Программа работает по следующему алгоритму:

1. Считываются входные данные и строится кодовое дерево (одно дерево для простого варианта кода, и N деревьев для контекстного, где N – количество различных символов во входных данных)
2. На основе полученных кодовых слов производиться кодирование и запись входного файла, информация об используемых кодовых словах записывается в заголовок файла.
3. Производится чтение закодированных данных и их декодирование.

В качестве тестового файла использовался файл, сгенерированный с помощью марковской цепи, описываемой следующей матрицей переходных вероятностей:

[ 0.00125126 0.563585 0.193304 0.174108 0.067751;

0.858943 0.0149846 0.0914029 0.00466933 0.0299997;

0.0089114 0.37788 0.531663 0.0570391 0.0245064;

0.607685 0.30195 0.0237434 0.0562151 0.0104068;

0.00878933 0.91879 0.0635395 0.0001384 0.00888097];

Энтропия данного источника равна 1.2463 бит/символ.

Исходный размер файла - 9999999(байт)

Размер файла при энтропийном кодировании 1 557 874,8 (байт)

Размер файла сжатого с помощью кода Хаффмана - 2 237 294(байт)

Размер файла сжатого с помощью контекстного кода Хаффмана - 1 827 467(байт)

Так же были проведены тесты с файлом, сгенерированным с помощью марковской цепи, описываемой матрицей переходных вероятностей, составленной самостоятельно:

[0.9 0.02 0.08 0.04 0.06;

0.01 0.9 0.09 0.08 0.02;

0.02 0.08 0.9 0.03 0.07;

0.05 0.02 0.05 0.9 0.08;

0.04 0.06 0.05 0.05 0.9];

Энтропия данного источника равна 0.9448 бит/символ.

Исходный размер файла - 9999999(байт)

Размер файла при энтропийном кодировании 1 180 999,8 (байт)

Размер файла сжатого с помощью кода Хаффмана - 2 040 990 (байт)

Размер файла сжатого с помощью контекстного кода Хаффмана - 1 243 975 (байт)

При увеличении размеров файла различия между теоретическими и практическими значениями вероятностей будут стремиться к нулю, однако при кодировании файлов маленького размера эффективнее использовать вероятности, полученные при анализе входных данных.