**Data Structure\_2071035 Lee Somin**

**Technical Report – huffman.cpp**

*Theorical Explanation of Functions in ‘huffman.cpp’*

**typedef struct TreeNode**

the structure contains the character(char data), frequency of the data(int key), Huffman codeword and its size, and the addresses of left and right child of the node.

**typedef struct bits\_stream**

the structure contains the bit stream which represents the encoded string, and the length of the stream.

**huffman\_traversal**

Inputs: TreeNode \*node, int digit, int code

Return: non

This function is made for making the codeword for each character. When the Huffman tree is given, the function recursively travels and adds digit to each codeword by copying parent’s codeword by memcpy() function and adding a new digit decided by its left or right movement. When it reaches the leaf node, it determines the character of the node and adds the codeword to the proper row of ‘m\_LUT’. Then, it assigns the bit size decided at the front of the function.

**huffman\_encoding**

Inputs: char \*str, bits\_stream \*bits\_str

Return: non

This function is for encoding the given string. When it reads a character from the string, the code word in ‘m\_LUT’ is copied to the bit stream through while loop. ‘int wtcode’ indicates which index to start writing. When writing is over, the size of codeword is added to the length of the bit stream. When this operates within all characters in the string, it prints out the length of bit stream and the actual bit stream.

**huffman\_decoding**

Inputs: bits\_stream \*bits\_str, TreeNode \*node, char \*decoded\_str

Return: non

This function is for decoding the given bit stream. The outermost while statement loops until the reading of the code is over. Inside of the second while loop, ‘tnode’ is updated following the address of Huffman tree given and as the function reads bits of the stream. When it meets the leaf node, it adds ‘char data’ of ‘tnode’ to the decoded string. When all the loop is over, it prints out the number of decoded characters and the decoded string.

(Result Screen Shot Continued)

**Result:**

텍스트이(가) 표시된 사진

자동 생성된 설명