Федеральное государственное бюджетное образовательное учреждение
высшего образования
Московский государственный университет имени М.В. Ломоносова
Факультет вычислительной математики и кибернетики

УТВЕРЖДАЮ
декан факультета
вычислительной математики и кибернетики

_____ /И.А. Соколов /
«14» октября 2021г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Наименование дисциплины (модуля):

Теория информации и теория кодирования

Уровень высшего образования:

магистратура

Направление подготовки / специальность:

01.04.02 "Прикладная математика и информатика" (3++)

Направленность (профиль) ОПОП:

Искусственный интеллект в кибербезопасности

Форма обучения:

очная

Рабочая программа рассмотрена и утверждена
на заседании Ученого совета факультета ВМК
(протокол №7, от 29 сентября 2021 года)

Москва 2021

Рабочая программа дисциплины (модуля) разработана в соответствии с Федеральным государственным образовательным стандартом высшего образования (ФГОС ВО) для реализуемых основных профессиональных образовательных программ высшего образования по направлению подготовки 01.04.02 "Прикладная математика и информатика" утвержденного Приказом Министерства образования и науки РФ от 10 января 2018 г. N 13.

1. Место дисциплины (модуля) в структуре ОПОП ВО:
Дисциплина (модуль) относится к части дисциплин основной профессиональной образовательной программы, формируемых участниками образовательных отношений.

2. Входные требования для освоения дисциплины (модуля), предварительные условия:
учащиеся должны владеть знаниями по математическому анализу, линейной алгебре и теории вероятностей в объеме, соответствующем программе обучения основных образовательных программ бакалавриата по укрупненным группам направлений и специальностей 01.00.00 «Математика и механика», 02.00.00 «Компьютерные и информационные науки» и другим направлениям подготовки бакалавриата.

3. Результаты обучения по дисциплине (модулю), соотнесенные с требуемыми компетенциями выпускников.

Планируемые результаты обучения по дисциплине (модулю)		
Содержание и код компетенции.	**Индикатор (показатель) достижения компетенции**	**Планируемые результаты обучения по дисциплине, сопряженные с индикаторами достижения компетенций**
ОПК-1. Способен решать актуальные задачи фундаментальной и прикладной математики	ОПК-1.1. Обладает фундаментальными знаниями, полученными в области математики и информатики.	ОПК-1.1 З-1. Знает архитектурные принципы построения систем искусственного интеллекта, методы декомпозиции основных подсистем (компонентов) и реализации их взаимодействия на основе методологии предметно-ориентированного проектирования
ОПК-2. Способен совершенствовать и реализовывать новые математические методы решения прикладных задач	ОПК-2.2. Умеет выстраивать архитектуру системы искусственного интеллекта математические методы и системы программирования для разработки и реализации алгоритмов.	ОПК-2.2 У-1. Умеет выстраивать архитектуру системы искусственного интеллекта, осуществлять декомпозицию основных подсистем (компонентов) и реализации их взаимодействия на основе методологии предметно-ориентированного проектирования

ОПК-3. Способен разрабатывать математические модели и проводить их анализ при решении задач в области профессиональной деятельности	ОПК-3.1. Знает возможности современных инструментальных средств и систем программирования в области профессиональной деятельности. ОПК-3.2. Умеет проводить сравнительный анализ и осуществлять выбор инструментальных средств для решения задач в области профессиональной деятельности.	ОПК-3.1 З-1. Знает возможности современных инструментальных средств и систем программирования для решения задач машинного обучения ОПК-3.2 У-1. Умеет проводить сравнительный анализ и осуществлять выбор инструментальных средств для решения задач машинного обучения
ОПК-7. Способен использовать методы научных исследований и математического моделирования в области проектирования и управления системами искусственного интеллекта	ОПК-7.1. Применяет логические методы и приемы научного исследования, методологические принципы современной науки, направления, концепции, источники знания и приемы работы с ними, основные особенности научного метода познания, программно-целевые методы решения научных проблем в профессиональной деятельности	ОПК-7.1. З-1. Знает логические методы и приемы научного исследования; методологические принципы современной науки, направления, концепции, источники знания и приемы работы с ними; основные особенности научного метода познания; программно-целевые методы решения научных проблем; основы моделирования управленческих решений; динамические оптимизационные модели; математические модели оптимального управления для непрерывных и дискретных процессов, их сравнительный анализ;

		многокритериальные методы принятия решений в профессиональной деятельности ОПК-7.1. У-1. Умеет применять логические методы и приемы научного исследования; методологические принципы современной науки, концепции, источники знания и приемы работы с ними; основные метода научного познания; программно-целевые методы решения научных проблем; основы моделирования управленческих решений; динамические оптимизационные модели; математические модели оптимального управления для непрерывных и дискретных процессов, их сравнительный анализ; многокритериальные методы принятия решений в профессиональной деятельности

4. Объем дисциплины (модуля) составляет 4 з.е., в том числе 72 академических часа, отведенных на контактную работу обучающихся с преподавателем, 72 академических часов на самостоятельную работу обучающихся.

5. Содержание дисциплины (модуля), структурированное по темам (разделам) с указанием отведенного на них количества академических часов и виды учебных занятий:

Наименование и краткое содержание разделов и	**Всего**	В том числе

тем дисциплины (модуля), Форма промежуточной аттестации по дисциплине (модулю)	(часы)	Контактная работа (работа во взаимодействии с преподавателем) Виды контактной работы, часы			Самостоятельная работа обучающегося, часы *(виды самостоятельной работы – эссе, реферат, контрольная работа и пр. – указываются при необходимости)*
		Занятия лекционного типа	Практические занятия	**Всего**	

1. Дискретные ансамбли и источники: дискретный источник, стационарный источник, источник без памяти, примеры стационарных, нестационарных источников, источников с памятью и без памяти	**5**	2	2	**1**	**0**
2. Количество информации в сообщении. Энтропия как количество собственной информации в сообщении. Энтропия на сообщение дискретного стационарного источника: первая теорема о пределе энтропии, вторая теорема о пределе энтропии, определение энтропии стационарного источника на сообщение.	**5**	2	2	**1**	**4**
3. Задача кодирования дискретных источников равномерными кодами: понятие кода, равномерный и неравномерный код, мощность кода, утверждение о верхней оценки мощности равномерного и неравномерного кода, кодирование сообщения, кодирование сообщений ансамбля, побуквенное кодирование, кодирование по словарю, множество однозначно кодируемых и декодируемых блоков, множество неоднозначно кодируемых и декодируемых блоков, ошибка равномерного кодирования, скорость равномерного кодирования, скорость создания информации при равномерном кодировании, прямая и обратные теоремы кодирования.	**6**	2	0	**2**	**4**
4. Кодирование стационарных источников без памяти. Теорема о высоковероятных множествах дискретного источника без памяти. Высоковероятные множества для двоичного источника без памяти с различными	**8**	4	2	**4**	**4**

параметрами распределения вероятности. Прямая и обратная теоремы кодирования стационарных источников без памяти равномерными кодами.					
5. Эргодические дискретные источники: среднее арифметическое условной собственной информации сообщения, вычисленная по множеству источников и вычисленная по реализации одного источника, роль закона больших чисел, определение класса эргодических источников, теорема об эргодичности дискретного стационарного источника без памяти, пример неэргодического источника. Количество информации, порождаемое эргодическим источником: лемма Мак-Миллана, прямая теорема кодирования, обратная теорема кодирования.	**6**	2	0	**2**	**4**
6. Неравномерное кодирование дискретных источников: постановка задачи, средняя скорость неравномерного кодирования, однозначно декодируемый код, префиксный код, суффиксный код, свойство однозначно декодируемого кода (ограничение на длины кодовых слов), кодовое дерево, неравенство Крафта. Теорема о средней длине кодовых слов однозначно декодируемого кода для одноместного ансамбля, оптимальный код, теорема о существовании оптимального D-ичного кода для одноместного ансамбля, средняя длина кода для произвольного ансамбля, среднее количество кодовых слов и средняя скорость кодирования для произвольного ансамбля,	**12**	4	4	**8**	**4, написание практической программы на языке Python**

обратная теорема кодирования для произвольного ансамбля, прямая теорема кодирования для произвольного ансамбля. Оптимальные неравномерные коды. Метод Шенно-Фано, метод Хаффмена.					
7. Двоичный симметричный канал, его энтропия, пропускная способность. Задача помехоустойчивого декодирования. Определение двоичного блокового кода. Определение линейного кода. Определение двоичного линейного блокового кода. Определение порождающей матрицы кода. Определение скорости передачи линейного кода. Утверждение о связи порождающих матриц одного и того же кода. Определение систематического кода. Задача декодирования. Декодирование по методу максимума правдоподобия и его обоснование. Стратегия декодирования в ближайшее кодовое слово и её обоснование. Ошибка декодирования. Прямая теорема Шеннона	**8**	4	0	**4**	**4**
8. Смежный класс по линейному коду. Лидеры смежного класса. Ошибка декодирования при использовании стандартного расположения. Кодовое расстояние. Связь кодового расстояния с исправляющей способностью кода. Приближения вероятности ошибки для кода, исправляющего t- ошибок. Совершенный и квазисовершенный коды.	**6**	2	2	**4**	**4**
9. Дуальный код. Проверочная матрица. Определение линейного кода через проверочную матрицу. Связь кодового расстояния и	**6**	2	2	**4**	**4, написание практической**

проверочной матрицы кода. Три свойства дуального кода. Слабо самодуальный и самодуальный коды. Синдром вектора. Стандартное расположение и синдром лидеров смежного класса. Декодирование, используя синдром. Синдромный критерий того, что код исправляет t-ошибок.					**программы на языке Python**
10. Код, исправляющий одну ошибку. Код Хэмминга: порождающая и проверочная матрица. Два способа кодирования кода Хэмминга. Декодирование кода Хэмминга. Граница Хэмминга. Граница Варшамова–Гильберта как аналог прямой теоремы Шеннона для линейных кодов. Граница Синглтона.	**12**	2	6	**8**	**4**
11. Код, исправляющий две ошибки. Циклические коды. Проверочная матрица циклического кода. Минимальный многочлен элемента расширения конечного поля и его свойства. Циклотомический класс и построение минимальных многочленов. Цикличность кода Хэмминга. Цикличность кода, исправляющего две ошибки.	**12**	2	6	**8**	**4**
12. Коды БЧХ как обобщение кодов Хэмминга. Задача декодирования кодов БЧХ. Локаторы ошибок. Многочлен локаторов ошибок. Уравнения декодирования. Последовательность, полученная на линейном регистре сдвига. Присоединённый многочлен. Линейная сложность последовательности (минимальная длина регистра сдвига). Теорема Месси. Алгоритм Берлекемпа–Месси и оценка его сложности. Использование алгоритма	**34**	8	10	**18**	**4**

Берлекемпа-Месси для декодирования кодов БЧХ					
Промежуточная аттестация: зачет	**36**	0	0	**0**	**36**
Итого	**144**	**36**	**36**	**72**	**72**

6. Фонд оценочных средств (ФОС, оценочные и методические материалы) для оценивания результатов обучения по дисциплине (модулю).

6.1. Типовые контрольные задания или иные материалы для проведения текущего контроля успеваемости, критерии и шкалы оценивания

Типовые задачи для практических занятий

Задание 1: Использование оптимальных кодов для сжатия данных: ZIPmeHuffman

Требования к программе
Требуется реализовать программу сжатия данных по методу Хаффмана
Программа должна работать в двух режимах.

Режим 1. На вход подаётся бинарный файл.
Программа сжимает файл, сохраняя результат в файл с тем же именем, что и входящий, но с расширением zmh.
Сжатие выполняется двухпроходным методом Хаффмена.
В первый проход строится модель, а во второй проход выполняется сжатие данных.
Формат сжатого файла определяется разработчиком программы.

Режим 2. На вход подаётся файл с расширением zmh.
Программа выполняет разархивирование файла. Результат записывается в файл с тем же именем, что и входной файл, но без расширения zmh.
Предусмотреть возможно обработки файлов, которые повреждены или имеют некорректный формат. В этом случае надо уведомлять пользователя.

Предпочтительный язык программирования — python. В случае использования другого языка нужно создать простую сборочную систему на основе make-файлов для трёх операционных системы: Windows 10, MacOS 10.15+, Linux.

Пример работы программы. Пусть на вход программе был подан текстовый файл в UTF-8 кодировке:

```bash
мама мыла раму
```

Бинарное представление этого файла следующее:

```bash
d0bc("м") d0b0("a") d0bc("м") d0b0("a") 20(" ") d0bc("м") d18b("ы") d0bb("л") d0b0("a") 20(" ") d180("р") d0b0("a") d0bc("м") d183("у")
```

Разбиваем файл на байты

```bash
d0 bc d0 b0 d0 bc d0 b0 20 d0 bc d1 8b d0 bb d0 b0 20 d1 80 d0 b0 d0 bc d1 83
```

Строим модель — словарь, содержащий сколько раз какой байт встречается в файле

```json
{
  "d0": 9,
  "bc": 4,
  "b0": 4,
  "d1": 3,
  "20": 2,
  "8b": 1,
  "bb": 1,
```

```
  "80": 1,
  "83": 1
}
```

Строим код Хаффмена. Например такой,

```json
{
  "d0": 0,
  "bc":  101,
  "b0":  110,
  "d1":  1111,
  "20":  1110,
  "8b":  10011,
  "bb":  10010,
  "80":  10001,
  "83": 10000
}
```

Получаем кодирование входного сообщений

```bash
0 101 0 110 0 101 0 110 1110 0 101 1111 10011 0 10010 0 110 1110 1111 10001 0 110 0 101 1111 10000
```

Или в байтах

```json
01010110
01010110
```

```
11100101
11111001
10100100
11011101
11110001
01100101
11111000
0
```

Что тоже самое:

```bash
56 56 e5 f9 a4 dd f1 65 f8 00
```

Таким образом, на входе

```text
d0bcd0b0d0bcd0b020d0bcd18bd0bbd0b020d180d0b0d0bcd183
```

получим сжатую версию файла:

```text
5656e5f9a4ddf165f800
```

НО!

1. Нужно ещё добавить к этому построенный код Хаффмена, который представляет собой таблицу из девяти пар байтов! Соответственно нужно результат сжатия и таблицу «интегрировать» в один бинарный файл. Постарайтесь сделать это максимально компактно.

2. Последний байт является неполным! Надо это учесть. В противном случае при декодировании в файле появятся лишние байты и файл в итоге может быть испорчен.

На файлах очень маленького размера (десятки байт) накладные расходы на представление самого кода могут перекрывать выигрыш, полученный при сжатии содержимого. Это естественный процесс.

Требования к выполнению задания и принцип его оценивания.

1. Предполагается, что к программе будет приложен файл пояснительной записки с описанием формата zmh, объяснением как были решены проблемы с интеграцией кода в файлы и борьбы с возможными лишними битами в последнем байте. Также записка должна содержать анализ того на сколько можно добиться сжатия различных типов файлов. Необходимо провести аналитическое (ответить на вопрос почему наблюдаются такие результаты, как их можно интерпретировать) сравнение с известными архиваторами (zip, rar, 7z и т.д). Вклад в оценку — 25 %.
2. Оценивается правильность и скорость (!) работы программы (вклад в оценку — 70 %) и её эргономика (вклад в оценку — 5 %).
3. Плагиат кода или пояснительной записки — обнуляет выполнение задачи. И она не засчитывается. Плагиатом не является любое заимствование с указанием его авторства. Заимствованный участок кода не включается в оценивание. Например, Вы не смогли реализовать функцию и заимствовали её у друга, указав это в программе. Реализация этой функции исключается из вклада в оценку правильности работы программы на основе «важности» (принципиальности) этой функции в программе. Важность вещь разумно-субъективная, определяемая лектором курса. Так, если Вы заимствовали реализацию алгоритма сжатия, то, ясно, что это важная функция, т. к. она проверяет знания по теме курса. И её вклад может быть до 20 % вклада кода программы в оценку. В этом случае, этот вклад вычитается из общего вклада в 50 % и получается 30 % вклада программы. Если же была заимствована функция реализации интерфейса командной строки, то она не относится к тем функциям, которые призваны проверить знания по теме курса, поэтому её принципиальность может быть оценена не более, чем в 5 %.
4. Для получения оценки «отлично» нужно выполнить задание более, чем на 79 %, оценка «хорошо» ставится за выполнение на 65 % — 79 %, за 50 % — 65 % — «тройка», и при выполнении задания менее, чем на 50 % ставится — «неудовлетворительно».

Задание 2. Требуется реализовать программу, реализующую схему декодирования на основе стандартного расположения.

Программа должна работать в трёх режимах.
Режим 1 или режим генерации кода. На вход подаётся:
R — скорость передачи кода;

n — длина блока сообщения;

p — вероятность ошибки в двоичном симметричном канале связи.

По выходным параметрам генерирует случайный линейный код длины n со скоростью передачи по меньшей мере R, строит его порождающую матрицу в систематическом виде, строит словарь для декодирования сообщений с границей числа ошибок t. На выходе программы появляется файл, содержащий необходимую информацию для работы остальных режимов программы.

Если такого кода не существует, то выдать объяснение того почему такой код не существует.

На выходе в файл записывается информация для кодера, а также информация для декодера. На экран выводится оценка вероятности ошибки декодера для этого кода на ошибках кратности не более t.

Число ошибок определяется исходя из входных параметров.

Режим 2 или режим кодирования. На вход подаётся:

Файл с описание кода и параметров канала связи, полученный в результате запуска программы в режиме 1;

Сообщение m для кодирования.

Случайно генерируется ошибка e в соответствии с вероятностью ошибки в двоичном симметричном канале. Вероятность берётся из входного файла. На экран выводится результат кодирования и наложения ошибки, а также значение вектора ошибки для контроля правильности работы программы.

Режим 3 или режим декодирования. На вход подаётся:

Файл с описание кода и параметров канала связи, полученный в результате запуска программы в режиме 1;

Сообщение y для декодирования.

На экран выводится значение ошибки, кодовый вектор и декодированное сообщение.

Предпочтительный язык программирования — python. В случае использования другого языка нужно создать простую сборочную систему на основе make-файлов для трёх операционных системы: Windows 10, MacOS 10.15+, Linux.

Требования к выполнению задания и принцип его оценивания.

Предполагается, что к программе будет приложен файл пояснительной записки с описанием формата файла с описание кода и с параметрами канала связи. Вклад в оценку — 5 %.

Оценивается правильность работы программы (вклад в оценку — 50 %), скорость и правильность построения стандартного расположения (вклад в оценку — 30 %) и её эргономика (вклад в оценку — 15 %).

Плагиат кода или пояснительной записки — обнуляет выполнение задачи. И она не засчитывается. Плагиатом не является любое заимствование с указанием его авторства. Заимствованный участок кода не включается в оценивание. Например, Вы не смогли реализовать функцию и заимствовали её у друга, указав это в программе. Реализация этой функции исключается из вклада в оценку правильности работы программы на основе «важности» (принципиальности) этой функции в программе. Важность вещь разумно-субъективная, определяемая

лектором курса. Так, если Вы заимствовали реализацию алгоритма кодирования, то, ясно, что это важная функция, т. к. она проверяет знания по теме курса. И её вклад может быть до 20 % вклада кода программы в оценку. В этом случае, этот вклад вычитается из общего вклада в 50 % и получается 30 % вклада программы. Если же была заимствована функция реализации интерфейса командной строки, то она не относится к тем функциям, которые призваны проверить знания по теме курса, поэтому её принципиальность может быть оценена не более, чем в 5 %.

Для получения оценки «отлично» нужно выполнить задание более, чем на 79 %, оценка «хорошо» ставится за выполнение на 65 % — 79 %, за 50 % — 65 % — «тройка», и при выполнении задания менее, чем на 50 % ставится — «неудовлетворительно».

Вопросы к зачету.

Разделены на две части: для ответа без подготовки и для ответа с подготовкой.

На зачете каждый студент получает 1 вопрос для ответа с подготовкой. При подготовке и ответе на вопрос он может пользоваться любыми материалами.

В начале зачета студент отвечает на вопрос. Самый главный вопрос от экзаменатора: почему это так? Обоснуйте свои выводы?

После ответа на первый вопрос экзаменатор выбирает по очереди любые 3 вопросы из части для ответа без подготовки и просит студента на них ответить. Перед ответом студенту даётся 30 секунд, чтобы собраться с мыслями. Пользоваться при ответе на вопросы никакими материалами нельзя.

Контрольные вопросы

1. Определение дискретного вероятностного ансамбля
2. Статистически независимые/зависимые дискретные вероятностные ансамбли
3. Условная вероятность
4. Понятие дискретного источника: стационарный, без памяти
5. Примеры: нестационарного источника без памяти/с памятью, стационарного без памяти/ с памятью
6. Дискретная случайная величина: математическая ожидание, дисперсия, центральные моменты
7. Теорема о линейности математического ожидания случайной величины (с доказательством)
8. Неравенство Чебышёва с доказательством
9. Понятие корреляционного момента и теорема о коррелляционном моменте статистически независимых случайных величин
10. Закон больших чисел в форме Чебышёва с доказательством
11. Количество собственной информации и её свойства
12. Понятие энтропии: свойства энтропии, верхняя граница энтропии с доказательством
13. Понятие энтропии: свойства энтропии, аддитивность энтропии с доказательством

14. Условная энтропия относительно сообщения
15. Условна собственная энтропия относительностей дискретного вероятностного ансамбля
16. Формулировка теоремы о свойствах собственной условной энтропии
17. Формулировка и доказательство теорему о выражении энтропии совместно заданных дискретных вероятностных ансамблей через сумму условных энтропий
18. Теорема о пределе последовательности H(X_n|X^{n-1}) с доказательством
19. Энтропия стационарного источника на сообщение
20. Понятия кода, кодового символа, кодового алфавита, кодового слова, мощности кода, равномерного и неравномерного кода, кодирование сообщений
21. Утверждение о максимальной мощности равномерного и неравномерного кода с доказательством
22. Ошибка и скорость равномерного кодирования, скорость создания информации дискретным источником
23. Формулировка прямой и обратной теорем кодирования для произвольного источника
24. Формулировка теоремы о высоковероятных множествах дискретного стационарного источника без памяти
25. Прямая теорема кодирования для дискретных стационарных источников без памяти с доказательством
26. Понятие эргодического источника
27. Пример неэргодического источника с обоснованием
28. Теорема о высоковероятных множествах эргодического источника
29. Прямая теорема кодирования эргодических источников
30. Обратная теорема кодирования эргодических источников
31. Средняя скорость неравномерного кодирования
32. Однозначно декодируемый неравномерный код, префиксный код, суффиксный код
33. Необходимое условие однозначной декодируемости кода с доказательством
34. Кодовое дерево и префиксные коды
35. Неравенство Крафта для префиксных кодов с доказательством достаточности
36. Понятие оптимального кода, теорема о минимально возможной средней длине кодовых слов однозначно декодируемого кода с доказательством
37. Верхняя граница минимальной средней длины кодовых слов однозначно декодируемого кода с доказательством
38. Прямая теорема кодирования для неравномерных кодов с доказательством
39. Обратная теорема кодирования для неравномерных кодов с доказательством
40. Метод Шеннона—Фано с обоснованием

41. Общая схема метода Хаффмена
42. Двоичный симметричный канал связи, его пропускная способность
43. Минимальный многочлен элемента конечного поля, его свойства с доказательством
44. Суть стратегии декодирования по методу максимума правдоподобия, понятие ошибки декодирования
45. Понятие кода, исправляющего ошибки, длина кода, скорость его передачи, кодовое расстояние, кодирование
46. Понятие линейного кода, линейный блоковый [n,k]q-код, длина, размерность и кодовое расстояние линейного блокового кода
47. Порождающая матрица линейного кода, утверждение о том, когда две матрицы порождают один и тот же код с доказательством, утверждение о числе порождающих матриц кода с доказательством, утверждение о числе [n,k]q-кодов с доказательством
48. Систематический кодер линейного кода
49. Формулировка прямой теоремы Шеннона для помехоустойчивого кодирования
50. Понятие стандартного расположения, ошибка декодирования при использовании стандартного расположения
51. Когда декодер, работающий в соответствии со стандартным расположением, исправляет ошибки кратности t с вероятностью 1?
52. Понятие совершенного кода и квазисовершенного кода
53. Граница Хемминга с доказательством
54. Граница Синглтона с доказательством
55. Граница Варшамова—Гильберта с доказательством
56. Понятие проверки на чётность для кода, дуальный код, проверочная матрица и её свойства
57. Синдром вектора, использование синдрома для декодирования по стандартному расположению. Когда код, исправляет все ошибки кратности не более t с вероятностью 1, в терминах синдромов векторов?
58. Как построить код, исправляющий одну ошибку? Параметры кода, исправляющего одну ошибку. Схема его декодирования.
59. Понятие циклического кода, понятие идеала кольца R_n, понятие главного идеала, интерпретация циклического кода через идеалы кольца R_n
60. Понятие проверочного многочлена.
61. Граница БЧХ (с доказательством).
62. Понятие кода БЧХ и его параметры.
63. Понятие многочлена локаторов ошибок.
64. Понятие регистра сдвига с линейными обратными связями, присоединенный многочлен регистра сдвига.

6.2. Типовые контрольные задания или иные материалы для проведения промежуточной аттестации по дисциплине, критерии и шкалы оценивания

Вопросы к зачету

1. Теорема о свойствах собственной условной энтропии и её доказательство.
2. Доказательство теоремы о монотонности условной собственной энтропии
3. Теорема о пределе последовательности \frac{1}{n}H(X^n).
4. Теорема о высоковероятных множествах дискретного стационарного источника без памяти
5. Обратная теорема кодирования для дискретных стационарных источников без памяти с доказательством
6. Теореме об эргодичности стационарного источника без памяти
7. Лемма Мак-Миллана
8. Метод Хаффмена с обоснованием его корректности
9. Основная теорема теории помехоустойчивого кодирования
10. Теорема Шеннона о существовании хороших линейных кодов
11. Построение кода, исправляющего две ошибки. Циклические коды, теорема о свойствах циклического кода.
12. Циклические коды, теорема о свойствах циклического кода, построение проверочной матрицы кода с обоснованием, доказательство теоремы о цикличности кода, дуального к циклическому коду.
13. Получение уравнения для декодирования кода БЧХ. Постановка задачи восстановления регистра сдвига. Теорема Месси и следствие из неё.
14. Алгоритм Берлекемпа—Месси с обоснование его корректности и оценкой сложности.

ШКАЛА И КРИТЕРИИ ОЦЕНИВАНИЯ результатов обучения (РО) по дисциплине (модулю)				
Оценка РО и соответствующие виды оценочных средств	2 (не зачтено)	3 (зачтено)	4 (зачтено)	5 (зачтено)

Знания *Зачет*	Отсутствие знаний	Фрагментарные знания	Общие, но не структурированные знания	Сформированные систематические знания
Умения *Практические задания*	Отсутствие умений	В целом успешное, но не систематическое умение	В целом успешное, но содержащее отдельные пробелы умение (допускает неточности непринципиального характера)	Успешное и систематическое умение
Навыки (владения, опыт деятельности) *Зачет, практические занятия*	Отсутствие навыков (владений, опыта)	Наличие отдельных навыков (наличие фрагментарного опыта)	В целом, сформированные навыки (владения), но используемые не в активной форме	Сформированные навыки (владения), применяемые при решении задач

Соответствие результатов обучения и компетенций, в развитии которых участвует дисциплина (модуль)	
Результаты обучения	Компетенция, с частичным формированием которой связано достижение результата обучения
Знать: 1. Математические модели дискретных источников; 2. Общую теорию кодирования дискретных стационарных источников равномерными и неравномерными кодами; 3. Общую теорию кодирования эргодических источников; 4. Основы теории помехоустойчивого кодирования для двоичного симметричного канала связи; 5. Основы алгебраической теории кодирования. **Уметь:**	ОПК-1 ОПК-2 ОПК-3

1. Применять на практике теории алфавитного кодирования для задач сжатия сигнала; 2. Применять на практике теорию помехоустойчивого кодирования для двоичного симметричного канала связи; 3. Строить коды БЧХ и их декодировать.	

7. Ресурсное обеспечение:

7.1. Перечень основной и дополнительной литературы

Основная литература:

1. Сидельников В.М. Теория кодирования.ФИЗМАТЛИТ,Москва, 2008, с. 322.
2. Логачев О.А., Сальников А.А., Смышляев С.В., Ященко В.В. Булевы функции в теории кодирования и криптологии. ЛЕНАНД, Москва, 2015, с. 576.

Дополнительная литература:

1. Э. Берлекэмп. Алгебраическая теория кодирования. Москва «Мир», 1971.
2. Т. Касами, Н. Токура, Ё. Ивадари, Я. Инагаки. Теория кодирования. Москва «Мир», 1978.

7.2.Перечень лицензионного программного обеспечения, в том числе отечественного производстваПри реализации дисциплины может быть использовано следующее программное обеспечение:

1. Операционная система Ubuntu 18.04.
2. Программный продукт Python 3.5.1 (64-bit) Python Software Foundation
3. Операционная система Microsoft Windows 10 Education академическая лицензия

7.3. Перечень профессиональных баз данных и информационных справочных систем

1. http://www.edu.ru – портал Министерства образования и науки РФ
2. http://www.ict.edu.ru – система федеральных образовательных порталов «ИКТ в образовании»
3. http://www.openet.ru - Российский портал открытого образования

4. http://www.mon.gov.ru - Министерство образования и науки Российской Федерации
5. http://www.fasi.gov.ru - Федеральное агентство по науке и инновациям

7.4. Перечень ресурсов информационно-телекоммуникационной сети «Интернет»

1. Math-Net.Ru [Электронный ресурс] : общероссийский математический портал / Математический институт им. В. А. Стеклова РАН ; Российская академия наук, Отделение математических наук. - М. : [б. и.], 2010. - Загл. с титул. экрана. - Б. ц.
URL: http://www.mathnet.ru
2. Университетская библиотека Online [Электронный ресурс] : электронная библиотечная система / ООО "Директ-Медиа" . - М. : [б. и.], 2001. - Загл. с титул. экрана. - Б. ц. URL: www.biblioclub.ru
3. Универсальные базы данных East View [Электронный ресурс] : информационный ресурс / East View Information Services. - М. : [б. и.], 2012. - Загл. с титул. экрана. - Б. ц.
URL: www.ebiblioteka.ru
4. Научная электронная библиотека eLIBRARY.RU [Электронный ресурс] : информационный портал / ООО "РУНЭБ" ; Санкт-Петербургский государственный университет. - М. : [б. и.], 2005. - Загл. с титул. экрана. - Б. ц.
URL: www.eLibrary.ru

7.5. Описание материально-технического обеспечения.

Факультет ВМК, ответственный за реализацию данной Программы, располагает соответствующей материально-технической базой, включая современную вычислительную технику, объединенную в локальную вычислительную сеть, имеющую выход в Интернет. Используются специализированные компьютерные классы, оснащенные современным оборудованием. Материальная база факультета соответствует действующим санитарно-техническим нормам и обеспечивает проведение всех видов занятий (лабораторной, практической, дисциплинарной и междисциплинарной подготовки) и научно-исследовательской работы обучающихся, предусмотренных учебным планом.

8. Соответствие результатов обучения по данному элементу ОПОП результатам освоения ОПОП указано в Общей характеристике ОПОП.

9. Разработчик (разработчики) программы.

доцент факультета ВМК МГУ И. В. Чижов.

10. Язык преподавания - русский.