

Формат закодированного файла

Файл, закодированный с помощью алгоритма Хаффмана, несет в себе дополнительную информацию о структуре дерева кодирования для того, чтобы можно было раскодировать записанное сообщение.

Представление файла

Закодированный файл является простым .txt файлом. Ввиду того, что в него побитово записывается сначала дерево кодирования, а потом закодированное сообщение, полученные байты чаще всего непредставимы в виде стандартных латинских ASCII символов, то есть букв a-zA-Z, цифр и знаков препинания. Ввиду этого прочтение закодированного сообщения без раскодировщика не представляется возможным. Пример закодированного сообщения:

Tİ>µ'Q•њ~g -іц°яКьџёК ЪШЊЦІ—йхџџ{ЧІ,є]Ль д@STWІр±тpy8K» нј•b[Vv
SHЄууб 7У™,R'zЭЛЬ;Д^=Тµг5іџNєН...ы`уW™ђЭ1^ оКЭыІFШехуњ е*|ЕгФаVjOA—
ЦІцъ,,8G®AdпэнJzФй ль Т'Зњ°® Ю9•Itf

Секции закодированного файла

Файл разделен на несколько частей:

1. Расширение кодируемого файла. Это необходимо для того, чтобы знать, в файл какого формата необходимо декодировать закодированное сообщение.
2. Количество байтов в кодируемом сообщении. Так как каждый байт кодируется не кратным 8 число бит, может оказаться, что записанный текст не укладывается в целое число байт, и на конце будут незначащие биты. В закодированный файл записывается число байт в исходном сообщении, чтобы при декодировке не прочесть лишние биты.
3. Число элементов в дереве Хаффмана. Дерево Хаффмана отделено от закодированного сообщения пустым байтом (00), но так как этот же пустой байт может оказаться одним из кодируемых байтов в дереве Хаффмана, в файл записывается число элементов в дереве, чтобы не прекратить считывание элементов преждевременно.
4. Дерево Хаффмана в его битовом представлении. Записывается кодируемый символ, длина его кода и сам код.
5. Закодированное сообщение. Каждый символ исходного сообщения преобразуется посредством дерева Хаффмана в последовательность из 0 и 1 и записывается в том же порядке в закодированный файл.