ABSTRAK

Pada penyimpanan Memori, ukuran data yang besar membutuhkan jumlah memori yang besar juga. Selain itu, waktu yang digunakan untuk mengunduh data tersebut menjadi lama. Salah satu solusi untuk mengatasinya adalah dengan melakukan kompresi data. Salah satu algoritma kompresi data yang terkenal adalah algoritma LZW. Walaupun algoritma LZW ini cukup tekenal, algoritma ini memiliki kekurangan. Salah satunya yaitu waktu yang dibutuhkan untuk membuat data kamus pada algoritma LZW tergolong lama. Algoritma LZW bisa ditingkatkan dengan cara membuat jumlah bit secara dinamis. Algortima ini diberi nama Algoritma OLZW. Namun, algoritma OLZW ini tidak memiliki batasan jumlah kamus sehingga mempengaruhi kecepatan saat melakukan teknik kompresi dan dekompresi.

Solusinya yaitu menambahkan batasan kamus pada Algoritma OLZW. Selain itu, penggunaan Struktur Data *Binary Search Tree* (BST) juga dapat meningkatkan performa Algortima tersebut. Khususnya pada saat melakukan teknik kompresi data.

Hasil dari penelitian ini dapat disimpulkan bahwa Algoritma LZW memiliki durasi yang paling cepat dibandingkan dengan algoritma lainnya. Sementara itu, Algoritma OLZW dengan batasan maksimal kamus sebesar 20 bit memiliki tingkat efisiensi rata-rata paling tinggi. Penggunaan Struktur Data BST juga dapat meningkatkan kecepatan pada saat melakukan kompresi data jika dibandingkan dengan menggunakan *Array*.

Kata Kunci: Kompresi Data Text, OLZW, *Binary Search Tree*, Batasan Kamus