Дисциплина «Алгоритмы решения прикладных задач»
Рабочая тетрадь 5.1.
Дерево Хаффмана.

Теоретический материал

Обычно для хранения данных и передачи сообщений используются коды фиксированной длины, например, код ASCII.

Множество символов представляются некоторым количеством кодовых слов равной длины, которая для кода ASCII равна восьми битам (1 байт). При этом для всех сообщений с одинаковым количеством символов требуется одинаковое количество битов при хранении и одинаковая ширина полосы пропускания при передаче.

Конечно, если сообщение написано, скажем, на английском языке, то вероятность появления в нем букв "z" намного меньше, чем вероятность появления букв "e". Это означает, что если для представления буквы "e" использовать более короткое кодовое слово, чем для представления буквы "z", то можно ожидать, что в среднем для хранения сообщения потребуется меньше памяти, а для его передачи - меньшая ширина канала.

В коде ASCII сообщение "easily" кодируется следующим образом:

01100101	01100001	01110011	01101001	01101100	01111001
e	a	s	i	l	y

для чего требуется 48 бит, в то время как при использовании кода со следующим представлением символов

1001	0	1010	11001	11010	1011
a	e	i	l	s	y

то же самое сообщение можно закодировать следующим образом

```
0 1 0 0 1 1 1 0 1 0 1 0 1 0 1 1 0 0 1 1 0 1 1
e a       s         i       l         y
```

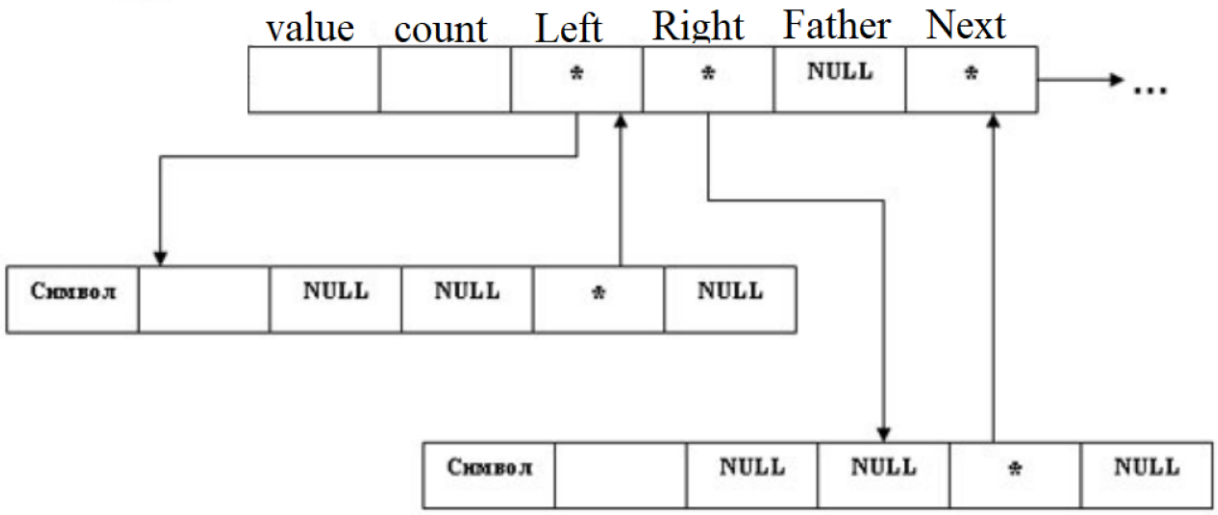
используя только 23 бита.

Кодовые слова должны быть выбраны так, чтобы никакое из них не было префиксом любого другого кодового слова. Благодаря этому условию гарантируется возможность однозначного декодирования определённого закодированного текста.

В классической статье, опубликованной в 1952 г. Дэвид Хаффман описал алгоритм поиска множества кодов, которые минимизируют ожидаемую длину сообщений при условии, что известны вероятности появления каждого символа.

Существенно, что в этом методе при определении длины кодовых слов символам, имеющим меньшую вероятность появления, ставятся в соответствие более длинные кодовые слова. После этого остается образовать некоторый однозначно декодируемый код с кодовыми словами надлежащей длины.

Хаффман в заключительном разделе работы отождествляет однозначное множество кодовых слов с двоичным деревом. Каждый лист дерева соответствует одному из символов. Глубина этого листа, т.е. его расстояние от корня, - это длина кодового слова соответствующего символа.



Цифры кодового слова являются адресом этого листа, т.е. последовательностью инструкций для продвижения от корня к листу, например, команда "0"- - двигаться влево, а "1" - двигаться вправо. Тогда каждой вершине дерева будет приписано двоичное слово, описывающее, как добраться к этой вершине от корня. Самому корню соответствует пустое слово "0".

Хаффман пишет: "Так как объединение сообщений в составные сообщения подобно слиянию струек, ручейков и речушек в одну большую реку, описанную выше процедуру можно рассматривать по аналогии с расстановкой знаков жуком-плавунцом в каждом месте впадения притоков по пути его перемещения вниз по течению ... искомым будет код, который должен помнить плавунец, чтобы совершить обратный путь против течения".

Алгоритм Хаффмана:

1 этап:

Множество символов располагается в порядке уменьшения вероятностей их появления. Каждому из символов будет соответствовать лист дерева, следовательно, можно представить себе этот этап процесса как построение линейного списка, содержащего листья будущего дерева.

2 этап:

Производится повторяющееся сокращение числа максимальных непересекающихся поддеревьев посредством объединения двух "легчайших" деревьев для получения нового составного дерева.

Листья любого бинарного дерева образуют префиксный код, и, наоборот, для всякого префиксного кода существует такое дерево, что слова кода соответствуют его листьям.

Задача:

Создать дерево Хаффмана. Реализовать следующие методы:

а) Метод построения дерева;

б) Метод построения списка частот символов;

в) Метод шифрования (сжатия);

г) Метод дешифровки;

г*) подсчет коэффициента сжатия.

Пример работы программы представлен ниже

```
Введите текст, содержащий не менее двух символов...
мама мыла раму
Полученный список:
м (0.285714) --> а (0.285714) -->   (0.142857) --> ы (0.0714286) --> л (0.0714286) --> р (0.0714286) --> у (0.0714286) -->
Построим дерево...
        л (0.0714286)
     * (0.142857)
        ы (0.0714286)
  * (0.428571)
     а (0.285714)
* (1)
     м (0.285714)
  * (0.571429)
        (0.142857)
     * (0.285714)
        у (0.0714286)
     * (0.142857)
        р (0.0714286)
-----------------------------------------
Приступим к кодировке введенного текста...
Код перед Вами... 10011001110100010000111011101101110
Коэффициент сжатия: 32.1429%
Ранее было зашифровано... мама мыла раму
Расшифровано...мама мыла раму
```

Решение:

Ответ: