Analysis of Algorithms (CIS -606)

Extra Credit Assignment (Huffman's Algorithm)

Due date: 05/09/2023 (5 points)

Maitri Radhesh
CSU ID# 2831749

What is Huffman Coding?

Huffman coding is a lossless data compression algorithm. The idea is to assign variable-length codes to input characters, lengths of the assigned codes are based on the frequencies of corresponding characters.

The variable-length codes assigned to input characters are Prefix Codes, means the codes (bit sequences) are assigned in such a way that the code assigned to one character is not the prefix of code assigned to any other character. This is how Huffman Coding makes sure that there is no ambiguity when decoding the generated bitstream.

There are mainly two major parts in Huffman Coding:

1. Build a Huffman Tree from input characters.
2. Traverse the Huffman Tree and assign codes to characters.

Algorithm:

The method which is used to construct optimal prefix code is called Huffman coding.

This algorithm builds a tree in bottom-up manner. We can denote this tree by T

Let, |c| be number of leaves

|c| -1 are number of operations required to merge the nodes. Q be the priority queue which can be used while constructing binary heap

In my code, I am using the ***Heap*** data structure in order to analyze the tree of the "Huffman Code". It makes it easier to iterate it if I am considering it as a heap as I can mode towards the right-sub-tree and left-sub-tree easily.

What is a Heap data structure?

A Heap is a special Tree-based data structure in which the tree is a complete binary tree.

Operations of Heap Data Structure:

- Heapify: a process of creating a heap from an array.
- Insertion: process to insert an element in existing heap time complexity O(log N).
- Deletion: deleting the top element of the heap or the highest priority element, and then organizing the heap and returning the element with time complexity O(log N).
- Peek: to check or find the most prior element in the heap, (max or min element for max and min heap).

Types of Heap Data Structure:

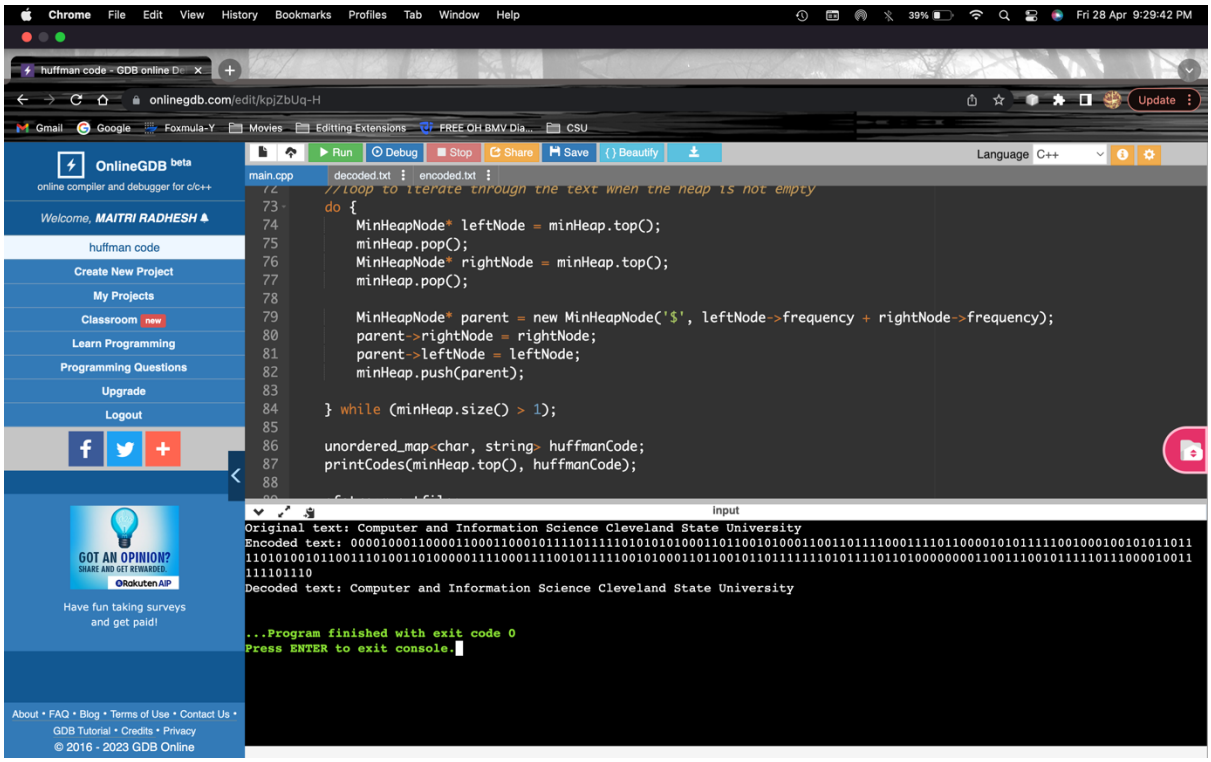
Generally, Heaps can be of two types:

1. Max-Heap: In a Max-Heap the key present at the root node must be greatest among the keys present at all of its children. The same property must be recursively true for all sub-trees in that Binary Tree.
2. Min-Heap: In a Min-Heap the key present at the root node must be minimum among the keys present at all of its children. The same property must be recursively true for all sub-trees in that Binary Tree.

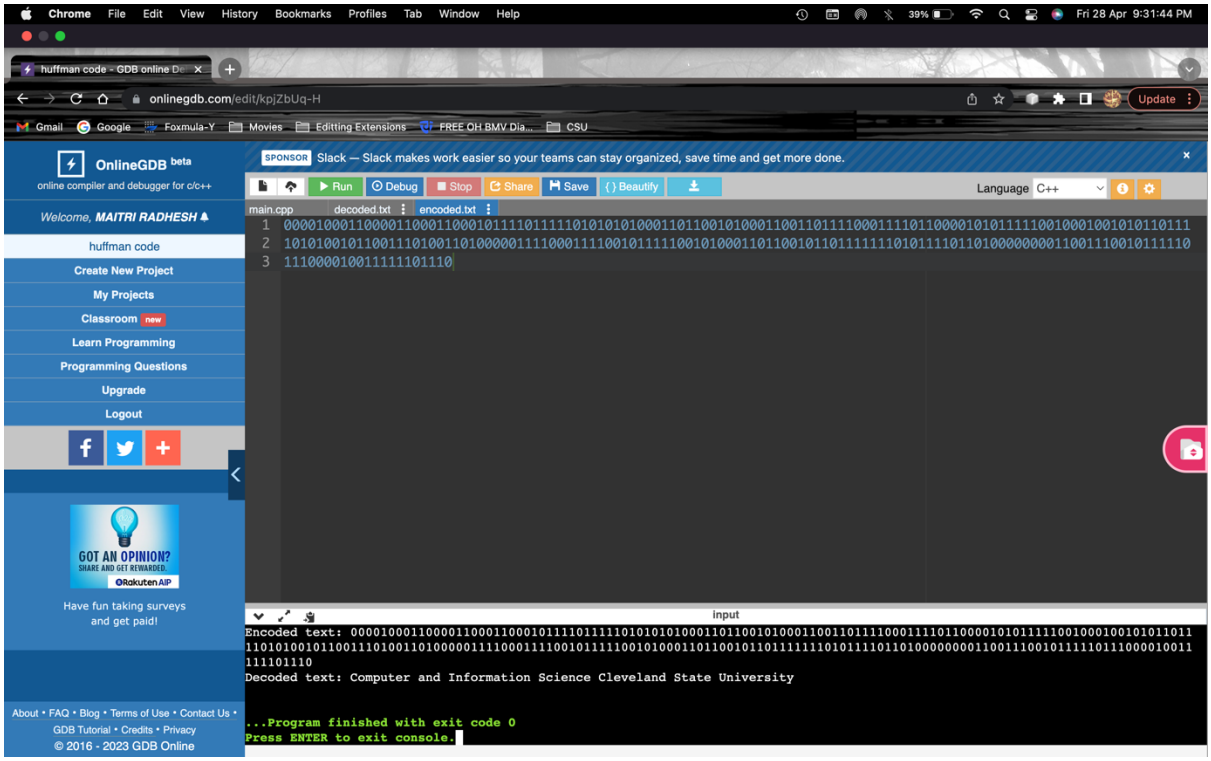
I am making use of the a ***MinHeap***.

Outputs:

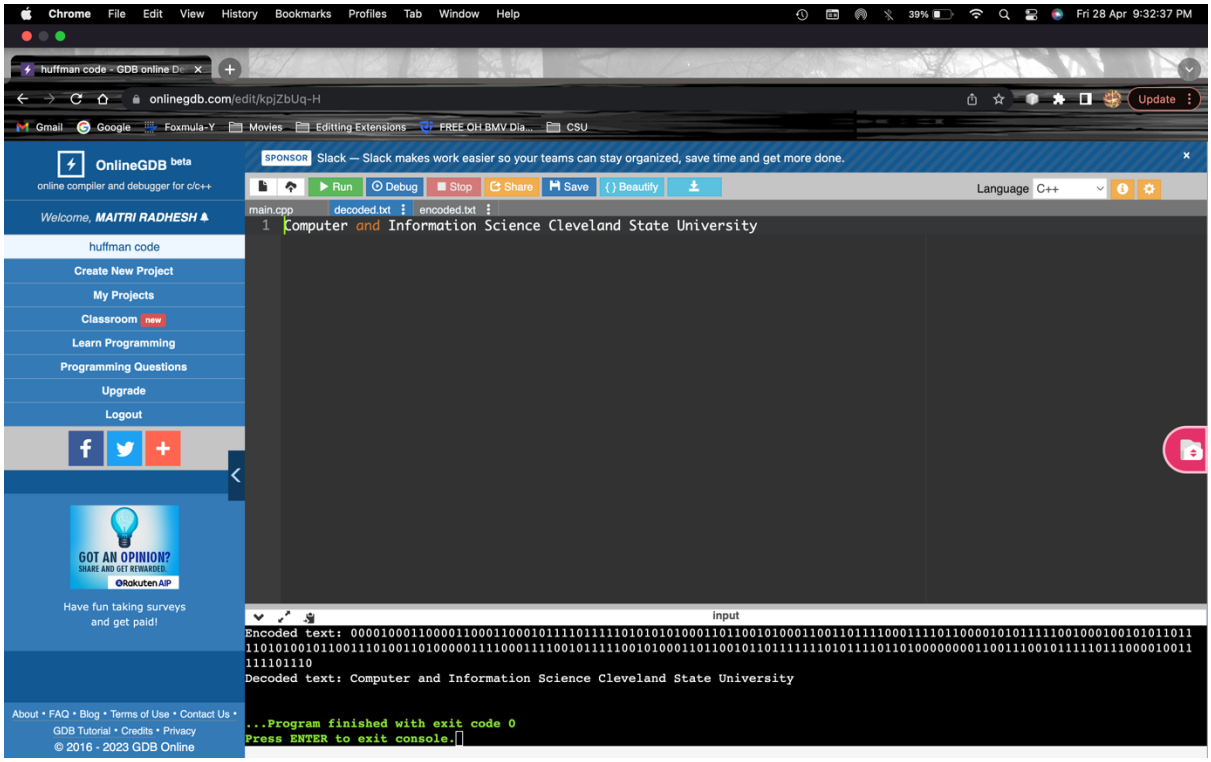
On the console:



Encoded.txt:



Decoded.txt:



I have made use of an online compiler.
Link: https://www.onlinegdb.com/edit/kpjZbUq-H