The Huffman algorithm

Bitstream

String codetable

Char* buffer

String carryover

Uses:

Reads and writes bits from files.

Huffman tree

Weight

Height

Uses:

Has 3 constructors.

Makes a new tree given a char and its frequency.

Makes a new tree without parameters.

Makes a new tree by joining two other trees.

Frequency Counter

Array of size 257

Uses:

Counts the occurrence of the chars using the concept of hash-tables.

Main()

Arguments : argv[] and argc[]

Checks if the user gives a valid input. If argv[1] == '-c', calls the compress function. If argc[2] == '-d', calls decompress.

Compress(){

Opens file and makes the frequency table based on the file. Counts the chars and makes a binary tree for them. Then it joins the binary tree until the Queue gets empty and finally encodes the data into the huff file.

}

Decompress(){

Open the huff file and push items into a priority queue based on their original order and populate the huff code table. The print the files back to the decompressed file and close both files.

}