Assignment 6 (Huffman coding):

Cpp code

```
#include <bits/stdc++.h>
#include <cstring>
using namespace std;

struct Node
{
    char data;
    unsigned freq;
    Node *left, *right;
    Node(char data, unsigned freq)
    {
        left = right = NULL;
        this->data = data;
        this->freq = freq;
    }
};

struct ComparisonType
{
    bool operator()(Node *l, Node *r)
    {
        return (l->freq > r->freq);
    }
};

void printchar_array(struct Node *root, string str)
{
    if (!root)
    {
        return;
    }
    if (root->data != '#')
    {
        cout << "\t"<<root->data << " : " << str << "\n";
    }
    printchar_array(root->left, str + "0");
    printchar_array(root->right, str + "1");
}

int main()
{
    string s;
    cout << "Enter string: ";
    cin >> s;
    int size = s.length();
    char char_array[size];
    int freq[size];
    strcpy(char_array, s.c_str());
    for (int i = 0; i < size; i++)
    {
```

```
        cout << "Enter frequency (" << char_array[i] <<"): ";
        cin >> freq[i];
    }
    priority_queue<Node *, vector<Node *>, ComparisonType> Heap;
    for (int i = 0; i < size; ++i)
    {
        Heap.push(new Node(char_array[i], freq[i]));
    }
    struct Node *left, *right, *top;
    while (Heap.size() != 1)
    {
        left = Heap.top();
        Heap.pop();
        right = Heap.top();
        Heap.pop();
        top = new Node('#', left->freq + right->freq);
        top->left = left;
        top->right = right;
        Heap.push(top);
    }
    cout << "\ncharacter" << " : " << "frequency"<< "\n";
    printchar_array(Heap.top(), " ");
    return 0;
}
```

- Test cases passed
- Completed on 24/10/22

Q/A:

1. How long did you spend on this assignment?
    a. 4hr
2. Based on your effort, what letter grade would you say you earned?
    a. On a scale of 1 to 10. I would grade this as 10/10.
3. Based on your solution, what letter grade would you say you earned?
    a. On a scale of 1 to 10. I would grade this as 9/10.
4. Provide a summary of what doesn't work in your solution, along with an explanation of how you attempted to solve the problem and where you feel you struggled?
    a. Create the structure of nodes to store data and frequency to create a min heap of all nodes.
    b. Extract two nodes with the minimum frequency from the min heap. Then create a new internal node with a frequency equal to the sum of the two nodes frequencies. First extract the left child and then extract the other node on the right. Add this node to the min heap.
    c. Repeat steps 3 until the heap contains only one node.