

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Смирнов Сергей Николаевич
Должность: врио ректора
Дата подписания: 18.10.2023 14:50:57
Уникальный программный ключ:
69e375c64f7e975d4e8830e7b4fcc2ad1bf35f08

Министерство науки и высшего образования Российской Федерации
ФГБОУ ВО «Тверской государственный университет»



Утверждаю:
Руководитель ООП

А.В. Язенин / А.В. Язенин /
«13» *сентября* 2020 года

Рабочая программа дисциплины (с аннотацией)

ТЕОРЕТИЧЕСКИЕ ОСНОВЫ ИНФОРМАТИКИ

Направление подготовки
02.03.02 ФУНДАМЕНТАЛЬНАЯ ИНФОРМАТИКА
И ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

Профиль подготовки
Инженерия программного обеспечения

Для студентов 1-го курса
Форма обучения – очная

Составитель:

к.ф.-м.н., доцент И.С.Солдатенко

Тверь, 2020

I. Аннотация

1. Цель и задачи дисциплины

Целью освоения дисциплины является:

Освоение базовых положений теории информации как теоретической и методологической основы для других дисциплин информационно-технологической подготовки, а также методов решения задач, связанных с представлением и обработкой дискретной информации. Получение базовых фундаментальных знаний о процессах получения, преобразования, хранения и использования информации.

Задачами освоения дисциплины являются:

Систематическое рассмотрение и практическое освоение базовых понятий теории информации, базовых принципов получения, хранения, обработки и использования информации, энтропийного подхода к определению количества информации в соответствии с теорией Клода Шеннона, элементов теории кодирования информации, в том числе помехоустойчивого кодирования, основных методов и алгоритмов сжатия информации, основных положений криптографической защиты информации.

2. Место дисциплины в структуре ООП

Дисциплина относится к части учебного плана, формируемой участниками образовательных отношений, раздел «Дисциплины профиля подготовки».

Дисциплина занимает важное место в процессе подготовки специалистов по профилю «Инженерия программного обеспечения», поскольку, с одной стороны, ее можно отнести к категории мировоззренческих, она призвана сформировать представление о единой информационной картине мира, значении информации и информационных процессов в соответствующих областях человеческой деятельности, а также о существующих научных методах их описания. С другой стороны, она служит основой для освоения других разделов информатики, программирования и информационных технологий, прямо или косвенно касающихся сферы программной инженерии.

Предварительные знания и навыки:

Для успешного освоения дисциплины необходимы знания и навыки, полученные в ходе школьного образования.

Дальнейшее использование:

Полученные знания частично используются при изучении предметов: «Методы программирования», «Алгоритмы и анализ сложности». Знания, умения и навыки, полученные при изучении дисциплины, закрепляются во время лабораторных занятий на дисциплине «Практикум на ЭВМ».

3. Объем дисциплины:

4 зачетных единицы, 144 академических часа, в том числе:

контактная аудиторная работа:

лекции 45 часов; практические занятия 15 часов;

контактная внеаудиторная работа: контроль самостоятельной работы 10 часов, в том числе расчетно-графическая работа 10 часов;

самостоятельная работа:

74 часа, в том числе контроль 36 часов.

4. Планируемые результаты обучения по дисциплине, соотнесенные с планируемыми результатами освоения образовательной программы

Планируемые результаты освоения образовательной программы (формируемые компетенции)	Планируемые результаты обучения по дисциплине
ПК-1 Способен понимать и применять в научно-исследовательской и прикладной деятельности современный математический аппарат, современные языки программирования и программное обеспечение, операционные системы и сетевые технологии	ПК-1.1 Обладает базовыми знаниями в области математических и естественных наук, программирования и информационных технологий ПК-1.2 Применяет полученные знания в области фундаментальных научных основ теории информации и решает стандартные задачи в собственной научно-исследовательской деятельности ПК-1.3 Реализовывает численные методы решения прикладных задач в профессиональной сфере деятельности, пакеты программного обеспечения, операционные системы, электронные библиотеки, сетевые технологии

5. Форма промежуточной аттестации и семестр прохождения: экзамен и РГР в 1-м семестре.

6. Язык преподавания русский.

II. Содержание дисциплины, структурированное по темам (разделам) с указанием отведенного на них количества академических часов и видов учебных занятий

Учебная программа – наименование разделов и тем	Всего (час.)	Контактная работа (час.)				Самостоятельная работа, в том числе Контроль (час.)	
		Лекции		Практические занятия			
		всего	в т.ч. практическая подготовка	всего	в т.ч. практическая подготовка		
<p>1. Измерение информации</p> <p>Информатика: история возникновения, роль и место среди других наук. Понятия информации и данных. Основные понятия теории информации: сигнал, сообщение, код. Свойства информации. Подходы к измерению информации. Вероятностный подход к измерению информации: понятие энтропии, частная энтропия, формула Хартли. Информационная энтропия системы. Формула Шеннона. Свойства и значение информационной энтропии. Информация и энтропия. Информация и алфавит. Объемный подход к измерению информации.</p>	19	4		2		5	8
<p>2. Теория кодирования</p> <p>Основные понятия теории кодирования. Равномерное кодирование информации. Кодирование символьной информации. Представление целых и вещественных чисел в компьютере. Точность представления чисел.</p> <p>Неравномерное кодирование. Избыточность. Асимптотически эффективное кодирование. Префиксные коды. Условие Фано. Алгоритмы построения префиксных кодов Шеннона-Фано и Хаффмана. Оценка избыточности кода. Формулы минимальной и средней длины кода. Повышение эффективности кодирования.</p>	19	4		2		5	8

3. Сжатие информации Энтропийные методы: арифметические кодирование. Методы словарного сжатия: LZ77 и LZ78.	37	12		3			22
4. Помехоустойчивое кодирование Передача информации. Модель канала связи, характеристики канала. Теоремы Шеннона. Помехоустойчивое кодирование информации, способы повышения помехоустойчивости. Теоремы об обнаруживающих и корректирующих кодах. Код Хэмминга. Полиномиальные коды. Циклический избыточный код (CRC).	26	8		2			16
5. Защита информации Основные понятия криптографической защиты информации. Методы подстановок и перестановок. Классические шифры. Шифры Цезаря и Виженера. Модульная арифметика. Расширенный алгоритм Евклида. Функция Эйлера. Асимметричное шифрование. Криптография с открытым ключом. RSA. Симметричное шифрование. DES.	43	17		6			20
ИТОГО	144	45		15		10	74

III. Образовательные технологии

Учебная программа – наименование разделов и тем	Вид занятия	Образовательные технологии
1. Измерение информации	<ul style="list-style-type: none"> • Лекция • Практические занятия 	<ul style="list-style-type: none"> • традиционные (фронтальная лекция, решение упражнений), • цифровые (показ презентаций, выполнение компьютерных лабораторных работ, расчетно-графической работы), • технология проблемного обучения, • групповая работа
2. Теория кодирования		
3. Сжатие информации		
4. Помехоустойчивое кодирование		
5. Защита информации		

IV. Оценочные материалы для проведения текущей и промежуточной аттестации

№	Результат (индикатор)	Примерная формулировка заданий	Вид/способ	Критерии оценивания
МАТЕРИАЛЫ ДЛЯ ПРОВЕДЕНИЯ ТЕКУЩЕЙ АТТЕСТАЦИИ				
1	ПК-1.1 ПК-1.2	Расчетно-графическая работа «Расчет информационных характеристик текста» Самостоятельная исследовательская работа по анализу информационных характеристик выбранного текста. Условие задания приведено в разделе VI.	вид: самостоятельная лабораторная работа способ: на компьютере результаты: отчет.	Максимум – 10 б. Критерии оценки: <ul style="list-style-type: none"> • корректно выполненная и оформленная работа – макс. балл, • за каждый некорректно рассчитанную характеристику снимается 1 балл, • оформление отчета не по заданному шаблону – минус 2 балла.
2	ПК-1.1 ПК-1.2 ПК-1.3	Домашнее задание №1 «Graphics Interchange Format» Самостоятельная лабораторная работа по изучению графического формата GIF и практического применения алгоритма сжатия LZW. Условие задания приведено в разделе VI.	вид: самостоятельная лабораторная работа способ: на компьютере результаты: отчет.	Максимум – 8 б. Критерии оценки: <ul style="list-style-type: none"> • выполнена первая часть задания «Кодирование» - 4 балла, • выполнена вторая часть задания «Декодирование» - 4 балла, • оформление отчета не по заданному шаблону – минус 1 балл. • отсутствие результата выполнения второй части в виде GIF-файла – минус 1 балл.
3	ПК-1.2 ПК-1.3	Домашнее задание №2 «Защита информации» Самостоятельная групповая лабораторная работа, направленная на изучение основ объектно-ориентированного проектирования систем на примере системы защиты информации. Условие задания приведено в разделе VI.	вид: самостоятельная групповая лабораторная работа способ: на компьютере результаты: отчет, код.	Максимум – 5 б. Критерии оценки: <ul style="list-style-type: none"> • задание выполнено в группе полностью и корректно – макс. балл, • код не содержит указанной в задании архитектуры классов – минус 1 балл, • некорректно работает система кодиро-

				<p>вания или декодирования – минус 2 балла,</p> <ul style="list-style-type: none"> отсутствует отчет – минус 1 балл. <p>Если задание выполнено не в группе, а индивидуально, то полученная сумма баллов делится на два.</p>
4	ПК-1.1 ПК-1.2 ПК-1.3	<p>Домашнее задание №3 «Алгоритм RSA»</p> <p>Самостоятельная лабораторная работа по изучению системы шифрования с открытым ключом RSA. Условие задания приведено в разделе VI.</p>	<p>вид: самостоятельная лабораторная работа</p> <p>способ: на компьютере</p> <p>результаты: отчет.</p>	<p>Максимум – 8 б.</p> <p>Критерии оценки:</p> <ul style="list-style-type: none"> задание выполнено полностью и корректно – макс. балл, ключ подобран корректно, но текст не расшифрован – минус 3 балла, отсутствует отчет – минус 1 балл.
5	ПК-1.1 ПК-1.2	<p>Модульная контрольная работа 1</p> <p>Пример задания приведен в разделе VI.</p>	<p>вид: контрольная работа</p> <p>способ: письменно</p> <p>результаты: выполненные задания</p>	<p>Максимум – 15 б.</p> <p>Критерии оценки:</p> <p>Контрольная состоит из 7 заданий. Каждое задание оценивается в 1, 2 или 3 балла.</p>
6	ПК-1.1 ПК-1.2	<p>Модульная контрольная работа 2</p> <p>Пример задания приведен в разделе VI.</p>		<p>Максимум – 14 б.</p> <p>Критерии оценки:</p> <p>Контрольная состоит из 8 заданий. Каждое задание оценивается в 1, 2, 3 или 5 баллов.</p>
МАТЕРИАЛЫ ДЛЯ ПРОВЕДЕНИЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ				
7	ПК-1.1 ПК-1.2	<p>Программа экзамена приведена в разделе VI.</p> <p>Пример вопроса:</p> <p>Информатика, информация, данные. Энтропия и количество информации по Хартли и по Шеннону.</p>	<p>вид: традиционный экзамен</p> <p>способ: устно/письменно</p> <p>результаты: устные ответы и выполненные упражнения</p>	<p>Максимум – 40 б.</p> <p>3 вопроса по 5 баллов каждый и 5 задач по 5 баллов каждая.</p> <p>Критерии оценки:</p> <ul style="list-style-type: none"> - ответ дан полностью и корректно – максимальный балл; - ответ дан полностью, допущены 1-2 ошибки – 4 балла; - ответ дан частично, но корректно – 2 б.

Шкала оценивания выполнения индикаторов:

Индикатор считается выполненным, если либо во время текущей, либо промежуточной аттестации студент набрал как минимум пороговое количество баллов за те виды активности, которые отвечают за данный индикатор. Типовые оценочные материалы с привязкой к отдельным индикаторам приведены в таблице выше.

№	Индикатор	Текущая аттестация		Промежуточная аттестация (экзамен)	
		Порог	Максимум	Порог	Максимум
1	ПК-1.1	22	56	16	40
2	ПК-1.2	24	60	16	40
3	ПК-1.3	8	21	–	–

Шкала и критерии выставления оценок за дисциплину:

Шкала и критерии выставления оценок «отлично», «хорошо», «удовлетворительно» и «неудовлетворительно» описаны в локальной нормативной документации Тверского государственного университета (Положение о рейтинговой системе обучения студентов ТвГУ). Положительная оценка может быть выставлена только в том случае, если выполнены все индикаторы.

V. Учебно-методическое и информационное обеспечение дисциплины

1) Рекомендуемая литература

а) Основная литература:

1. Теоретические основы информатики: учебник / Р.Ю. Царев, А.Н. Пупков, В.В. Самарин и др.; Министерство образования и науки Российской Федерации, Сибирский Федеральный университет. - Красноярск: Сибирский федеральный университет, 2015. - [Электронный ресурс]. - Режим доступа: <http://biblioclub.ru/index.php?page=book&id=435850>
2. Забуга А. А. Теоретические основы информатики. - Новосибирск: НГТУ, 2013. - [Электронный ресурс]. - Режим доступа: <http://biblioclub.ru/index.php?page=book&id=258592>
3. Балюкевич, Э.Л. Основы теории информации: учебно-практическое пособие / Э.Л. Балюкевич. - М.: Евразийский открытый институт, 2008. - 216 с.; То же [Электронный ресурс]. – Режим доступа: <http://biblioclub.ru/index.php?page=book&id=90955>

б) Дополнительная литература:

1. Баранова, Е. К. Основы информатики и защиты информации [Электронный ресурс]: учебное пособие / Е. К. Баранова. - М.: РИОР: ИНФРА-М, 2013. - 183 с. - (Высшее образование: Бакалавриат). - ISBN 978-5-16-006484-0 (ИНФРА-М).-Режим доступа: <http://znanium.com/go.php?id=415501>
2. Горелик, В.А. Пособие по дисциплине «Теоретические основы информатики»: учебное пособие / В.А. Горелик, О.В. Муравьева, О.С. Трембачева; Министерство образования и науки Российской Федерации, Московский педагогический государственный университет. - М.: МПГУ, 2015. - 120 с.: ил. - Библиогр. в кн. - ISBN 978-5-4263-0220-4; [Электронный ресурс]. – Режим доступа: <http://biblioclub.ru/index.php?page=book&id=472092>
3. Грацианова, Т.Ю. Программирование в примерах и задачах [Электронный ресурс]: учеб. пособие — Электрон. дан. — Москва: Издательство "Лаборатория знаний", 2016. — 373 с. — Режим доступа: <https://e.lanbook.com/book/90242>
4. Гуменюк А.С. Прикладная теория информации [Электронный ресурс]: учебное пособие/ Гуменюк А.С., Поздниченко Н.Н.– Омск: Омский государственный технический университет, 2015 .– 189 с.– Режим доступа: <http://www.iprbookshop.ru/58097>
5. Душин В.К. Теоретические основы информационных процессов и систем [Электронный ресурс]: учебник/ Душин В.К.– М.: Дашков и К, 2014.– 348 с.– Режим доступа: <http://www.iprbookshop.ru/24764>
6. Панин В.В. Основы теории информации [Электронный ресурс]: учебное пособие для вузов/ Панин В.В.– М.: БИНОМ. Лаборатория знаний, 2012.– 438 с.– Режим доступа: <http://www.iprbookshop.ru/6521>
7. Бескид П.П. Криптографические методы защиты информации. Часть 1. Основы криптографии [Электронный ресурс]: учебное пособие/ Бескид П.П., Тагарникова Т.М.– СПб.: Российский государственный гидрометеорологический университет, 2010.– 95 с.– Режим доступа: <http://www.iprbookshop.ru/17925>

2) Программное обеспечение

Компьютерный класс №2 факультета ПМиК № 249 (170002, Тверская обл., г.Тверь, Садовый переулок, д.35)	Перечень программного обеспечения (со свободными лицензиями): Adobe Acrobat Reader DC, Google Chrome, Kaspersky Endpoint Security для Windows, ONLYOFFICE Desktop Editors 7.1 (x64), Python 3.10.7, R for Windows 3.6.1, RStudio Desktop, Visual Studio Community 2022, VLC media player, Unreal Commander v3.57x64
--	---

3) Современные профессиональные базы данных и информационные справочные системы

- ЭБС «ZNANIUM.COM» www.znanium.com;
- ЭБС «IPRBooks» <http://www.iprbookshop.ru>;
- ЭБС «Университетская библиотека онлайн» <https://biblioclub.ru>;
- ЭБС «Лань» <http://e.lanbook.com>

4) Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины

- Электронная образовательная среда ТвГУ <http://lms.tversu.ru>
- Научная библиотека ТвГУ <http://library.tversu.ru>

VI. Методические материалы для обучающихся по освоению дисциплины

1. Структура рейтинговых баллов

1-й семестр

Название работы	Кол-во баллов
ТЕКУЩАЯ АТТЕСТАЦИЯ	
Первый модуль	
Расчетно-графическая работа «Расчет информационных характеристик текста»	10
Домашнее задание №2 «Защита информации»	5
Модульная контрольная 1	15
ИТОГО за первый модуль	30
Второй модуль	
Домашнее задание №1 «Graphics Interchange Format»	8
Домашнее задание №3 «Алгоритм RSA»	8
Модульная контрольная 2	14
ИТОГО за второй модуль	30
ПРОМЕЖУТОЧНАЯ АТТЕСТАЦИЯ	
Экзамен	40

2. Самостоятельная работа

2.1 Домашнее задание №1

Порядок выполнения домашней работы:

1. изучите источники из раздела «Литература» (если вы знаете английский язык, предпочтительнее – источник №4) или любые другие, которые сможете найти;
2. ознакомьтесь с двумя он-лайн сервисами из раздела «Инструменты»;
3. выполните два задания:

- Задание 1.1 «Декодирование»: разобрать вручную все поля gif-файла с помощью HEX-редактора и распаковать исходное изображение, сжатое с помощью LZW-алгоритма.
 - Задание 1.2 «Кодирование»: придумать изображение (количество цветов выбрать самостоятельно, минимальный размер – 5x5 пикселей), а затем вручную закодировать его в gif-файл.
4. результаты работы оформите в виде отчета.

Инструменты


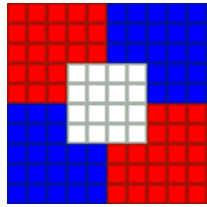
1. [Шестнадцатеричный он-лайн редактор.](#)
2. [Калькулятор RGB-цветов.](#)

Литература

1. [Цветовые модели - RGB и CMYK.](#)
2. [Краткое описание формата GIF.](#)
3. [Сжатие по методу LZW.](#)
4. [Project: What's In A GIF \(англ.\).](#)

Шаблон отчета

Задание 1.1 «Декодирование»

Исходное изображение	
Увеличенное исходное изображение	
Размеры	10x10
Размер несжатых данных	300 байт
Размер сжатых данных (без учета накладных расходов)	22 байта
Коэффициент сжатия	93%

HEX-запись последовательности байт:

47 49 46 38 39 61 0A 00 0A 00 91 00 00 FF FF FF FF 00 00
 00 00 FF 00 00 00 21 F9 04 00 00 00 00 00 2C 00 00 00 00
 0A 00 0A 00 00 02 16 8C 2D 99 87 2A 1C DC 33 A0 02 75 EC
 95 FA A8 DE 60 8C 04 91 4C 01 00 3B

Заголовок

47 49 46 38 39 61

Поле	Назначение	Значение
47 49 46	сигнатура	GIF
38 39 61	версия	89a

Дескриптор логического экрана

0A 00 0A 00 91 00 00

Поле	Назначение	Значение
0A 00	ширина канвы	10
0A 00	высота канвы	10
91	упакованные флаги	1 001 0 001 <ul style="list-style-type: none">• 1 – есть глобальная палитра• 001 – разрешение 64 цвета• 0 – палитра не сортирована по значимости• 001 – размер палитры: 4 цвета, 12 байт
00	цвет фона	0
00	соотношение сторон	0

Глобальная палитра

FF FF FF FF 00 00 00 00 00 FF 00 00 00

Номер цвета	Цвет
0	FF FF FF
1	FF 00 00
2	00 00 AA
3	00 00 00

Блок-расширение управления графикой

21 F9 04 00 00 00 00 00

Поле	Назначение	Значение
21	маркер начала блока	!
F9	тип блока	· (управление графикой)
04	размер блока	4
00	упакованные флаги	000 000 0 0 <ul style="list-style-type: none"> • 000 – зарезервировано • 000 – способ замены “на усмотрение браузера” • 0 – реакция пользователя не ожидается • 0 – прозрачности нет
00 00	время задержки	0
00	номер прозрачного цвета	0
00	терминатор блока	0

Дескриптор изображения

2C 00 00 00 00 0A 00 0A 00 00

Поле	Назначение	Значение
2C	разделитель	,
00 00	отступ слева от границы логического экрана	0
00 00	отступ сверху от границы логического экрана	0
0A 00	ширина картинка	10
0A 00	высота картинка	10
00	упакованные флаги	0 0 0 00 000 <ul style="list-style-type: none"> • 0 – нет локальной палитры • 0 – построчная развертка • 0 – не исп., т.к. нет локальной палитры • 00 – зарезервировано • 000 – не исп., т.к. нет локальной палитры

Графический блок

02 16 8C 2D 99 87 2A 1C DC 33 A0 02 75 EC 95 FA A8 DE 60
8C 04 91 4C 01 00

Поле	Назначение	Значение
02	стартовый размер кода	2 бита
16	размер под-блока	22 байта
8C ... 01	данные	сжатая последовательность
00	маркер конца блока	0

Последовательность бит сжатых данных:

10001100 00101101 10011001 10000111 00101010 00011100
11011100 00110011 10100000 00000010 01110101 11101100
10010101 11111010 10101000 11011110 01100000 10001100
00000100 10010001 01001100 00000001

Транспонированная последовательность бит/кодов:

0000 000101 001100 100100 010000 010010 001100 011000
001101 11101 01010 00111 11010 10010 10111 10110 00111
01010 00000 10101 00000 00110 01111 01110 00001 1100
0010 1010 1000 0111 1001 1001 0010 110 110 001 100

Последовательность кодов:

4 1 6 6 2 9 9 7 8 10 2 12 1 14 15 6 0 21 0 10 7 22 23
18 26 7 10 29 13 24 12 18 16 36 12 5

Словарь		Декодированная последовательность
Код	Значение	
#0	0	1 1 1 1 1 2 2 2 2 2
#1	1	1 1 1 1 1 2 2 2 2 2
#2	2	1 1 1 1 1 2 2 2 2 2
#3	3	1 1 1 0 0 0 0 2 2 2
#4	маркер очистки словаря	1 1 1 0 0 0 0 2 2 2
#5	маркер конца данных	2 2 2 0 0 0 0 1 1 1
#6	1, 1	2 2 2 0 0 0 0 1 1 1
#7	1, 1, 1	2 2 2 2 2 1 1 1 1 1
#8	1, 1, 2	2 2 2 2 2 1 1 1 1 1
#9	2, 2	2 2 2 2 2 1 1 1 1 1
#10	2, 2, 2	
#11	2, 2, 1	
#12	1, 1, 1, 1	
#13	1, 1, 2, 2	
#14	2, 2, 2, 2	

#15	2, 1
#16	1, 1, 1, 1, 1
#17	1, 2
#18	2, 2, 2, 2, 2
#19	2, 1, 1
#20	1, 1, 0
#21	0, 0
#22	0, 0, 0
#23	0, 2
#24	2, 2, 2, 1
#25	1, 1, 1, 0
#26	0, 0, 0, 0
#27	0, 2, 2
#28	2, 2, 2, 2, 2, 0
#29	0, 0, 0, 0, 1
#30	1, 1, 1, 2
#31	2, 2, 2, 0
#32	0, 0, 0, 0, 1, 1
#33	1, 1, 2, 2, 2
#34	2, 2, 2, 1, 1
#35	1, 1, 1, 1, 2
#36	2, 2, 2, 2, 2, 1
#37	1, 1, 1, 1, 1, 2
#38	2, 2, 2, 2, 2, 1, 1

Маркер конца файла

3B

Поле	Назначение	Значение
3B	Маркер конца файла	;

Задание 1.2 «Кодирование»

Эскиз исходного изображения	
Размеры	5x5
Размер несжатых данных	25 байт

Размер сжатых данных (без учета накладных расходов)	...
Коэффициент сжатия	...

(оформить по образцу задания 1.1)

2.2 Домашнее задание №2

Порядок выполнения домашней работы:

1. ознакомьтесь с алгоритмом шифрования, который вам достался (**все** алгоритмы имеют подробное описание на русском языке на Википедии);
2. разработайте криптографическое приложение, содержащее следующую систему классов (описаны только интерфейсы классов, реализацию придумать самим):
 - **класс Ключ** - вычисляет (в зависимости от потребностей алгоритма) и хранит ключ шифрования:
 - конструктор по-умолчанию `__init__(self)` - создает пустой ключ,
 - конструктор с параметром `__init__(self, value)` - создает готовый ключ,
 - метод `set(self, value)` - установка/изменение ключа,
 - перегруженный метод `__str__(self)` - вывод ключа на печать,
 - **класс Ключ[НазваниеШифра]**, наследуемый от класса Ключ - переопределяет методы базового класса для работы с конкретным алгоритмом шифрования,
 - **класс КриптоСистема** - производит всю работу по шифрованию и дешифрованию сообщений:
 - конструктор по-умолчанию `__init__(self)` - создает пустую криптосистему,
 - конструктор с параметром класса Ключ `__init__(self, key)` - создает готовую криптосистему с ключом,
 - метод `setKey(self, key)` - установка/изменение ключа,
 - метод `encrypt(self, string)` - шифрование строки `string`,
 - метод `decrypt(self, string)` - дешифрование строки `string`,
 - **класс Криптосистема[НазваниеШифра]**, наследуемый от класса Криптосистема - переопределяет методы базового класса для работы с конкретным алгоритмом шифрования,
 - **класс КриптоМенеджер** - класс-контейнер или класс-оболочка, использующий конкретный объект криптосистемы для шифрования и дешифрования данных из разных источников:
 - конструктор по-умолчанию `__init__(self)` - создает пустой крипто-менеджер,

- конструктор с параметром класса Криптосистема `__init__(self, crypto)` - создает готовый к работе крипто-менеджер,
 - метод `setCrypto(self, crypto)` - установка/изменение крипто-системы,
 - метод `encryptFile(self, inputFileName, outputFileName)` - шифрование данных из файла с именем `inputFileName` с помещением результата в файл с именем `outputFileName`,
 - метод `decryptFile(self, inputFileName, outputFileName)` - дешифрование данных из файла с именем `inputFileName` с помещением результата в файл с именем `outputFileName`,
 - метод `encryptString(self, string)` - шифрование строки `string`,
 - метод `decryptString(self, string)` - дешифрование строки `string`.
- Пример фрагмента кода, демонстрирующего использование классов, для шифров Виженера и Цезаря:


```

o cm = CryptoManager()
o k = KeyVigenere()
o k.set("Secret password")
o cm.setCrypto(CryptoVigenere(k))
o cm.encryptFile("input.txt", "input-vigenere.txt")
o cm.decryptFile("input-vigenere.txt", "input-vigenere-output.txt")
o
o # Меняем криптосистему на шифр Цезаря
o crypto = CryptoCaesar()
o crypto.setKey(KeyCaesar(4))
o cm.setCrypto(crypto)
o eStr = cm.encryptString("Some string")
o dStr = cm.decryptString(eStr)
o print("Encrypted string: ", eStr)
o print("Decrypted string: ", dStr)

```

3. Зашифруйте и расшифруйте текстовый файл, с которым вы работали в РГР.
4. Результаты работы оформите в виде отчета в произвольной форме. Обязательные пункты отчета:
 - описание **алгоритма** шифрования и дешифрования на русском языке,
 - результат работы алгоритма на небольшом фрагменте текста (одно предложение).

2.3 Домашнее задание №3

В таблице ниже приведены открытый ключ и зашифрованное текстовое сообщение, которое необходимо расшифровать методом подбора параметров RSA.

Порядок выполнения домашней работы:

1. напишите программу, которая будет раскладывать число n из данного вам открытого ключа на простые множители p и q (в зависимости от мощности компьютера и написанного кода подбор может осуществляться от 2 до 12 часов);
2. после того, как p и q будут подобраны, найдите n и $\phi(n)$;
3. используя число e из данного вам открытого ключа и найденное $\phi(n)$, найдите закрытый ключ d ;
4. расшифруйте данный вам шифротекст с помощью найденного закрытого ключа;
5. результаты работы оформите в виде отчета.

Данные для выполнения домашней работы

№	ФИО	Публичный ключ	Зашифрованный текст
1		(65537, 1780531566236248668433)	[1223326826427156564127, 906118110307211134973, 581901781744724142165, 380324793728421578921, 1373109080413466323755]

Шаблон отчета

Объект исследования

Полученный публичный ключ:

(65537, 2081179901116607342141)

Полученный зашифрованный текст, упакованный в 64-битные числа:

[185285271794656132703,
446880134334342609101,
770812668851906545623,
835918421902467517605,
537242871735767354588]

Результаты декодирования

Приватный ключ	1462735685201091269057
Числа p и q	48240376309, 43141867049
Число $\phi(n)$	2081179901025225098784
Расшифрованный текст, упакованный в 64-битные числа	7161126247200745061, 8247902303536838003, 7522544236193215336,

	7597121167947753839, 7235448851096674336
Распакованный текст	canal fervor pusher mush- iness iodize

Программный код

Код для подбора закрытого ключа

```
# ...
```

Код для декодирования текста

```
# ...
```

Общие процедуры упаковки и распаковки символов

Получает на вход строку и возвращает список 64-битных чисел

```
def packString(s):
    padding = 8 - len(s) % 8
    if padding > 0: s += (' ') * padding
    slist = list(map(ord, [letter for letter in s]))
    chunks = len(slist) // 8
    res = []
    for i in range(chunks):
        res.append(reduce(lambda x, y: x*256+y, slist[i*8:i*8+8]))
    return res
```

Получает на вход список 64-битных чисел и возвращает строку

```
def unpackString(s):
    chunks = len(s)
    masks = [8*i for i in range(7, -1, -1)]
    result = []
    sres = ""
    for i in range(chunks):
        slst = list(map(lambda x: (s[i] >> x) % 256, masks))
        sres += "".join(list(map(chr, slst)))
    return sres
```

3. Расчетно-графическая работа

Порядок выполнения РГР:

1. выберите любое произведение размером не менее 100 Кб из библиотеки Мошкова (см. раздел «Литература») или любого другого источника;
2. выполните расчет информационных характеристик и оформите результат в виде отчета.

Особые условия

- Задание выполняется **индивидуально**.
- Все тексты должны быть **различными**.
- При выполнении РГР **обязательно** написание программы на языке Python, вычисляющей основные информационные характеристики выбранного

текста: таблицы частот, энтропии по Хартли и Шеннону, количества информации по Хартли и Шеннону. Отсутствие программного кода в РГР ведет к снижению количества баллов на 5.

- Оптимальные коды и остальные характеристики могут рассчитываться как вручную, так и с помощью самостоятельно написанных программ. Это не приводит ни к снижению, ни к повышению баллов. Программный код включать не обязательно.
- Частоты символов указываются с точностью до седьмого знака после запятой.

Литература

1. [Библиотека Максима Мошкова \(Ссылки на внешний сайт.\)](#)[Ссылки на внешний сайт.](#)

Шаблон отчета

Объект исследования

Для выполнения расчетно-графической работы было выбрано литературное произведение Адамса Дугласа «Детективное агентство Дирка Джентли».

Ссылка на исходный текст: http://lib.ru/ADAMS/gently_1.txt

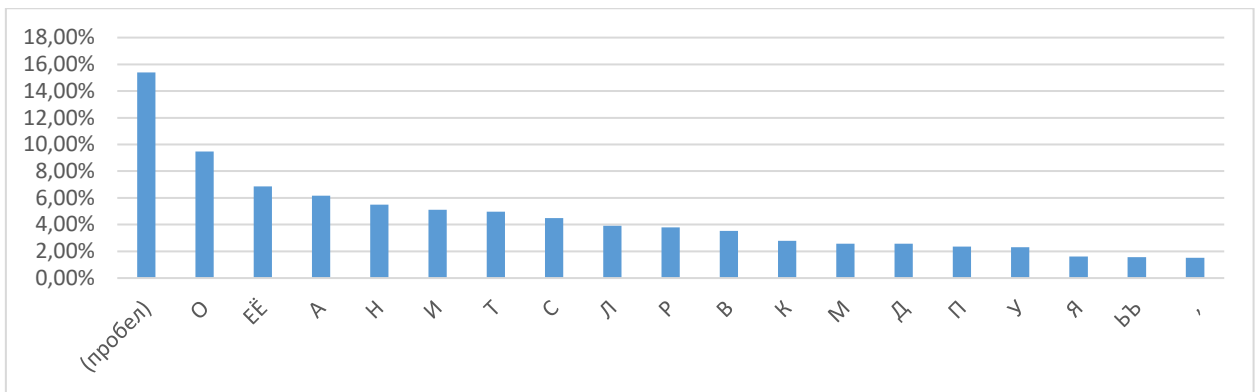
Информационные характеристики

Частотная характеристика текста:

№	Символ	Количество	Вероятность	%
1	(пробел)	82902	0,1539184	15,39%
2	О	51012	0,0947105	9,47%
3	ЕЁ	36961	0,0686229	6,86%
4	А	33252	0,0617367	6,17%
5	Н	29630	0,0550120	5,50%
6	И	27535	0,0511223	5,11%
7	Т	26706	0,0495832	4,96%
8	С	24132	0,0448042	4,48%
9	Л	21121	0,0392139	3,92%
10	Р	20422	0,0379161	3,79%
11	В	19026	0,0353243	3,53%
12	К	14968	0,0277901	2,78%
13	М	13877	0,0257645	2,58%
14	Д	13833	0,0256828	2,57%
15	П	12742	0,0236572	2,37%
16	У	12393	0,0230092	2,30%

17	Я	8728	0,0162047	1,62%
18	ЬЪ	8378	0,0155549	1,56%
19	,	8186	0,0151984	1,52%
20	Ы	7855	0,0145838	1,46%
21	З	7506	0,0139359	1,39%
22	Г	7418	0,0137725	1,38%
23	Ч	7418	0,0137725	1,38%
24	Б	6895	0,0128015	1,28%
25	-	6228	0,0115631	1,16%
26	Й	4756	0,0088301	0,88%
27	Ж	4276	0,0079390	0,79%
28	Ш	3753	0,0069679	0,70%
29	-	3322	0,0061677	0,62%
30	Х	3229	0,0059951	0,60%
31	Ю	2618	0,0048607	0,49%
32	Э	2007	0,0037263	0,37%
33	Ц	1396	0,0025919	0,26%
34	Щ	1353	0,0025120	0,25%
35	Ф	1222	0,0022688	0,23%
36	?	655	0,0012161	0,12%
37	“	426	0,0007909	0,08%
38	:	155	0,0002878	0,03%
39	!	141	0,0002618	0,03%
40	1	30	0,0000557	0,01%
41	2	27	0,0000501	0,01%
42	3	26	0,0000483	0,00%
43	0	16	0,0000297	0,00%
44	7	15	0,0000278	0,00%
45	6	13	0,0000241	0,00%
46	4	12	0,0000223	0,00%
47	8	11	0,0000204	0,00%
48	;	10	0,0000186	0,00%
49	5	10	0,0000186	0,00%
50	9	7	0,0000130	0,00%

Гистограмма распределения самых частых символов (не менее 2%):



Основные показатели:

Мощность алфавита	50 символов
Размер текста	538 610 символов
Размер текста в битах (N)	4 308 880 бит
Энтропия по Хартли	5.643856
Количество информации по Хартли	3 039 837 бит
Энтропия по Шеннону (I_A)	4.516293
Количество информации по Шеннону	2 432 521 бит
Минимальная длина кода ($K_{min} = \frac{I_A}{I_B} = \frac{I_A}{1} = I_A$)	4.516293

Оптимальный равномерный код

№	Символ	Код	№	Символ	Код	№	Символ	Код
1	(пробел)	000000	18	ЪЪ	010001	35	Ф	100010
2	О	000001	19	,	010010	36	?	100011
3	ЕЁ	000010	20	Ы	010011	37	“	100100
4	А	000011	21	З	010100	38	:	100101
5	Н	000100	22	Г	010101	39	!	100110
6	И	000101	23	Ч	010110	40	1	100111
7	Т	000110	24	Б	010111	41	2	101000
8	С	000111	25	-	011000	42	3	101001
9	Л	001000	26	Й	011001	43	0	101010
10	Р	001001	27	Ж	011010	44	7	101011
11	В	001010	28	Ш	011011	45	6	101100
12	К	001011	29	-	011100	46	4	101101
13	М	001100	30	Х	011101	47	8	101110
14	Д	001101	31	Ю	011110	48	;	101111
15	П	001110	32	Э	011111	49	5	110000
16	У	001111	33	Ц	100000	50	9	110001
17	Я	010000	34	Щ	100001			

Основные показатели:

Длина оптимального равномерного кода (K_1)	6 бит
Размер закодированного текста (N_1)	3 231 660 бита
Относительная избыточность кода ($Q = \frac{K_1 - K_{min}}{K_{min}}$)	0,3285
Коэффициент сжатия ($\frac{N_1}{N} * 100$)	75%

Оптимальный неравномерный префиксный код

Для построения оптимального префиксного кода был выбран алгоритм Хаффмана.

№	Символ	Код	№	Символ	Код	№	Символ	Код
1	(пробел)	110	18	БЪ	100010	35	Ф	101110110
2	О	000	19	,	100001	36	?	101111110
3	ЕЁ	1010	20	Ы	100000	37	“	1011111110
4	А	1001	21	Э	011101	38	:	101111111111
5	Н	0110	22	Ч	010111	39	!	101111111110
6	И	0100	23	Г	011100	40	1	1011111111000
7	Т	0010	24	Б	001101	41	2	10111111110110
8	С	11110	25	Й	1011110	42	3	10111111110101
9	Л	11101	26	Ж	1011100	43	0	10111111110010
10	Р	11100	27	Ш	0101101	44	7	101111111101111
11	В	10110	28	-	0011001	45	6	101111111101110
12	К	01111	29	-	0101100	46	4	101111111101001
13	М	01010	30	Х	0011000	47	8	101111111101000
14	Д	00111	31	Ю	10111110	48	;	101111111100111
15	П	111111	32	Э	10111010	49	5	1011111111001101
16	У	111110	33	Ц	101111110	50	9	1011111111001100
17	Я	100011	34	Щ	101110111			

Основные показатели:

Средняя длина оптимального префиксного кода $\left(K_2 = \sum_{i=1}^N n_i p_i \right)$	4,5479623 бита
Размер закодированного текста (N_2)	2 449 578 бита
Относительная избыточность кода ($Q = \frac{K_2 - K_{min}}{K_{min}}$)	0,007
Коэффициент сжатия ($\frac{N_2}{N} * 100$)	56,8%

Сжатие словарным методом

Размер файла до сжатия	538 610 байт
Размер файла после сжатия методом LZW (zip-архив)	156 786 байт
Коэффициент сжатия	29.1%

Программный код

...

4. Примеры модульных контрольных работ

4.1 Пример первой модульной контрольной

Пожалуйста, впишите только ответы. Ход решения записывать не надо.

1. Укажите бит, где произошла ошибка в коде Хэмминга: **100010001010001** (2 балла)
2. Используя полином-генератор: $x^3 + x + 1$, построить CRC-4 код для сообщения: **1101111100** (2 балла)
3. Для кодирующей матрицы:

00011

01101

выписать (5, 2)-код и основные характеристики полученных кодов: минимальное расстояние между словами кода; максимальную кратность ошибок, до которой включительно они все исправляются или обнаруживаются (2 балла)

4. Основываясь на единственном сообщении «**sesenntnsn**», запишите: приблизительную энтропию источника и количество информации в сообщении по Хартли (1 балл)
5. Запишите префиксный код, построенный по сообщению из упр. 4 методом Хаффмана. Выпишите получившееся в ходе работы алгоритма дерево. (2 балла)
6. Пусть дано сообщение **BAAA**, полученное от д.с.в. X со следующим распределением вероятностей: $P(A) = 1/5$, $P(B) = 1/4$, $P(C) = 11/20$. Выпишите его арифметическое
7. Закодировать последовательность **adaacababaaadac** алгоритмом LZ77. Размер словаря и буфера принять равным 16. (3 балла)

4.2 Пример второй модульной контрольной

Пожалуйста, впишите только ответы. Ход решения записывать не надо.

1. Вычислите (2 балла):

$$\begin{array}{l} 27 \bmod -9 \quad \boxed{} \quad 37 \bmod 5 \quad \boxed{} \\ -37 \bmod 10 \quad \boxed{} \quad 29 \bmod 4 \quad \boxed{} \end{array}$$

2. Вычислите (2 балла):

$$(117+263) * (838+497) \text{ в } Z_7 \quad (189-548) * (921-905) \text{ в } Z_5$$

3. Рассчитать обратное значение (2 балла):

$$68 \pmod{24} \quad 29 \pmod{21}$$

4. Распишите шаги расширенного алгоритма Евклида для Упр. 3 (2 балла):

5. Вычислить (2 балла): 7^{2160} в Z_{10}

6. Запишите разложение на простые множители и вычислите по формуле $\phi(1067)$ (2 балла):

7. Пользователь системы RSA, выбравший $p = 13$, $q = 7$ и $e = 25$, получил зашифрованное сообщение $y = 10$. Написать, чему равен закрытый ключ, и дешифровать y (3 балла).

Ответ: $d =$, $\text{dec}(y) =$

8. У пользователя системы RSA есть открытая ключевая пара $(65, 23)$. Зашифровать с ее помощью сообщение $x = 9$, а также взломать код и расшифровать сообщение $y = 28$ (5 баллов).

Ответ: $p =$, $q =$, $\phi =$, $d =$, $\text{enc}(x) =$, $\text{dec}(y) =$

5. Программа экзамена

5.1 Определения

- информация
- данные
- энтропия
- код, длина кода, основание кода
- обратимость кодирования (обратимое, принципиально необратимое, обратимое с помощью дополнительной информации)
- формула средней длины кода
- формула минимальной длины кода
- формула относительной избыточности кода
- префиксный код
- условие Фано
- помехоустойчивый код

- обнаруживающий и корректирующий коды
- информационные и проверочные символы кода
- блоковые и непрерывные коды
- делимые и неделимые коды
- расстояние Хэмминга
- (n, m) -код
- криптология, криптография, криптоанализ
- симметричное шифрование
- асимметричное шифрование
- моноалфавитные и полиалфавитные шифры
- принцип Керкгоффса
- остаток от деления a на m
- класс эквивалентности по модулю m
- целочисленное кольцо (поле) вычетов по модулю m
- результат работы алгоритма Евклида (расширенного алгоритма Евклида)
- уравнение Безу, коэффициенты Безу
- функция Эйлера
- лавинный эффект
- сеть Фейстеля

5.2 Вопросы

1. Информатика, информация, данные. Энтропия и количество информации по Хартли и по Шеннону.
2. Теория кодирования. Задача кодирования, код, кодирование/декодирование, кодировщик/декодировщик, длина и основание кода. Обратимость кодирования. Вывод формулы средней длины кода, вывод формулы минимальной длины кода. Относительная избыточность кода.
3. Асимптотически оптимальный код. Первая теорема Шеннона. Варианты построения оптимального кода. Теорема Шеннона для двоичного кодирования.
4. Коды переменной и фиксированной длины. Префиксный код. Условие Фано. Алгоритм Шеннона-Фано.
5. Коды переменной и фиксированной длины. Префиксный код. Условие Фано. Алгоритм Хаффмана.
6. Коды переменной и фиксированной длины. Префиксный код. Арифметическое кодирование.
7. Алгоритм сжатия LZ77.
8. Алгоритм сжатия LZ78.
9. Помехоустойчивое кодирование, корректирующие и обнаруживающие коды, информационные/избыточные символы, делимые и неделимые коды. Расстояние Хэмминга. Теоремы об обнаруживающем и корректирующем кодах. (n, m) -коды, линейный код. Построение кода Хэмминга, обнаружение и исправление ошибки в коде Хэмминга.
10. Помехоустойчивое кодирование, корректирующие и обнаруживающие коды, информационные/избыточные символы, делимые и неделимые коды. Расстояние Хэмминга. Теоремы об обнаруживающем и корректирующем кодах. (n, m) -коды, линейный код. Двоичный полином. CRC- n код.
11. Моноалфавитные и полиалфавитные шифры. Шифр Цезаря и шифр Виженера. Шифр из ДЗ №2.

12. Простой и расширенный алгоритмы Евклида. Уравнение Безу. Использование в модульной арифметике.
13. Остаток от деления, эквивалентность по модулю, классы эквивалентности, модульная арифметика, кольца и поля вычетов по модулю m . Быстрое возведение в степень.
14. Функция Эйлера. Теорема для вычисления функции Эйлера. Малая теорема Ферма, теорема Эйлера.
15. Ассиметричное шифрование. Алгоритм RSA. Генерация ключей, шифрование, дешифрование. Доказательство работы алгоритма.
16. Симметричное шифрование. Основные характеристики алгоритма DES. Операция XOR. Сеть Фейстеля. Шифрация и дешифрация сетью Фейстеля.

5.3 Письменные упражнения

- первая модульная контрольная:
 - упр. 1 или 2 или 3,
 - упр. 5 или 6,
 - упр. 7,
- вторая модульная контрольная:
 - упр. 4 или 5,
 - упр. 7 или 8.

6. Указания для обучающихся

Организуя свою учебную работу, студенты должны, во-первых, выявить рекомендуемый режим и характер учебной работы по изучению теоретического курса, практическому применению изученного материала, по выполнению заданий для самостоятельной работы, по использованию информационных технологий и т.д. Во-вторых, ознакомиться с указанным в методическом материале по дисциплине перечнем учебно-методических изданий, рекомендуемых студентам для подготовки к занятиям и выполнения самостоятельной работы, а также с методическими материалами на бумажных и/или электронных носителях, выпущенных кафедрой своими силами и предоставляемые студентам во время занятий.

Самостоятельная работа студентов, предусмотренная учебным планом, должна соответствовать более глубокому усвоению изучаемого курса, формировать навыки исследовательской работы и ориентировать студентов на умение применять теоретические знания на практике.

1. Работа с учебными пособиями.

Для полноценного усвоения курса студент должен, прежде всего, овладеть основными понятиями этой дисциплины. Необходимо усвоить определения и понятия, уметь приводить их точные формулировки, приводить примеры объектов, удовлетворяющих этому определению. Кроме того, необходимо знать круг фактов, связанных с данным понятием. Требуется также знать связи между понятиями, уметь устанавливать соотношения между классами объектов, описываемых различными понятиями.

2. Самостоятельное изучение тем.

Самостоятельная работа студента является важным видом деятельности, позволяющим хорошо усвоить изучаемый предмет и одним из условий достижения необходимого качества подготовки и профессиональной переподготовки специалистов. Она предполагает самостоятельное изучение студентом рекомендованной учебно-методической литературы, различных справочных материалов, написание рефератов, выступление с докладом, подготовку к лекционным и практическим занятиям, подготовку к зачёту и экзамену.

3. Подготовка к практическим занятиям.

При подготовке к практическим занятиям студентам рекомендуется следовать методическим рекомендациям по работе с учебными пособиями, приведенным выше.

4. Составление конспектов.

В конспекте отражены основные понятия темы. Для наглядности и удобства запоминания используются схемы и таблицы.

VII. Материально-техническое обеспечение

Для аудиторной работы

Учебная аудитория № 212 (170002, Тверская обл., г.Тверь, Садовый переулок, д.35)	Набор учебной мебели, мультимедийный комплекс (доска, проектор, панель управления, переносной ноутбук).
--	---

Для самостоятельной работы

Помещение для самостоятельной работы обучающихся: Компьютерный класс факультета прикладной математики и кибернетики № 46 (170002, Тверская обл., г.Тверь, Садовый переулок, д.35)	Компьютер, экран, проектор, кондиционер.
--	--

VIII. Сведения об обновлении рабочей программы дисциплины

№ п.п.	Обновленный раздел рабочей программы дисциплины	Описание внесенных изменений	Дата и протокол заседания кафедры, утвердившего изменения
1.	3. Объем дисциплины	Выделение часов на практическую подготовку	От 29.10.2020 года, протокол № 3 ученого совета факультета

2.	II. Содержание дисциплины, структурированное по темам (разделам) с указанием отведенного на них количества академических часов и видов учебных занятий	Выделение часов на практическую подготовку по темам	От 29.10.2020 года, протокол № 3 ученого совета факультета
3.	V. Учебно-методическое и информационное обеспечение дисциплины 2) Программное обеспечение	Внесены изменения в программное обеспечение	От 29.09.2022 года, протокол № 2 ученого совета факультета
4.	VII. Материально-техническое обеспечение	Внесены изменения в материально-техническое обеспечение аудиторий	От 29.09.2022 года, протокол № 2 ученого совета факультета
5.			