МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ

Учреждение образования «Полоцкий государственный университет»

Факультет информационных технологий

Кафедра технологий программирования

**ЛАБОРАТОРНАЯ РАБОТА №1**

по дисциплине: **«Теория информации»**

на тему: «**Сжатие без потерь**»

ВЫПОЛНИЛ студент группы

Яковлев Н.А.

ПРОВЕРИЛ ст. преподаватель

Захарова И.Ю.

Новополоцк, 2020 г.

**Цель работы**: изучить основные понятия, связанные с сжатием без потерь. Разобрать и научиться использовать на практике алгоритм Хаффмана.

**Теоретическая часть**

Метод оптимального побуквенного кодирования был разработан в 1952 г. Д. Хаффманом. Оптимальный код Хаффмана обладает минимальной средней длиной кодового слова среди всех побуквенных кодов для данного источника с алфавитом *А=*{*a*1*,…,an*} и вероятностями *pi =P*(*ai*).

Рассмотрим алгоритм построения оптимального кода Хаффмана, который основывается на утверждениях лемм предыдущего параграфа.

1. Упорядочим символы исходного алфавита *А*={*a*1*,…,an*} по убыванию их вероятностей *p1≥p2≥…≥pn*.
2. Если *А*={*a*1*,a2*}, то *a*10, *a*21.
3. Если *А*={*a*1*,…,aj,…,an*} и известны коды <*aj*  *bj*>, *j = 1,…,n*, то для алфавита {*a1,…aj/, aj//…,an*} с новыми символами *aj/ и aj//* вместо *aj*, и вероятностями *p(aj)=p(aj/)+ p(aj//),* код символа *aj* заменяется на коды  *aj/   bj0, aj// bj1*.

**Пример.** Пусть дан алфавит A={*a*1*, a*2*, a*3*, a*4*, a*5*, a*6} с вероятностями

*p*1=0.36, *p*2=0.18, *p*3=0.18, *p*4=0.12, *p*5=0.09, *p*6=0.07.

Здесь символы источника уже упорядочены в соответствии с их вероятностями. Будем складывать две наименьшие вероятности и включать суммарную вероятность на соответствующее место в упорядоченном списке вероятностей до тех пор, пока в списке не останется два символа. Тогда закодируем эти два символа 0 и 1. Далее кодовые слова достраиваются, как показано на рисунке 4.

*a*1 0.36 0.36 0.36 0.36 0.64 0

*a*2 0.18 0.18 0.28 0.36 0.36 1

*a*3 0.18 0.18 0.18 0.28 00

*a*4 0.12 0.16 0.18 000 01

*a*50.09 0.12 010 001

*a*60.07 0100 011

0101

Рисунок 1 Процесс построения кода Хаффмана

Таблица 5 Код Хаффмана

|  |  |  |  |
| --- | --- | --- | --- |
| *ai* | *pi* | *Li* | кодовое слово |
| *a*1  *a*2  *a*3  *a*4  *a*5  *a*6 | 0.36  0.18  0.18  0.12  0.09  0.07 | 2  3  3  4  4  4 | 1  000  001  011  0100  0101 |

Посчитаем среднюю длину, построенного кода Хаффмана

*Lср(P)*=1**.**0.36 + 3**.**0.18 + 3**.**0.18 + 3**.**0.12 + 4**.**0.09 + 4**.**0.07 =2.44,

при этом энтропия данного источника

*H*(*p*1,…,*p*6)= − 0.36**.**log0.36 − 2**.**0.18**.**log0.18 −

− 0.12**.**log0.12 − 0.09**.**log0.09 − 0.07log0.07=2.37



0

1

*а1*

00

01

000

001

010

011

0100

0101

*а2*

*а3*

*а4*

*а6*

*а5*

Рисунок 1 Кодовое дерево для кода Хаффмана

Код Хаффмана обычно строится и хранится в виде двоичного дерева, в листьях которого находятся символы алфавита, а на «ветвях» – 0 или 1. Тогда уникальным кодом символа является путь от корня дерева к этому символу, по которому все 0 и 1 собираются в одну уникальную последовательность (рис. 5).

**Прeфиксный код** в теории кодирования — **код** со словом переменной длины, имеющий такое свойство (выполнение условия Фано): если в **код** входит слово a, то для любой непустой строки b слова ab в коде не существует.

**Теорема** (Крафт). Для того, чтобы существовал побуквенный двоичный префиксный код с длинами кодовых слов L1,…,Ln необходимо и достаточно, чтобы

.

**Энтропия** характеризует меру неопределенности выбора для данного источника.

**Основной характеристикой неравномерного кода** является количество символов, затрачиваемых на кодирование одного сообщения.

**Избыточностью кода** называется разность между средней длиной кодового слова и предельной энтропией источника сообщений

.

Побуквенный разделимый код называется **оптимальным**, если средняя длина кодового слова минимальна среди всех побуквенных разделимых кодов для данного распределения вероятностей символов.

**Порядок выполнения работы**

1. Изучить теоретический материал.

2. Реализовать процедуру построения оптимального кода Хаффмана.

3. Построить код Хаффмана для текста на языке, обозначенном преподавателем, использовать файл не менее 1 Кб. Распечатать полученную кодовую таблицу в виде:

|  |  |  |
| --- | --- | --- |
| Символ | Вероятность | Кодовое слово |

4. Проверить выполнение неравенства Крафта-МакМиллана для полученного кода.

5. Вычислить энтропию исходного файла и сравнить со средней длиной кодового слова.

6. Восстановить исходный текст.

**Практическая часть**

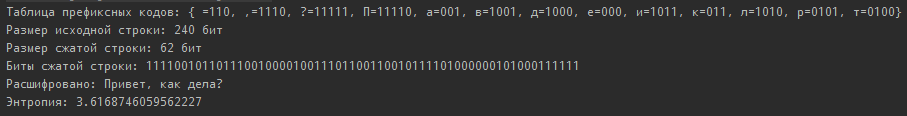
Дли примера возьмем передаваемое сообщение «Привет, как дела?».

Посчитаем вероятность и коды для каждого символа.

|  |  |  |
| --- | --- | --- |
| Символ | Вероятность, р | Код в двоичной системе |
| П | 0.058 | 11110 |
| р | 0.058 | 0101 |
| и | 0.058 | 1011 |
| в | 0.058 | 1001 |
| е | 0.117 | 000 |
| т | 0.058 | 0100 |
| « » | 0.117 | 110 |
| к | 0.117 | 011 |
| а | 0.117 | 001 |
| д | 0.058 | 1000 |
| л | 0.058 | 1010 |
| , | 0.058 | 1110 |
| ? | 0.058 | 11111 |

Коды получаются программно, по такому же принципу как и из теоритеческой части. Скриншот функции для определения кода находиться в приложении.

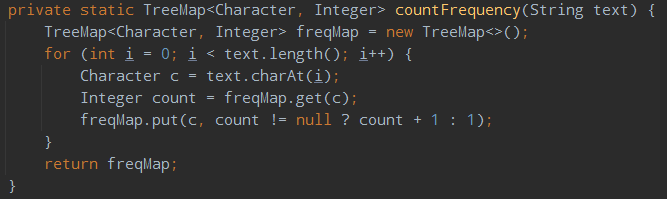
**Результат работы программы**



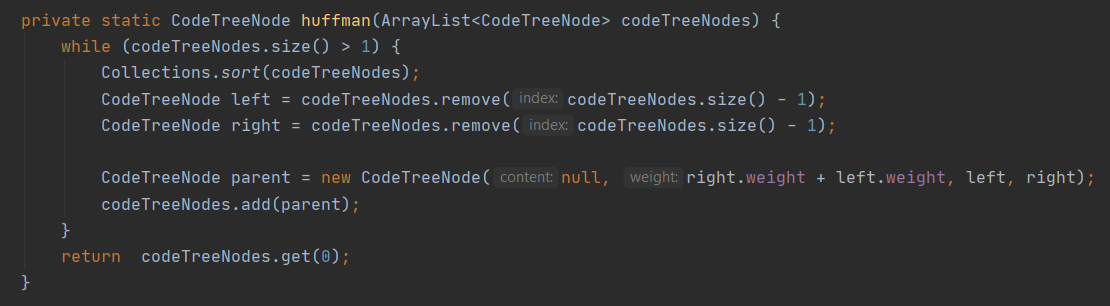
**Вывод :** Для тестирования работы программы был использован файл с текстом на русском языке размером 1,2 Мб. В ходе работы программы файл был сжат до 446 Кб. Было соблюдено правило префиксности и разделяемости. Энтропия в текстовом документе составила 3,6168.При декодировании информации ничего не потерялось.

**Приложение**

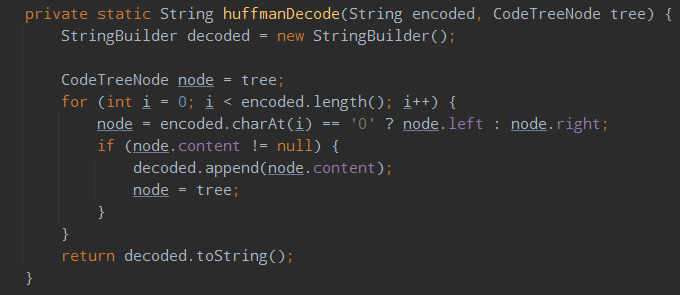
Метод, подсчитывающий кол-во символов в сообщении :



Метод, реализующий алгоритм Хаффмана :



Метод декодирования сообщения :



Метод, реализующий вычисление энтропии :

