**Сжатие данных**

**Равномерное кодирование со словарём**

Происходит анализ сообщения с записью всех символов, а потом — кодирование.  
В словаре передаются количество символов и коды всех из них. Располагаются они по возрастанию. Длина словаря всегда будет M + 1 байт. M = 2i.   
Эффективность сжатия показывает коэффициент оного. Он — отношение размеров исходного и сжатого файлов.  
Такой тип архивирования не подходит для любого типа информации. К примеру, изображения с огромным колчиеством оттенков будут сжиматься крайне слабо почти всегда.

**RLE — Run Length Encoding**Копировавние цепочек одинаковых символов.   
Собираются близкие элементы и собираются в один. К примеру, пиксели близких оттенков можно объединять. Посмотрим на примере текста;  
AAAAAA (100)BBBBBBBB(100)  
Занимает 200 байтов;  
При сжатии получится 100A 100B; K = 50;  
После сжатия появляются **управляющие байты.** Они идут перед каждым сиволом, показыая, сколько раз надо повторять сиволы. Если первый бит управляющего байта — единица, **следующий** байт повторяется количество раз, указанное в остальных 7 битах. Если байт начинается с 0, остальные 7 бит показывают **количество следующих байтов, которые не меняются**.

**Сравнение алгоритмов**Равномероне кодирование восьмибитным кодом\_ 140; О 68; Е 68; Н 64; Т 60;  
(140 + 68 + 68 + 64 + 160) \* 4 = 3200 бит  
Равномерное кодирование трёхбитным кодом  
(140 + 68 + 68 + 64 + 160) \* 3 + 48 (словарь) = 1248 бит  
Неравномерное кодирование   
Код Шеннона-Фано — 924 бита  
Код Хаффмана — 920 бит