```
*************************************************************
* PROGRAMMED BY : Wesley Chok
* CLASS         : CS1D
* SECTION       : MW 2:30 - 4:50p
* ASN #15       : Huffman Coding
*************************************************************

/******************************************************************************
* OUTPUT (This is a comment, not part of the output)
* ------------------------------------------------------------------------------
* This outputs the frequency table, huffman code for each letter of the
* Gettysberg address, encoded Gettysburg address and the decoded Gettysburg
* address. The first output lists all of the characters on the left and
* frequencies on the right. The second output outputs the entire encoded
* huffman code. Finally the third output outputes the decoded huffman code.
* ------------------------------------------------------------------------------
/******************************************************************************

Character With there Frequencies:
  00
, 100011
- 1100111
. 11110110
B 11110101000
F 11110101001
G 11110101010
I 111101000
L 11110101100
N 11110101101
T 1111010011
W 1111010010
a 1011
b 1010110
c 111100
d 11010
e 011
f 110000
g 110010
h 0101
i 11011
k 1111010111
l 10000
m 1010111
n 11111
o 1001
p 1100110
q 11110101011
r 0100
s 10100
t 1110
u 100010
v 101010
w 110001
y 11110111

Encoded Huffman Code:1111010100110011000100100001010011110010010100011001011111
                     1110100010100011101010011111100111101110111011010010100001
```

```
0111100101001001001100010010000110000101111100101011010010 1
```

```
                     1011011111000110001011001010010111111011100010101101000111O
```

a final resting place for those who here gave their lives that that nation might live. It is altogether fitting and proper that we should do this.

But, in a larger sense, we cannot dedicate -- we cannot consecrate -- we cannot hallow -- this ground. The brave men, living and dead, who struggled here, have consecrated it, far above our poor power to add or detract. The world will little note, nor long remember what we say here, but it can never forget what they did here. It is for us the living, rather, to be dedicated here to the unfinished work which they who fought here have thus far so nobly advanced. It is rather for us to be here dedicated to the great task remaining before us -- that from these honored dead we take increased devotion to that cause for which they gave the last full measure of devotion -- that we here highly resolve that these dead shall not have died in vain -- that this nation, under God, shall have a new birth of freedom -- and that government of the people, by the people, for the people, shall not perish from the earth.

```
#include <bits/stdc++.h>
#define MAX_TREE_HT 256
using namespace std;

/******************************************************************************
 * ASN #15 - Huffman Coding
 * ----------------------------------------------------------------------------
 * This program will output the class heading
 * ----------------------------------------------------------------------------
 *
 *****************************************************************************/

map<char, string> codes;    // holds the huffman value

map<char, int> freq;        // holds the frequency of the character via input
                            // data


struct MinHeapNode
{
    char data;                  // holds the input character
    int freq;                   // holds the frequency of the character
    MinHeapNode *left;          // left child
    MinHeapNode *right;         // right child

    /*****************************************************************************
     * Constructor MinHeapNode
     *  ____________________________________________________________________________
     *  Instantiates the variables given in the struct.
     *  _________________________________________________________________________
     * ***********************************************************************/
    MinHeapNode(char data, int freq)
    {
        left = right = NULL;
        this->data = data;
        this->freq = freq;
    }
};


struct compare
{
    /*****************************************************************************
    * ____________________________________________________________________________
    *   Function for the priority queue, used for utility purposes only.
    *   _________________________________________________________________________
    * ***********************************************************************/
    bool operator()(MinHeapNode* l, MinHeapNode* r)
    {
        return (l->freq > r->freq);
    }
};

/****************************************************************************
* FUNCTION printCodes
* ____________________________________________________________________________
*   This function prints out the characters and their respective huffman
*   values, used for utility purposes only.
*   ________________________________________________________________________
* ***********************************************************************/
void printCodes(struct MinHeapNode* root, string str)
{
    if (!root)
    {
        return;
    }
    if (root->data != '$')
    {
        cout << root->data << ": " << str << "\n";
    }

    printCodes(root->left, str + "0");
    printCodes(root->right, str + "1");
}


/***************************************************************************
* FUNCTION storeCodes
* ___________________________________________________________________________
*   This function stores the characters and their respective huffman values
*   into a hash table.
*   ________________________________________________________________________
* ***********************************************************************/
void storeCodes(struct MinHeapNode* root, string str)
{
    if (root==NULL)
    {
        return;
    }
    if (root->data != '$')
    {
        codes[root->data]=str;
    }

    storeCodes(root->left, str + "0");
    storeCodes(root->right, str + "1");
}

priority_queue<MinHeapNode*, vector<MinHeapNode*>, compare> minHeap;
                                            // stores the heap tree

/***************************************************************************
* FUNCTION HuffmanCodes
* ___________________________________________________________________________
*   This function will construct the huffman tree and store it in minHeap.
*   The for loop will push the heap values until it reaches the end of the
*   frequency map.
*   ________________________________________________________________________
* ***********************************************************************/
void HuffmanCodes(int size)
{
    struct MinHeapNode *left, *right, *top;
    for (map<char, int>::iterator v = freq.begin(); v != freq.end(); v++)
    {
        minHeap.push(new MinHeapNode(v->first, v->second));
    }
    while (minHeap.size() != 1)
    {
        left = minHeap.top();
        minHeap.pop();
        right = minHeap.top();
        minHeap.pop();
        top = new MinHeapNode('$', left->freq + right->freq);
        top->left = left;
        top->right = right;
        minHeap.push(top);
    }
    storeCodes(minHeap.top(), "");
}

/***************************************************************************
* FUNCTION calcFreq
* _____________________________________________________________________________
*   This function will store the map of each character with its respective
*   frequency in the input string located at int main()
*   __________________________________________________________________________
* ***********************************************************************/
void calcFreq(string str, int n)
{
    for (int i=0; i<str.size(); i++)
    {
        freq[str[i]]++;
    }
}

/***************************************************************************
* FUNCTION decode_file
* _____________________________________________________________________________
*   This function will iterate the string value. For instance, if string_value
*   equals 1 then it will move the node to the right, if string_value equals
*   0 then it will move the node to the left.
*   __________________________________________________________________________
* ***********************************************************************/
string decode_file(struct MinHeapNode* root, string string_value)
{
    string ans = "";
    struct MinHeapNode* curr = root;
    for (int i = 0; i < string_value.size(); i++)
    {
        if (string_value[i] == '0')
        {
           curr = curr->left;
        }
        else
        {
           curr = curr->right;
        }


        if (curr->left==NULL and curr->right==NULL)
```

```cpp
        {
            ans += curr->data;
            curr = root;
        }
    }

    return ans+'\0';
}


int main()
{

    /**************************************************************************
        * CONSTANT
        * -------------------------------------------------------------------------
        * OUTPUT - USED FOR CLASS HEADING
        * -------------------------------------------------------------------------
        * PROGRAMMER : Wesley Chok
        * CLASS      : CS 1D
        * SECTION    : MW 2:30p - 4:50p
        * ASN_NUM    : 15
        * ASN_NAME   : Huffman Coding
        **************************************************************************/
    string str = "Four score and seven years ago our fathers brought forth on "
            "this continent, a new nation, conceived in Liberty, and dedicated "
            "to the proposition that all men are created equal. "
            ""
            "Now we are engaged in a great civil war, testing whether that "
            "nation, or any nation so conceived and so dedicated, can long "
            "endure. We are met on a great battle-field of that war. We have "
            "come to dedicate a portion of that field, as a final resting place "
            "for those who here gave their lives that that nation might live. "
            "It is altogether fitting and proper that we should do this. "
            ""
            "But, in a larger sense, we cannot dedicate -- we cannot "
            "consecrate -- we cannot hallow -- this ground. The brave men, "
            "living and dead, who struggled here, have consecrated it, far "
            "above our poor power to add or detract. The world will little note, "
            "nor long remember what we say here, but it can never forget what"
            " they did here. It is for us the living, rather, to be dedicated "
            "here to the unfinished work which they who fought here have thus "
            "far so nobly advanced. It is rather for us to be here dedicated to "
            "the great task remaining before us -- that from these honored dead "
            "we take increased devotion to that cause for which they gave the "
            "last full measure of devotion -- that we here highly resolve that "
            "these dead shall not have died in vain -- that this nation, under"
            " God, shall have a new birth of freedom -- and that government of "
            "the people, by the people, for the people, shall not perish from "
            "the earth.";

    string encodedString;       // holds data for the encoded string
    string decodedString;       // holds data for the decoded string

    calcFreq(str, str.length());
    HuffmanCodes(str.length());

    cout << "Character With there Frequencies:";

    cout << endl;

    for (auto v=codes.begin(); v!=codes.end(); v++)
    {
        cout << v->first <<' ' << v->second << endl;
    }

    for (auto i: str)
    {
        encodedString+=codes[i];
    }

    cout << endl;

    cout << "Encoded Huffman Code:" << encodedString << endl << endl;

    decodedString = decode_file(minHeap.top(), encodedString);

    cout << endl;

    cout << "Decoded Huffman Code:" << decodedString << endl << endl;


    return 0;
}
```