

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РФ  
ФГБОУ ВО «ДАГЕСТАНСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ  
УНИВЕРСИТЕТ»

РЕКОМЕНДОВАНО К

УТВЕРЖДЕНИЮ:

Декан, председатель совета

факультета КТ, ВТиЭ

Ш.А.Юсуфов  
19 09 2018 г.

УТВЕРЖДАЮ:

Проректор по учебной работе,  
председатель методического  
совета ДГТУ

Н.С.Суракатов  
19 11 2018 г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЬ)

Дисциплина Б1.В.ОД.10 Программирование  
для направления 09.03.04 – Программная инженерия  
по профилю Разработка программно-информационных систем  
факультет Компьютерных технологий, вычислительной техники и энергетики  
наименование факультета, где ведется дисциплина  
кафедра Программного обеспечения вычислительной техники и автоматизированных систем  
наименование кафедры, за которой закреплена дисциплина  
Квалификация выпускника (степень) бакалавр

Форма обучения очная; курс 2; семестр(ы) 3, 4;

Всего трудоемкость в зачетных единицах (часах) 8 ЗЕТ(288);

Лекции 51 (час); Экзамен 3,4 семестры 2 ЗЕТ (72ч.)

Практические (семинарские) занятия - (час); Зачет - (семестр);

Лабораторные занятия 51 (час); Курсовая работа 4 (семестр);

Самостоятельная работа 114 (час).

Зав. кафедрой В.Б.Мелехин

Начальник УО Э.В.Магомаева

Э.В.Магомаева

Программа составлена в соответствии с требованиями ФГОС ВО с учетом рекомендаций ООП ВО по направлению подготовки бакалавров 09.03.04 – Программная инженерия и профилю «Разработка программно – информационных систем».

Программа одобрена на заседании выпускающей кафедры ПОВТиАС от 12.09 2018 г., протокол № 1

Зав. выпускающей кафедрой по данному направлению  В.Б.Мелехин

**ОДОБРЕНО:**

Методической комиссией по  
укрупненным группам  
специальностей и направлений  
подготовки бакалавриата  
09.00.00 – Информатика и вычислительная  
техника

Председатель МК

  
подпись /А.М.Абдулгалимов/  
И.О.Ф.

13.09 2018г.

**АВТОР(Ы) ПРОГРАММЫ:**

И.В.Шишова, ст.преп.  
И.О.Ф., уч. степень, ученое звание

  
подпись



### **1. Цели освоения дисциплины**

Учебная дисциплина «Программирование» реализует требования федерального государственного образовательного стандарта высшего образования по направлению подготовки 09.03.04 – Программная инженерия (уровень бакалавриата) и профилю «Разработка программно-информационных систем».

Цель дисциплины – ознакомить обучаемых с основными тенденциями и направлениями развития современных технологий программирования и обработки данных, с основными принципами и методологией разработки прикладного программного обеспечения.

Задача дисциплины - привить обучаемым навыки использования типовых способов организации программных данных, а также типовых способов разработки программных алгоритмов.

### **2. Место дисциплины в структуре ООП бакалавриата**

Дисциплина «Программирование» входит в цикл обязательных дисциплин (вариативная часть).

Для успешного усвоения данной дисциплины необходимо, чтобы обучаемый владел знаниями, умениями и навыками, сформированными в процессе изучения дисциплин «Теоретическая информатика» и «Информатика и программирование».

Знания, полученные обучаемыми по дисциплине «Программирование», используются при изучении дисциплин «Алгоритмы и структуры данных» и «Объектно – ориентированное программирование».

### **3. Компетенции обучающегося, формируемые в результате освоения дисциплины «Программирование»**

В результате освоения дисциплины у обучающихся должна быть сформирована следующая общекультурная компетенция:

- способность к самоорганизации и самообразованию (ОК – 7).

В результате освоения дисциплины у обучающихся должны быть сформированы следующие общепрофессиональные и профессиональные компетенции:

- готовность применять основы информатики и программирования к проектированию, конструированию и тестированию программных продуктов (ОПК – 3);
- готовность применять основные методы и инструменты разработки программного обеспечения (ПК – 1);
- владеть навыками чтения, понимания и выделения главной идеи прочитанного исходного кода, документации (ПК-21);
- способность оформления методических материалов и пособий по применению ПС (ПК- 27).

В результате изучения дисциплины обучаемый должен:

#### **Знать:**

- методы поиска, хранения и обработки информации;
- основные принципы и методологию разработки прикладного ПО, типовые способы организации программных данных, а также типовые способы разработки программных алгоритмов;
- синтаксис и семантику универсального алгоритмического языка программирования высокого уровня.



**Уметь:**

- использовать математические инструментальные средства для обработки, анализа и систематизации информации по теме исследования;
- решать на персональной ЭВМ задачи программной обработки данных;
- использовать универсальные программные средства в процессе разработки и сопровождения программных продуктов;
- согласовано решать задачи разработки эффективных моделей данных и алгоритмов их обработки при создании прикладного обеспечения, а также получать программные реализации полученных решений на универсальном алгоритмическом языке высокого уровня.

**Владеть навыками:**

- методами и инструментальными средствами разработки программ



#### 4. Структура и содержание дисциплины (модуля) «Программирование»

##### 4.1. Содержание дисциплины

№ п/п	Раздел дисциплины Тема лекции и вопросы	Семестр	Неделя семестра	Виды учебной работы, включая самостоятельную работу студентов и трудоемкость (в часах)				Формы текущего контроля (по срокам текущей аттестации)
				ЛК	ПЗ	ЛР	СРС	
1	2	3	4	5	6	7	8	9
<b>3 семестр</b>								
1.	<b>Лекция 1.</b> <b>Тема: Указатели</b> Адреса и указатели. Арифметические операции над указателями. Операции отношения. Указатели на указатели. Массивы указателей. Строки и указатели. Ссылки	3	1	2	-	-	8	Вх.контр. работа
2.	<b>Лекция 2.</b> <b>Тема: Динамическая память</b> Средства резервирования динамической памяти. Средства освобождения динамической памяти. Динамические массивы	3	3	2	-	4	10	КР№1
3.	<b>Лекция 3.</b> <b>Тема: Модульное программирование</b> Прототип функций. Определение функций. Вызов функций. Глобальные и локальные переменные. Классы памяти. Параметры функций. Значения по умолчанию.	3	5	2	-	-	8	КР№1
4.	<b>Лекция 4.</b> <b>Тема: Модульное программирование</b> Массивы как параметры функций. Строки как параметры функций. Указатели на функции. Рекурсивные функции. Встраиваемые функции. Параметры функции main()	3	7	2	-	4	10	КР№2
5.	<b>Лекция 5.</b> <b>Тема: Библиотечные функции</b> Символьные функции: алфавитная и числовая проверка, проверка специальных символов, функции преобразования символов. Строковые функции: проверки и обработки строк, преобразования строк в числа. Числовые функции. Функции генерации случайных чисел.	3	9	2	-	4	10	Кр№2



1	2	3	4	5	6	7	8	9
6.	<b>Лекция 6.</b> <b>Тема: Функции текстового и графического режима</b> Представление экрана в текстовом режиме. Функции заголовочного файла conio.h. Представление экрана в графическом режиме. Функции заголовочного файла graphics.h	3	11	2	-	-	8	Кр№3
7.	<b>Лекция 7.</b> <b>Тема: Поточковый ввод-вывод</b> Стандартные потоки. Манипуляторы и форматирование ввода-вывода. Функции символьного ввода-вывода. Функции строкового ввода-вывода. Ошибки потоков	3	13	2	-	-	10	КР№3
8.	<b>Лекция 8.</b> <b>Тема: Работа с файлами</b> Типы доступа к файлам на диске. Открытие и закрытие файлов. Запись в файл и чтение из файла. Открытие файлов с произвольным доступом. Функции позиционирования. Двоичные файлы	3	15	2	-	5	10	
9.	<b>Лекция 9.</b> <b>Тема: Ввод-вывод нижнего уровня</b> Открытие/закрытие файла. Чтение и запись данных. Произвольный доступ к файлу.	3	17	1	-	-	-	
<b>Итого за 3 семестр</b>				17		17	74	Экзамен (13ЕТ-36)
<b>4 семестр</b>								
1	2	3	4	5	6	7	8	9
1.	<b>Лекция 1.</b> <b>Тема: Пользовательские типы данных. Перечисления</b> Типы данных, создаваемые пользователем. Ключевое слово typedef. Перечислимый тип данных. Синтаксис объявления перечисления. Переменные перечислимого типа и операции над ними	4	1	2	-	4	2	КР№1
2.	<b>Лекция 2.</b> <b>Тема: Структуры</b> Синтаксис объявления структуры. Псевдонимы структур. Присваивание значений структурным переменным. Операции, допустимые над переменными структурного типа	4	2	2	-	4	2	КР№1



1	2	3	4	5	6	7	8	9
3.	<b>Лекция 3.</b> <b>Тема: Структуры, массивы и указатели</b> Массивы и структуры в качестве элементов структур. Массивы структур. Указатели на структуры. Указатели как средство доступа к элементам структур. Операции над указателями на структуры	4	3	2	-	4	2	Кр№1
4.	<b>Лекция 4.</b> <b>Тема: Объединения и битовые поля</b> Синтаксис объявления объединения. Размещение объединения в памяти. Операции с объединениями. Битовые поля. Функции работы с датой и временем	4	4	2	-		2	Кр№1
5.	<b>Лекция 5.</b> <b>Тема: Динамические структуры данных</b> Понятие о самоссылочных структурах. Классификация самоссылочных структур. Области применения	4	5	2	-	-	2	Кр№1
6.	<b>Лекция 6.</b> <b>Тема: Односвязные списки</b> Формирование очереди. Формирование стека. Добавление и удаление элементов в односвязных списках	4	6	2	-	4	2	Кр№2
7.	<b>Лекция 7.</b> <b>Тема: Двусвязные списки</b> Формирование двусвязных списков. Добавление и удаление элементов в двусвязных списках. Реализация списков с помощью массивов	4	7	2	-	4	2	Кр№2
8.	<b>Лекция 8.</b> <b>Тема: Бинарные деревья</b> Понятие бинарного дерева. Особенности дерева двоичного поиска. Основные операции с деревьями. Сортировка на основе деревьев	4	8	2	-		4	Кр№2
9.	<b>Лекция 9.</b> <b>Тема: Примеры разработки программ. Обработка списков в основной памяти</b> Постановка задачи. Разработка функций, реализующих операции по обслуживанию базы данных. Сохранение базы данных на диске	4	9	2	-		2	Кр№2
10.	<b>Лекция 10.</b> <b>Тема: Классы и объекты</b> Описание классов. Уровни доступа к элементам класса. Описание объектов. Функции в составе класса. Указатель this	4	10	2	-	4	2	Кр№2



11.	<b>Лекция 11.</b> <b>Тема: Конструкторы и деструкторы</b> Конструкторы по умолчанию. Конструкторы с параметрами. Конструкторы копирования. Деструкторы	4	11	2	-	4	2	Кр№3
12.	<b>Лекция 12.</b> <b>Тема: Стандартный класс string</b> Класс string. Конструкторы строк. Изменение величины строки и ее емкости. Сравнение строк. Операции поиска	4	12	2	-	-	2	Кр№2
13.	<b>Лекция 13.</b> <b>Тема: Контейнерные классы</b> Классификация контейнеров. Последовательные контейнеры: векторы, стеки, очереди, деки. Ассоциативные контейнеры	4	13	2	-	4	4	КР№3
14.	<b>Лекция 14.</b> <b>Тема: Универсальные алгоритмы</b> Немодифицирующие и модифицирующие операции с последовательностями. Алгоритмы, связанные с сортировкой	4	14	2	-	2	4	КР№3
15.	<b>Лекция 15.</b> <b>Тема: Обработка ошибок</b> Виды ошибок. Возвращаемое значение функции как признак ошибок. Обработка исключительных ситуаций. Синтаксис исключений. Перехват исключений. Иерархии исключений	4	15	2	-	-	2	КР№3
16.	<b>Лекция 16.</b> <b>Тема: Препроцессорные средства</b> Включение текстов из файлов. Условная компиляция. Макроподстановки средствами препроцессора. Отличие макросов от функций. Вспомогательные директивы. Встроенные макроимена	4	16	2	-	-	2	
17.	<b>Лекция 17.</b> <b>Тема: Компиляция и компоновка программ</b> Компилируемые и интерпретируемые языки. Компоновка нескольких файлов в одну программу. Проблема использования общих функций и имен	4	17	2	-	-	2	
	<b>Итого за 4 семестр</b>			<b>34</b>		<b>34</b>	<b>40</b>	Экзамен (13ЕТ-36ч.)
	<b>Итого за 3 и 4 семестры</b>			<b>51</b>	-	<b>68</b>	<b>114</b>	



## 4.2. Содержание лабораторных занятий

№ п/п	№ лекции из рабочей программы	Наименование лабораторного занятия	Количество часов	Рекомендуемая литература и методические разработки(№источника из списка литературы)
<b>3 семестр</b>				
1	№1	Разработка программ обработки данных с помощью указателей	4	№1 - №7
2	№3	Разработка программ с использованием функций	4	№1 - №5
3	№4, №5	Разработка функций обработки массивов данных	4	№1 - №6
4	№7, №8	Разработка программ обработки файлов данных	4	№1-№4, №6
<b>Итого за 3 семестр</b>			<b>17</b>	
<b>4 семестр</b>				
1	№1	Разработка программ с использованием перечислений	4	№1-5
2	№2	Разработка программ обработки структур данных	4	№1-№6
3	№2, №3	Разработка программ с применением указателей для работы со структурами	4	№1-№7
4	№6	Разработка программ работы с односвязными динамическими структурами данных	4	№1, №2, №6
5	№7	Разработка программ работы с двусвязными динамическими структурами данных	4	№1-№6
6	№10	Разработка программ с классами	4	№4
7	№13	Разработка классов с конструкторами и деструкторами	4	№1-№4
8	№13	Разработка программ с использованием контейнерных классов	6	№1-№4
<b>Итого за 4 семестр</b>			<b>34</b>	
<b>Итого за 3 и 4 семестры</b>			<b>51</b>	



### 4.3. Тематика для самостоятельной работы студента

№ п/п	Тематика по содержанию дисциплины, выделенная для самостоятельного изучения	Кол-во часов из содержания дисциплины	Рекомендуемая литература и источники информации	Форма контроля СРС
<b>3 семестр</b>				
1	Массивы указателей и моделирование многомерных массивов	8	№1-№6	Кр№1
2	Функции резервирования и освобождения динамической памяти malloc(), calloc() и free()	10	№1, №2, №7	КР№1
3	Функции с переменным числом параметров. Функций с переменным числом параметров	8	№1, №2, №5	КР№1
4	Указатели на функции. Указатели на функции как параметры. Указатель на функцию как возвращаемое функцией значение	10	№1-№4	КР№2
5	Функции генерации случайных чисел. Функции задержки и выдачи звукового сигнала	10	№6	КР№2
6	Функции заголовочного файла conio.h	8	№1-№4	КР№3
7	Манипуляторы ввода-вывода	10	№1-№4	КР№3
8	Двоичные файлы. Организация доступа к элементам двоичных файлов	10	№7	Зачетная Кр
<b>Итого за 3 семестр</b>		<b>74</b>		
<b>4 семестр</b>				
1	Применение перечислений	2	№1-№6	Кр№1
2	Вложенные структуры	2	№1-№7	КР№1
3	Структуры и функции	2	№2	КР№1
4	Функции работы с датой и временем	2	№1, №2, №5	КР№1
5	Применение самоссылочных структур	2	№2	КР№1
6	Сортировка односвязных списков	2	№1, №2, №5	КР№2
7	Сортировка двусвязных списков	2	№4, №7	КР№2
8	Сортировка на основе бинарного дерева	4	№3, №7	КР№2
9	Поиск данных в структурах	2	№3, №4, №5	КР№2
10	Дружественные функции и классы	2	№1, №3, №5	КР№2
11	Указатели в составе классов	2	№1, №6	КР№3
12	Функции обработки строк класса string	2	№2, №7	КР№3
13	Ассоциативные контейнеры: карты, множества	4	№1, №2, №4	КР№3
14	Численные алгоритмы	4	№1- №5	КР№3
15	Классы исключений	2	№5	КР№3
16	Препроцессорные обозначения строк. Пустая директива. Прагмы	2	№1, №7	Экзамен
17	Сборки	2	№1, №4, №5	Экзамен
<b>Итого за 4 семестр</b>		<b>40</b>		
<b>Итого за 3 и 4 семестры</b>		<b>114</b>		



#### **4.4. Тематика курсовых работ**

Цель курсовой работы - приобретение студентами навыков самостоятельной разработки, отладки и документирования программ на языке программирования высокого уровня (C++).

В задачи курсовой работы входят:

- анализ задания;
- обзор возможностей используемого языка программирования;
- разработка алгоритма;
- программирование задачи;
- тестирование программы и анализ полученных результатов.

Студентам выдаются индивидуальные задания по одной из следующих тематик: программирование игр, базы данных, программы – эмуляторы, программы, работающие с графикой.

#### **5. Образовательные технологии**

В соответствии с требованиями ФГОС ВО по направлению подготовки реализация компетентного подхода должна предусматривать широкое использование в учебном процессе активных и интерактивных форм проведения занятий в сочетании с внеаудиторной работой с целью формирования и развития профессиональных навыков обучающихся.

Удельный вес занятий, проводимых в интерактивных формах, определяется главной целью программы, особенностью контингента обучающихся и содержанием дисциплины, и в целом в учебном процессе они составляют не менее 20% аудиторных занятий.

При проведении занятий по учебной дисциплине рекомендуется следовать и традиционным технологиям, в частности, в каждом разделе курса выделять наиболее важные моменты, акцентировать на них внимание обучаемых.

При чтении лекций по всем разделам программы иллюстрировать теоретический материал большим количеством примеров, что позволит сделать изложение наглядным и продемонстрировать обучаемым приемы программирования.

При изучении всех разделов программы добиться точного знания обучаемыми основных исходных понятий.



**6. Оценочные средства для текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины и учебно– методическое обеспечение самостоятельной работы студентов**

**Вопросы входного контроля**

1. Формат программы на языке C++
2. Основные типы данных C++
3. Константы. Именованные константы
4. Операции C++. Поразрядные операции
5. Выражения. Преобразования типов данных
6. Операторы выбора
7. Операторы цикла
8. Операторы переходов
9. Массивы
10. Символьные массивы

**3 СЕМЕСТР**

**Контрольная работа №1**

1. Понятие об адресном пространстве
2. Объявление указателей
3. Разыменование указателей
4. Нулевые указатели и указатели на тип void
5. Операции над указателями
6. Указатели и отношения
7. Указатели на указатели
8. Массивы указателей
9. Указатели и массивы
10. Указатели и многомерные массивы
11. Строковые указатели
12. Операторы new и delete
13. Функции malloc(), calloc() и free()
14. Прототип функции
15. Определение функции
16. Вызов функции
17. Ссылки

**Контрольная работа №2**

1. Способы передачи параметров в функцию
2. Передача массивов посредством указателей
3. Строки как аргументы функций
4. Функции с переменным числом параметров
5. Рекурсивные функции
6. Параметры функций со значениями по умолчанию
7. Глобальные и локальные переменные
8. Классы памяти
9. Встраиваемые функции
10. Функция main() с параметрами
11. Функции заголовочного файла `ctype.h`



11. Доступ к элементам структур с помощью указателей
12. Операции над указателями на структуры
13. Объединения. Размещение объединения в памяти
14. Структуры с битовыми полями
15. Функции работы с датой и временем

### **Контрольная работа №2**

1. Классификация динамических структур данных
2. Очередь. Работа с очередью
3. Стек. Работа со стеком
4. Двусвязные списки
5. Бинарные деревья
6. Сортировка в односвязных списках
7. Сортировка в двусвязных списках
8. Сортировка бинарных деревьев
9. Определение классов
10. Объявление объектов и массивов объектов

### **Контрольная работа №3**

1. Свойства конструктора. Конструктор по умолчанию
2. Конструкторы с параметрами
3. Свойства деструктора
4. Конструкторы копирования
5. Последовательные контейнеры
6. Ассоциативные контейнеры
7. Универсальные алгоритмы
8. Препроцессорная директива #define
9. Условная компиляция
10. Макроподстановки средствами препроцессора
11. Встроенные макроимена
12. Отличие макросов от функций
13. Вспомогательные директивы препроцессора
14. Прагмы
15. Препроцессорные обозначения строк

### **Экзаменационные вопросы**

1. Перечисления. Примеры использования перечислимого типа
2. Переименования типов (typedef). Псевдонимы структур
3. Объявление структур. Присваивание значений структурным переменным
4. Операции над структурными переменными
5. Массивы и структуры как элементы структур
6. Функции и структуры
7. Объединения и операции с ними
8. Структуры с битовыми полями
9. Указатели на структуры. Операции над указателями на структуры
10. Классификация динамических структур данных
11. Очередь. Работа с очередью
12. Стек. Работа со стеками
13. Двусвязные списки. Работа с двусвязными списками
14. Бинарные деревья
15. Сортировка динамических структур
16. Реализация динамических структур с помощью массивов



17. Определение классов и объектов
18. Конструкторы. Свойства конструкторов
19. Деструкторы. Свойства деструкторов
20. Конструкторы с параметрами. Конструкторы копирования
21. Стандартный класс string
22. Последовательные контейнеры
23. Ассоциативные контейнеры
24. Универсальные алгоритмы
25. Виды ошибок. Возвращаемое значение функции как признак ошибки
26. Обработка исключительных ситуаций
27. Директива #define. Цепочка подстановок
28. Условная компиляция. Директивы ветвлений
29. Макроподстановки средствами препроцессора. Отличие макросов от функций. Встроенные макроимена
30. Вспомогательные директивы. Препроцессорные обозначения строк. Пустая директива. Прагмы
31. Компилируемые и интерпретируемые языки
32. Компоновка нескольких файлов в одну программу

#### **Вопросы для проверки остаточных знаний студентов**

1. Основные типы данных C++
2. Основные группы операторов C++
3. Массивы
4. Указатели
5. Перечислимый тип данных
6. Структуры
7. Объединения
8. Функции
9. Библиотечные функции
10. Поточковый ввод-вывод. Манипуляторы потоков
11. Файловый ввод/вывод
12. Динамическое распределение памяти
13. Динамические структуры данных
14. Директивы препроцессора



## 7. Учебно-методическое и информационное обеспечение дисциплины (модуля)

### 7.1. Рекомендуемая литература и источники информации (основная и дополнительная)

№	Виды занятий (лк, пр, лб, срс)	Комплект необходимой учебной лит-ры по дисциплинам (наименование учебника, пособия)	Авторы	Издат-во и год издания	Кол-во пособий, учебников и прочей литературы	
					в библиот.	на кафедре
<b>ОСНОВНАЯ</b>						
1.	Лк, лб, срс	Программирование на языке С++ [Электронный ресурс] : учебное пособие	Павловская, Т.А.	Москва : 2016. — 154 с.	<a href="https://e.lanbook.com/book/100409">https://e.lanbook.com/book/100409</a>	
2.	Лк, лб, срс	Языки программирования: учебник для студ. учреждений высш. проф. образования	Баженова И.Ю.	М.: Изд. дом «Академия», 2012		1
3.	Лк, лб, срс	Программирование на языке С++ [Электронный ресурс] : методические указания	С.В. Борисов [и др.].	Москва : МГТУ им. Н.Э. Баумана, 2017. — 76 с.	<a href="https://e.lanbook.com/book/103498">https://e.lanbook.com/book/103498</a>	
4.	Лк, лб, срс	Решение задач на языках программирования Си и Си++ [Электронный ресурс]	Быков, А.Ю.	Москва : МГТУ им. Н.Э. Баумана, 2017. — 248 с.	<a href="https://e.lanbook.com/book/103505">https://e.lanbook.com/book/103505</a>	
5.	Лк, лб, срс	Объектно-ориентированное программирование в С++: лекции и упражнения. Учебное пособие для вузов	Ашарина И.В.	М.: Горячая линия-Телеком, 2008	2	1
<b>ДОПОЛНИТЕЛЬНАЯ</b>						
6.	Лк, лб, срс	Начальный курс С и С++: Учебное пособие	Березин Б.И.	М: Диалог – МИФИ 2007г.	2	1
7.	Лк, лб, срс	Программирование на языке С++ в среде Visual Studio CLR Windows Forms [Электронный ресурс] : учебное пособие /.	Рязанова, Н.Ю. К.Л. Тассов, М.В. Филиппов Н.Ю. Рязанова	Москва : МГТУ им. Н.Э. Баумана, 2017	<a href="https://e.lanbook.com/book/103509">https://e.lanbook.com/book/103509</a>	