Отчёт 10 задание 1 вариант

**RLEDeflate**

Этот метод принимает входной массив src беззнаковых символов и его длину len. Затем он сжимает входной массив, используя алгоритм Run Length Encoding (RLE). Метод ищет серии подряд идущих идентичных символов во входном массиве и заменяет их на один символ, за которым следует длина серии. Затем метод сохраняет сжатые данные в выходной массив dst.

**RLEInflate**

Этот метод принимает входной массив src сжатых данных и его длину len. Он распаковывает входной массив, используя RLE алгоритм и сохраняет распакованные данные в выходной массив dst. Метод читает сжатые данные парами по два: первый байт - это символ, который должен быть повторен, а второй байт - количество раз, которое он должен быть повторен. Затем метод повторяет символ указанное количество раз и сохраняет его в выходном массиве.

**Кодирование COBS**

В этом проекте метод кодирования COBS (Consistent Overhead Byte Stuffing) применяется к сжатым данным, полученным с помощью алгоритма RLE. COBS используется для устранения наличия нулевых байтов в сжатых данных, так как нулевые байты могут вызывать проблемы при работе с некоторыми протоколами связи.

## Эффективность сжатия методом RLE

Эффективность сжатия методом RLE зависит от типа данных, которые сжимаются. Если в данных много серий подряд идущих идентичных символов, то соотношение сжатия будет высоким. Однако, если данные не содержат повторяющихся шаблонов, то соотношение сжатия будет низким. В целом, алгоритм RLE наиболее эффективен для сжатия изображений и аудиоданных, где часто встречаются длинные серии идентичных данных.  
  
В моем примере кодирование оказалось не эффективным, так как длина закодированного файла оказалась больше, чем исходного.