BLG 372E – Analysis of Algorithms, Spring 2010

Assignment 2 – Huffman Coding
Handed out: 09.04.2010
Due: 22.04.2010

Problem:
Find the Huffman Coding for a given text. See the supplementary file "a2-supp.pdf" at Ninova for details of the algorithm.
A text consists of lowercase letters in the Latin alphabet(a-z), space characters( ) and periods(.).
Submit your source code and a report through Ninova.

In your report:
- Explain how Huffman Coding algorithm works step by step. (10 pts)
- Analyse the time and space complexity of the algorithm. (10 pts)
- Discuss if the time or space complexity of your algorithm could be lowered by using other data structures. (10 pts)
- Discuss the optimality of the algorithm. (10 pts)
- Compress the sample files one by one with huffman coding and gzip respectively. Compare the compression ratios of these algorithms. Explain the differences of gzip and huffman coding algorithms. (10 pts)

Implementation Details:
- All your code must be written in C++ and able to compile and run on linux/unix using g++.
- You are not allowed to use any library except STL.
- Your should write a Makefile with a defined "all" and "test" targets. "all" target will compile all the necessary files in your project and "test" target will run your project with the test input.
- Your submission should only contain the necessary files (sources, headers and a makefile)

Example run:

Input format: Input file contains the text to be coded:

```
lobortis tation ludus causa iaceo camur ibidem te immitto olim iusto os. adsum quis
duis ullamcorper amet laoreet capio regula loquor indoles quae praesent sit. usitas
gilvus humo ludus vero brevitas facilisis vero torqueo sed velit tristique saepius
iriure. illum esse augue virtus jugis vel eum patria. qui plaga suscipit esse
damnum praemitto huic mara nimis blandit minim.
```

Output format: Output file "freq.txt" should contain the frequences of the characters in the original text, "hcodes.txt" should contain the Huffman Coding, "encoded.txt" should contain the encoded text, and "stats.txt" file should include the maximum bits required for coding, length of the encoded text in bits, and the compression ratio. Original text is in ASCII format (each character is represented with 8 bits).

```
freq.txt:          hcodes.txt          encoded.txt                    stats.txt

a 27               a 1001              000101010000010010100111100    8 // Max bits required
b 4                b 000010            001111101010110001001100001    1523 // length of the
c 8                c 110011            101011100011010001111110 10    encoded text
d 9                d 01000             001111111101011100111001 11    0.50 // compression ratio
e 29               e 1101              11111010011010111001110 01
f 1                f 00001110          11101010110111001110010 01
g 5                g 010011            01111001110101100001001 10
h 2                h 0000110           10001101001010110001101 10
i 40               i 011               10110010001001110001000 01
j 1                j 00001111          01101010100010110010101 01
l 18               l 0001              11111111010000101101010 11
m 18               m 0010              11001001010110010100011 10
n 7                n 110001            11110010101110000111101 11
o 20               o 0101              11010101000111101111101 01
p 8                p 00000             11110001000110010010110 01
q 6                q 110000            10101001100000110100111 01
r 19               r 0011              10010010110110001010001 10
s 31               s 1110              01010100111101110110001 01
t 23               t 1000              11001110010000001101011 01
u 31               u 1111              00111101010011111100011 00
v 7                v 110010            11010001010111000011110 10
space 57           space 101           10011101011110001010000 10
. 5                . 010010            10001110111101011100 0...
```