МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РФ

Федеральное государственное бюджетное образовательное учреждение высшего образования

«ДАГЕСТАНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

Факультет **Информатики и информационных технологий**

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

УПРАВЛЕНИЕ ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТЬЮ

Кафедра информатики и информационных технологий

Образовательная программа

09.03.02 «Информационные системы и технологии»

Профиль подготовки
Общий профиль

Уровень высшего образования
Бакалавриат

Форма обучения
очная

Статус дисциплины: **вариативная обязательная**

Махачкала, 2018

Рабочая программа дисциплины «Управление информационной безопасностью» составлена в 2018 году в соответствии с требованиями ФГОС ВО по направлению подготовки 09.03.02 «Информационные системы и технологии», профиль подготовки «Общий профиль» (уровень бакалавриат), утвержденного приказом Минобрнауки РФ от 12 марта 2015 г. № 219_, вступил в силу 30 марта 2015 г.

Разработчик: Абдуллаев Габид Шаванович, к.э.н., доцент кафедры информатики и информационных технологий

Рабочая программа дисциплины одобрена:

На заседании кафедры ИиИТ от «2» 07 2018г., протокол № 12

Зав. Кафедрой _____ Ахмедов С.А.

(подпись)

На заседании Методической комиссии факультета ИиИТ

от «3» июня 2018г., протокол № 10.

Председатель _____ Камилов К.Б.

(подпись)

Рабочая программа дисциплины согласована с учебно – методическим управлением «04» 07 2018г. _____

(подпись)

Аннотация рабочей программы дисциплины

Дисциплина «Управление информационной безопасностью» является дисциплиной по выбору образовательной программы бакалавриата по направлению 09.03.02 «Информационные системы и технологии».

Дисциплина реализуется на факультете Информатики и ИТ кафедрой Информатики и ИТ.

Содержание дисциплины охватывает круг вопросов, связанных с изучением современных подходов, методов и методик создания системы управления информационной безопасностью предприятий и организаций, приобретением практических навыков по разработке надежной системы управления информационной безопасности.

Дисциплина нацелена на формирование следующих компетенций выпускника: профессиональных – ПК22, ПК - 24, ПК - 34, ПК - 35, ПК - 36, ПК - 37.

Преподавание дисциплины предусматривает проведение следующих видов учебных занятий: *лекции, практические занятия, лабораторные занятия, самостоятельная работа.*

Рабочая программа дисциплины предусматривает проведение следующих видов контроля успеваемости в форме *контрольной работы или тестирования* и промежуточный контроль в форме зачета.

Объем дисциплины 3 зачетных единиц, в том числе в академических часах по видам учебных занятий

Семестр		Учебные занятия							Форма промежуточной аттестации (зачет, дифференцированный зачет, экзамен
		в том числе							
	Всего	Контактная работа обучающихся с преподавателем						СРС, в том числе экзамен	
		Всего	из них						
			Лекции	Лабораторные занятия	Практические занятия	КСР	консультации		
7	108	50	16	18	16	2	2	54	зачет

1. Цели освоения дисциплины

Целями изучения дисциплины «Управление информационной безопасностью» является:

формирование навыков организации и методологии обеспечения информационной безопасности в коммерческих организациях и организациях банковской системы РФ;

создание представления о функциях, структурах и штатах подразделения информационной безопасности; об организационных основах, принципах, методах и технологиях и управлении информационной безопасностью в коммерческих организациях и организациях банковской системы РФ;

развитие способностей по использованию существующей системы управления информационной безопасности.

2. Место дисциплины в структуре ОПОП бакалавлиата

Дисциплина «Управление информационной безопасностью» входит в блок дисциплин по выбору образовательной программы бакалавриата по направлению 09.03.02 «Информационные системы и технологии».

Изучение дисциплины базируется на знаниях, полученных студентами при изучении дисциплин «Надежность информационных систем», «Информационная безопасность и защита информации»,

«Методы и средства проектирования информационных систем и технологий». Изучение дисциплины позволяет овладеть как теоретической базой, так и конкретными практическими навыками по организации и управлению информационной безопасностью.

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

Дисциплина «Управление информационной безопасностью» обеспечивает инструментарий формирования следующих профессиональных (ПК) компетенций:

Код компетенции из ФГОС ВО	Наименование компетенции из ФГОС ВО	Планируемые результаты обучения
ПК-22	способностью проводить сбор, анализ научно-технической информации, отечественного и зарубежного опыта по тематике исследования	Знает: - элементарные логические методы и приемы научного исследования; - основные методологические теории и принципы современной науки. Умеет: - осуществлять сбор и анализ научно-технической информации, полученной из отечественных и зарубежных источников и литературы, в том числе посвященных информационным системам и технологиям Владеет: - элементарными методами научного поиска и интеллектуального анализа научной информации при решении новых задач; - основными методами сбора, обработки и анализа научно-технической информации, полученной из отечественных и зарубежных источников и литературы, посвященных информационным системам и технологиям;
ПК-24	способностью обосновывать правильность выбранной модели, сопоставляя результаты экспериментальных данных и полученных решений	Знает: - способы обработки, анализа и сравнения результатов, полученных в ходе теоретического расчёта, а также численного и натурного моделирования с помощью стандартных научных пакетов; - методы анализа, теоретического и экспериментального исследования для разработки, проектирования, эксплуатации и поддержки информационных систем. Умеет: - проводить оценку границ применимости физических моделей; - корректно подходить к решению проблемы выбора аналитической и численной моделей, организации вычислительного эксперимента; - оценить достоинства, недостатки и рамки применимости того или иного метода на практике; Владеет: - навыками подбора адекватных методов для составления математических моделей физических явлении и их решения; - методами корректной компьютерной обработки и последующего анализа результатов математического моделирования;
ПК-34	способностью к инсталляции, отладке программных и настройке технических средств для ввода	Знает: структуру программного и технического обеспечения, их основные функции и характеристики, методы инсталляции, отладку программных и

	информационных систем в опытную и промышленную эксплуатацию	настройку технических средств, механизмы администрирования, тенденции их развития (управление распределением памяти для объектов ИС, установление квот памяти для пользователей ИС, управления доступностью данных, включая режимы (состояния)). Умеет: выполнять процедуры настройки технических средств информационных систем. Владеет: средствами и средой программирования, современной технологиями программирования, методами настройки и отладки осуществления перехода от управления функционированием отдельных устройств к анализу трафика в отдельных участках сети.
ПК-35	способностью проводить сборку информационной системы из готовых компонентов	Знает: структуру, состав и свойства информационных процессов, систем и технологий, методы анализа информационных систем, модели представления проектных решений, конфигурации информационных систем; структуру, принципы реализации и функционирования информационных технологий, используемых при создании информационных систем, базовые и прикладные информационные технологии, инструментальные средства информационных технологий, состав и свойств готовых компонентов, принципы их адаптации. Умеет: использовать архитектурные и детализированные решения при проектировании систем; применять готовые компоненты информационные технологии и систем при проектировании информационных систем. Владеет: средствами разработки архитектуры информационных систем на основе готовых компонентов; технологиями адаптации типовых проектных решений.
ПК-36	способностью применять основные приемы и законы создания и чтения чертежей и документации по аппаратным и программным компонентам информационных систем	Знает: основные приемы и законы создания и чтения документации по компонентам информационных систем Умеет; применять основные приемы и законы создания и чтения документации по компонентам информационных систем Владеет: практическими навыками применения основных приемов и законов создания и чтения чертежей и документации по программным компонентам информационных систем
ПК-37	способностью выбирать и оценивать способ реализации информационных систем и устройств (программно-, аппаратно- или программно-аппаратно-) для решения поставленной задачи	Знает: технологии выбора и оценки способов реализации ИС Умеет: выбирать и оценивать существующие технологии разработки ИС для решения поставленной задачи Владеет: практическими навыками выбора и оценки современных технологий проектирования и разработки ИС для решения поставленной задачи

4. Объем, структура и содержание дисциплины.

4.1. Объем дисциплины составляет 3 зачетных единиц, 108 академических часов.

4.2. Структура дисциплины.

№ п/п	Разделы и темы дисциплины	Семестр	Неделя семестра	Виды учебной работы, включая самостоятельную работу студентов и трудоемкость (в часах)				Самостоятельная работа	Формы текущего контроля успеваемости *(по неделям семестра)* Форма промежуточной аттестации *(по семестрам)*
				Лекции	Практические занятия	Лабораторные занятия	Контроль самост. раб.		
	Модуль 1. *Введение в управление информационной безопасностью предприятия*								
1	Основы построения систем обеспечения информационной безопасности на предприятии.	7	1	2	2	2		8	
2	Обеспечение информационной безопасности бизнеса.	7	2-3	2	2	2		10	Выполнение и защита лаборат. работ 1-2
3	Система управления информационной безопасностью бизнеса.	7	4-5	2	2	2			
	Итого по модулю 1:			6	6	6		18	Тестирование по мод
	Модуль 2. Анализ и оценка у*правления информационными рисками*								
4	Анализ и оценка управленческих и экономических показателей системы управления информационной безопасностью бизнеса.	7	6-7	2	2	2		8	Выполнение и защита лаборат. работ 3-6
5	Управление жизненным циклом информационных активов. Анализ влияния состояния информационных активов на деятельность организации.	7	8-9	2	2	2		8	
6	Методы управления информационными рисками. Анализ влияния	7	10-11	2	2	2		8	

	информационного риска на деятельность организации.								
	Итого по модулю 2:			6	6	6		24	Тестирование по мод
	Модуль 3. *Аудит системы управления ИБ организации*								
7	Планирование деятельности по обработке рисков обеспечения информационной безопасности организации.	7	12-13	2	2	2		8	Выполнение и защита лаборат. работ 7-9
8	Аудит методов и средств обеспечения информационной безопасности организации.	7	14-17	2	2	4		8	
	Итого по модулю 3:			4	4	6		16	Тестирование по мод
	ИТОГО:			16	16	18		58	

Содержание дисциплины

4.3. Содержание дисциплины, структурированное по темам (разделам).

4.3.1. Содержание лекционных занятий по дисциплине

Модуль 1. ***Введение в управление информационной безопасностью предприятия***

Тема 1. Основы построения систем обеспечения информационной безопасности на предприятии

Деятельность по обеспечению информационной безопасности. Предметная направленность деятельности по обеспечению информационной безопасности. Цель деятельности по обеспечению информационной безопасности. Принципы и форма деятельности по обеспечению информационной безопасности. Методы деятельности по обеспечению информационной безопасности. Средства обеспечения информационной безопасности. Субъекты обеспечения информационной безопасности.

Тема 2. Обеспечение информационной безопасности бизнеса Информационная сущность бизнеса. Роль руководства организации в обеспечении информационной безопасности. Определение информационной безопасности. Правовая среда бизнеса и ее свойства. Внутренняя нормативная база организации. Модель информационной безопасности бизнеса. Обобщенная модель распределения ресурсов организации в условиях рисков. Ущерб и негативные последствия. Риск- ориентированный подход к обеспечению информационной безопасности бизнеса. Общая модель обеспечения ИБ бизнеса.

Тема 3. Система управления информационной безопасностью бизнеса

Модели непрерывного совершенствования и корпоративное управление. Модели непрерывного совершенствования и международные стандарты. Шаги реализации стандартной системы управления информационной безопасностью организации. Модели COSO, COBIT, ITIL. Контроль и аудит.

Модуль 2. Анализ и оценка у*правления информационными рисками*

Тема 4. Анализ и оценка управленческих и экономических показателей системы управления информационной безопасностью бизнеса

Способы оценки информационной безопасности. Основные элементы процесса

оценки информационной безопасности. Способы измерения атрибутов объекта оценки информационной безопасности. Применение типовых моделей оценки на основе оценки процессов и уровней зрелости процессов для оценки информационной безопасности.
Модель оценки информационной безопасности на основе оценки процессов. Риск-ориентированная оценка информационной безопасности.

Тема 5. Управление жизненным циклом информационных активов. Анализ влияния состояния информационных активов на деятельность организации

Жизненный цикл искусственно созданных объектов. Теротехнология. Информационный актив. Распределение информационных систем по целевым направлениям. Совместное использование данных в организации. Формирование стоимости владения информационным активом. Модель BSI PAS 99. Ценность и стоимость информационного актива. Уровни критичности информационного актива. Понятие «Анализ воздействия на бизнес». Структура управления операционным риском организации.

Тема 6. Методы управления информационными рисками. Анализ влияния информационного риска на деятельность организации.

Рискообразующие факторы. Структура информационного риска. Понятие «Риск информационной безопасности». Методика анализа риска информационной безопасности. Обработка рисков информационной безопасности. Процесс «Управление рисками информационной безопасности». Место управления рисками информационной безопасности в структуре управления операционными рисками организации. Место управления рисками информационной безопасности в структуре управления информационной безопасностью организации.

Модуль 3. ***Аудит системы управления ИБ организации***

Тема 7. Подходы к формированию нормативного обеспечения системы информационной безопасности организации

Анализ нормативной базы организации. Делопроизводство и документооборот организации. Процесс управления документацией. Защищённый документооборот. Информационный обмен. Жизненный цикл документа. Структура документации системы управления информационной безопасностью. Понятие «запись». Процесс уничтожения документации.

Тема 8. Аудит методов и средств обеспечения информационной безопасности организации

Аудит информационной безопасности. Стандарты и практики аудита информационной безопасности. Международный стандарт ISO 19011. Методы организации, подготовки и проведения аудита информационной безопасности. Обработка результатов аудита. Понятие
«Корректирующие» и «Превентивные» мероприятия. Аудит интегрированных систем управления. Психологические аспекты подготовки аудитора информационной безопасности. Понятие «независимая оценка» состояния информационной безопасности организации. Ответственность за результаты аудиторской проверки. Место аудита информационной безопасности в структуре управления информационной безопасностью организации.

4.3.2. *Содержание лабораторно-практических занятий по дисциплине.*

№ п/п	№ темы дисцип-лины	Тематика практических занятий (семинаров)	Технология проведения	Трудоём-кость в часах

1	**1-4**	Анализ и оценка управленческих и экономических показателей системы управления обеспечением информационной безопасности бизнеса.	Теоретическая справка с кратким изложением основных понятий. Проведение ситуационных моделей происходит в интерактивной форме для отработки навыков управления информационной безопасностью Выступления студентов с докладами и презентациями. Аудиторные самостоятельные работы для качественной оценки пройденного материала (15-20 мин.).	6
2	**5-6**	Управление жизненным циклом информационных активов.		4
3	**7-8**	Анализ влияния информационного риска на деятельность организации.		6
		ИТОГО		16

Лабораторный практикум

1. ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ

1.1. Способы контроля правильности передачи данных

1.1.1. Код с проверкой на четность

Контроль целостности информации при передачи от источника к приемнику может осуществляться с использованием корректирующих кодов.

Простейший корректирующий код – *код с проверкой на четность*, который образуется добавлением к группе информационных разрядов одного избыточного, значение которого выбирается таким образом, чтобы сумма единиц в кодовой комбинации (то есть вес кодовой комбинации) была всегда четна.

Пример. Рассмотрим код с проверкой на четность, образованный добавлением контрольного разряда к простому двоичному коду.

	Информационные разряды	*Контрольный разряд*
0	000	0
1	001	1
2	010	1
3	011	0
4	100	1
5	101	0
6	110	0
7	111	1

Такой код обнаруживает все одиночные ошибки и групповые ошибки нечетной кратности, так как четность количества единиц в этом случае будет также нарушаться.

Следует отметить, что при кодировании целесообразно число единиц в кодовой комбинации делать нечетным, и осуществлять контроль на нечетность, в этом случае любая комбинация, в том числе и изображающая нуль, будет иметь хотя бы одну единицу, что дает возможность отличить полное отсутствие информации от передачи нуля.

1.1.2. Коды Хэмминга

N - значный код Хэмминга имеет *m* - информационных разрядов и *k* - контрольных. Число контрольных разрядов должно удовлетворять соотношению:

$$k >= log_2 (n+1),$$

откуда

$$m <= n - log_2 (n + 1).$$

Код Хэмминга строится таким образом, что к имеющимся информационным разрядам кодовой комбинации добавляется вычисленное по вышеприведенной формуле количество контрольных разрядов, которые формируются путем подсчета четности суммы единиц для определенных групп информационных разрядов. При приеме такой кодовой комбинации из полученных информационных и контрольных разрядов путем аналогичных подсчетов четности составляют корректирующее число, которое равно нулю, при отсутствии ошибки, либо указывает номер ошибочного разряда.

Рассмотрим подробнее процесс кодирования.

Пусть первый контрольный разряд имеет нечетный порядковый номер, установим его при кодировании таким образом, чтобы сумма единиц всех разрядов с нечетными порядковыми номерами была равна 0. Такая операция может быть записана в виде соотношения:

$$E_1 = a_1 \oplus a_3 \oplus a_5 \oplus a_7 \ldots = 0,$$

где a_1, a_3, a_5 , a_7 - двоичные символы, размещенные в разрядах с номерами

1, 3, 5, 7,…

Появление единицы во втором разряде (справа) корректирующего числа означает ошибку в тех разрядах кодовой комбинации, порядковые номера которых (2, 3, 6, 7, …) в двоичном изображении имеют единицу во втором справа разряде. Поэтому вторая операция кодирования, позволяющая найти второй контрольный разряд, имеет вид:

$$E_2 = a_2 \oplus a_3 \oplus a_6 \oplus a_7 \ldots = 0,$$

Рассуждая аналогичным образом:

$$E_3 = a_4 \oplus a_5 \oplus a_6 \oplus a_7 \ldots = 0, E_4 = a_8 \oplus a_9 \oplus a_{10} \oplus a_{11} \ldots = 0,$$

После приема кодовой комбинации, совместно со сформированными контрольными разрядами, выполняются те же операции подсчета, что были описаны выше, а полученное число:

$$E_k \; E_{k-1} \ldots E_2 \; E1$$

считается корректирующим, причем при $E_k \; E_{k-1} \ldots E_2 \; E_1 = 0$ ошибки отсутствуют, а при наличии ошибок неравными нулю оказываются те суммы Ei , в образовании которых участвовал ошибочный разряд.

Корректирующее число при этом будет равно порядковому номеру этого разряда.

Выбор места для контрольных разрядов в каждой из кодовых комбинаций определяется таким образом, чтобы контрольные разряды участвовали только в одной операции подсчета четности. Это упрощает процесс кодирования, такими позициями являются целые степени двойки: 1, 2, 4, 8, 16,… .

Пример. Составим шестизначный код Хэмминга

$n = 6, k >= log_2 7 , k = 3, m = n - k = 3$

Цифра	Простой двоичный код	Код Хэмминга *6 5 4 3 2 1*
0	000	0 0 0 0 0 0
1	001	0 0 0 1 1 1
2	010	0 1 1 0 0 1
3	011	0 1 1 1 1 0
4	100	1 0 1 0 1 0
5	101	1 0 1 1 0 1
6	110	1 1 0 0 1 1
7	111	1 1 0 1 0 0

Варианты исправления ошибок

Принят код: 111100 *исправлено* 110100 - *ошибка по корректирующему числу в разряде 4;*

111010 *исправлено* 101010 - *ошибка по корректирующему числу в разряде 5;*

100000 *исправлено* 000000 - *ошибка по корректирующему числу в разряде 6*

1.1.3. Циклические коды

Циклические коды являются разновидностью систематических кодов и поэтому обладают всеми их свойствами. Характерной особенностью циклического кода, определяющей его название, является то, что если *n*-значная кодовая комбинация $a_0\, a_1\, a_2 \ldots a_{n-1}\, a_n$ принадлежит данному коду, то и комбинация $a_n\, a_0\, a_1\, a_2 \ldots a_{n-1}$, полученная циклической перестановкой знаков, также принадлежит этому коду.

Идея построения циклических кодов базируется на использовании *неприводимых многочленов*. Неприводимым называется многочлен, который не может быть представлен в виде произведения многочленов низших степеней, то есть такой многочлен, который делится только на самого себя или на 1.

Неприводимые многочлены при построении циклических кодов играют роль так называемых образующих полиномов, от вида которых, собственно и зависят основные характеристики полученного кода: избыточность и корректирующая способность. В таблице 1.1 указаны неприводимые многочлены со степенями *k=1, 2, 3, 4.*

Таблица 1.1

	$P(x)$	$P(1,0)$
$k=1$	$x+1$	11
$k=2$	x^2+x+1	111
$k=3$	x^3+x+1	1011
	x^3+x^2+1	1101
$k=4$	x^4+x+1	10011
	x^4+x^3+1	11001
	$x^4+x^3+x^2+x+1$	11111

Основные принципы кодирования в циклическом коде заключаются в следующем. Двоично-кодированное *n*-разрядное число представляется полиномом *(n–1)*-й степени некоторой переменной *x,* причем коэффициентами полинома являются двоичные знаки соответствующих разрядов. Запись, чтение и передача кодовых комбинаций в циклическом коде производятся, начиная со старшего разряда. В соответствии с этим правилом, в дальнейшем сами числа и соответствующие им полиномы будем записывать так, чтобы старший разряд оказывался справа.

0 1 2 3 4 5

Пример. Число 110101 (нумерация разрядов согласно выше приведенному правилу, ведется слева направо от 0 до 5) будет представлено полиномом пятой степени: $1 + x + x^3 + x^5$.

Следует отметить, что циклическая перестановка разрядов в двоичном представлении числа соответствует умножению полинома на *x*, при котором x^n заменяется 1 и переходит в начало полинома.

Пример. Произведем умножение полинома, полученного в предыдущем примере, на *x*. Новый полином $x + x^2 + x^4 + x^6$, преобразуем, заменив x^6 на 1.

Окончательно получим $1 + x + x^2 + x^4$, что соответствует числу 111010.

Циклический код *n*-значного числа, как всякий систематический код, состоит их *m* информационных и *k* контрольных знаков, причем последние занимают *k* младших разрядов. Так как последовательная передача кодовых комбинаций производится, как мы уже указывали, начиная со старших разрядов, контрольные знаки передаются в конце кода.

Как уже указывалось, образование кода производится при помощи так называемого *порождающего полинома*, *P(x),* степени *k,* видом которого определяются основные свойства кода – избыточность и корректирующая способность.

Кодовым полиномом, F(x), является полином степени, меньшей *(m+k),* если он делится без остатка на порождающий полином *P(x).* После передачи сообщения, декодирование состоит в выполнении деления полинома *H(x),* соответствующего принятому коду, на *P(x).* При отсутствии ошибок *H(x)=F(x),* и деление выполняется без остатка. Наличие ненулевого остатка указывает на то, что при передаче или хранении

произошли искажения информации.

Для получения систематического циклического кода используется следующее соотношение:

$$F(x) = x^k\, G(x) \oplus R(x),$$

где $G(x)$ – полином, представляющий информационные символы

(информационный полином);

$R(x)$ – остаток от деления $x^k\, G(x)$ на $P(x)$.

Пример. Рассмотрим кодирование 8-значного числа 10110111. Пусть для кодирования задан порождающий полином 3-й степени $P(x) = 1 + x + x^3$.

Делим $x^3\, G(x)$ на $P(x)$: $G(x) = 1 + x^2 + x^3 + x^5 + x^6 + x^7$;

$$x^3\, G(x) = x^3 + x^5 + x^6 + x^8 + x^9 + x^{10};$$

$$\begin{array}{l} x^{10} + x^9 + x^8 + x^6 + x^5 + x^3 \\ \underline{x^{10} + x^8 + x^7} \end{array} \qquad \begin{array}{l|l} 1 + x + x^3 & \\ \hline & x^7 + x^6 + x^2 \end{array}$$

$$\begin{array}{l} x^9 + x^7 + x^6 + x^5 + x^3 \\ \underline{x^9 + x^7 + x^6} \end{array}$$

$$\begin{array}{l} x^5 + x^3 \\ \underline{x^5 + x^3 + x^2} \end{array}$$

$$R(x) = x^2$$

Используя соотношение для получения систематического циклического кода, находим $F(x)$:

$$F(x) = (x^3 + x^5 + x^6 + x^8 + x^9 + x^{10}) \oplus x^2.$$

Таким образом, окончательно кодовая комбинация, соответствующая $F(x)$, имеет вид

$$\begin{array}{r} 00010110111 \\ \underline{001} \end{array}$$

контрольные разряды → 001 10110111

Практически, применяемая процедура кодирования еще более проста, так как нас интересует только остаток, а частное в конечном результате не используется, то можно производить последовательное вычитание по модулю 2 делителя из делимого и полученных разностей до тех пор, пока разность не будет иметь более низкую степень, чем делитель. Эта разность и есть остаток. Такой алгоритм может быть реализован аппаратно при помощи k-разрядного сдвигающего регистра, имеющего

обратные связи. Очевидно, что полученный таким способом циклический код будет являться систематическим.

Однако существует и второй вариант получения циклического кода, когда очередная кодовая комбинация получается путем умножения кодовой комбинации *C(x)* простого *n*-значного кода на образующий полином *P(x).*

При втором способе образования циклических кодов информационные и контрольные символы в комбинациях циклического кода не отделены друг от друга, что затрудняет процесс декодирования. Поэтому этот способ кодирования применяется реже, чем первый. Рассмотрим пример кодирования с использованием второго варианта, причем при выполнении операций будем использовать непосредственно записи исходных кодовых комбинаций в двоичном виде.

Пример. Дан порождающий полином вида *P(1,0)=1101*. Требуется построить циклический код из простого четырехзначного кода вторым способом. В качестве примера для построения используем исходную комбинацию *C(1,0)=0011*. Операция умножения этой комбинации на образующий полином запишется следующим образом:

$$\begin{array}{r} 0011 \\ 1101 \\ \hline \\ 0011 \\ \oplus\ 0011 \\ 0011 \\ \hline \end{array}$$

0010111 – это и есть циклический код для 0011.

Аналогичным образом получены остальные кодовые комбинации несистематического циклического кода, приведенные в таблице 1.2.

Таблица 1.2

Простой четырехсимвольный код C(x)	*Циклический (7,4) – код C(x) P(x), где P(1,0)=1101*
0000	*0000000*
0001	*0001101*
0010	*0011010*
0011 *	*0010111* *
0100	*0110100*
0101	*0111001*
0110	*0101110*
0111	*0100011*
1000	*1101000*
1001	*1100101*
1010	*1110010*
1011	*1111111*
1100	*1011100*
1101	*1010001*
1110	*1000110*
1111	*1001011*

Проблема обнаружения различного типа ошибок при помощи циклического кода, как уже указывалось, сводится к нахождению нужного порождающего полинома..

1.2. Эффективное кодирование информации

1.2.1. Общие положения

Напомним, что *кодирование* – это представление сообщений в форме, удобной для передачи по данному каналу, а *декодирование* – восстановление информации по принятому сигналу. Одним из важнейших вопросов кодирования является повышение его эффективности, то есть путем устранения избыточности снижение среднего числа символов, требующихся на букву сообщения. Такое кодирование получило название *эффективное кодирование.* Из выше сказанного следует, что эффективное кодирование решает задачу максимального *сжатия информации*.

Учитывая статистические свойства источника информации можно минимизировать среднее число двоичных символов, требующихся для выражения одной буквы сообщения, что при отсутствии помех позволяет уменьшить время передачи сообщения и его объем.

Эффективное кодирование базируется на *основной теореме Шеннона для канала без помех,* суть которой сводится к следующему.

Суть теоремы Шеннона

Сообщения, составленные из букв некоторого алфавита, можно закодировать так, что среднее число двоичных символов на букву сколь угодно близко к энтропии источника этих сообщений, но не меньше этой величины.

Наличие в сообщениях "избыточности" позволяет ставить вопрос о сжатии данных, то есть о передаче того же количества информации с помощью последовательностей символов меньшей длины – *методы сжатия без потерь.* Для этого используются специальные алгоритмы сжатия, уменьшающие избыточность. Эффект сжатия оценивают *коэффициентом сжатия*

$$K = n/q,$$

где n – число минимально необходимых символов для передачи сообщения;

q – число символов в сообщении, сжатом данным методом.

Так, при эффективном двоичном кодировании n равно энтропии источника информации.

Наряду с методами сжатия, не уменьшающими количество информации в сообщении, применяются методы сжатия, основанные на потере малосущественной информации – *методы сжатия с потерями.* Заметим, что применение методов сжатия с потерями неприемлемо для некоторых видов информации, например, текстовой или числовой.

Рис. 1.1. Методы сжатия информации

Простейший способ сжатия числовой информации, представленной в коде *ASCII,* заключается в использовании сокращенного кода с четырьмя битами на символ вместо восьми, так как в *ASCII*-кодировке цифры, а также символы *"точка"*, *"запятая"* и *"пробел"* в качестве первого кодового символа имеют нуль, который может быть отброшен.

Среди широко используемых из за своей простоты алгоритмов сжатия наиболее известен *алгоритм RLE (Run Length Encoding),* где вместо передачи цепочки из одинаковых символов передаются символ и значение длины цепочки. Этот метод эффективен при передаче растровых изображений, особенно монохромных, где имеется большое число цепочек из 1 и 0, но малополезен при передаче текста. Алгоритм *RLE* реализован в формате *PCX.*

Пример использования алгоритма RLE
Исходный текст (9 байт): 77 77 77 77 11 11 11 01 FF

В исходной последовательности:

1. Выделяется повторяющаяся последовательность байт
2. Заменяется счетчиком повторов (04) и кодирующим байтом (77)
3. 00 – нет повторений
4. 02 - сколько байт не повторяется
5. Какие байты не повторяются (01 FF)

Результат сжатия (8 байт): 04 77 03 11 00 02 01 FF

Недостатками алгоритма *RLE* являются:

- низкая степень сжатия;
- сильная зависимость от количества повторов байт в исходной информационной последовательности.

Достоинство – простота реализации.

Основная сфера использования методов сжатия с потерями графические, аудио и

видео файлы, где имеется возможность, частично пожертвовав качественными характеристиками информации, добиться значительного уменьшения ее объемов.

Наибольшее распространение среди методов сжатия информации без потерь нашли статистические алгоритмы сжатия, учитывающие вероятность появления отдельных символов в информационном потоке. В первую очередь, это алгоритмы Шеннона-Фано и Хаффмена.

1.2.2. Алгоритм Шеннона-Фано

Эффективное кодирование методом Шеннона-Фано базируется на основной теореме Шеннона для канала без помех.

Алгоритм эффективного кодирования по Шеннону-Фано

- Буквы алфавита сообщений выписываются в таблицу в порядке убывания вероятностей.
- Затем они разделяются на две группы так, чтобы суммы вероятностей в каждой группе были по возможности одинаковы.
- Всем буквам верхней половины в качестве первого символа приписывается 1, а всем нижним – 0.
- Каждая из полученных групп в свою очередь разбивается на две подгруппы с одинаковыми суммарными вероятностями и т. д., процесс повторяется до тех пор, пока в каждой подгруппе останется по одной букве.

Пример. Рассмотрим алфавит из 7 букв, приписав каждой букве вероятность ее появления в сообщении p_i (существуют специальные таблицы вероятности появления букв русского или латинского алфавитов в сообщениях, например, см. приложение 1.2.1).

Таблица 1.3

Буква	Вероятность, p_i	Процесс получения кода	Код Шеннона
A	1/4	1 1 II	11
E	1/4	1 0 I	10
F	1/8	0 1 1 III	011
C	1/8	0 1 II 0	010
B	1/8	0 0 1 III	001
D	1/16	0 0 0 1 IV	0001
G	1/16	0 0 0 0	0000

$$H(\xi) = \sum_{k=1}^{N} p_k \log_2 1/p_k = 2/4 + 2/4 + 3/8 + 3/8 + 3/8 + 4/16 + 4/16 = 2{,}625.$$

Ясно, что при обычном двоичном кодировании, не учитывая статистических характеристик, для представления каждой буквы потребуется три символа. Наибольший эффект сжатия получается в случае, когда вероятности букв представляют собой

целочисленные отрицательные степени двойки. Среднее число символов на букву в этом случае точно равно энтропии.

Среднее число символов на букву

$$L = \sum_{i=1}^{N} p_i n_i ,$$

где p_i – вероятность появления i-го символа алфавита;

n_i – количество символов в кодовой комбинации i-го символа алфавита.

Для рассмотренного в таблице примера

$L=1/4 \times 2 + 1/4 \times 2 + 1/8 \times 3 + 1/8 \times 3 + 1/8 \times 3 + 1/16 \times 4 + 1/16 \times 4 = 2{,}625.$

Разность величин $(L – H)$ – есть *избыточность кода,* а величина

$(L – H)/L$ – *относительная избыточность.*

Избыточность может трактоваться как мера бесполезно совершаемых альтернативных выборов. Так как в практических случаях отдельные знаки никогда не встречаются одинаково часто, то кодирование с постоянной длиной кодовых слов в большинстве случаев избыточно. Несмотря на это, такое кодирование применяется достаточно часто, руководствуясь техническими соображениями.

1.2.3. Алгоритм Хаффмена

Алгоритм эффективного кодирования по Хаффмену

Буквы алфавита сообщений выписываются в основной столбец таблицы в порядке убывания вероятностей.

- Две последние буквы объединяются в одну вспомогательную букву, которой приписывается суммарная вероятность.
- Вероятности букв, не участвовавших в объединении, и полученная суммарная вероятность снова располагаются в порядке убывания вероятностей, а две последние объединяются, и т. д. до тех пор, пока не получают единственную вспомогательную букву с вероятностью 1.
- Далее для построения кода используется бинарное дерево, в корне которого располагается буква с вероятностью 1, при ветвлении ветви с большей вероятностью присваивается код 1, а с меньшей – код 0 (возможно левой – 1, а правой – 0).

  Пример. Рассмотрим условный алфавит из восьми букв, каждой из которых приписана соответствующая вероятность ее появления в сообщении.

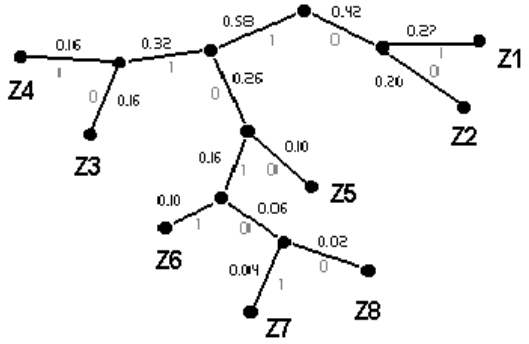
Таблица 1.4

Буквы	**Вероятности**	**Вспомогательные столбцы вероятностей**	**Код Хаффмена**
Z1	0,22	0,22 0,26 0,32 0,42 0,58 1	01
Z2	0,20	0,20 0,20 0,20 0,22 0,26 0,32 0,42	00
Z3	0,16	0,16 0,16 0,16 0,20 0,22 0,26	110
Z4	0,16	0,16 0,16 0,16 0,16 0,20	111
Z5	0,10	0,10 0,16 0,16	100
Z6	0,10	0,10 0,10	1011
Z7	0,04	0,06	10101
Z8	0,02		10100

L=0,22 × 2 + 0,20 × 2 + 0,16 × 3 + 0,16 × 3 + 0,10 × 3 + 0,10 × 4 + 0,04 × 5 + 0,02× 5 = 2,8; H=2,76;
L – H = 0,04.

Для сравнения: в коде Шеннона с таким же распределением вероятностей L – H = 0,08.

Построениебинарногодерева



В *приложении 1.2.2* приводятся задания для самостоятельной работы по кодированию информации с использованием алгоритмов Шеннона-Фано и Хаффмена.

Приложение 1.2.1

Вероятности отдельных букв в русском языке

Буква	Вероятность	Буква	Вероятность
_	0,175	**Я**	0,018
О	0,090	**Ы**	0,016
Е	0,072	**З**	0,016
А	0,062	**Ъ,Ь**	0,014
И	0,062	**Б**	0,014
Т	0,053	**Г**	0,013
Н	0,053	**Ч**	0,012
С	0,045	**Й**	0,010
Р	0,040	**Х**	0,009
В	0,038	**Ж**	0,007
Л	0,035	**Ю**	0,006
К	0,028	**Ш**	0,006
М	0,026	**Ц**	0,004
Д	0,025	**Щ**	0,003
П	0,023	**Э**	0,003
У	0,021	**Ф**	0,002

Приложение 1.2.2

Задания для самостоятельной работы

1. *Написать процедуру, обеспечивающую эффективное кодирование: вариант а) код Шеннона; вариант б)*

код Хаффмена,

и подсчет среднего количества информации на знак H;

избыточности (L - H); и относительной избыточности получаемого кода (L – H) / L.

Для тестирования программы использовать два контрольных вариантa:

Вариант 1

Буквы	Вероятности	Код Шеннона
Z1	0,22	11
Z2	0,20	101
Z3	0,16	100
Z4	0,16	01
Z5	0,10	001
Z6	0,10	0001
Z7	0,04	00001
Z8	0,02	00000 H = 2,76 L = 2,84 L – H = 0,08

Вариант 2

Буквы	Вероятности	Код Шеннона
Z1	1/2	1
Z2	1/4	01
Z3	1/8	001
Z4	1/16	0001
Z5	1/32	00001
Z6	1/64	000001
Z7	1/128	0000001
Z8	1/128	0000000 H = 1 63/64 L = 1 63/64 L - H = 0

2. *Источник сообщений порождает знаки A, B, C с вероятностями 0,7; 0,2; 0,1. Написать процедуру позволяющую:*

- *определить энтропию источника;*
- *указать для этих трех знаков оптимальное бинарное кодирование:*

  вариант а) код Шеннона;
  вариант б) код Хаффмена,

  и определить среднюю длину кодовых комбинаций;
- *закодировать все пары AA, AB, …;*
- *построить для этих девяти пар оптимальный бинарный код:*

  вариант а) код Шеннона; вариант б) код Хаффмена.
- *увеличить блочность кода до трехсимвольных комбинаций, и построить оптимальный бинарный код:*

  вариант а) код Шеннона; вариант

б) код Хаффмена.

Сделать вывод об изменении избыточности кода с увеличением блочности. Как это влияет на эффективности кода?

Буква	Вероятность	Код	Знак	Вероятность	Код пары
A	0,7	0	**AA**	0,49	1
B	0,2	11	**AB**	0,14	011
C	0,1	10	**BA**	0,14	010
			AC	0,07	0011
			CA	0,07	0010
			BB	0,04	0001
			BC	0,02	00001
			CB	0,02	000001
			CC	0,01	000000

$H = 1{,}1571;\ L_{знак} = 2{,}33;$

$L = 1{,}3;\ L_{символ} = L_{знак} / 2 = 1{,}165.$

Вывод: при кодировании большими блоками удается уменьшить избыточность кода, теоретически минимум избыточности может быть достигнут при кодировании блоков, включающих бесконечное число букв:

$$\lim_{N \to \infty} L = H.$$

1.2.3. LZW – сжатие

Одним из наиболее широко используемых в настоящее время методов сжатия без потерь является алгоритм Лемпела-Зива-Уэлча *(Lempel-Ziv-Welch).*

Метод сжатия был предложен в 1977 году, усовершенствованный (*Terry Welch*) вариант был опубликован в 1984 году, и в дальнейшем неоднократно модернизировался. Этот алгоритм используется, в частности, стандартной процедурой *UNIX Compress*. Алгоритмы группы *LZ (LZ77, LZ78, LZSS)*, лежат в основе почти всех современных программных и аппаратных средств сжатия информации. *PKZIP, LHA, Stacker, SuperStor, Dblspace* и многие другие архиваторы используют ту или иную модификацию алгоритма *LZ*.

Идея сжатия, предложенная Лемпелом и Зивом заключается в следующем: если в тексте сообщения появляется последовательность из двух ранее уже встречавшихся символов, то эта последовательность объявляется новым символом, для нее назначается код, который при определенных условиях может быть значительно короче исходной последовательности. В дальнейшем, в сжатом сообщении вместо исходной последовательности записывается назначенный код. При декодировании повторяются аналогичные действия и потому становятся известными последовательности символов для каждого кода.

Процесс кодирования

1. Каждому символу исходного алфавита присваивается определенный код (здесь код – порядковый номер, начиная с 0).
2. Выбирается первый символ сообщения и заменяется на его код.
3. Выбираются следующие два символа и заменяются своими кодами.

Одновременно этой двухсимвольной комбинации присваивается свой код (обычно это порядковый номер, равный числу уже использованных кодов). Так, если алфавит включает 8 символов, имеющих коды от 000 до 111, то первая двухсимвольная комбинация получит код 1000, следующая – код 1001 и так далее.

4. Выбираются из исходного текста очередные 2, 3, ... *N*-символьные комбинации до тех пор, пока не образуется еще не встречавшаяся комбинация. Тогда этой комбинации присваивается очередной код, и поскольку совокупность из первых *N-1* символов уже встречалась, то она имеет свой код, который и записывается вместо этих *N-1* символов. Каждый акт введения нового кода назовем шагом кодирования.
5. Процесс продолжается до исчерпания исходного текста.

Процесс декодирования

При *декодировании* код первого символа, а затем второго и третьего заменяются на символы алфавита. При этом становится известным код комбинации второго и третьего символов. В следующей позиции могут быть только коды уже известных символов и их комбинаций. Процесс декодирования продолжается до исчерпания сжатого текста.

Пример.

Пусть исходный текст представляет собой двоичный код (первая строка таблицы 2.3), то есть исходный алфавит *A={1,0}*. Коды этих символов соответственно также 0 и 1.

Образующийся по методу Лемпела-Зива *LZ*-код показан во второй строке таблицы 2.3. В третьей строке отмечены шаги кодирования, после которых происходит переход на представление кодов *A* с увеличенным числом разрядов *r*. Так, на первом шаге вводится код 10 для комбинации 00 и поэтому на следующих двух шагах *r*=2, после третьего шага *r*=3, после седьмого шага *r*=4. В общем случае *r*=*k* после шага 2^{k-1}–1.

В приведенном примере *LZ*-код оказался даже длиннее исходного кода, так как обычно короткие тексты с небольшим количеством повторяющихся символов не дают эффекта сжатия. Эффект сжатия проявляется в достаточно длинных текстах и особенно заметен, например, в графических файлах.

Таблица 1.5

Исходный текст	0.00.000. 01. 11. 111.1111. 110. 0000.00000. 1101. 1110.
***LZ*-код**	0.00.100.001.0011.1011.1101. 1010.00110.10010.10001.10110.
r	2 3 4
Вводимые коды	- 10 11 100 101 110 111 1000 1001 1010 1011 1100

Усовершенствованный алгоритм сжатия и распаковки по методу Лемпела-Зива-Уэлча приведен на рис. 1.2. На рис. 1.3 приведен пример программной реализации алгоритма *LZW*.

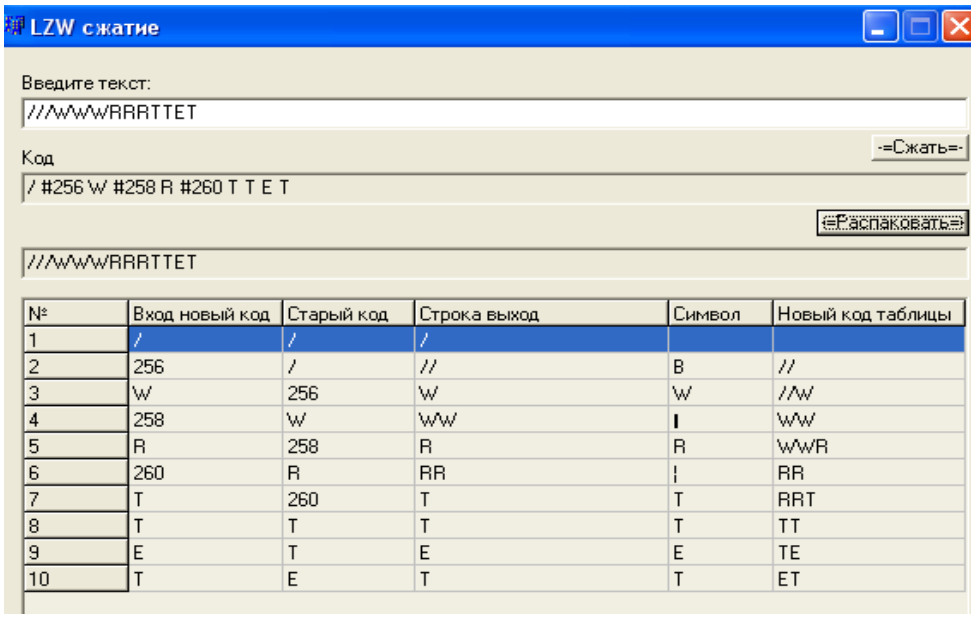
Алгоритм компрессии

```
СТРОКА = очередной символ из входного потока
WHILE входной поток  не пуст DO
    СИМВОЛ = очередной символ из входного  потока
    IF СТРОКА + СИМВОЛ в таблице строк THEN
       СТРОКА = СТРОКА + СИМВОЛ
    ELSE
       вывести в выходной поток код для СТРОКА
       добавить в таблицу строк СТРОКА + СИМВОЛ
       СТРОКА = СИМВОЛ
    END of IF
END of WHILE
вывести в выходной поток код для СТРОКА
```

Алгоритм декомпрессии

```
Читать СТАРЫЙ_КОД
вывести СТАРЫЙ_КОД
WHILE входной поток не пуст DO
   читать НОВЫЙ_КОД
   СТРОКА = перевести НОВЫЙ_КОД
   вывести СТРОКУ
   СИМВОЛ = первый символ СТРОКИ
   добавить в таблицу  перевода СТАРЫЙ_КОД + СИМВОЛ
   СТАРЫЙ_КОД = НОВЫЙ_КОД
END of WHILE
```

Рис.1.2. Алгоритм *LZW*-компрессии и декомпрессии



№	Вход новый код	Старый код	Строка выход	Символ	Новый код таблицы
1	/	/	/		
2	256	/	//	B	//
3	W	256	W	W	//W
4	258	W	WW	▮	WW
5	R	258	R	R	WWR
6	260	R	RR	¦	RR
7	T	260	T	T	RRT
8	T	T	T	T	TT
9	E	T	E	E	TE
10	T	E	T	T	ET

Рис.1.3. Пример программной реализации алгоритма *LZW*

ЛАБОРАТОРНАЯ РАБОТА № 9
КОРРЕКТИРУЮЩИЕ КОДЫ. КОДЫ ХЕММИНГА

Цель работы: Ознакомление с общими принципами построения и использования корректирующих кодов для контроля целостности информации, распространяемой по телекоммуникационным каналам.

Примечание. Для выполнения лабораторной работы на компьютере необходимо установить файл

Hemming.exe, который находится в архиве *Код Хэмминга.rar*.

1. Описание лабораторной работы

Программа предназначена для кодирования символов по алгоритму Хэмминга. Главное окно программы представлено на рис.9.1.

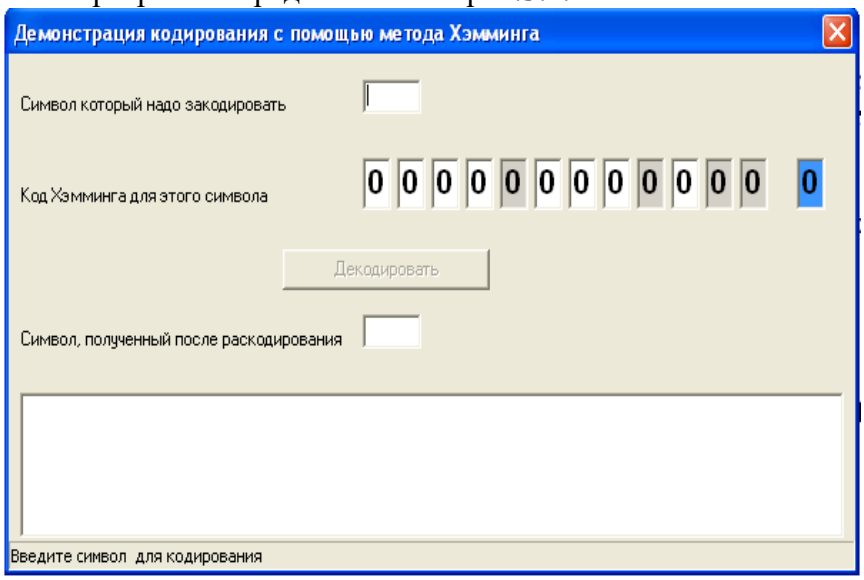


Рис. 9. 1. Главное окно программы

Кодируются любые символы *ASCII*-кодировки, с помощью тринадцатисимвольного кода Хэмминга, содержащего 8 информационных разрядов, 4 контрольных разряда и разряд общей четности.

Процессы кодирования и декодирования представлены на рис.9.2

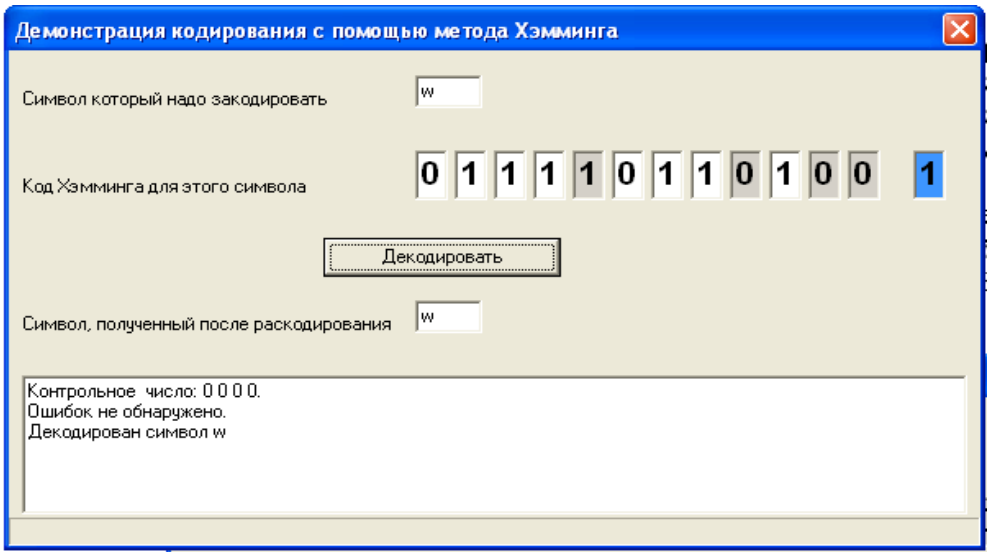


Рис. 9. 2. Процесс кодирования и декодирования

Серым цветом отмечены контрольные биты, голубым – бит общей четности. В полученный код можно вносить ошибки. Одиночные ошибки могут быть исправлены, двойные – обнаруживаются без исправления. Если ошибка была исправлена, то

указывается, в каком бите она была допущена. Затем символ декодируется. Процесс исправления одиночной ошибки представлен на рис. 9.3.

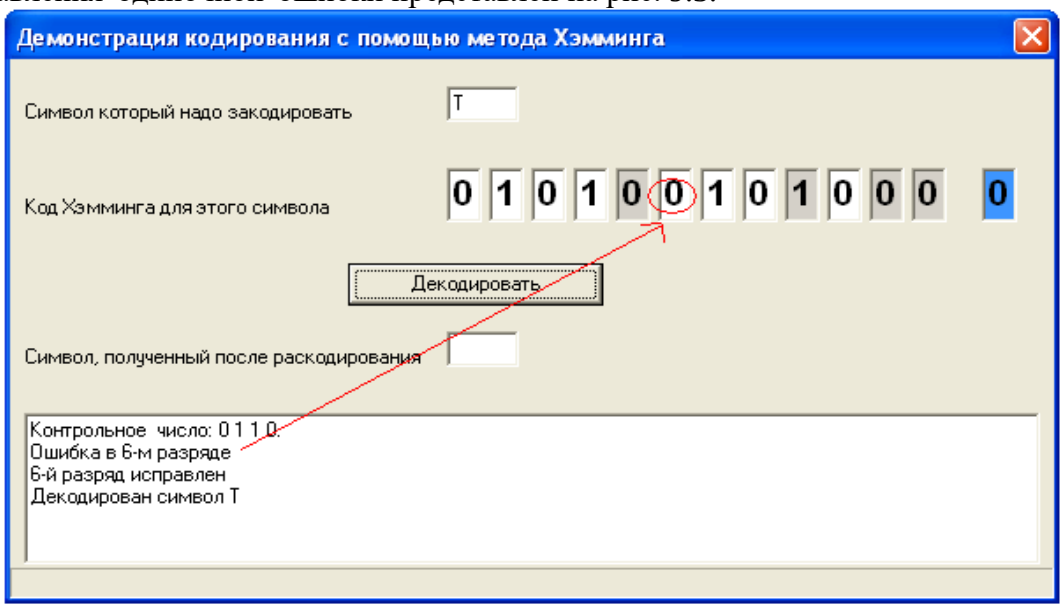


Рис. 9. 3. Процесс исправления одиночной ошибки

Если возникает ошибка двойной или более кратности – выводится сообщение о невозможности исправления кода.

2. Задание

1. Закодировать с помощью кода Хэмминга предложенный алфавит (варианты указаны в Таблице 9.1).
2. В каждую строку таблицы с закодированной информацией внести одиночную ошибку, зафиксировать в кодовой таблице результат декодирования.
3. В последние две строки таблицы с закодированной информацией внести двойные ошибки, зафиксировать в кодовой таблице результат декодирования.
4. Проанализировать полученные результаты и сформулировать аргументированные выводы.
   Описать полученный код Хэмминга :

   - количество контрольных и информационных разрядов и их номера;
   - избыточность кода;
   - относительная избыточность;
   - минимальное кодовое расстояние;
   - оценить корректирующую способность полученного кода.

5. Составить из предложенного алфавита слово длиной не менее 5 символов и закодировать его с помощью полученного кода Хэмминга.Подсчитать длину исходного текста (кодировка *ASCII*) и закодированного текста (код Хэмминга).
6. Представить краткий теоретический материал, в котором описать способ кодирования и декодирования информации с помощью кода Хэмминга.
7. Привести итоги проведенных экспериментов по кодированию с помощью

программы *Hemming.exe*.

8. Оценить результаты обнаружения и исправления одиночных и обнаружения двойных ошибок. Сделать выводы о корректирующей способности исследуемого кода.
9. Привести в отчете ответы на контрольные вопросы, в соответствии с номером варианта, указанным преподавателем.

Варианты заданий представлены в Таблице 9.1

Таблица 9.1

Номера вариантов	**Исходный алфавит**
1,5,9,13,17	Кириллица А...М
2,6,10,14,18,22	Кириллица О..Я
3,7,11,15,19,23,27	Латиница A..N
4,8,12,16,20,24,28	Латиница O..Z
21,25,29,26,30	Десятичные цифры 0..9

Номер варианта	**Контрольные вопросы**
1,5,7, 3,9,18,28	Что представляет собой и как используется код Хэмминга?
2,4,6,8, 20,22,24, 26,30	Как составляется код Хэмминга?
11,13,15, 10,17,19, 27	Какие ошибки можно исправить с помощью кода Хэмминга? От каких свойств кода зависит его корректирующая способность?
12,14,16 21,23.25, 29	Какие ошибки могут возникнуть при составлении кода Хэмминга?

ЛАБОРАТОРНАЯ РАБОТА № 10
КОРРЕКТИРУЮЩИЕ КОДЫ. ЦИКЛИЧЕСКИЕ КОДЫ

Цель работы: Ознакомление с общими принципами построения и использования корректирующих кодов для контроля целостности информации, распространяемой по телекоммуникационным каналам. Изучение принципов построения циклических кодов.

Примечание. Для выполнения лабораторной работы на компьютере необходимо установить файл
CyclicCode74.exe, который находится в архиве *Циклический код.rar*.

.

1. Описание лабораторной работы

1. Изучить сведения о программе кодирования информации с использованием циклического кода (папка *Документация*). Запустить программу кодирования (папка *Исполняемый модуль*, файл *CyclicCode74.exe*), рис. 10.1.

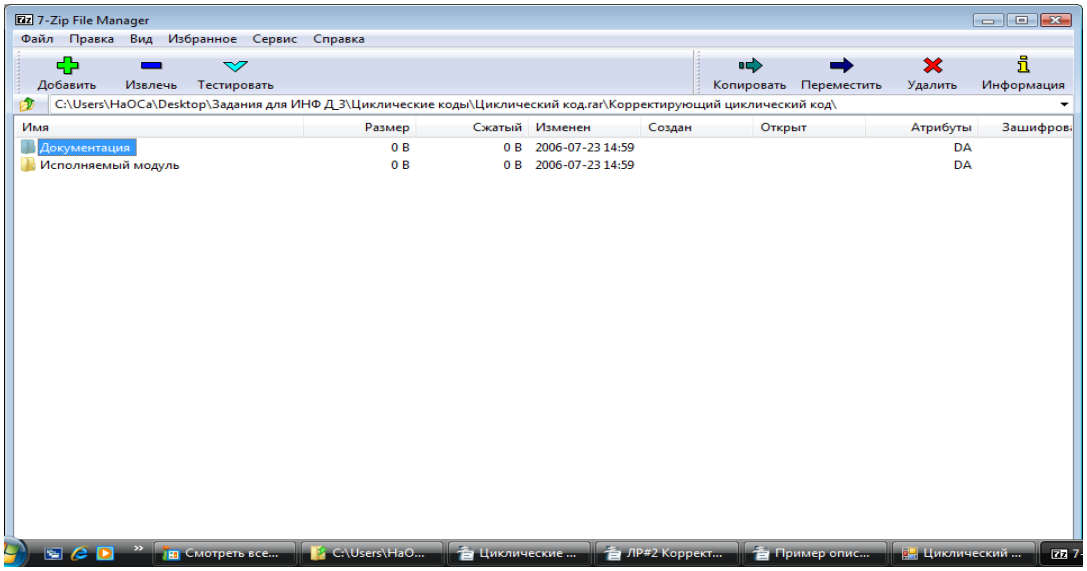


Рис. 10.1. Запуск программы *CyclicCode74*

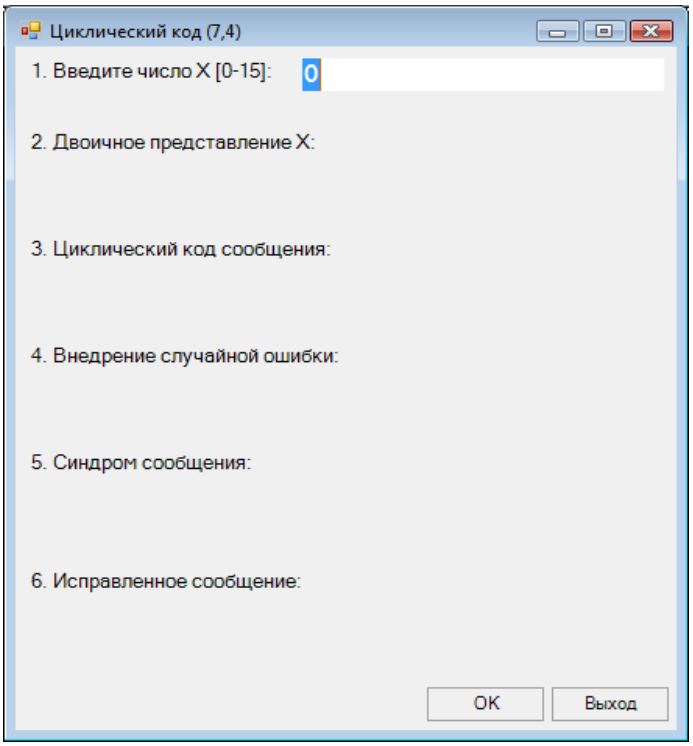


Рис.10. 2. Главное окно программы

2.Закодировать с помощью циклического кода предложенный вариант (*см. варианты заданий*). Для этого в строку *"1. Ведите число X [0-15]"* вводим заданное число. Кодовую таблицу сохранить и перенести в личную папку (для этого нажимаем одновременно Ctrl+Alt+PrtScSysRq затем Ctrl+V).

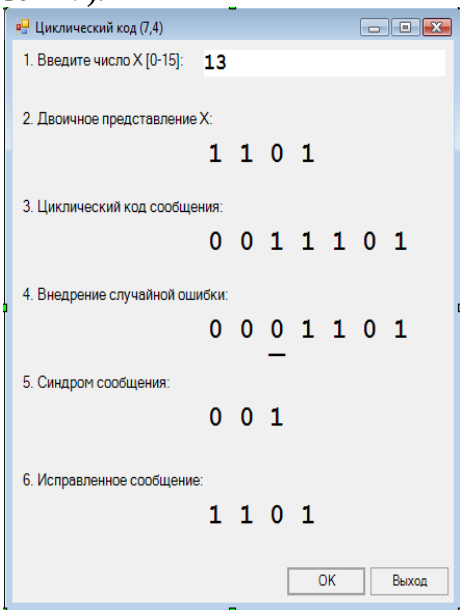


Рис. 10. 3. Пример заполнения полей главного окна программы

3. Проанализировать полученные результаты и сформулировать аргументированные выводы.

2. Задание

1. Получить систематический циклический код, используя приведенные в

Таблице 10.1 порождающие полиномы, в соответствии с вариантом, указанным преподавателем:

Таблица 10.1

Вариант Порождающий полином 3-ей степени	
1,7,13,19,25	1011
2,8,14,20,26	1101
3,9,15,21,27	1011
4,10,16,22,2	1101
5,11,17,23,2	1011
6,12,18,24,3	1101

0 1 2 3 4 5 6 7

Пример: Рассмотрим кодирование восьмизначного числа 00011111. Пусть для кодирования задан порождающий полином 3-ей степени

$P(1,0) = 1101,$

$P(x) = 1 + x + x^3$

Делим $x^3 G(x)$ на $P(x)$, где информационный полином $G(x) = x^3 + x^4 + x^5 + x^6 + x^7$

$x^3 G(x) = x^6 + x^7 + x^8 + x^9 + x^{10}$

$x^{10} + x^9 + x^8 + x^7 + x^6$ | $x^3 + x + 1$

$x^{10} + x^8 + x^7$ | $x^7 + x^6 + x^4 + x^2 + x + 1$

$x^9 + x^6$

$x^9 + x^7 + x^6$

x^7

$x^7 + x^5 + x^4$

$x^5 + x^4$

$x^5 + x^3 +$

x^2

$x^4 +$

$x^3 + x^2$

$x^4 +$

$x^2 + x$

$x^3 + x$

$x^3 + x + 1$

$R(x) = 1$

Находим значение кодового полинома *F(x):*

$$F(x) = (x^6 + x^7 + x^8 + x^9 + x^{10}) \oplus 1$$

Таким образом, окончательно кодовая

F(x) имеет вид : 10000011111
Контрольные разряды 100

10000011111

2. Представить в отчете краткий теоретический материал, в котором описать способ кодирования и декодирования информации с помощью циклического кода.
3. Привести итоги проведенных экспериментов по кодированию с помощью программы *CyclicCode74.exe* .
4. Оценить результаты обнаружения и исправления одиночных ошибок. Сделать выводы о корректирующей способности исследуемого кода.
5. Привести в отчете ответы на контрольные вопросы, в соответствии с номером варианта, указанным преподавателем.

Варианты заданий приведены в Таблице 10.2.

Таблица 10.2

Вариант	*Число*
1,1	1
2,1	2
3,1	3
4,1	4
5,2	5
6,2	6
7,2	7
8,23	8
9,2	9
10,2	10
11,2	11
12,2	12
13,2	13
14,2	14
15,3	15

Номер варианта	**Контрольные вопросы**
1,5,7, 3,9,18,28	Укажите различия понятий: *кодирование информации, шифрование информации.*
2,4,6,8, 20,22,24, 26,30	Определите понятия: *равномерные коды*, *неравномерные коды*, *несистематические коды.*
11,13,15, 10,17,19, 27	От каких характеристик кода зависит его корректирующая способность?
12,14,16 21,23.25, 29	Сравните коды Хэмминга и циклические коды по эффективности целостности информации.

ЛАБОРАТОРНАЯ РАБОТА № 11 МЕТОДЫ СЖАТИЯ ПО ШЕННОНУ И ХАФФМЕНУ

Цель работы: Ознакомление со статистическими принципами сжатия информации с использованием методов Шеннона-Фано и Хаффмена.

Примечание. Для выполнения лабораторной работы на компьютере необходимо установить файл *Shannon-Huffman.exe*, который находится в архиве *Методы сжатия по Шеннону и Хаффмену.rar*.

1. Описание лабораторной работы

Работа выполняется на персональном компьютере с использованием программы *Shannon-Huffman.exe*. Программа предназначена для демонстрации методов сжатия информации с использованием алгоритмов Шеннона-Фано и Хаффмена (рис.11.1).

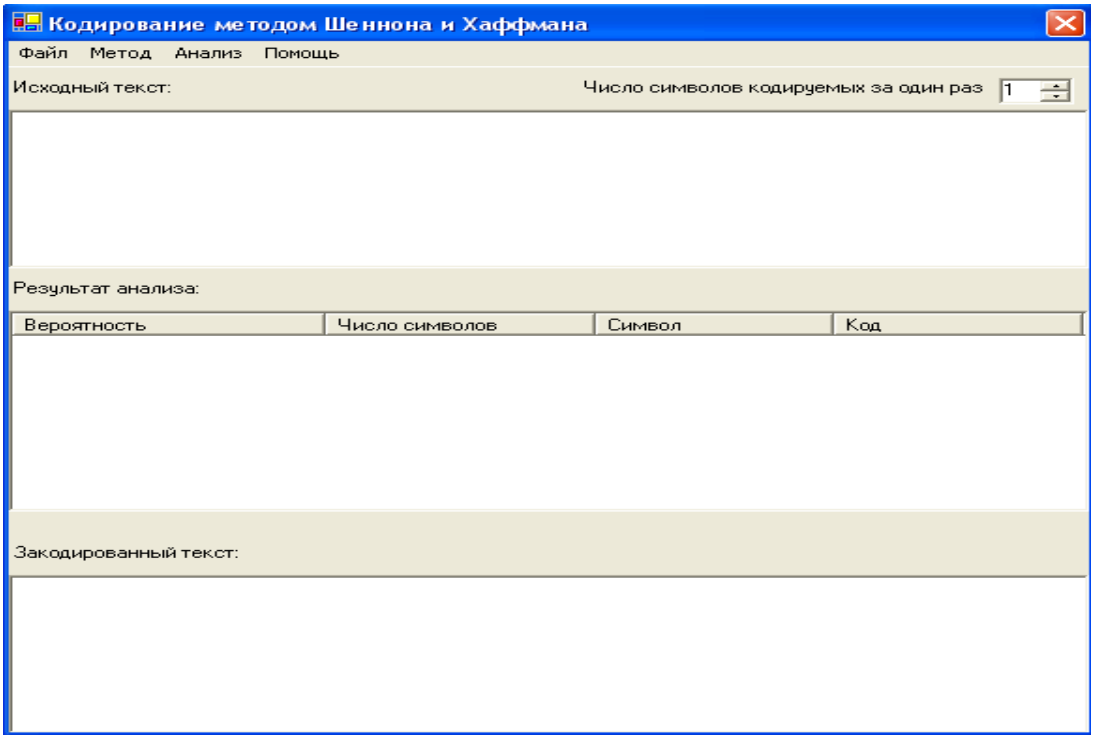


Рис. 11. 1. Главное окно программы

Для работы с программой пользователем выбирается требуемый метод сжатия, рис. 11.2.

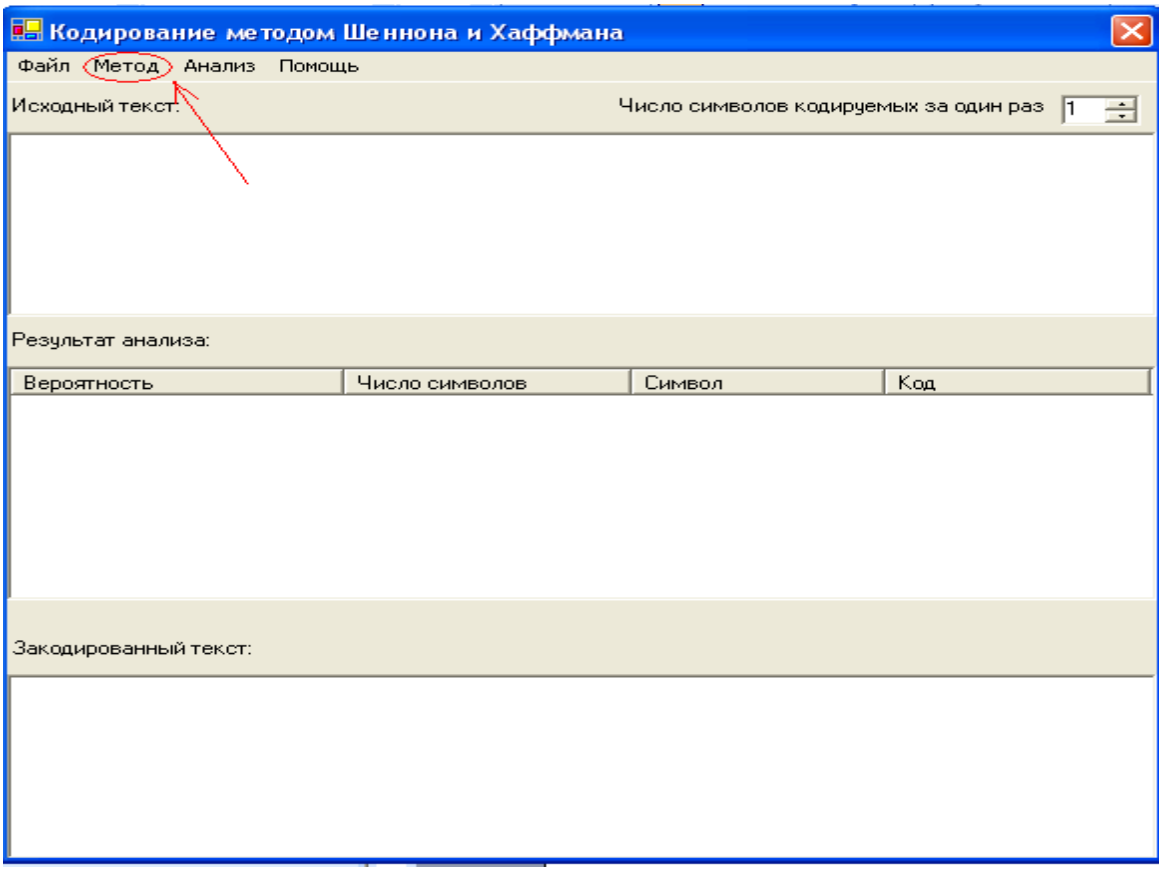


Рис. 11. 2. Выбор метода сжатия

Далее в окне программы *Исходный текст* записывается сообщение произвольной длины (или загружается сообщение из заранее подготовленного файла в формате *.txt*). После этого необходимо указать число символов, кодируемых за один раз.

Для подсчета вероятности появления букв во введенном сообщении *P*, определения энтропии источника сообщений *H*, среднего числа символов при кодировании одной буквы сообщения *L*, необходимо выбрать закладку *Анализ.*

Для определения эффективности кодирования рассчитывается избыточность кода *(L-H).*

Кратко ознакомится со сведениями о программе вы можете, выбрав закладку *Помощь* (рис.11.3).

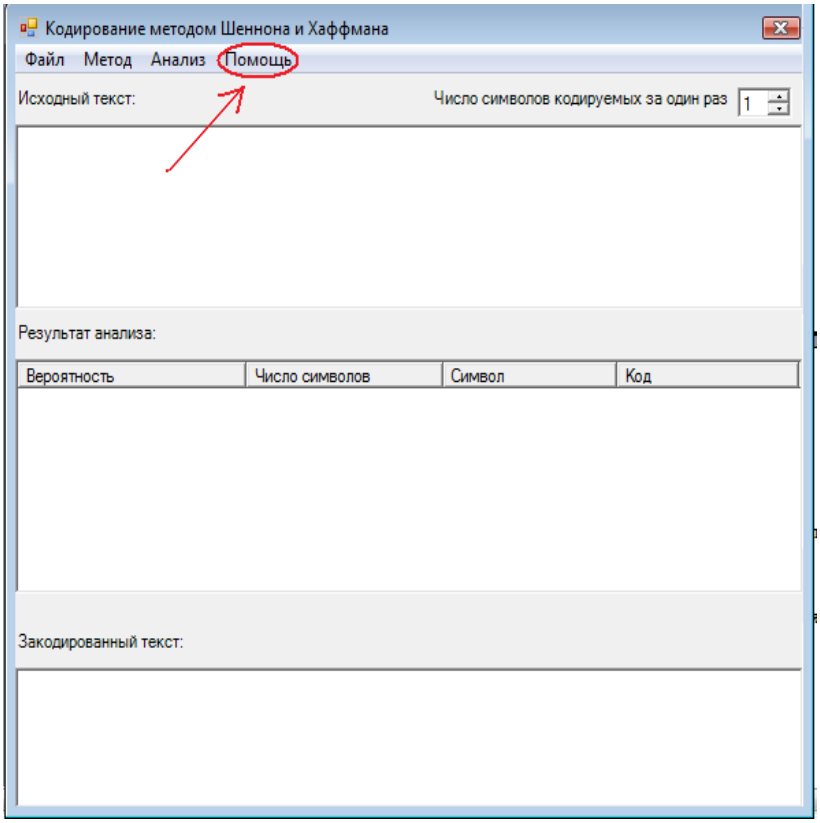


Рис. 11. 3. Просмотр сведений о программе

Чтобы выйти из данной программы достаточно выбрать закладку *Файл* и далее *Выход.*

Пример 1.

Исходное сообщение *аббввггггдддддеееее*, состоящее из символов шестибуквенного алфавита

$A = \{а, б, в, г, д, е\}$, закодируем, используя метод Хаффмена. Определяем вероятность появления символа в исходном сообщении

$$P = n/N,$$

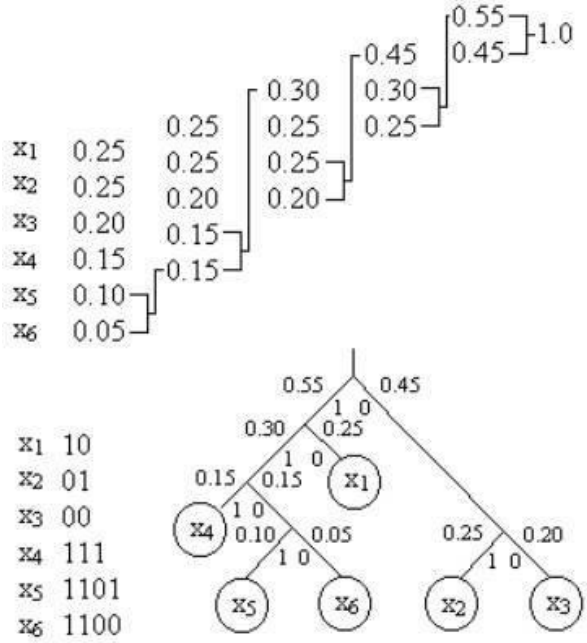
где n – число повторов символа в сообщении, N – длина сообщения.

Выписываем в столбец все символы алфавита в порядке убывания вероятности их появления в тексте (Таблица 11.1).

Таблица 11.1

симво	число повторений в данном	вероятнос
д	5	0,2
е	5	0,2
г	4	0,2
в	3	0,1
б	2	0,1
а	1	0,0

Последовательно объединяя два символа с наименьшими вероятностями появления в новый составной символ, вероятность появления которого полагается равной сумме вероятностей составляющих его символов, в конце концов мы построим дерево, каждый узел которого имеет суммарную вероятность всех узлов, находящихся ниже него. Прослеживаем путь к каждому листу дерева помечая направление к каждому узлу (например, направо - 0, налево - 1).



Среднее количество символов на букву сообщения:

$$L = \sum_{i=1}^{n} P(i)n(i) = 2*2*0.25+2*0.20+3*0.15+4*(0.10+0.05)=2.45,$$

где

n(i) – количество знаков в кодовой комбинации *i*-го символа алфавита

P(i) – вероятность появления *i*-го символа алфавита.

Энтропия:

$$H = \sum_{i=1}^{n} P(i)\log\frac{1}{P(i)} = 2.425$$

Значение избыточности кода (L-H)
= 0.025.

Сравниваем полученные данные с результатами работы программы (рис.11.4)

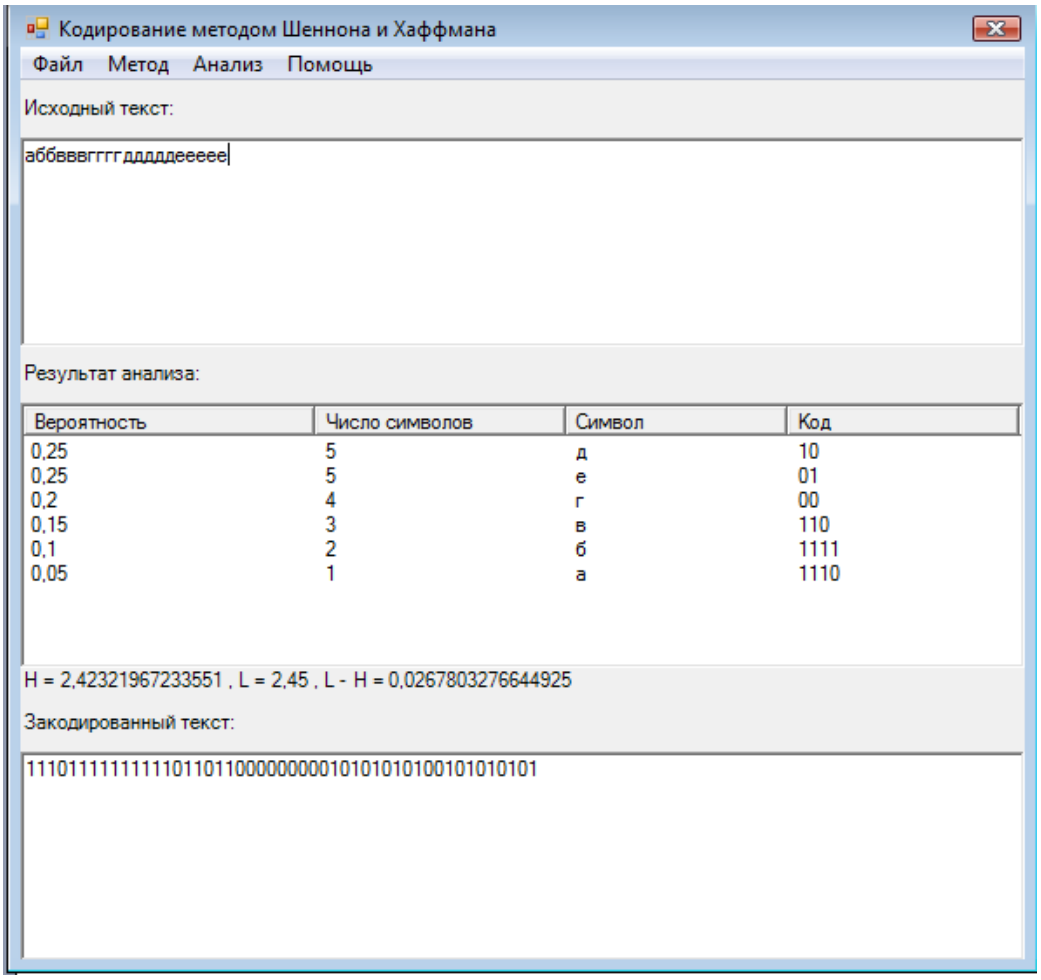


Рис. 11. 4. Пример кодирования

Пример 2.

Закодируем исходное сообщение (см. *Пример 1*), но используя метод Шеннона.

Выписываем в столбец все символы в порядке убывания вероятности появления их в тексте;

Последовательно делим множество символов на два подмножества так, чтобы сумма вероятностей появления символов одного подмножества была примерно равна сумме вероятностей появления символов другого. Для нижнего подмножества каждому символу приписываем "0", для верхнего – "1". Дальнейшие разбиения повторяются до тех пор, пока все подмножества не будут состоять из одного элемента (Таблица 11.2).

Таблица 11.2

символ	число повторений в данном сообщении	вероятность	Код Шеннона		
д	5	0,25	1	1	
е	5	0,25	1	0	
г	4	0,20	0	1	1

в	3	0,15	0	1	0
б	2	0,10	0	0	1
а	1	0,05	0	0	0

Средняя длина полученного кода будет равна
Среднее количество символов на букву сообщения:

$$L = \sum_{i=1}^{n} P(i)n(i) = 2 * 2)0.25 + 2 * 0.20 + 3 * 0.15 + 4 * (0.10 + 0.05) = 2.45$$

где
n(i) – количество знаков в кодовой комбинации *i*-го символа алфавита
P(i) – вероятность появления *i*-го символа алфавита.

Энтропия:

$$H = \sum_{i=1}^{n} \left(\mathrm{P(i)} \log \frac{1}{P(i)} \right) = 2/425$$

Значение избыточности кода (L-H) = 0.075.
Сравниваем полученные данные с результатами работы программы (рис.11.5)

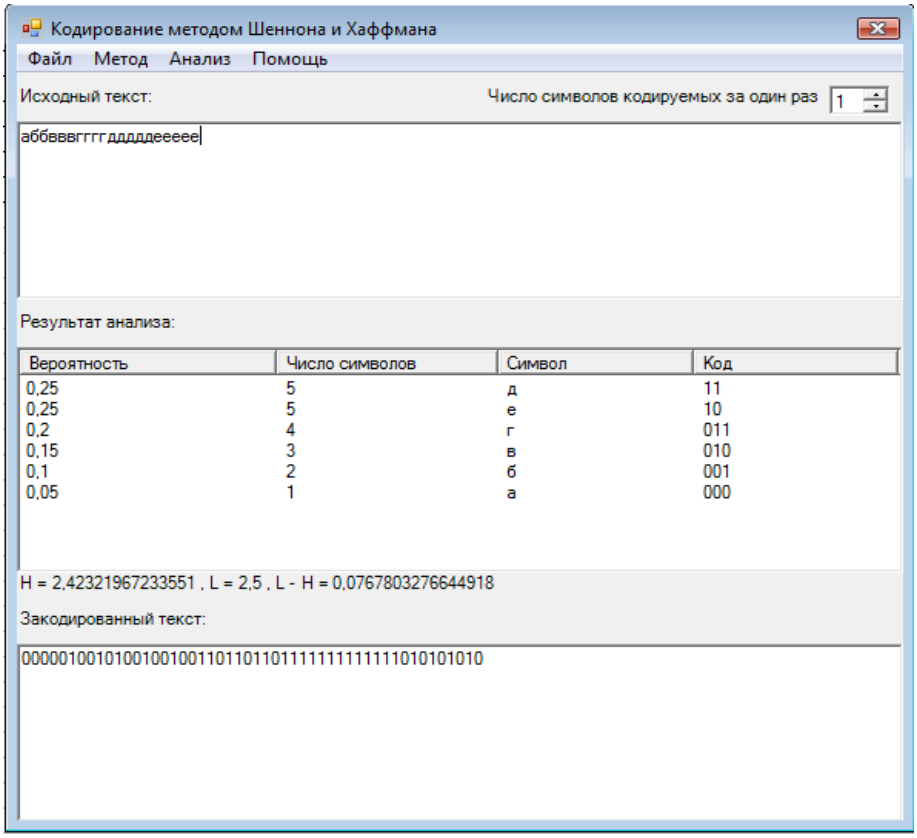


Рис. 11. 5. Пример кодирования

2. Задание

1.Ознакомиться со сведениями о программе *Shannon-Huffman.exe*, и рассмотреть примеры эффективного кодировании информации.

2. Запустить программу *Shannon-Huffman.exe*, предназначенную для демонстрации

методов сжатия информации с использованием алгоритмов Шеннона-Фано и Хаффмена.

3. Выполнить предложенные задания:

1). Число символов алфавита $k = m$ (m– номер варианта).

Составить такое исходное сообщение, чтобы

a) символы алфавита встречались в сообщении с равными вероятностями;
b) символы алфавита встречались в сообщении с разными вероятностями.

2). Ввести произвольный связный текст на русском языке.

Это может быть пословица, стихотворение или произвольный текст. Используя результаты работы программы, следует проанализировать алфавит введенного сообщения: подсчитать количество символов алфавита, значение энтропии H, среднее количество символов на знак L при целочисленном кодировании.

4. Сохранить в отчете экранные формы, демонстрирующие процесс сжатия информации.
5. Отчет по лабораторной работе должен содержать результаты анализа программы *Shannon-Huffman.exe* при кодировании; алгоритм построения кода Хаффмена и Шеннона с применением знаний по данной тематике; результаты сравнения различных методов кодирования; выводы об эффективности используемых методов сжатия.
6. Привести в отчете ответы на контрольные вопросы, в соответствии с номером варианта, указанным преподавателем.

Включить в отчет о лабораторной работе выполненные задания 1, 2, 3.

Варианты заданий

Задание1.

Сообщение состоит из последовательности двух букв A и B, вероятности появления каждой из которых не зависят от того, какая была передана раньше, и равны 0,8 и 0,2 соответственно. Произведите кодирование по методу Шеннона: а) отдельных букв; б) блоков, состоящих из двухбуквенных сочетаний; в) блоков, состоящих из трехбуквенных сочетаний. Сравните полученные коды по их эффективности.

Задание2.

Составьте текст, который бы соответствовал данным, приведенным в Таблице 11.3.

Используя программу *Shannon-Huffman.exe* закодируйте текст методом Хаффмена.

Таблица 11.3

Номер Варианта	вероятность	символ	число символов
1,5,7,	0,333333333	o	2

3,9,12,14,18,28	0,166666667	г	1
	0,166666667	р	1
	0,166666667	д	1
	0,166666667	а	1
2,4,6,8,16,21 20,22,24, 30	0,25	е	2
	0,25	т	2
	0,125	о	1
	0,125	п	1
	0,125	р	1
	0,125	а	1
11,13,15, 10,17,19,23,25	0,25	р	2
	0,25	а	2
	0,25	с	2
	0,125	т	1
	0,125	е	1

Задание3.

Для вариантов А),Б),В) (см.рис.11.6 и Таблица 4) составьте код Хаффмена. Сделайте подсчет среднего количества символов на знак; избыточности *(L – H)*; и относительной избыточности полученного кода *(L – H) / L*. Сравните полученные значения с *L, H, (L-H)* для кода Шеннона, сделайте выводы.

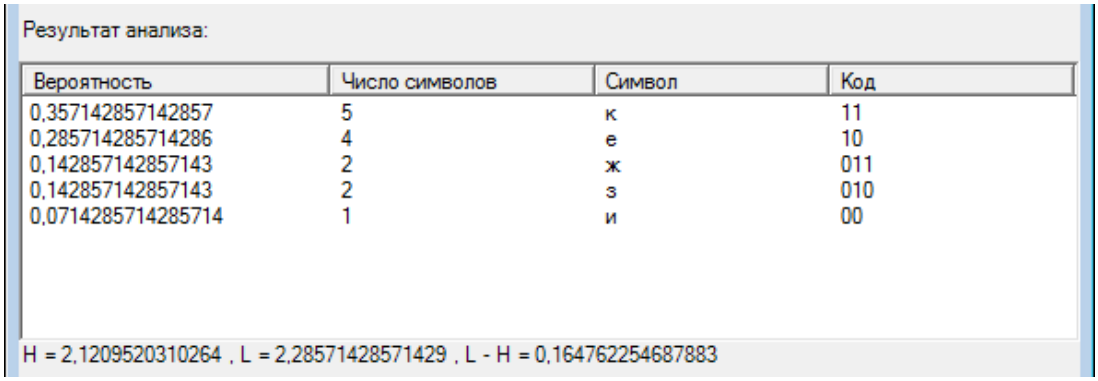
А)

Результат анализа:

Вероятность	Число символов	Символ	Код
0,4375	7	а	11
0,1875	3	б	10
0,1875	3	г	01
0,125	2	в	001
0,0625	1	д	000

H = 2,05242128382936 , L = 2,1875 , L - H = 0,135078716170635

Б)



В)

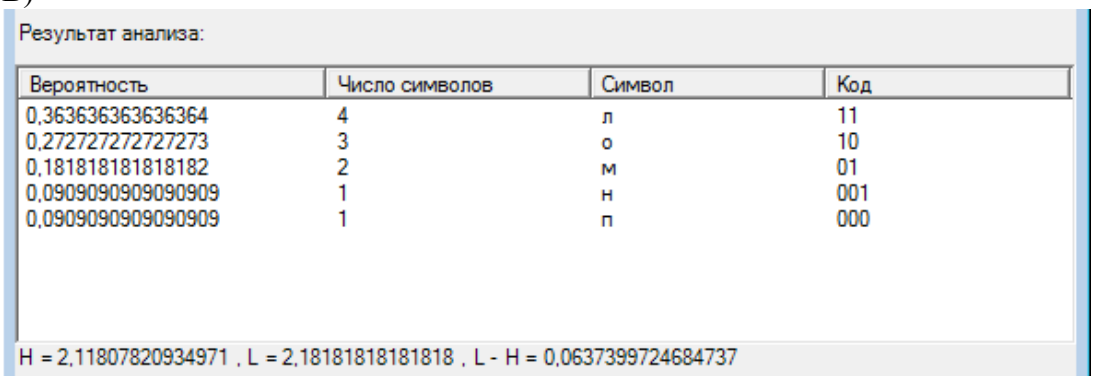


Рис. 11. 6. Варианты заданий

Таблица 11.4

Номер варианта	Задание
1,5,7, 3,9, 12,14,18, 28	*Задание 3*. Вариант А).
2,4,6,8, 20,22,23, 24,25 30	*Задание 3*. Вариант Б).
11,13,15, 16 21,10,17, 19	*Задание 3*. Вариант В).

Номер варианта	**Контрольные вопросы**

1,5,7, 3,9,18,28	Какие коды позволяют производить однозначное декодирование даже без использования разделительного символа? Приведите примеры таких кодов.
2,4,6,8, 20,22,24, 26,30	Назовите условия построения оптимальных кодов.
11,13,15, 10,17,19, 27	С какой целью используются эффективные коды, и какие из них вам известны?
12,14,16 21,23.25, 29	Перечислите основные методы сжатия информации без потерь.

ЛАБОРАТОРНАЯ РАБОТА № 12 LZW - СЖАТИЕ

Цель работы: Ознакомление с принципами сжатия информации с использованием метода *LZW (Lempel-Ziv-Welch).*

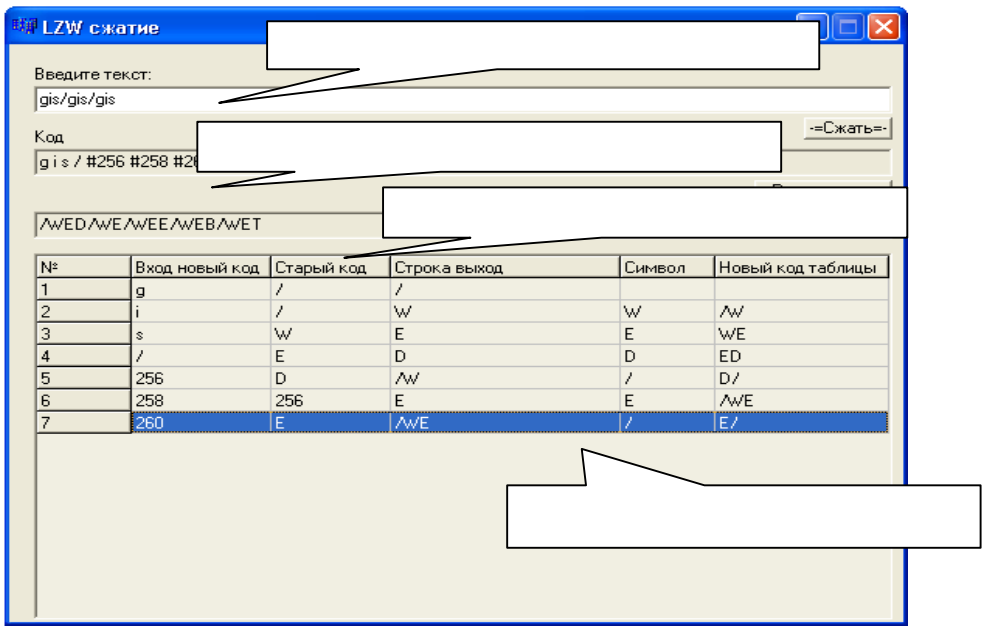
Примечание. Для выполнения лабораторной работы на компьютере необходимо установить файл *LZW.exe.*

1. Описание лабораторной работы

Лабораторная работа выполняется на персональном компьютере в среде программы *LZW.exe.*

При запуске программы *LZW.exe* появляется окно, рис. 12.1, содержащее следующие компоненты:

- строка текста, где вводится текст для сжатия;
- строка кода, где выводится код сжатого текста;
- строка кода, где выводится распакованный текст;
- таблица выполнения кодирования программой.
-



строка текста, где вводится текст для сжатия

строка кода, где выводится код сжатого текста

строка кода, где выводится распакованный текст

таблица выполнения кодирования программой

Рис. 12. 1. Главное окно программы

Программа позволяет сжимать и распаковывать текст любой длины, записанный как на кириллице, так и на латинице, а также просматривать процесс сжатия и распаковки текста.

Для сжатия какого-либо текста следует выполнить следующие действия:

1. В строку текста «Введите текст» ввести текст для сжатия.
2. Нажать кнопку «Сжать».
3. В стоке «Код» появится код сжатого текста.
4. Нажать кнопку «Распаковать».
5. В следующей строке появится распакованный текст.
6. Просмотреть таблицу, выполнения программы сжатия и распаковки текста.

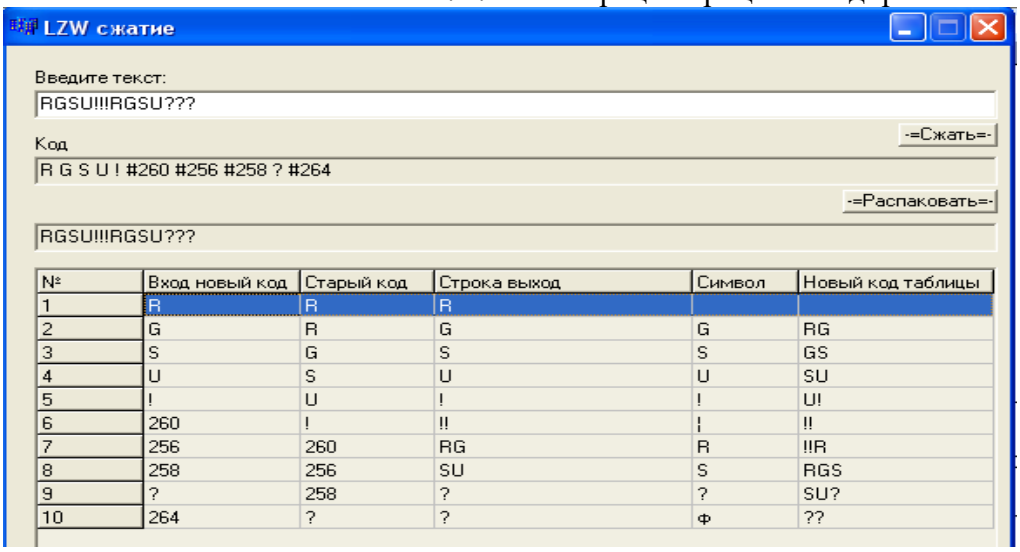
Пример:

- Исходный текст: **RGSU!!!RGSU???**
- Код сжатого текста: **R G S U ! #260 #256 #258 ? #264**
- Распакованный текст: **RGSU!!!RGSU???**

Процесс кодирования проиллюстрирован на рис.12.2 и рис.12.3.

№	Вход новый код	Старый код	Строка выход	Символ	Новый код таблицы
1	R	R	R		
2	G	R	G	G	RG
3	S	G	S	S	GS
4	U	S	U	U	SU
5	!	U	!	!	U!
6	260	!	!!	¦	!!
7	256	260	RG	R	!!R
8	258	256	SU	S	RGS
9	?	258	?	?	SU?
10	264	?	?	ф	??

Рис. 12. 2. Иллюстрация процесса кодирования



№	Вход новый код	Старый код	Строка выход	Символ	Новый код таблицы
1	R	R	R		
2	G	R	G	G	RG
3	S	G	S	S	GS
4	U	S	U	U	SU
5	!	U	!	!	U!
6	260	!	!!	¦	!!
7	256	260	RG	R	!!R
8	258	256	SU	S	RGS
9	?	258	?	?	SU?
10	264	?	?	ф	??

Рис. 12. 3. Иллюстрация процесса кодирования

Примечание LZW-сжатие выделяется среди прочих, когда встречается с потоком данных, содержащим повторяющиеся строки любой структуры, уровень сжатия может достигать 50% и выше.

2. Задание

1. Сжать и затем распаковать исходные строки символов латинского алфавита, которые набираются непосредственно в окне программы. Сохранить в отчете экранные формы, демонстрирующие процесс сжатия и распаковки информации. Проанализировать сделанную работу и сделать выводы об эффективности алгоритма сжатия и распаковки.
2. Сжать и затем распаковать исходные строки символов кириллического алфавита, которые

набираются непосредственно в окне программы. Сохранить в отчете экранные формы, демонстрирующие процесс сжатия и распаковки информации. Проанализировать сделанную работу и сделать выводы об эффективности алгоритма сжатия и распаковки.

3. Сжать и затем распаковать исходные строки чисел, которые набираются непосредственно в окне программы. Сохранить в отчете экранные формы, демонстрирующие процесс сжатия и распаковки информации. Проанализировать сделанную работу и сделать выводы об эффективности алгоритма сжатия и распаковки.

4. Сжать и затем распаковать исходные строки математических функций и формул, которые набираются непосредственно в окне программы. Сохранить в отчете экранные формы, демонстрирующие процесс сжатия и распаковки информации. Проанализировать сделанную работу и сделать выводы об эффективности алгоритма сжатия и распаковки.
5. Сжать и затем распаковать исходные строки произвольных символов, которые набираются непосредственно в окне программы. Сохранить в отчете экранные формы, демонстрирующие процесс сжатия и распаковки информации. Проанализировать сделанную работу и сделать выводы об эффективности алгоритма сжатия и распаковки.
6. Включить в отчет:

- результаты сжатия и распаковки текстов (варианты указаны в

  Таблице 12.1) с помощью программы LZW-сжатия;

- сохранить в отчете кодовую таблицу в виде таблицы или рисунка;

- проанализировать полученные результаты и сформулировать аргументированные выводы.

7. Привести в отчете ответы на контрольные вопросы, в соответствии с номером варианта, указанным преподавателем.

Таблица 12.1

№ варианта					
1	Program	Информация	612534712	X+Y+Z-12=5	!!A/b/C/dE@?
2	Processor	Математика	954723890	2*X+Y+Z=2	ЪС<л>*ш/dE?
3	Operation	Язы	410125935	X*X+2*Y-3	[4Pu/t%/3$1$
4	Information	Вычисления	181952471	Y*Y+Z=4*X	Ё1?QяЧ*/!Д/;
5	Mathematics	Общество	365412389	X+4Y+Z-2+1	ЙрТиV{G}%
6	Language	Технология	113212598	X+2Y+5*Z=	ёЮВ&<h>@
7	Company	Кодирование	725847391	5*X+Y+8Z=	ЖО[Д]m/я/12
8	Encoding	Информатика	820191856	X-6*Y-3*Z=2	ВН!(И)13№
9	Informatics	Обеспечение	100258741	X-2*Y+7*Z-9	Б!15fЛ2ыф»
10	Software	Операция	101296752	8*X/Y+Z/2=5	у4?-+/25/8?8
11	Editor	Редактор	496123578	5X/2+Y/12+	uФ/z-1/2+3#
12	Table	Процессор	123457890	X/5+5Y+Z/1	Г@14^5D/G
13	Security	Таблица	154789123	7*X>Y+2*8Z	г13#&7Fоэ,
14	Cybernetics	Безопасность	571236984	X*X4*Y<Z+5	ЩП/’ы’45*7
15	Knowledge	Знания	432987067	X+3*Y>Z+2	i5(B)<px)\Й

16	Computer	Кибернетика	433129851	X/2+2*Y>3*	шЁ8впо/тп5?
17	System	Компьютер	184569347	7X+Y-	bРлQas156+
18	Swindle	Система	189242578	Y<5*X/3+Z-	Z~+258|=[f]
19	Logic	Мошенничество	213459632	2*Z>4*X+5Y	С8&8вАъ/ё
20	Programming	Логика	214357980	X/5+Y/4+Z=	n-q6&8%^
21	Hardware	Программирование	896124587	Y/4-	S?dkошй^*
22	Intellec	Техника	654932140	X+2>Z/4Y+2	p$&Ghgdku
23	Internet	Интеллект	781258743	X+7*Z<2Y+	j5%jkЖЁ%
24	Translator	Интернет	834107892	2*X-7Z=2+Y	Fd&5&%№
25	Driver	Транслятор	417625877	6X*X+2*Y-3	Dfj&5*6-2

Номер варианта	**Контрольные вопросы**
1,5,7, 3,9,18,28	В чем состоит принцип LZW-сжатия?
2,4,6,8, 20,22,24, 26,30	Какие форматы файлов наиболее эффективно архивировать с помощью LZW-сжатия ?
11,13,15, 10,17,19,	В чем особенность процесса распаковки при использовании алгоритма LZW?
12,14,16 21,23.25, 29	Какими достоинствами и недостатками обладает алгоритм LZW?

5. **Образовательные технологии**

Основными образовательными технологиями проведения курса «Управление информационной безопасность» являются:

- Лекции, сопровождаемые компьютерными презентациями;
- практические и лабораторные работы, в рамках которых составляются и тестируются программы,иллюстрирующие теоретический материал лекций;
- самостоятельная работа студентов, включающая усвоение теоретического материала, поиск дополнительного материала и эффективных способов выполнения заданий, завершение выполнения лабораторных работ; оформление и подготовка к защите лабораторных работ, подготовка к текущему контролю знаний и к итоговому экзамену;
- разработанные индивидуальные задания для самостоятельной работы;
- рейтинговая технология контроля учебной деятельности студентов для обеспечения их ритмичной работы в течение семестра
- консультирование студентов по вопросам учебного материала и выполнения курсового заданий.

6. Учебно-методическое обеспечение самостоятельной работы студентов

При изучении дисциплины «Управление информационной безопасностью» обязательными являются следующие виды самостоятельной работы:

- разбор теоретического материала по учебным пособиям и конспектам лекций;
- самостоятельное изучение указанных теоретических вопросов; подготовка к проведению ситуационных моделей в интерактивной форме;

№ темы дисциплины	Форма самостоятельной работы	Трудоемкость в часах
1–8	Работа с учебной литературой. Разбор вопросов по теме занятия. Работа с источниками и поиск информаций в Интернете. Подготовка устного доклада. Подготовка к самостоятельной проверочной работе.	32
1-8	Выполнение контрольной работы.	16
4, 6	Подготовка к интерактивному занятию	10
Итого:		**58**

Контроль результатов освоения дисциплины

Текущий контроль успеваемости осуществляется путем оценки результатов выполнения заданий лабораторных, самостоятельной работ, посещения лекций.

Промежуточная аттестация осуществляется в форме экзамена, который выставляется по результатам проверки выполнения тестов и заданий.

Оценочные средства результатов освоения дисциплины, критерии оценки выполнения заданий представлены в разделе «Фонды оценочных средств для проведения промежуточной аттестации» и фонде оценочных средств образовательной программы.

7. Фонд оценочных средств для проведения текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины.

7.1. Перечень компетенций с указанием этапов их формирования в процессе освоения образовательной программы.

Перечень компетенций с указанием этапов их формирования приведен в описании образовательной программы.

Код компетенции из ФГОС ВО	Наименование компетенции из ФГОС ВО	Планируемые результаты обучения	Процедура освоения
ПК-22	способностью проводить сбор, анализ научно-технической информации, отечественного и зарубежного опыта по тематике исследования	Знает: - элементарные логические методы и приемы научного исследования; - основные методологические теории и принципы современной науки. Умеет: - осуществлять сбор и анализ научно-технической информации, полученной из отечественных и зарубежных источников и литературы, в том числе посвященных информационным системам и технологиям Владеет: - элементарными методами научного поиска и интеллектуального анализа научной информации при решении новых задач; - основными методами сбора, обработки и анализа научно-технической	Устный опрос, письменный опрос

		информации, полученной из отечественных и зарубежных источников и литературы, посвященных информационным системам и технологиям;	
ПК-24	способностью обосновывать правильность выбранной модели, сопоставляя результаты экспериментальных данных и полученных решений	Знает: - способы обработки, анализа и сравнения результатов, полученных в ходе теоретического расчёта, а также численного и натурного моделирования с помощью стандартных научных пакетов; - методы анализа, теоретического и экспериментального исследования для разработки, проектирования, эксплуатации и поддержки информационных систем. Умеет: - проводить оценку границ применимости физических моделей; - корректно подходить к решению проблемы выбора аналитической и численной моделей, организации вычислительного эксперимента; - оценить достоинства, недостатки и рамки применимости того или иного метода на практике; Владеет: - навыками подбора адекватных методов для составления математических моделей физических явлении и их решения; - методами корректной компьютерной обработки и последующего анализа результатов математического моделирования;	Устный опрос, письменный опрос
ПК-34	способностью к инсталляции, отладке программных и настройке технических средств для ввода информационных систем в опытную и промышленную эксплуатацию	Знает: структуру программного и технического обеспечения, их основные функции и характеристики, методы инсталляции, отладку программных и настройку технических средств, механизмы администрирования, тенденции их развития (управление распределением памяти для объектов ИС, установление квот памяти для пользователей ИС, управления доступностью данных, включая режимы (состояния)). Умеет: выполнять процедуры настройки технических средств информационных систем. Владеет: средствами и средой программирования, современной технологиями программирования, методами настройки и отладки осуществления перехода от управления функционированием отдельных устройств к анализу трафика в отдельных участках сети.	Устный опрос, письменный опрос
ПК-35	способностью проводить сборку информационной системы из готовых компонентов	Знает: структуру, состав и свойства информационных процессов, систем и технологий, методы анализа информационных систем, модели представления проектных решений, конфигурации информационных систем; структуру, принципы реализации и функционирования информационных технологий, используемых при создании информационных систем, базовые и прикладные информационные технологии,	Устный опрос, письменный опрос

		инструментальные средства информационных технологий, состав и свойств готовых компонентов, принципы их адаптации. Умеет: использовать архитектурные и детализированные решения при проектировании систем; применять готовые компоненты информационные технологии и систем при проектировании информационных систем. Владеет: средствами разработки архитектуры информационных систем на основе готовых компонентов; технологиями адаптации типовых проектных решений.	
ПК-36	способностью применять основные приемы и законы создания и чтения чертежей и документации по аппаратным и программным компонентам информационных систем	Знает: основные приемы и законы создания и чтения документации по компонентам информационных систем Умеет; применять основные приемы и законы создания и чтения документации по компонентам информационных систем Владеет: практическими навыками применения основных приемов и законов создания и чтения чертежей и документации по программным компонентам информационных систем	Устный опрос, письменный опрос
ПК-37	способностью выбирать и оценивать способ реализации информационных систем и устройств (программно-, аппаратно- или программно-аппаратно-) для решения поставленной задачи	Знает: технологии выбора и оценки способов реализации ИС Умеет: выбирать и оценивать существующие технологии разработки ИС для решения поставленной задачи Владеет: практическими навыками выбора и оценки современных технологий проектирования и разработки ИС для решения поставленной задачи	Устный опрос, письменный опрос

7.2. Типовые контрольные задания

7.3. Методические материалы, определяющие процедуру оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций.

Общий результат выводится как интегральная оценка, складывающая из текущего контроля - 30% и промежуточного контроля - 70%.

Текущий контроль по дисциплине включает:

- посещение занятий - _0__ баллов,
- участие на практических занятиях - 20 баллов,
- выполнение лабораторных заданий – 60 баллов,
- выполнение домашних (аудиторных) контрольных работ – 20 баллов.

Промежуточный контроль по дисциплине включает:

- устный опрос - 30 баллов,
- письменная контрольная работа - 30 баллов,
- тестирование - 40 баллов.

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

а) основная литература:

1. **Правовое обеспечение информационной безопасности** : [учеб. пособие для вузов по специальностям 075200 "Компьютер. безопасность", 075500 "Комплекс. обеспечение информ. безопасности и автоматизир. систем", 075600 "Информ. безопасность телекоммуникац. систем" / С.Я.Казанцев и др.]; под ред. С.Я.Казанцева. - М. : Academia, 2005. - 239 с. : ил. ; 22 см. - (Высшее профессиональное образование. Информационная безопасность). - Библиогр.: с. 235-237. - Допущено УМО. - ISBN 5-7695-1209-1 : 129-47

2. **Филин С. А.** Информационная безопасность : учеб. пособие / Филин, Сергей Александрович. - М. : Альфа-Пресс, 2006. - 411 с. - ISBN 5-94280-163-0 : 129-03.

3. **Галатенко В. А.** Стандарты информационной безопасности : курс лекций: учеб. пособие / Галатенко, Владимир Антонович ; под ред. В.Б.Бетелина; Интернет-ун-т информ. технологий. - 2-е изд. - М. : ИНТУИТ.ру, 2006. - 263 с. - (Основы информационных технологий). - ISBN 5-9556-0053-1 : 176-00.

4. **Уколов В. Ф.** Теория управления : учеб. для вузов / Уколов, Владимир Фёдорович, А. М. Масс, И. К. Быстряков. - 3-е изд., доп. - М. : Экономика, 2007. - 696 с. - Допущено МО РФ. - ISBN 978-5-282-02698-6 : 260-00

5. **Галатенко В. А.** Основы информационной безопасности : учеб. пособие для студентов вузов, обуч. по специальности 351400 "Прикл. информ." / Галатенко, Владимир Антонович. - 4-е изд. - М. : Изд-во Интернет-Ун-та Информ. Технологий: БИНОМ. Лаб. знаний, 2016, 2008, 2006. - 205 с. - (Основы информационных технологий). - Рекомендовано УМО. - ISBN 978-5-94774-821-5 : 230-00.

6. Петров С. В. Информационная безопасность : учеб. пособие /. - Новосибирск: М. : АРТА, 2012. - 439-77.

б) дополнительная:

1. ISO/IEC 27001:2005 Information technology – Security techniques – Information security management systems – Requirements.Международный стандарт. ISO/IEC 27000:2005 Информационные технологии. Методы обеспечения безопасности. Определения и основные принципы./ http://www.27000.org/
2. Аудит информационной безопасности. Под ред. А.П.Курило. – М: БДЦ-Пресс, 2014.
3. Международный стандарт. ISO/IEC 27001:2005 Информационные технологии. Методы обеспечения безопасности. Системы управления информационной безопасностью. Требования (BS 7799-2:2005)./ http://www.27000.org/
4. Международный стандарт. ISO/IEC 27002:2005 Информационные технологии. Методы обеспечения безопасности. Практические правила управления информационной безопасностью./ http://www.27000.org/Международный стандарт. ISO/IEC 27003:2005 Информационные технологии. Методы обеспечения безопасности. Руководство по внедрению системы управления информационной безопасностью./ http://www.27000.org/
5. Международный стандарт. ISO/IEC 27004:2005 Информационные технологии. Методы обеспечения безопасности. Измерение эффективности системы управления информационной безопасностью./ http://www.27000.org/
6. Международный стандарт. ISO/IEC 27005:2005 Информационные технологии. Методы обеспечения безопасности. Управление рисками информационной безопасности./ http://www.27000.org/
7. Международный стандарт. ISO/IEC 27006:2005 Информационные технологии. Методы обеспечения безопасности. Требования к органам аудита и сертификации систем управления информационной безопасностью./ http://www.27000.org/
8. Международный стандарт. ISO/IEC 27007:2005 Информационные технологии. Методы обеспечения безопасности. Руководство для аудитора систем управления информационной безопасностью./ http://www.27000.org/

9. Петренко С., Симонов С. Управление информационными рисками. Экономически оправданная безопасность. — М.: АйТи-Пресс, 2012.
10. Петренко С.А., Курбатов В.А. Политики информационной безопасности. - М.: ДМК пресс, 2013.
11. Репин В., Елиферов В. Процессный подход к управлению. Моделирование бизнес-процессов. М.: Стандарты и качество, 2014.
12. Романов О.А., Бабин С.А., Жданов С.Г. Организационное обеспечение информационной безопасности. – М.: Академия, 2008 г. – 192 стр.
13. Золотарев Управление информационной безопасностью. Ч. 1. Анализ информационных рисков - Красноярск: Сибирский государственный аэрокосмический университет имени академика М. Ф. Решетнева, 2010.

9. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины.

1. ЭБС IPRbooks: http://www.iprbookshop/ru/
2. Электронно-библиотечная система «Университетская библиотека онлайн»(архив):www.biblioclub.ru
3. Единое окно доступа к образовательным ресурсам. http://window.edu.ru/
4. http://www.microsoft.com/msf
5. http://www.uml.org
6. http://www.wikipedia.org

10. Методические указания для обучающихся по освоению дисциплины.

Критерии и показатели сформированности компетенций

Степень (уровень) сформированности компетенций на этапе изучения дисциплины «Управление информационной безопасностью» оценивается по следующим критериям: мотивационно-ценностный, когнитивный, операционно-деятельностный. Показателями критериев являются результаты обучения по дисциплине (дескрипторы) таблицы 1. Инструментарий, этапы измерения показателей и критериев компетенции представлены в таблице.

Критерии и показатели сформированности компетенции ПК-37

Критерии сформированности компетенции	Способы оценки	
	Этапы контроля	Средства оценки
Мотивационно-ценностный критерий	4, 6, 7, экзамен	1
Когнитивный критерий	4, 5, 6, 7	1
Операционно-деятельностный критерий	4, 5, 7	1
	6, экзамен	
Интегральная оценка	Экзамен	1

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ДЛЯ ПРОВЕДЕНИЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ

Этапы контроля: раздел 2 (самостоятельная работа), раздел 3 (самостоятельная работа), раздел 4 (самостоятельная работа), раздел 5 (самостоятельная работа), раздел 6 (самостоятельная работа), раздел 7 (самостоятельная работа), экзамен.

Время на выполнение: 60 мин.

Метод оценивания: автоматизированный

Критерии оценки результатов выполнения: менее 50% правильных ответов - неудовлетворительно, менее 65% - удовлетворительно, менее 86% хорошо, 86% и более – отлично.

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

Информационные технологии

Образовательный процесс осуществляется с применением локальных и распределенных информационных технологий (таблица 4, 5).

Таблица 4 – Локальные информационные технологии

Группа программных средств	Наименование программного продукта
Офисные программы	Microsoft Office
	Libre Office
Распознавание текста и речи	ABBYY FineReader 2010
Средства разработки	MicroSoft Visual Studio 2015 MicroSoft SQL Server 2012 VipNet Client 4 Dallas Lock 8.0 КриптоПро CSP

Таблица 5 – Распределенные информационные технологии

Группа	Наименование
Система тестирования	Система сетевого компьютерного тестирования ДГУ www.ts.icc.dgu.ru
Библиотеки и образовательные ресурсы	Электронная библиотека ДГУ http://www.elib.dgu.ru
	Кафедральные сайты ДГУ http://cafedra.dgu.ru
	Сайте электронных образовательных ресурсов ДГУ http://eor.dgu.ru
Система электронного обучения	Сервер электронного обучения moodle http://moodle.dgu.ru

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

Таблица 6 – Материально-техническая база

Помещения для осуществления образовательного процесса	Перечень основного оборудования (с указанием кол-ва посадочных мест)	Адрес (местоположение)
Аудитории для проведения лекционных занятий		
Лекционные аудитории	Интерактивная доска, ноутбук; проектор. Количество посадочных мест – 30.	Ауд. 3-14, 4-16, 2-10, учебный корпус № 8, г.Махачкала, ул. Джержинского, 12.
Аудитории для проведения лабораторных занятий, контроля успеваемости		
Компьютерный класс	Компьютеры с выходом в Интернет и доступом в электронную информационно-образовательную среду вуза. Количество посадочных мест – 15.	Компьютерный зал № 2 учебный корпус № 3, г.Махачкала, ул. Джержинского, 12.
Помещения для самостоятельной работы		
Компьютерные классы	Компьютеры с выходом в Интернет и доступом в электронную информационно-образовательную	Компьютерный зал № 1, учебный

	среду вуза. Количество посадочных мест – 15	корпус № 3, г. Махачкала, ул. Джержинского, 12.
Читальный зал библиотеки ДГУ	Компьютеры с выходом в Интернет и доступом в электронную информационно-образовательную среду вуза. Количество посадочных мест – 30.	Электронный читальный зал научной библиотеки ДГУ, г. Махачкала, ул. Батырая, 4