

Обязательное и дополнительное задания к практической работе №1
"Алгоритмы сжатия двоичной информации"
группа 14-502

Исходные данные:

Последовательность десятичных чисел, представляющих номера уровней квантования ($L=0\div 15$) выборок, полученная в результате аналого-цифрового преобразования некоторого информационного сигнала

Для выполнения обязательного задания необходимо:

I. Выполнить сжатие информационной последовательности в соответствии со статическим алгоритмом Хаффмана.

1. Определить объем исходной информационной последовательности в битах до сжатия.
2. Подсчитать статистику появления каждого из символов в информационной последовательности.
3. В соответствии с подсчитанной в п.1.2 статистикой построить кодовое дерево Хаффмана;
4. Получить набор кодов переменной длины для каждого из символов исходной последовательности;
5. Оценить объем сжатой последовательности, подсчитать полученный коэффициент сжатия.

II. Выполнить сжатие информационной последовательности в соответствии со словарными алгоритмами семейства Лемпеля-Зива LZ77 и LZSS.

1. Алгоритм LZ77

- а) выбрать размер словаря и размер буфера. Заполнить буфер символами исходной информационной последовательности;
- б) сформировать код записи для первого символа в буфере, а также представить его в виде двоичной комбинации. Сместить окно (словарь и буфер) на один символ вправо относительно входной информационной последовательности;
- в) проверить наличие в словаре символа, находящегося на первой позиции в текущем буфере. Если символ в словаре отсутствует, сформировать код записи для этого символа и представить его в виде двоичной комбинации, а также сместить окно на один символ вправо. Если же символ есть в словаре, создать ссылку на словарь для текущего символа и следующего за ним символа и также представить ссылку в виде двоичной комбинации. Сместить окно на два символа вправо;
- г) для всех последующих символов выполнить операции, аналогичные пункту в). Однако если текущий символ в буфере присутствует в словаре, необходимо также проверить и следующие за ним символы в буфере и создать ссылку на весь повторяющийся блок. Сместить окно в этом случае необходимо на величину всего повторяющегося блока вправо;
- д) подсчитать объем сжатого текста, вычислить коэффициент сжатия;
- е) выполнить распаковку всех сформированных кодов для полного восстановления исходного текста.

2. Алгоритм LZSS

Выполнить пункты I.1а) – I.1е) с учетом отличий в формировании кодов от алгоритма LZ77.

Для выполнения дополнительного задания необходимо:

I. Выполнить сжатие информационной последовательности в соответствии с динамическим (адаптивным) алгоритмом Хаффмана.

1. Представить весь набор символов исходной информационной последовательности в равномерной четырехразрядной кодировке.
2. Создать пустое дерево, содержащее единственный лист – символ "+" с весом "0".
3. Взять первый символ информационной последовательности, проверить его наличие в дереве. Сформировать выходной код для первого символа в равномерной кодировке.
4. Добавить первый символ информационной последовательности с весом "0" в пустое дерево на позицию младшего листа дерева, объединить его с символом "+" и сформировать родительский узел, вес которого равен сумме весов листьев, входящих в него. Пронумеровать в порядке возрастания все листья дерева начиная с крайнего левого листа в нижнем ярусе, а также обозначить каждое из ребер двоичными символами "0" и "1".
5. Выполнить процедуру увеличения веса добавленного листа и всех его родительских узлов.
6. Выполнить процедуру обновления дерева путем проверки веса соседних по порядковому номеру узлов и изменением структуры дерева в случае нарушения свойства упорядоченности.
7. Взять второй символ последовательности. В случае, если текущий символ встречается впервые, сформировать код по принципу: код символа "+" в текущем дереве Хаффмана и код поступившего символа в равномерной кодировке. Далее выполнить пп.4-6. Если символ уже встречался в прошедшем кодировании информационном потоке, сформировать код символа в зависимости от его положения в текущем дереве. Далее выполнить пп.5-6.
8. Выполнить пункт 7 для всех последующих символов последовательности. После выполнения кодирования всех символов привести двоичную последовательность, полученную в результате кодирования, а также подсчитать объем сжатого текста, сравнить с объемом исходной последовательности в равномерной кодировке и с объемом информации, полученном после сжатия в соответствии со статическим алгоритмом Хаффмана.
9. Показать первые пять итераций распаковки адаптивного кода Хаффмана путем построения динамического дерева:
 - выполнить п.2;
 - взять из двоичного потока четыре первых разряда, определить первый закодированный символ по таблице равномерных кодов;
 - выполнить пп.4-6;
 - по текущему двоичному символу кодовой последовательности и сформированному кодовому дереву определить, является ли второй информационный декодируемый символ новым или уже встречавшимся ранее. Восстановить закодированный информационный символ и выполнить пп.5-6. Повторить процедуру распаковки еще для трех закодированных символов. Показать полученное после пяти итераций декодирования дерево Хаффмана, сравнить восстановленный информационный поток с исходной информационной последовательностью.

II. Выполнить сжатие информационной последовательности в соответствии со словарными алгоритмами семейства Лемпеля-Зива LZ78 и LZW.

1. Алгоритм LZ78

- а) выбрать размер словаря;
- б) сформировать код записи для первого символа во входной информационной последовательности, а также представить его в виде двоичной комбинации. Добавить обработанный символ в словарь просмотренных фраз под номером 1;
- в) перейти ко второму входному информационному символу. Проверить наличие текущего символа в словаре просмотренных фраз. Если текущий символ в словаре отсутствует, сформировать код записи для этого символа, представить его в виде двоичной комбинации и записать его в словарь просмотренных фраз под номером 2. Если символ есть в словаре, перейти к следующему входному символу, создать ссылку на соответствующий номер записи в словаре, представить ее в виде двоичной комбинации и записать в словарь просмотренных фраз под номером 2 комбинацию из существующего в словаре и следующего за ним символа;

- г) для всех последующих символов выполнить операции, аналогичные пункту в). По мере увеличения количества записей в словаре необходимо увеличивать поле в двоичном коде, предназначенное для хранения ссылки на позицию фразы в словаре;
- д) после выполнения кодирования всех символов подсчитать объем сжатого текста, вычислить коэффициент сжатия;
- е) выполнить распаковку всех сформированных кодов для полного восстановления исходного текста путем составления словаря просмотренных фраз на основе полученных кодов для распаковки.

2. Алгоритм LZW

- а) определить размер начального словаря одиночных символов;
- б) сформировать код записи для первого символа во входной информационной последовательности, представить его в виде двоичной комбинации. Взять следующий за ним информационный символ и добавить полученную символьную комбинацию в словарь, присвоив ей соответствующий номер;
- в) перейти ко второму входному информационному символу. Взять следующий за ним информационный символ, проверить наличие этой символьной комбинации в словаре просмотренных фраз. Если текущая комбинация символов в словаре отсутствует, сформировать код записи для первого символа, представить его в виде двоичной комбинации и записать символьную комбинацию в словарь просмотренных фраз. Если символьная комбинация есть в словаре, перейти к следующему входному символу, создать код ссылки на соответствующую запись в словаре и занести всю просмотренную фразу в словарь;
- г) для всех последующих символов выполнить операции, аналогичные пункту в). В случае необходимости следует увеличить поле в двоичном коде, предназначенное для хранения ссылки на позицию фразы в словаре;
- д) после выполнения кодирования всех символов подсчитать объем сжатого текста, вычислить коэффициент сжатия;
- е) выполнить распаковку всех сформированных кодов для полного восстановления исходного текста путем составления словаря просмотренных фраз на основе полученных кодов для распаковки.

№ варианта	Информационная последовательность
1.	1 2 3 5 7 7 6 5 7 7 9 6 7 7 9 7
2.	4 2 3 4 2 4 3 3 1 4 2 3 4 2 2 0 1
3.	14 11 9 9 13 14 11 9 9 12 14 11 9 10 7 14
4.	3 3 1 2 3 3 1 5 4 3 3 1 2 5 5 4 7 6 3 3 1
5.	13 10 9 8 7 8 8 10 9 8 7 9 9 8 7 10 9 5
6.	7 7 6 6 7 6 8 7 7 6 9 8 7 7 6 4
7.	5 3 4 3 5 5 6 3 4 3 5 5 6 4 5 6 3
8.	11 10 11 10 11 9 11 10 9 11 8 7 9 11 10 11
9.	6 2 2 4 6 2 2 3 3 4 6 4 3 3 4 5 2 2 4 1
10.	9 12 11 12 10 11 12 9 13 12 11 12 10 9 12 11
11.	8 6 7 8 9 10 8 9 7 8 9 10 8 6 7 7 9 10 8 8
12.	8 8 10 9 7 7 6 8 10 9 8 7 9 8 10 9 8 7 6 8
13.	11 12 9 10 12 11 12 13 9 12 11 10 12 11 12 9
14.	1 4 2 2 5 4 3 3 4 6 4 3 3 2 2 6 4 2 2 6
15.	11 10 11 9 7 8 11 9 10 11 9 11 10 11 10 11
16.	3 6 5 4 6 5 5 3 4 3 6 5 5 3 4 3 5
17.	4 6 7 7 8 9 6 7 7 8 6 7 6 6 7 7
18.	5 9 10 7 8 9 9 7 8 9 10 8 8 7 8 9 10 13
19.	1 3 3 6 7 4 5 5 2 1 3 3 4 5 1 3 3 2 1 3 3
20.	14 7 10 9 11 14 12 9 9 11 14 13 9 9 11 14
21.	1 0 2 2 4 3 2 4 1 3 3 4 2 4 3 2 4
22.	7 9 7 7 6 9 7 7 5 6 7 7 5 3 2 1