ASSIGNMENT 5 DESIGN DOCUMENT

Chris Moon

February 2023

1 Goal

Implement the Lempel-Ziv method of file compression and decompression, utilizing tries and words ADTs.

2 Pseudocode

TRIES

-WHAT IS A TRIE:
-tree type data structure, made up of connected individual nodes
-first letter is the parent, all other letters following are children
-ex: A is a parent, N and T are children, for the words AN and AT
-each node is a struct, containing an array and a value
-the value is referred to as a code
-the array contains pointers to 256 nodes, one per ASCII character
-HOW TO IMPLEMENT CONSTRUCTION AND DELETION:
-trie create uses struct syntax (arrows for parameters)
-should allocate memory for the node first, and then the array, using malloc
-takes a code value as a parameter, and sets the code of the node using arrow syntax
-trie node delete should free the array pointer first, and then the pointer to the node
-create a root first, which is a node with an empty code
-TREE FUNCTIONS
-implement the function, trie create, which will just use the trie constructor with the given EMPTY CODE value as the code parameter
-this will create the root node of the trie tree
-trie delete should take a node, and clear that node and all its children
-recursively call trie node delete here, starting from the lowest children, and working up, as to not lose the ability to reference lower children
-trie reset should delete the trie starting at the root node
-call trie delete(root)

WORDS

-allows translation of codes to symbols
-word type contains a list of symbols and a length
-also has a constructor and delete function
-an array of words, or a word table, can also be created