

## Index of words

The program accepts a set of text files, creates the index and determines the location of each word, the row and the column where the first character of each word appears in each file. The words will be stored and searched using a 1024-position hash table, which is an index table, using the following function:

$$h(s) = \sum_{k=0}^{ak} 2^{ak-k} * s_k \text{ modulo } 1024$$

In this function,  $ak$  is the length of the word and  $s_k$  is the ASCII code number to which the  $k$  character of the word corresponds.

To continue, suppose that WORD1, WORD2, WORD3 and WORD4 must all be placed in position 2 of the table. The WORD1 will then be placed at the root of a binary search tree, while the other words will be placed in the tree according to their lexical order. Three pointers will start from the node, which contains each word. Consider for example the WORD1 of the above example. Two of the indexes of the node containing the WORD1 will point to the children of this node, and the third index will show the beginning of a simply linked list. Each node in the list will contain the name of the file where the word was found. From the FILE1 node, another simply linked list starts and contains the row and column numbers in each file where the word is found. For example, the WORD1 exists in FILE1, in positions (3,25) and (9,12). It also exists in FILE2 in places that will appear in the list that starts from the corresponding node.

So, the user gives a word and the program will display the location of that word in all the files that come with the program.