Министерство образования и науки Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего профессионального образования
«Российский государственный профессионально-педагогический университет»
Институт электроэнергетики и информатики
Кафедра микропроцессорной управляющей вычислительной техники

ЗАДАНИЯ И МЕТОДИЧЕСКИЕ УКАЗАНИЯ К ВЫПОЛНЕНИЮ КОНТРОЛЬНОЙ РАБОТЫ ПО ДИСЦИПЛИНЕ «ОСНОВЫ КОДИРОВАНИЯ ИНФОРМАЦИИ»

для студентов всех форм обучения
направления подготовки 051000.62 Профессиональное обучение (по отраслям)
профиля подготовки «Энергетика»
профилизации «Компьютерные технологии автоматизации и управления»

Екатеринбург
РГППУ
2013

Задания и методические указания к выполнению контрольной работы по дисциплине «Основы кодирования информации». Екатеринбург, ФГАОУ ВПО «Российский государственный профессионально-педагогический университет», 2013. 21 с.

Авторы: ст. преподаватель Т.П. Телепова
канд. техн. наук, доцент А.А. Карпов

Одобрена на заседании кафедры микропроцессорной управляющей вычислительной техники. Протокол от 31 января 2013 г. № 5.

Заведующий кафедрой
микропроцессорной управляющей
вычислительной техники С.Б. Петров

Рекомендована к печати методической комиссией Института электроэнергетики и информатики РГППУ. Протокол от 11 февраля 2013 г. №6.

Председатель методической комиссии ЭлИн А.О. Прокубовская

СОГЛАСОВАНО
Зав. сектором инспектирования ИМО УМУ С.В. Пеннер

И.о. директора ЭлИн А.А. Карасик

ВВЕДЕНИЕ

Контрольная работа по дисциплине «Основы кодирования информации» предназначена для студентов, обучающихся по направлению подготовки 051000.62 Профессиональное обучение (по отраслям) профиль «Энергетика», профилизация «Компьютерные технологии автоматизации и управления».

Выполнение контрольной работы является одной из форм самостоятельной работы студентов и имеет важное значение при изучении дисциплины «Основы кодирования информации».

Целью контрольной работы является закрепление у студентов изученного теоретического материала и формирование практических умений применения математических методов представления информации, методов кодирования.

Контрольная работа включает в себя выполнение трёх заданий, охватывающих наиболее важные вопросы дисциплины. В каждом из предложенных заданий содержится 30 вариантов исходных данных, номера которых задаются преподавателем.

Задания по контрольной работе сопровождаются методическими указаниями, краткими теоретическими сведениями, необходимыми для выполнения задания, и примерами конкретного выполнения.

При выполнении контрольной работы необходимо выполнить требования по её оформлению:

– контрольная работа может быть оформлена в тетради или распечатана на листах А4 (титульный лист в приложении А);

– выполнению каждого задания должны предшествовать постановка задачи и исходные данные, соответствующие заданному варианту;

– работа должна быть оформлена грамотно и аккуратно;

– контрольная работа должна быть зарегистрирована в деканате.

ЗАДАНИЕ 1

1. Определить энтропию и избыточность источника независимых сообщений A, кодируемых с помощью символов a_1, a_2, a_3, a_4,. Вероятности появления символов равны соответственно p_1, p_2, p_3, p_4.

2. Определить условную энтропию источника сообщений A, кодируемых с помощью символов a_1, a_2, a_3, a_4. Зависимость между символами задана матрицей условных вероятностей $M(p_{(aj/ai)})$. Безусловные вероятности символов равны соответственно p_1, p_2, p_3, p_4. Каждая строка матрицы условных вероятностей формируется из значений p_1, p_2, p_3, p_4 в соответствии с указанными в таблице вариантами. Например, для варианта *1* строки матрицы формируются: первая строка – *0,5 0,1 0,3 0,1* (из варианта 2); вторая строка – *0,6 0,2 0,1 0,1* (из варианта 3); третья строка – *0,25 0,3 0,02 0,43* (из варианта 4); четвертая строка – *0,2 0,35 0,25 0,2* (из варианта 5).

Методические указания к выполнению задания

Под дискретным источником сообщений можно рассматривать устройство, выдающее некоторое дискретное сообщение в виде последовательности символов a_i $(i = 1, 2, \ldots, m_1)$, принадлежащих исходному кодируемому алфавиту A. Величина m_1 называется объемом алфавита источника. Если символы на выходе появляются независимо друг от друга, то источник называется источником без памяти.

Если информацию рассматривать как меру снятой неопределённости, то количество информации I при равновероятном появлении символов вычисляют как произведение устранённой неопределенности H, снимаемой одним символом, полученным от источника, на число переданных символов k.

$$I = H_i * k$$

Если каждый элемент сообщения появляется на выходе источника с вероятностью p_i, то количество информации I, содержащееся в символе a_i, определяется выражением:

$$I_i = -\log p_i.$$

Таблица 1 – Исходные числа к выполнению задания 1

Номер варианта	p_1, p_2, p_3, p_4	Матрица условных вероятностей			
		1 строка	2 строка	3 строка	4 строка
1	**2**	**Номер варианта**			
1	0,1 0,2 0,3 0,4	2	3	4	5
2	0,5 0,1 0,3 0,1	6	7	8	9
3	0,6 0,2 0,1 0,1	10	11	12	13
4	0,25 0,3 0,02 0,43	14	15	16	17
5	0,2 0,35 0,25 0,2	18	19	20	21
6	0,15 0,45 0,05 0,35	22	23	24	25
7	0,1 0,45 0,05 0,4	26	27	28	29
8	0,22 0,28 0,05 0,45	2	6	10	14
9	0,25 0,05 0,55 0,15	3	7	11	15
10	0,44 0,05 0,16 0,35	4	8	12	16
11	0,33 0,05 0,17 0,45	5	9	13	17
12	0,21 0,05 0,19 0,55	22	26	2	3
13	0,25 0,25 0,19 0,31	23	27	6	7
14	0,1 0,5 0,09 0,31	24	28	10	11
15	0,2 0,4 0,09 0,31	25	29	14	15
16	0,4 0,3 0,07 0,23	4	5	22	1
17	0,22 0,41 0,3 0,07	7	4	15	9
18	0,15 0,45 0,05 0,35	1	5	10	15
19	0,22 0,28 0,05 0,45	2	6	11	14
20	0,6 0,2 0,1 0,1	3	5	7	9
21	0,25 0,05 0,55 0,15	4	7	10	13
22	0,1 0,45 0,05 0,4	5	9	13	17
23	0,25 0,3 0,02 0,43	9	12	14	19
24	0,44 0,05 0,16 0,35	6	7	11	12
25	0,25 0,25 0,19 0,31	1	10	20	25
26	0,2 0,35 0,25 0,2	3	7	13	15
27	0,33 0,05 0,17 0,45	30	12	2	5
28	0,21 0,05 0,19 0,55	28	10	1	2
29	0,4 0,3 0,07 0,23	26	24	3	4
30	0,15 0,45 0,05 0,35	6	7	9	10

Основание логарифма определяет систему единицы измерения количества информации. В дальнейшем информацию будем измерять в битах, а в качестве основания логарифма использовать двоичную систему счисления. Общая сумма вероятностей появления символов равна 1:

$$\sum p_i=1 \ (i=1\ldots m_1)$$

При передаче больших массивов сообщений важно не количество информации в одном конкретном символе I_i, а среднее количество информации, приходящееся на один символ. Такой мерой измерения количества информации является математическое ожидание (или среднее значение) случайной величины I_i, которое определяется по следующе1 формуле:

$$H(A)= - \sum_{i=1}^{m1} p_i \log p_i = - (p_1 \log p_1 + p_2 \log p_2 + \ldots + p_{m1} \log p_{m1})$$

Величина $H(A)$ называется ***энтропией источника независимых сообщений***, а формула её нахождения – формулой Шеннона. Минус в формуле используется для того, чтобы энтропия не получалась с отрицательным знаком.

Таким образом, энтропия источника независимых сообщений характеризует среднее количество информации, содержащееся в одном случайно выбранном символе. При равновероятном появлении символов источника, т.е. при $p_i = 1/m_1$ (m_1 – количество символов исходного алфавита), формула Шеннона переходит в формулу Хартли. Энтропия такого источника максимальна.

$$H(A)_{max} = - m_1 p \log 1/m_1 = \log m_1$$

Для двух равновероятных символов энтропия равна 1, для 4–х равновероятных символа энтропия равна 2 и т.д.

Для учета статистической взаимозависимости появления символов на выходе источника, вводят понятие условной энтропии, которая вычисляется:

$$H(A)_{усл.} = - \sum_{i=1}^{m1} p_{(ai)} \sum_{j=1}^{m1} p_{(aj/ai)} \log p_{(aj/ai)}$$

где $p_{(aj/ai)}$ – вероятность появления символа a_j при условии, что перед ним на выходе источника был символ a_i.

Протяженность статистической связи между символами сообщений характеризует глубину памяти источника.

Из-за неравновероятности появления символов в сообщениях уменьшается количество информации, которое переносит один символ. Численно эти потери информации характеризуются коэффициентом избыточности:

$$R= (H(A)_{max} - H(A)) / H(A)_{max} *100\ \%$$

Т.е, если энтропия определяет информационную нагрузку на символ сообщения, то избыточность – недогруженность символов.

Пример. Определить энтропию источника независимых сообщений, кодируемых с помощью двух символов a_1 и a_2. Вероятности появления символов ровны соответственно $p_1 = 0{,}99$ и $p_2 = 0{,}01$.

Энтропию источника сообщений находим по формуле Шеннона:

$$H(A) = - (p_1 \log p_1 + p_2 \log p_2) = - (0{,}99 \log_2 0{,}99 + 0{,}01 \log_2 0{,}01) = 0{,}014 + 0{,}066 = 0{,}080 \text{ бит}$$

Для вычисления значений $(-p_i \log p_i)$ используем таблицу двоичных логарифмов (приложение А). Энтропия получилась маленькая, потому что вероятность появления символа a_1 велика.

Пример. Оценить избыточность источника независимых сообщений, кодируемых с помощью символов a_1 и a_2. Вероятность появления символов ровны соответственно $p_1 = 0{,}2$ и $p_2 = 0{,}8$

Избыточность источника независимых сообщений находим по формуле:

$$R= (H(A)_{max} - H(A)) / H(A)_{max} *100\ \%$$

По формуле Хартли $H(A)_{max}$ для двух равновероятных символов равна 1. Энтропию источника независимых сообщений $H(A)$ находим по формуле Шеннона:

$$H(A) = - 0{,}2\ p \log 0{,}2 - 0{,}8\ p \log 0{,}8 = 0{,}722 \text{ бит}$$

$$R= (1-0{,}722) *100 /1 = 27{,}8\ \%$$

Пример. Определить условную энтропию источника независимых сообщений, кодируемых с помощью трех символов a_1 , a_2 и a_3. Зависимость

между символами задана матрицей условных вероятностей $p_{(aj/ai)}$. Безусловные вероятности символов равны соответственно: $p_1 = 0,2$, $p_2 = 0,3$ *и* $p_3 = 0,5$.

j \ i	a_1	a_2	a_3
a_1	0.8	0.1	0.1
a_2	0.1	0.5	0.4
a_3	0.1	0.6	0.3

Находим условную энтропию для каждого символа:

$H(A/a_1)= -0,8\log 0,8 -0,1\log 0,1 -0,1\log 0,1= 0,26+0,33+0,33=0,92$ *бит*

$H(A/a_2)= -0,1\log 0,1 -0,5\ \log 0,5 -0,4\log 0,4= 0,33+0,5+0,53 =1,36$ *бит*

$H(A/a_3)= -0,1\log 0,2 -0,6\log 0,6 -0,3\log 0,3= 0,33+0,44+0,52 =1,29$ *бит*

$$H_{усл} = 0,2*0,92+0,3*1,36+ 0,5* 1,29=0,18+0,41+0,65=1,24$$

Энтропия приемника сообщений представляет собой неопределенность появления на входе приемника символа после ее появления на выходе источника сообщений. Если в канале связи не происходит потерь информации, то $H(A)= H(B)$.

$$H(B) = - \sum_{i=1}^{m_2} p(b_i) \log\ p(b_i)$$

В общем случае условная энтропия, обозначаемая как $H(A/B)$, определяет количество закодированной информации (представленной в символах вторичного алфавита), потерянной при передаче по каналу связи.

ЗАДАНИЕ 2

1. Построить неравномерный двоичный код с заданными длинами кодовых слов $w_1=w_2= w_3=2$, $w_4= 3$ *и* $w_5 = 4$ используя префиксное дерево.

2. Построить оптимальный код по алгоритму Шеннона–Фано. Статистические вероятности появления символов в сообщениях представлены в таблице к заданию (столбец 3). Рассчитаться среднюю длину кодового слова, энтропию источника и коэффициент сжатия.

3. Построить оптимальный код по алгоритму Хафмена. Статистические вероятности появления символов в сообщениях представлены в таблице к заданию (столбец 3). Рассчитаться среднюю длину кодового слова, энтропию источника и коэффициент сжатия.

Таблица 2 – Исходные данные к заданию 2

Номер варианта	Длина кода символа						Вероятности								
	w1	*w*	*w*	*w*	*w*	*w*	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**
1	1	2	3	4	4		0,011	0,041	0,071	0,072	0,101	0,131	0,161	0,191	0,221
2	2	2	2	3	4		0,012	0,042	0,064	0,072	0,102	0,132	0,162	0,192	0,222
3	1	2	4	4	4		0,013	0,043	0,056	0,073	0,103	0,133	0,163	0,193	0,223
4	1	3	3	4	4	4	0,014	0,044	0,048	0,074	0,104	0,134	0,164	0,194	0,224
5	1	3	3	3	4	4	0,015	0,040	0,045	0,075	0,105	0,135	0,165	0,195	0,225
6	2	2	2	3	4	4	0,016	0,032	0,046	0,076	0,106	0,136	0,166	0,196	0,226
7	1	3	3	3	4		0,017	0,024	0,047	0,077	0,107	0,137	0,167	0,197	0,227
8	2	2	2	4	4		0,016	0,018	0,048	0,078	0,108	0,138	0,168	0,198	0,228
9	2	2	3	3	3		0,008	0,019	0,049	0,079	0,109	0,139	0,169	0,199	0,229
10	2	2	3	3	4		0,020	0,050	0,080	0,110	0,140	0,170	0,200	0,230	
11	2	2	3	4	4		0,021	0,051	0,081	0,111	0,141	0,171	0,201	0,223	
12	2	3	3	3	3		0,022	0,052	0,082	0,112	0,142	0,172	0,202	0,216	
13	2	2	4	4	4		0,023	0,053	0,083	0,113	0,143	0,173	0,203	0,209	
14	2	3	3	3	4		0,024	0,054	0,084	0,114	0,144	0,174	0,202	0,204	
15	2	3	3	4	4		0,025	0,055	0,085	0,115	0,145	0,175	0,195	0,205	
16	2	3	4	4	4		0,026	0,056	0,086	0,116	0,146	0,176	0,188	0,206	
17	2	3	4	4	4	4	0,027	0,057	0,087	0,117	0,147	0,177	0,181	0,207	
18	2	2	4	4	4	4	0,028	0,058	0,088	0,118	0,148	0,174	0,178	0,208	
19	2	3	3	3	4	4	0,029	0,059	0,089	0,119	0,149	0,167	0,179	0,209	
20	2	2	3	4	4	4	0,030	0,060	0,090	0,120	0,150	0,160	0,180	0,210	
21	3	3	3	3	4		0,031	0,061	0,091	0,121	0,151	0,153	0,181	0,211	
22	2	4	4	4	4		0,032	0,062	0,092	0,122	0,146	0,152	0,182	0,212	
23	3	3	3	4	4		0,033	0,063	0,093	0,123	0,139	0,153	0,183	0,213	
24	3	3	4	4	4		0,034	0,064	0,094	0,124	0,132	0,154	0,184	0,214	
25	3	4	4	4	4		0,035	0,065	0,095	0,125	0,125	0,155	0,185	0,215	
26	3	3	3	4	4	4	0,036	0,066	0,096	0,118	0,126	0,156	0,186	0,216	
27	2	2	3	3	3	4	0,037	0,067	0,097	0,111	0,127	0,157	0,187	0,217	
28	2	2	3	3	4	4	0,038	0,068	0,098	0,104	0,128	0,158	0,188	0,218	
29	2	2	2	4	4	4	0,039	0,069	0,097	0,099	0,129	0,159	0,189	0,219	
30	3	3	3	4	4	4	0,040	0,070	0,090	0,100	0,130	0,160	0,190	0,220	

Методические указания к выполнению задания

В общем случае *кодирование* – процесс преобразования символов исходного алфавита $A=(a_1, a_2, \dots a_{m1})$ в символы вторичного алфавита $B=(b_1, b_2, \dots b_{m2})$. При этом *код* есть правило, закон, алгоритм, по которому осуществляется это преобразование. В качестве значений символов вторичного алфавита при передаче информации по каналам связи могут использоваться различные признаки передающего сигнала. В связи с тем, что в цифровой вычислительной технике используется двоичная система счисления, то в качестве носителя информации сигнала принимается его полярный признак – положительный и отрицательный импульсы, т.е. вторичный алфавит будет представлен двумя символами – $B=(0, 1)$.

Последовательность символов вторичного алфавита, которая в процессе кодирования присваивается каждому символу исходного алфавита источника сообщений A называется *кодовым словом*.

Множество кодовых слов вторичного алфавита B удобно представлять в виде графа. Такой граф называется *кодовым деревом* (рис.). Основными структурными элементами кодового дерева являются – корень, ветви, узел ветвления (обозначается кружком), терминальные узлы или листья – узлы, из которых не исходит ни одной ветви (прямоугольники). Из каждого узла ветвления двоичного дерева выходит 2 ветви. Величина 2 называется *степенью дерева* и численно ровна значности кода. Листья дерева отображают кодовые слова. Вид кодового слова определяется значениями соответствующих ветвей дерева. Обычно принято обозначать крайнюю левую ветвь нулем.

Корень имеет нулевой уровень, а уровень любого другого узла на 1 больше уровня своего родителя. Любой уровень в общем случае содержит m^k узлов, где k – номер уровня. В этом случае кодовое дерево является полным.

Высота кодового дерева определяется максимальным уровнем терминального узла k_{max}:

$$h=k_{max}$$

Коды, в которых сообщения представлены кодовыми словами с равным

количеством символов, называются *равномерными.*

,

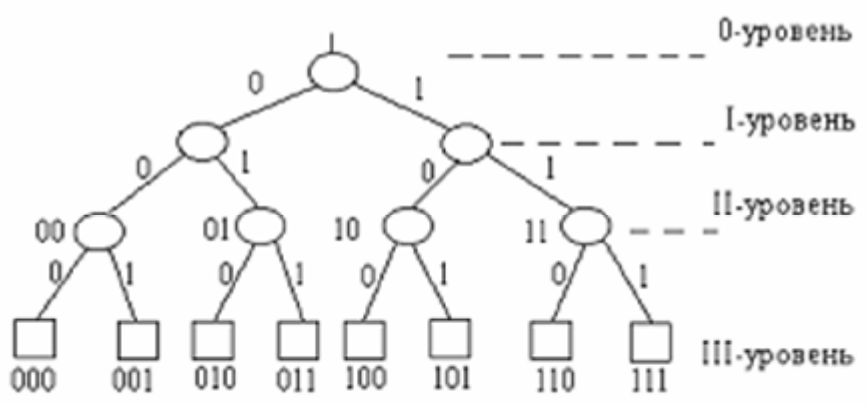

Рис. Полное двоичное кодовое дерево

Коды, в которых сообщения представлены кодовыми словами с разным количеством символов, называются *неравномерными*. Принцип неравномерного кодирования используется при построении оптимальных кодов.

Процедура кодирования, при которой число символов вторичного алфавита, затрачиваемое на кодирование одного символа источника, будет минимальным, называется оптимальным или эффективным кодированием. При этом должны выполниться следующие условия:

1) более вероятным символам источника соответствуют самые короткие кодовые комбинации, а менее вероятным – более длинные.

2) эффективный код необходимо строить так, чтобы ни одна кодовая комбинация не совпадала с началом другой, более длинной комбинацией.

Этим условиям отвечают префиксные коды, которые используются для кодирования исходных сообщений. Префиксный код является однозначно декодируемым.

Префиксом кодовой комбинации является любая последовательность, составленная из её начальной части. Та часть кодовой комбинации, которая дополняет префикс до самой кодовой комбинации, называется *суффиксом*.

Например, код – *00 01 101 010*, кодирующий символы a_1, a_2, a_3, a_4 не является префиксным кодом, так как комбинация, отображающая символ a_2, является префиксом кодового слова, отображающего символ a_4.

Основным параметром, характеризующим неравномерный код, является средняя длина кодовых слов, определяемая равенством:

$$L_{ср} = \sum_{i=1}^{m1} p(a_i)l(a_i)$$

где $p(a_i)$ – вероятность появления *i–го* символа алфавита источника сообщений; $l(a_i)$ – длина *i-го* кодового слова.

Пример. Алфавит источника сообщений содержит четыре символа – a_1, a_2, a_3, a_4 с вероятностями появления 0,5 0,375 0,0625 0,0625. Кодовая таблица имеет следующий вид: $a_1 = 0$, $a_2 = 1$, $a_3 = 10$, $a_4 = 11$. Наиболее частые сообщения a_1 и a_2 кодируются одним двоичным знаком.

$$L_{ср} = 1*0,5+1*0,375+2*0,0625+2*0,0625=1,125 \text{ сим}$$

Построение неравномерного префиксного кода с заданными длинами кодовых слов определяется неравенством Крафта:

$$\leq 1$$

где m – значность кода, k – число уровней кодового дерева, w_1, w_2, ... , w_k – заданные длины кодовых слов.

Пример: Построить неравномерный двоичный префиксный код с длинами кодовых слов $w_1 = w_2 = w_3 = 2$, $w_4 = 3$ *и* $w_5 = 4$.

Определим по неравенству Крафта, что код с заданными длинами кодовых слов существует:

$$= 0,25+0,25+0,25+ 0,125+ 0,0625 = 0,9375 < 1,$$

Для построения кода используем полное двоичное дерево с высотой $h = 4$. Отсекая три ветви дерева на втором уровне и одну на третьем, получаем дерево, изображённое на рис.

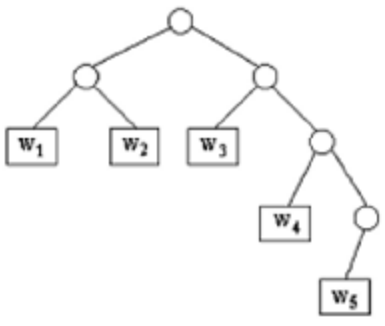

Рис. Неполное двоичное префиксное дерево

Построение оптимального префиксного кода по **алгоритму Шеннона-Фано** сводится к следующей процедуре.

На первом этапе кодирования 1) символы алфавита источника заносятся в таблицу в порядке убывания вероятностей; 2) символы разбиваются на две группы так, чтобы суммы вероятностей символов в каждой из них были по возможности одинаковы; 3) всем символам верхней группы присваивается нулевой старший разряд кодового слова, а всем нижним – единичный;

На втором этапе кодирования 1) каждая из полученных на первом этапе групп вновь разбивается на две равновероятные подгруппы; 2) следующему разряду кодового слова верхней группы присваивается 0, а нижней группы – 1;

Процесс кодирования продолжается до тех пор, пока в каждой подгруппе не останется по одному символу.

Рассмотрим алгоритм Шеннона-Фано на примере кодирования источника, алфавит которого состоит из 8 символов a_i, $(i = 1, 2, \ldots, 8)$. Рассмотрим процедуру разбиения символов на группы и подгруппы и образование кодовых слов.

На *первом этапе* делим символы на две равновероятные по сумме группы a_1 $(p_1=0{,}5)$ и a_2 a_3 a_4 a_5 a_6 a_7 a_8 $(\sum p_i=0{,}5,\ i=2\ldots8)$ и присваиваем разряду кодового слова первой группы – 0, а старшему разряду кодовых слов второй группы – 1.

На *втором этапе,* так как первая группа, полученная на первом этапе, состоит из одного символа, поэтому её деление на подгруппы больше не производится и код символ a_1 будет соответствовать 0. Вторую группу символов, полученную на первом этапе, дальше делим на две равновероятные по сумме подгруппы a_2 $(p_2=0{,}25)$ и a_3 a_4 a_5 a_6 a_7 a_8 $(\sum p_i=0{,}25,\ i=3\ldots8)$. Присваиваем следующему разряду кодового слова первой подгруппы – 0, а старшему разряду кодовых слов второй подгруппы – 1.

Деление на подгруппы символов равных вероятностей и присвоение

значений разрядам кодовых слов продолжаем до седьмого этапа, пока в подгруппах второй группы не останется по одному символу.

Таблица

A	p_i	1 этап	$\sum p_i$	2 этап	$\sum p_i$	3 этап	$\sum p_i$	4 этап	$\sum p_i$	5 этап	$\sum p_i$	6 этап	$\sum p_i$	7 этап							
a_1	0,5	0,5	**0,5**												0						
a_2	0,25	0,25		0,25	**0,25**										1	0					
a_3	0,125	0,125		0,125		0,125	**0,125**								1	1	0				
a_4	0,063	0,063		0,063		0,063		0,063	**0,063**						1	1	1	0			
a_5	0,031	0,031		0,031		0,031		0,031		0,031	**0,031**				1	1	1	1	0		
a_6	0,016	0,016		0,016		0,016		0,016		0,016		0,016	**0,016**		1	1	1	1	1	0	
a_7	0,008	0,008		0,008		0,008		0,008		0,008		0,008		0,008	1	1	1	1	1	1	0
a_8	0,008	0,008	**0,5**	0,008	**0,25**	0,008	**0,125**	0,008	**0,063**	0,008	**0,032**	0,008	**0,016**	0,008	1	1	1	1	1	1	1

Средняя длина кодового слова при кодировании заданного источника кодом Шеннона-Фано равна:

$$L_{ср} = \sum_{i=1}^{8} P(a_i) l(a_i) = 1/2 + (1/4) \times 2 + (1/8) \times 3 + (1/16) \times 4 +$$

$$+ (1/32) \times 5 + (1/64) \times 6 + (1/128) \times 7 + (1/128) \times 8 = 1\frac{63}{64},$$

а энтропия источника:

$$H(A) = -\sum_{i=1}^{m} p(a_i) \log p(a_i) = (1/2)\log(1/2) + (1/4)\log(1/4) +$$

$$(1/8)\log(1/8) + (1/16)\log(1/16) + (1/32)\log(1/32) + (1/64)\log(1/64) +$$

$$(1/128)\log(1/128) + (1/128)\log(1/128) = 256 / 128 = 1\frac{63}{64}$$

Таким образом, код Шеннона-Фано для заданного распределения вероятностей символов является оптимальным, так как среднее число бит на символ равно энтропии источника. При обычном кодировании, не учитывающем статистических свойств источника, для представления каждой буквы требуется 3 бита. Следовательно, коэффициент сжатия *Ксж* сообщения

за счет неравномерного кодирования Шеннона-Фано равен *Ксж = 2/3 = 0,666 = 67%.*

Алгоритм Шеннона-Фано не всегда приводит к однозначному построению кода, т.к. при разбиении на подгруппы можно сделать большей по вероятности как верхнюю, так и нижнюю. Поэтому среднее число бит на символ может быть разным. Более эффективным является алгоритм, предложенный Хаффменом в 1952 г., который позволяет построить оптимальный код с наименьшим средним числом бит на символ:

$$L_{cp\ min} = \sum_{i=1}^{m1} p(a_i)l(a_i)$$

Для двоичного кода **алгоритм Хаффмена** сводится к следующему.

1. Символы сообщения упорядочиваются по убыванию вероятностей таким образом, что $p(a_i) \geq p(a_j)$ для всех $i<j$ (табл.).

2. Два последних символа объединяются в один вспомогательный, вероятность которого равна сумме вероятностей составляющих его символов.

3. Все оставшиеся символы, вместе со вспомогательным снова располагаются по убыванию

Процедура продолжается до тех пор, пока не получится единственный вспомогательный символ, имеющий вероятность, равную единице. Код, построенный по рассмотренному алгоритму, получил название кода Хаффмена.

Таблица

Символы a_i	Вероятности $P(a_i)$	Дополнительные столбцы						
		1	2	3	4	5	6	7
a_1	0.22	0.22	0.22	0.26	0.32	0.42	0.58	1.0
a_2	0.20	0.20	0.20	0.22	0.26	0.32	0.42	
a_3	0.16	0.16	0.16	0.20	0.22	0.26		
a_4	0.16	0.16	0.16	0.16	0.20			
a_5	0.10	0.10	0.16	0.16				
a_6	0.10	0.10	0.10					
a_7	0.04	0.06						
a_8	0.02							

Для формирования кодовых комбинаций, соответствующих символам данного сообщения, необходимо проследить путь перехода символов по строкам и столбцам таблицы. Далее на основании этой таблицы строится дерево кодирования Хаффмана (Н-дерево).

Построение дерева (рис.) начинается с корневого узла, вероятность которого равна 1.

1. Из корня проводятся две ветви, ветви с большей вероятностью присваивается значение (бит) 1, а с меньшей вероятностью - 0.

2. Вновь образованные узлы могут отображать одиночный или вспомогательный символы. В последнем случае узел является промежуточным и из него снова проводятся две ветви.

Такое последовательное ветвление продолжается до тех пор, пока не будет достигнут узел, соответствующий вероятности символа алфавита (узел листа). Двигаясь по кодовому дереву от корня сверху вниз, для каждого символа записывают соответствующую ему кодовую комбинацию.

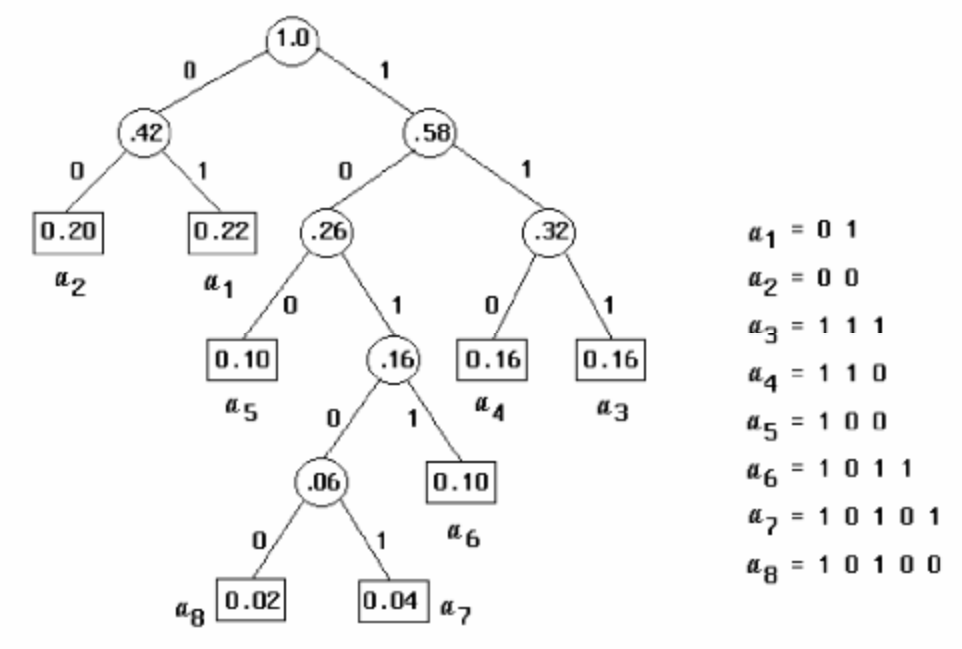

Рис.

Среднее число битов на символ при таком построении кода составляет:

$$l_{ср} = \sum P(a_i)\, l(a_i) = 0{,}22 \times 2 + 0{,}2 \times 2 + 0{,}16 \times 3 + 0{,}16 \times 3 +$$
$$+ \ 0{,}1 \times 3 + 0{,}1 \ \times 4 + 0{,}04 \times 5 + 0{,}02 \times 5 = 2{,}8 \ \text{бит}.$$

Энтропия источника сообщения равна:

$$H(A) = -\sum P(a_i) \log P(a_i) = -(0{,}22 \log 0{,}22 + 0{,}2 \log 0{,}2 +$$
$$+0{,}16 \log 0{,}16 + 0{,}16 \log 0{,}16 + 0{,}1 \log 0{,}1 + 0{,}1 \log 0{,}1 +$$
$$+ 0{,}04 \log 0{,}04 + 0{,}02 \log 0{,}02 = 2{,}754 \ \text{бит}.$$

Как видно из рассмотренного примера средняя длина кодовой комбинации и энтропия источника практически совпадают, т. е. полученный код является источника практически совпадают, т. е. полученный код является оптимальным.

Алгоритм Хаффмена является двухпроходным, т.е. при его реализации требуется дважды просматривать кодируемое сообщение:

1. При первом проходе вычисляются вероятности (или частоты) появления символов в сжимаемом сообщении и строится хаффменовское дерево.

2. При втором проходе осуществляется кодирование символов, поступивших от источника.

ЛИТЕРАТУРА

Основная литература

1. Кудряшов Б.Д. Теория информации: Учебник для вузов. – СПб.: Питер, 2009. – 320 с.: ил.

2. Симонович С.В. Информатика. Базовый курс: Учебник для вузов. 3-е изд. Стандарт третьего поколения. – СПб.: Питер, 2011. – 640 с.: ил.

3. Ткаченко Ф.А. Электронные приборы и устройства: Учебник для студентов вузов / Ф.А. Ткаченко. – М.: ИНФРА-М; Минск: Новое знание, 2011. – 681 с.: ил;

4. Хохлов Г.И. Основы теории информации: Учебное пособие. – М.: Академия, 2008. – 176 с.: ил.

Дополнительная литература

1. Горбоконенко В.Д. Кодирование информации: Методические указания / В.Д. Горбоконенко, В.Е. Шикина. – Ульяновск: УлГТУ, 2006. – 56 с.

2. Цымбал В.П. Теория информации и кодирование: Учебник. – 4-е изд., перераб. и доп. / В.П. Цымбал. – К.: Вища шк., 1992. – 263 с.

3. Дмитриев В.И. Прикладная теория информации: Учебное пособие для студентов ВУЗов / В.И. Дмитриев. – М.: Высш. шк., 1989. – 332с.

Приложение А

Значения величин $-p \log_2 p$

p	$-p \log_2 p$	p	$-p \log_2 p$	p	$-p \log_2 p$
0,00	—	0,36	0,5306	0,71	0,3508
0,01	0,0664	0,37	0,5307	0,72	0,3412
0,02	0,1129	0,38	0,5304	0,73	0,3314
0,03	0,1517	0,39	0,5298	0,74	0,3215
0,04	0,1857	0,40	0,5288	0,75	0,3113
0,05	0,2161				
0,06	0,2435	0,41	0,5274	0,76	0,3009
0,07	0,2686	0,42	0,5256	0,77	0,2903
0,08	0,2915	0,43	0,5236	0,78	0,2796
0,09	0,3127	0,44	0,5211	0,79	0,2687
0,10	0,3322	0,45	0,5184	0,80	0,2575
0,11	0,3503	0,46	0,5153	0,81	0,2462
0,12	0,3671	0,47	0,5120	0,82	0,2348
0,13	0,3826	0,48	0,5083	0,83	0,2231
0,14	0,3971	0,49	0,5043	0,84	0,2113
0,15	0,4105	0,50	0,5000	0,85	0,1993
0,16	0,4230	0,51	0,4954	0,86	0,1871
0,17	0,4346	0,52	0,4906	0,87	0,1748
0,18	0,4453	0,53	0,4854	0,88	0,1623
0,19	0,4552	0,54	0,4800	0,89	0,1496
0,20	0,4644	0,55	0,4744	0,90	0,1368
0,21	0,4728	0,56	0,4684	0,91	0,1238
0,22	0,4806	0,57	0,4623	0,92	0,1107
0,23	0,4877	0,58	0,4558	0,93	0,0974
0,24	0,4941	0,59	0,4491	0,94	0,0839
0,25	0,5000	0,60	0,4422	0,95	0,0703
0,26	0,5053	0,61	0,4350	0,96	0,0565
0,27	0,5100	0,62	0,4276	0,97	0,0426
0,28	0,5142	0,63	0,4199	0,98	0,0286
0,29	0,5179	0,64	0,4121	0,99	0,0140
0,30	0,5211	0,65	0,4040		
0,31	0,5238	0,66	0,3957		
0,32	0,5260	0,67	0,3871		
0,33	0,5278	0,68	0,3784		
0,34	0,5292	0,69	0,3694		
0,35	0,5301	0,70	0,3602		

ПРИЛОЖЕНИЕ Б

Образец оформления титульного листа

Министерство образования и науки Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего профессионального образования
«Российский государственный профессионально-педагогический университет»
Институт электроэнергетики и информатики

КОНТРОЛЬНАЯ РАБОТА
по дисциплине «Основы кодирования информации»
051000.62

Шифр направления подготовки и квалификации

Вариант № ____

Выполнил:
студент группы ЗКТэ – 201С Семин С.А.

Проверил:
старший преподаватель Нестеров В.И.

Екатеринбург
2013

Задания и методические указания к выполнению
контрольной работы по дисциплине
«Основы кодирования информации»

Подписано в печать _____. Формат 60×84/16. Бумага для множ. аппаратов.
Печать плоская. Усл. печ. л. ___. Уч.-изд. л.____. Тираж ____ экз. Заказ № ____.
ФГАОУ ВПО «Российский государственный профессионально-педагогический университет». Екатеринбург, ул. Машиностроителей, 11.

Ризограф ФГАОУ ВПО РГППУ. Екатеринбург, ул. Машиностроителей, 11.