средствами, поддерживающими создание программных продуктов и программных комплексов, их сопровождения, администрирования</p>	<p><i>Знает:</i> основные методы проектирования и производства программного продукта.</p> <p><i>Умеет:</i> использовать методы проектирования и производства программного продукта.</p> <p><i>Владеет:</i> практическим опытом применения указанных выше методов и технологий.</p>	



	<p>и развития (эволюции).  ПК-7.2. Умеет использовать методы проектирования и производства программного продукта, принципы построения, структуры и приемы работы с инструментальными средствами, поддерживающими создание программного продукта.  ПК-7.3. Имеет практический опыт применения указанных выше методов и технологий.</p>		
--	---	--	--

### 7.3. Типовые контрольные задания

#### ВАРИАНТ - 1

1. Порядковые типы данных.
2. Составной оператор, оператор выбора.
3. Вычислить:  $n!$
4. В двумерном вещественном массиве  $3 \times 4$  найти строку с наименьшим элементом.
5. Пусть заданы координаты:  $x_1, y_1; x_2, y_2; x_3, y_3$  вершин треугольника и координаты  $x, y$  любой точки на плоскости. Определить: лежит ли точка с заданными координатами  $x, y$  внутри треугольника?

#### ВАРИАНТ – 2

1. Строки. Процедуры и функции для работы со строками.
2. Условный оператор, оператор цикла с постусловием.

3. Вычислить:  $\sin(x) + \sin(x^n) + \dots + \sin(x^n)$ ;
4. В двумерном целочисленном массиве  $3 \times 4$  найти сумму элементов столбца с максимальным элементом.
5. Проверить корректность расстановки скобок в арифметическом выражении.

### ВАРИАНТ – 3

1. Множества. Описание и операции над множествами.
2. Оператор безусловного перехода. Оператор цикла с предусловием.
3. Вычислить:  $A * (A+1) * (A+2) * \dots * (A+n)$ ;
4. В заданной строке найти количество слов. Слова могут быть разделены и несколькими пробелами.
5. По некоторому каналу связи передается сообщение, имеющее вид последовательности нулей и единиц. Из-за помех возможен ошибочный прием некоторых сигналов: ноль может быть воспринят как единица и наоборот. Для повышения вероятности правильного приема сигналов было решено передавать каждый сигнал трижды. Теперь передатчик вместо 1 всегда передает 111, а вместо 0 всегда 000. Вам предлагается написать программу, которая будет восстанавливать исходное сообщение. При передаче могли произойти ошибки, поэтому вместо каждой тройки цифр программа должна вывести ту цифру, которая встречается в этой тройке по крайней мере два раза.

### ВАРИАНТ – 4

1. Тип запись. Тип массив.
2. Оператор цикла *for*.
3. Вычислить:  $\cos(x) + \cos(2*x) + \dots + \cos(n*x)$ ;
4. В двумерном целочисленном массиве  $3 \times 4$  найти сумму элементов главной диагонали.
5. Даны координаты центров столбов, вбитых в вершины выпуклого многоугольника. Известен диаметр столбов. Найти длину натянутой вокруг столбов нити.

#### **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 – 10 баллов,
- выполнение текущих практических заданий – 40 баллов,
- выполнение домашних (аудиторных) контрольных работ - 50 баллов.

Промежуточный контроль по дисциплине включает:

- устный опрос - 50 баллов,
- письменная контрольная работа - 50 баллов.

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

а) адрес сайта курса

<http://cathedra.dgu.ru/EducationalProcess.aspx?Value=18&id=6>

б) основная литература:

1. Мейер Б. Объектно-ориентированное программирование и программная инженерия [Электронный ресурс]/ Мейер Б.— Электрон. текстовые данные.— М.: Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Эр Медиа, 2019.— 285 с.— Режим доступа: <http://www.iprbookshop.ru/79706.html> . – ЭБС «IPRbooks»
2. Биллиг В.А. Основы программирования на С# [Электронный ресурс]/ Биллиг В.А.— Электрон. текстовые данные.— М.: Интернет-Университет Информационных Технологий (ИНТУИТ), 2016.— 574 с.— Режим доступа: <http://www.iprbookshop.ru/73695.html> .— ЭБС «IPRbooks»
3. Соколов, Александр Павлович. Системы программирования: теория, методы, алгоритмы: [учеб. пособие по направлению 654600 "Информатика и вычисл. техника"] / Соколов, Александр Павлович. - М.: Финансы и статистика, 2004. - 319, [1] с.: ил.; 21 см. - Библиогр.: с. 309-310.- Предм. указ.: с. 313-320. - Рекомендовано УМО. - ISBN 5-27902770-7: 101-97. Местонахождение: Научная библиотека ДГУ.
4. Пятибратов, Александр Петрович. Вычислительные системы, сети и телекоммуникации: [учеб. для вузов по специальности "Прикладная информатика в экономике"] / Пятибратов, Александр Петрович; Л.П.Гудыно, А.А.Кириченко; под ред. А.П. Пятибратова. - 3-е изд., перераб. и доп. - М.: Финансы и статистика, 2005, 2003. - 558, [1] с.; 25 см. - Библиогр.: с. 539-541. - Предм. указ.: с. 553-559. - Рекомендовано МО РФ. - ISBN 5-279-02779-0: 257-40. Местонахождение: Научная библиотека ДГУ.
5. Желонкин, Андрей Владимирович. Основы программирования в интегрированной среде DELPHI: практикум / Желонкин, Андрей Владимирович. - 2-е изд. - М.: БИНОМ. Лаб. знаний, 2006. - 236, [4] с.: ил. - ISBN 5-94774-417-1: 127-00. Местонахождение: Научная библиотека ДГУ.

в) дополнительная литература:

1. Роганов Е.А. Основы информатики и программирования [Электронный ресурс]/ Роганов Е.А.— Электрон. текстовые данные.— М.: Интернет-Университет Информационных Технологий (ИНТУИТ), 2016.— 392 с.— Режим па: <http://www.iprbookshop.ru/73689.html> .— ЭБС «IPRbooks»
2. Информатика : [учеб. для экон. специальностей вузов / Н.В.Макарова, Л.А.Матвеев, В.Л.Бройдо и др.]; под ред. Н.В.Макаровой. - 3-е изд., перераб. - М. : Финансы и статистика, 2007, 2005, 2001. - 765 с. : ил. ; 26 см. - Библиогр. в конце гл. - Предм. указ.: с. 748-758. - Рекомендовано МО РФ. - ISBN 978-5-279-02202-1 : 369-60. Местонахождение: Научная библиотека ДГУ.
3. Галатенко, Владимир Антонович. Основы информационной безопасности : учеб. пособие для студентов вузов, обуч. по специальности 351400 "Прикл. информ." / Галатенко, Владимир Антонович. - 4-е изд. - М. : Изд-во Интернет-Ун-та Информ. Технологий: БИНОМ. Лаб. знаний, 2016, 2008, 2006. - 205 с. - (Основы информационных технологий). - Рекомендовано УМО. - ISBN 978-5-94774-821-5 : 230-00. Местонахождение: Университетская библиотека
4. Гагарина, Лариса Геннадьевна. Технология разработки программного обеспечения : [учеб. пособие] / Гагарина, Лариса Геннадьевна, Е. В. Кокорева ; под ред. Л.Г.Гагариной. - М. : ФОРУМ: ИНФРА-М, 2009, 2008. - 399 с. - (Высшее образование). - Допущено УМО. - ISBN 978-5-8199-0342-1 (ИД "ФОРУМ") : 246-84.

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

1. ЭБС IPRbooks: <http://www.iprbookshop.ru/> Лицензионный договор № 2693/17 от 02.10.2017г. об оказании услуг по предоставлению доступа. Доступ открыт с с 02.10.2017 г. до 02.10.2018 по подписке( доступ будет продлен)
2. Электронно-библиотечная система «Университетская библиотека лайн» [www.biblioclub.ru](http://www.biblioclub.ru) договор № 55\_02/16 от 30.03.2016 г. об оказании информационных услуг. (доступ продлен до сентября 2019 года).
3. Moodle [Электронный ресурс]: система виртуального обучением: [базаданных] / Даг. гос. ун-т. - Махачкала, г. - Доступ из сети ДГУ или, после регистрации из сети ун-та, из любой точки, имеющей доступ в интернет. - URL: <http://moodle.dgu.ru/> (дата обращения: 22.03.2018).
4. Доступ к электронной библиотеки на <http://elibrary.ru> основании лицензионного соглашения между ФГБОУ ВПО ДГУ и «ООО» «Научная Электронная библиотека» от 15.10.2003. (Раз в 5 лет обновляется лицензионное соглашение)
5. Национальная электронная библиотека <https://нэб.рф/>. Договор 101/НЭБ/101/НЭБ/1597 от 1.08.2017г. Договор действует в течении 1года с момента его подписания.
6. Федеральный портал «Российское образование» <http://www.edu.ru/> (единое окно доступа к образовательным ресурсам).

7. Федеральное хранилище «Единая коллекция цифровых образовательных ресурсов» <http://school-collection.edu.ru/>
8. Российский портал «Открытого образования» <http://www.openet.edu.ru>
9. Сайт образовательных ресурсов Даггосуниверситета <http://edu.icc.dgu.ru>
10. Информационные ресурсы научной библиотеки Даггосуниверситета <http://elib.dgu.ru> (доступ через платформу Научной электронной библиотеки [elibrary.ru](http://elibrary.ru)).
11. Федеральный центр образовательного законодательства <http://www.lexed.ru>
12. <https://www.msu.ru/resources/msu-ws1.html#mm> – доступ к ресурсам мехмата МГУ
13. <https://www.msu.ru/resources/msu-ws1.html#cm> – доступ к ресурсам ВМК МГУ

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

1) При проведении аттестации студентов важно всегда помнить, что систематичность, объективность, аргументированность - главные принципы, на которых основаны контроль и оценка знаний студентов. Проверка, контроль и оценка знаний студента, требуют учета его индивидуального стиля в осуществлении учебной деятельности. Знание критериев оценки знаний обязательно для преподавателя и студента. Помимо выполнения заданий на лабораторных занятиях рекомендуется самостоятельно решить упражнения, предложенных к каждой лекции.

2) Планирование времени на самостоятельную работу, необходимого на изучение настоящей дисциплины, студентам лучше всего осуществлять на весь семестр, предусматривая при этом регулярное повторение пройденного материала. Самостоятельная работа студентов заключается в решении всех разобранных на занятиях упражнений, материала учебника и соответствующих форумов интернет, решения всех заданий из индивидуальных практических заданий, решения рекомендуемых задач, подготовки к сдаче промежуточных отчетов и экзамена и дополнительной работы в компьютерном классе самостоятельно.

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

Для проведения полноценных занятий необходимо следующее программное обеспечение: Операционная система Windows 7, 8.1 и 10, JDK, Microsoft Visual Studio Express, NetBeans, Ubuntu Linux.

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

Лекции по дисциплине читаются в классе, оборудованном проектором, к каждому занятию имеются презентации, лабораторные работы проходят в компьютерном классе, оборудованном необходимым аппаратными и программными средствами. Часть лекций предоставляется студенту в электронном формате. Практические занятия проводятся в компьютерных классах с современным аппаратным и программным обеспечением. На сайте кафедры размещаются учебные пособия и презентации к лекции.