



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

РЕКОМЕНДОВАНО
К УТВЕРЖЕНИЮ
Декан факультета КТВТиЭ

 Юсуфов Ш.А.
«02» 03 2020г.

УТВЕРЖДАЮ
Врио ректора ДГТУ,
Председатель методического
совета ДГТУ

 Суракатов Н.С.
«03» 03 2020г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Дисциплина Б1.В.ДВ.9 Теория вычислительных процессов
код и наименование дисциплины по ООП

для направления 09.03.01 «Информатика и вычислительная техника»
код и направление направления подготовки

по профилю Вычислительные машины комплексы системы и сети
наименование профиля подготовки

факультет Компьютерных технологий, вычислительной техники и энергетики
наименование факультета, где ведется дисциплина (практика)

кафедра Управление и информатика в технических системах и вычислительной техники
наименование кафедры, за которой закреплена дисциплина (практика)

Квалификация выпускника (степень) Бакалавр
бакалавр, магистр (специалист)

Форма обучения очная курс 4 семестр (ы) 8
очная, заочная, др.

Всего трудоемкость в зачетных единицах (часах) 3 ЗЕТ (108)

лекции 16 экзамен 8 (1 ЗЕТ – 36 ч.)
час семестр

практические (семинарские) занятия 16 зачет -
час семестр

лабораторные занятия - самостоятельная работа 40
час час

курсовой проект (работа, РГР) -
семестр

И.о. зав. кафедрой


подпись

Асланов Т.Г.

Начальник УО

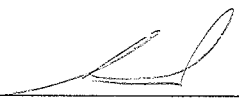

подпись

Магомаева Э.В.

Программа составлена в соответствии с требованиями ФГОС ВО с учетом рекомендаций ООП ВО по направлению подготовки 09.03.01 «Информатика и вычислительная техника»

Программа одобрена на заседании выпускающей кафедры
от «28» 02 2020 года, протокол № 6.

И.о. зав. кафедрой по
данному направлению



Асланов Т.Г.

ОДОБРЕНО

**Методической комиссией
по УГС(Н)
09.00.00 – Информатика и
вычислительная техника**


АВТОР ПРОГРАММЫ

К.т.н., ст. преп. Т.Г. Асланов



подпись

Председатель М.К.



Абдулгалимов А.М.

«28» 02 2020.

1. Цели освоения дисциплины

Целью изучения дисциплины «Теория вычислительных процессов» является формирование знаний об основных моделях вычислительных процессов, необходимых для осуществления профессиональной деятельности.

2. Место дисциплины в структуре ООП бакалавриата

Дисциплина «Теория вычислительных процессов» представляет собой вариативную часть дисциплин по выбору учебного плана.

Дисциплина «Теория вычислительных процессов» основывается на изучении таких дисциплин как: «ЭВМ и периферийные устройства», «Математическая логика и теория алгоритмов» и «Схемотехника ЭВМ и систем».

3. Компетенции обучающегося, формируемые в результате освоения дисциплины «Теория вычислительных процессов»

Выпускник, освоивший программу бакалавриата, должен обладать следующими компетенциями:

общекультурными (ОК):

- способностью использовать методы и средства физической культуры для обеспечения полноценной социальной и профессиональной деятельности (ОК-8);
- способностью использовать приемы оказания первой помощи, методы защиты в условиях чрезвычайных ситуаций (ОК-9).

профессиональными компетенциями (ПК):

- способностью разрабатывать компоненты аппаратно-программных комплексов и баз данных, используя современные инструментальные средства и технологии программирования (ПК-2);
- способностью готовить конспекты и проводить занятия по обучению работников применению программно-методических комплексов, используемых на предприятии (ПК-4);
- способностью сопрягать аппаратные и программные средства в составе информационных и автоматизированных систем (ПК-5);

В результате освоения дисциплины обучающийся должен:

Знать: методы и средства физической культуры; приемы оказания первой помощи, методы защиты в условиях чрезвычайных ситуаций; методы разработки компонентов аппаратно-программных комплексов и баз данных; методы подготовки конспектов и проведения занятий; способы сопряжения аппаратных и программных средства в составе информационных и автоматизированных систем;

Уметь: использовать методы и средства физической культуры; использовать приемы оказания первой помощи, методы защиты; разрабатывать компоненты аппаратно-программных комплексов и баз данных; готовить конспекты и проводить занятия по обучению работников применению программно-методических комплексов, используемых на предприятии; сопрягать аппаратные и программные средства в составе информационных и автоматизированных систем;

Владеть: навыками обеспечения полноценной социальной и

профессиональной деятельности; навыками защиты в условиях чрезвычайных ситуаций; навыками разработки компонентов аппаратно-программных комплексов и баз данных; навыками подготовки конспектов и проведения занятий; навыками сопряжения аппаратных и программных средства в составе информационных и автоматизированных систем;

4. Содержание дисциплины «Теория вычислительных процессов»

4.1 Содержание дисциплины

№	Раздел дисциплины Тема лекции и вопросы	Семестр	Неделя семестра	Виды учебной работы, включая самостоятельную работу студентов и трудоемкость (в часах)				Формы текущего контроля успеваемости (по срокам текущих аттестаций в семестре) Форма промежуточной аттестации (по семестрам)
				ЛК	ПЗ	ЛБ	СР	
а	б	в	г	д	е	ж	з	и
1	<p>Лекция 1</p> <p>ТЕМА: Математическая индукция как способ доказательства правильности программ</p> <p>1. Математическая индукция.</p> <p>Введение: основные направления исследований, связанные с доказательством правильности программ (методы доказательства, конструирование программ и языков, механизация процесса доказательства правильности).</p> <p>2. Простая индукция: принцип простой индукции, принцип модифицированной простой индукции, доказательство высказываний, относящихся к программам для вычислительных машин.</p> <p>3. Строгая версия математической индукции: принцип строгой индукции.</p> <p>4. Обобщенная индукция: принцип обобщенной индукции.</p> <p>5. Доказательство правильности блок-схем программ.</p> <p>6. Основные принципы доказательства правильности для блок-схем.</p> <p>7. Дополнительные примеры доказательства правильности блок-схем программ.</p> <p>8. Метод индуктивных утверждений (основные определения и теоремы, описание метода).</p> <p>9. Сокращенные доказательства правильности.</p> <p>10. Формализация доказательства с помощью индуктивных утверждений.</p>	8	1	2	2	0	5	Входная контрольная работа

2	<p>Лекция 1 ТЕМА: Математическая индукция как способ доказательства правильности программ</p> <ol style="list-style-type: none"> 1. Доказательство правильности программ, написанных на обычных языках программирования. 2. Примеры доказательства правильности программ написанных на Фортране. 3. Примеры доказательства правильности программ, написанных на ПЛ/1. 4. Аксиоматический подход к доказательству частичной правильности. 5. Доказательство частичной правильности как часть процесса программирования. 6. Доказательство правильности рекурсивных программ. 7. Упрощенный язык программирования для иллюстрации понятия рекурсии. 8. Структурная индукция. 9. Более трудные примеры доказательства правильности программ методом структурной индукции. 10. Структурная индукция для нерекурсивных программ. 	2	2	2	0	5	
3	<p>Лекция 1 ТЕМА: Математическая индукция как способ доказательства правильности программ</p> <ol style="list-style-type: none"> 1. Управляющие структуры и структуры данных. 2. Внешний синтаксис для управляющих структур: последовательные структуры, структуры с ветвлением, циклические структуры. 3. Внешний синтаксис структур данных: структуры поименованных данных (скаляр, массив, запись), структуры непоименованных данных (список, стек, множество, очередь, дерево, граф). 4. Внутренний синтаксис: выражения внутреннего синтаксиса, типы данных. 	3	2	2	0	5	
4	<p>Лекция 2 ТЕМА: Организация вычислительных процессов.</p> <ol style="list-style-type: none"> 1. Процессы и потоки 2. Мультипрограммирование. 3. Мультипрограммирование (или многозадачность). 4. Мультипрограммирование в системах пакетной обработки. 5. Мультипрограммирование в системах разделения времени. 6. Мультипрограммирование в системах реального времени. 7. Мультиобработка. 	4	2	2	0	5	

	8. Планирование процессов и потоков. 9. Понятие “процесс” и “поток”. 10. Создание процессов и потоков. 11. Планирование и диспетчеризация потоков. 12. Состояние потока. 13. Вытесняющие и невытесняющие алгоритмы планирования. 14. Алгоритмы планирования, основанные на квантовании. 15. Алгоритмы планирования, основанные на приоритетах. 16. Смешанные алгоритмы планирования. 17. Планирование в системах реального времени. 18. Моменты перепланирования.							
5	Лекция 2 ТЕМА: Организация вычислительных процессов. 1. Процессы и потоки 2. Мультипрограммирование на основе прерываний. 3. Назначение и типы прерываний. 4. Механизм прерываний. 5. Программные прерывания. 6. Диспетчеризация и приоритизация прерываний в ОС. 7. Функции централизованного диспетчера прерываний на примере Windows NT. 8. Процедуры обработки прерываний и текущий процесс. 9. Системные вызовы. 10. Синхронизация процессов и потоков. 11. Цели и средства синхронизации. 12. Необходимость синхронизации и гонки. 13. Критическая секция. 14. Блокирующие переменные. 15. Семафоры. 16. Тупики. 17. Синхронизирующие объекты ОС. 18. Сигналы.	5	2	2	0	5	Аттестационная контрольная работа 1	
6	Лекция 2 ТЕМА: Организация вычислительных процессов. 1. Процессы и потоки. 2. Обмен данными между процессами и потоками. 3. Конвейеры, именованные конвейеры, очереди сообщений. 4. Способ взаимодействия между процессами в распределенных системах (в отличие от централизованных). 5. Синхронизация. 6. Буферизация в примитивах передачи сообщений. 7. Надежные и ненадежные примитивы. 8. Способы адресации, интерфейсы и протоколы связывания клиента с сервером.	6	2	2	0	5		

7	Лекция 3 ТЕМА: Сети Петри 1. Сети Петри как инструмент моделирования систем. 2. Моделирование. 3. Природа систем. 4. Принципы построения сетей Петри. 5. Структура сетей Петри. 6. Графы сетей Петри. 7. Маркировка сетей Петри. 8. Пространство состояний сетей Петри Алгоритмы поведения сетей Петри.	7	2	2	0	5	
8	Лекция 3 ТЕМА: Сети Петри 1. Способы реализации сетей Петри. 2. Альтернативные формы определения сетей Петри. 3. Сети Петри: события и условия, одновременность и конфликт. 4. Моделирование сетями Петри аппаратного и программного обеспечения ЭВМ. 5. Задачи и методы анализа сетей Петри. 6. Области применения сетей Петри.	8	2	2	0	5	
Итого:			16	16	0	40	Экзамен (1 ЗЕТ – 36 ч.)

4.2 Содержание практических занятий

№ п/п	№ по содержанию дисциплины	Наименование практического занятия	Количество часов	Рекомендуемая литература и методические разработки (№ источника из списка литературы)
1	1	Доказательство правильности блок-схем программ.	2	1-5
2	2	Доказательство частичной правильности как часть процесса программирования.	2	1-5
3	3	Изучение внешнего синтаксиса для управляющих структур.	2	1-5
4	4	Создание процессов и потоков.	2	1-5
5	5	Изучение процедуры обработки прерываний и текущий процесс.	2	1-5
6	6	Изучение систем синхронизации.	2	1-5
7	7	Изучение сетей Петри	2	1-5
8	8	Задачи и методы анализа сетей Петри.	2	1-5
Итого:			16	

4.3 Содержание лабораторных занятий

Лабораторные занятия учебным планом не предусмотрены.

4.4 Тематика для самостоятельной работы студента

N п/п	Тематика по содержанию дисциплины, выделенная для самостоятельного изучения	Количество часов из содержания дисциплины	Рекомендуемая литература и источники информации	Формы контроля СРС
1	<p>Математическая индукция. Введение: основные направления исследований, связанные с доказательством правильности программ (методы доказательства, конструирование программ и языков, механизация процесса доказательства правильности). Простая индукция: принцип простой индукции, принцип модифицированной простой индукции, доказательство высказываний, относящихся к программам для вычислительных машин. Строгая версия математической индукции: принцип строгой индукции. Обобщенная индукция: принцип обобщенной индукции. Доказательство правильности блок-схем программ. Основные принципы доказательства правильности для блок-схем. Дополнительные примеры доказательства правильности блок-схем программ. Метод индуктивных утверждений (основные определения и теоремы, описание метода). Сокращенные доказательства правильности. Формализация доказательства с помощью индуктивных утверждений.</p>	5	1-5	Опрос
2	<p>Доказательство правильности программ, написанных на обычных языках программирования. Примеры доказательства правильности программ, написанных на Фортране. Примеры доказательства правильности программ, написанных на ПЛ/1. Аксиоматический</p>	5	1-5	Опрос

	<p>подход к доказательству частичной правильности. Доказательство частичной правильности как часть процесса программирования. Доказательство правильности рекурсивных программ. Упрощенный язык программирования для иллюстрации понятия рекурсии. Структурная индукция. Более трудные примеры доказательства правильности программ методом структурной индукции. Структурная индукция для нерекурсивных программ.</p>			
3	<p>Управляющие структуры и структуры данных. Внешний синтаксис для управляющих структур: последовательные структуры, структуры с ветвлением, циклические структуры. Внешний синтаксис структур данных: структуры поименованных данных (скаляр, массив, запись), структуры непоименованных данных (список, стек, множество, очередь, дерево, граф). Внутренний синтаксис: выражения внутреннего синтаксиса, типы данных.</p>	5	1-5	Опрос
4	<p>Мультипрограммирование. Мультипрограммирование (или многозадачность). Мультипрограммирование в системах пакетной обработки. Мультипрограммирование в системах разделения времени. Мультипрограммирование в системах реального времени. Мультиобработка. Планирование процессов и потоков. Понятие “процесс” и “поток”. Создание процессов и потоков. Планирование и диспетчеризация потоков. Состояние потока. Вытесняющие и невытесняющие алгоритмы</p>	5	1-5	Опрос

	<p>планирования. Алгоритмы планирования, основанные на квантовании. Алгоритмы планирования, основанные на приоритетах. Смешанные алгоритмы планирования. Планирование в системах реального времени. Моменты перепланирования.</p>			
5	<p>Мультипрограммирование на основе прерываний. Назначение и типы прерываний. Механизм прерываний. Программные прерывания. Диспетчеризация и приоритизация прерываний в ОС. Функции централизованного диспетчера прерываний на примере Windows NT. Процедуры обработки прерываний и текущий процесс. Системные вызовы. Синхронизация процессов и потоков. Цели и средства синхронизации. Необходимость синхронизации и гонки. Критическая секция. Блокирующие переменные. Семафоры. Тупики. Синхронизирующие объекты ОС. Сигналы.</p>	5	1-5	Опрос
6	<p>Обмен данными между процессами и потоками. Конвейеры, именованные конвейеры, очереди сообщений. Способ взаимодействия между процессами в распределенных системах (в отличие от централизованных). Синхронизация. Буферизация в примитивах передачи сообщений. Надежные и ненадежные примитивы. Способы адресации, интерфейсы и протоколы связывания клиента с сервером.</p>	5	1-5	Опрос
7	<p>Сети Петри как инструмент моделирования систем. Моделирование. Природа систем. Принципы построения</p>	5	1-5	Опрос

	сетей Петри. Структура сетей Петри. Графы сетей Петри. Маркировка сетей Петри. Пространство состояний сетей Петри. Алгоритмы поведения сетей Петри.			
8	Способы реализации сетей Петри. Альтернативные формы определения сетей Петри. сети Петри: события и условия, одновременность и конфликт. Моделирование сетями Петри аппаратного и программного обеспечения ЭВМ. Задачи и методы анализа сетей Петри. Области применения сетей Петри.	5	1-5	Опрос
Итого:		40		

5. Образовательные технологии

В ходе проведения занятий используются такие методы обучения как презентация, применение компьютерной техники.

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

6.1 Перечень вопросов по проверке входных знаний студентов

1. Блокирующие переменные.
2. Блок-схема программ.
3. Вычислительные машины.
4. Графы.
5. Математическая индукция.
6. Понятие “поток”.
7. Понятие “процесс”.
8. Прерывания.
9. Природа систем.
10. Семафоры.
11. Сети Петри.
12. Сигналы.
13. Синхронизация.
14. Системные вызовы.
15. Системы реального времени.
16. Тупики.

6.2. Задания для текущих аттестаций

1. Математическая индукция.

2. Введение: основные направления исследований, связанные с доказательством правильности программ (методы доказательства, конструирование программ и языков, механизация процесса доказательства правильности).

3. Простая индукция: принцип простой индукции, принцип модифицированной простой индукции, доказательство высказываний, относящихся к программам для вычислительных машин.

4. Строгая версия математической индукции: принцип строгой индукции.

5. Обобщенная индукция: принцип обобщенной индукции.

6. Доказательство правильности блок-схем программ.

7. Основные принципы доказательства правильности для блок-схем.

8. Дополнительные примеры доказательства правильности блок-схем программ.

9. Метод индуктивных утверждений (основные определения и теоремы, описание метода).

10. Сокращенные доказательства правильности.

11. Формализация доказательства с помощью индуктивных утверждений.

12. Доказательство правильности программ, написанных на обычных языках программирования.

13. Примеры доказательства правильности программ написанных на Фортране.

14. Примеры доказательства правильности программ, написанных на ПЛ/1.

15. Аксиоматический подход к доказательству частичной правильности.

16. Доказательство частичной правильности как часть процесса программирования.

17. Доказательство правильности рекурсивных программ.

18. Упрощенный язык программирования для иллюстрации понятия рекурсии.

19. Структурная индукция.

20. Более трудные примеры доказательства правильности программ методом структурной индукции.

21. Структурная индукция для нерекурсивных программ.

22. Управляющие структуры и структуры данных.

23. Внешний синтаксис для управляющих структур: последовательные структуры, структуры с ветвлением, циклические структуры.

24. Внешний синтаксис структур данных: структуры поименованных данных (скаляр, массив, запись), структуры непоименованных данных (список, стек, множество, очередь, дерево, граф).

25. Внутренний синтаксис: выражения внутреннего синтаксиса, типы данных.

26. Мультипрограммирование.

27. Мультипрограммирование (или многозадачность).

28. Мультипрограммирование в системах пакетной обработки.

29. Мультипрограммирование в системах разделения времени.

30. Мультипрограммирование в системах реального времени.

31. Мультиобработка.

32. Планирование процессов и потоков.

33. Понятие “процесс” и “поток”.

34. Создание процессов и потоков.
35. Планирование и диспетчеризация потоков.
36. Состояние потока.
37. Вытесняющие и невытесняющие алгоритмы планирования.
38. Алгоритмы планирования, основанные на квантовании.
39. Алгоритмы планирования, основанные на приоритетах.
40. Смешанные алгоритмы планирования.

6.3. Перечень вопросов по проверке остаточных знаний

1. Математическая индукция.
2. Мультипрограммирование.
3. Состояние потока.
4. Планирование в системах реального времени.
5. Моменты перепланирования.
6. Механизм прерываний.
7. Программные прерывания.
8. Системные вызовы.
9. Сигналы.
10. Синхронизация.
11. Природа систем.
12. Сети Петри.

6.4. Задания для промежуточной аттестации (экзамена)

6.4.1 Контрольные вопросы для проведения экзамена

1. Математическая индукция.
2. Введение: основные направления исследований, связанные с доказательством правильности программ (методы доказательства, конструирование программ и языков, механизация процесса доказательства правильности).
3. Простая индукция: принцип простой индукции, принцип модифицированной простой индукции, доказательство высказываний, относящихся к программам для вычислительных машин.
4. Строгая версия математической индукции: принцип строгой индукции.
5. Обобщенная индукция: принцип обобщенной индукции.
6. Доказательство правильности блок-схем программ.
7. Основные принципы доказательства правильности для блок-схем.
8. Дополнительные примеры доказательства правильности блок-схем программ.
9. Метод индуктивных утверждений (основные определения и теоремы, описание метода).
10. Сокращенные доказательства правильности.
11. Формализация доказательства с помощью индуктивных утверждений.
12. Доказательство правильности программ, написанных на обычных языках программирования.
13. Примеры доказательства правильности программ написанных на Фортране.

14. Примеры доказательства правильности программ, написанных на ПЛ/1.
15. Аксиоматический подход к доказательству частичной правильности.
16. Доказательство частичной правильности как часть процесса программирования.
17. Доказательство правильности рекурсивных программ.
18. Упрощенный язык программирования для иллюстрации понятия рекурсии.
19. Структурная индукция.
20. Более трудные примеры доказательства правильности программ методом структурной индукции.
21. Структурная индукция для нерекурсивных программ.
22. Управляющие структуры и структуры данных.
23. Внешний синтаксис для управляющих структур: последовательные структуры, структуры с ветвлением, циклические структуры.
24. Внешний синтаксис структур данных: структуры поименованных данных (скаляр, массив, запись), структуры непоименованных данных (список, стек, множество, очередь, дерево, граф).
25. Внутренний синтаксис: выражения внутреннего синтаксиса, типы данных.
26. Мультипрограммирование.
27. Мультипрограммирование (или многозадачность).
28. Мультипрограммирование в системах пакетной обработки.
29. Мультипрограммирование в системах разделения времени.
30. Мультипрограммирование в системах реального времени.
31. Мультиобработка.
32. Планирование процессов и потоков.
33. Понятие “процесс” и “поток”.
34. Создание процессов и потоков.
35. Планирование и диспетчеризация потоков.
36. Состояние потока.
37. Вытесняющие и невытесняющие алгоритмы планирования.
38. Алгоритмы планирования, основанные на квантовании.
39. Алгоритмы планирования, основанные на приоритетах.
40. Смешанные алгоритмы планирования.
41. Планирование в системах реального времени.
42. Моменты перепланирования.
43. Мультипрограммирование на основе прерываний.
44. Назначение и типы прерываний.
45. Механизм прерываний.
46. Программные прерывания.
47. Диспетчеризация и приоритизация прерываний в ОС.
48. Функции централизованного диспетчера прерываний на примере Windows NT.
49. Процедуры обработки прерываний и текущий процесс.
50. Системные вызовы.
51. Синхронизация процессов и потоков.
52. Цели и средства синхронизации.

53. Необходимость синхронизации и гонки.
54. Критическая секция.
55. Блокирующие переменные.
56. Семафоры.
57. Тупики.
58. Синхронизирующие объекты ОС.
59. Сигналы.
60. Обмен данными между процессами и потоками.
61. Конвейеры, именованные конвейеры, очереди сообщений.
62. Способ взаимодействия между процессами в распределенных системах (в отличие от централизованных).
63. Синхронизация.
64. Буферизация в примитивах передачи сообщений.
65. Надежные и ненадежные примитивы.
66. Способы адресации, интерфейсы и протоколы связывания клиента с сервером.
67. Сети Петри как инструмент моделирования систем.
68. Природа систем.
69. Принципы построения сетей Петри.
70. Структура сетей Петри.
71. Графы сетей Петри.
72. Маркировка сетей Петри Пространство состояний сетей Петри Алгоритмы поведения сетей Петри.
73. Способы реализации сетей Петри.
74. Альтернативные формы определения сетей Петри.
75. Моделирование сетями Петри аппаратного и программного обеспечения ЭВМ.
76. Задачи и методы анализа сетей Петри.
77. Области применения сетей Петри.

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

Рекомендуемая литература и источники информации (основная и дополнительная)

Зав. библиотекой



№ п/п	Необходимая учебная, учебно-методическая (основная и дополнительная) литература, программное обеспечение и Интернет ресурсы	Автор(ы)	Издательство и год издания	Количество изданий	
				В библиотеке	На кафедре
ОСНОВНАЯ					
1	Теория вычислительных процессов. Часть 2. Теория сетей Петри и моделирование систем : учебное пособие	Веретельникова Е.Л.	Новосибирск : Новосибирский государственный технический университет, 2010	IPR BOOKS iprbookshop.ru/ 47720.html	
2	Теория вычислительных процессов : учебник	Кузнецов А.С.	Красноярск : Сибирский федеральный университет, 2015	IPR BOOKS iprbookshop.ru/ 84154.html	
ДОПОЛНИТЕЛЬНАЯ					
3	Теория вычислительных процессов и структур : учебное пособие	Егоров Д.Л.	Казань : Казанский национальный исследовательский технологический университет, 2018	IPR BOOKS iprbookshop.ru/ 95042.html	
4	Теория информационных процессов и систем : учебное пособие	Чернышев А.Б.	Ставрополь : Северо-Кавказский федеральный университет, 2015	IPR BOOKS iprbookshop.ru/ 63140.html	
5	Теория информационных процессов и систем : учебник	Громов Ю.Ю.	Тамбов : Тамбовский государственный технический университет, ЭБС АСВ, 2014	IPR BOOKS iprbookshop.ru/ 63907.html	

8. Материально-техническое обеспечение дисциплины (модуля)

Семинарские занятия по дисциплине проводятся в аудитории с презентационной техникой и учебной мебелью.

Программа составлена в соответствии с требованиями ФГОС ВО с учетом рекомендаций ООП ВО по направлению и профилю подготовки 09.03.01 – Информатика и вычислительная техника. Рецензент от выпускающей кафедры по направлению _____ Меркухин Е.Н.