Abstract

The Lempel Ziv Welch (LZW) Algorithm used dictionary technique to compress data. Dictionary is the source to keep many kind of symbols combination, but the array based dictionary has weakness in searching and inserting string by brute force way so that Binary Search Tree is implemented to the LZW's dictionary. Both of that two methods have similarity which implement initialization process to 256 ASCII's characters. Whereas The Huffman Algorithm use statistic technique to compress data. The first phase of Huffman is counting the frequency of each simbol appearance then making binary tree which contains sequence of each simbol from the most till the least.The Huffman code for each simbol is compression output that kind of code chain read from leaf till its root.

This final assignment will analyze the performance of two compression algorithm which are Huffman Algorithm and LZW Binary Search Tree Algorithm in compressing and decompressing process of plaintext and Rijndael ciphertext file by observing parameters, ratio, compression time and decompression time.

According to the result of this experiment, it can be concluded that implementing binary search tree structure on LZW algorithm's dictionary has not yet optimal if compared with Huffman algorithm.

Key Words : compression, LZW, Binary Search Tree, Dictionary, Huffman, ratio, time