# Huffman Coding

Name________________________________

## CS 240 Fall 2016

### Exercise 1

Consider the following string of characters (ignoring spaces):

U U D D L R L R B A S U U U U D D L A

If you were to encode this string using the same number of bits per character, how many bits would be required per character? For the entire string?

Compute the frequency of each of the characters in the string:

U D L R B A S

Next, build the Huffman tree (when creating a new node, the smaller frequency goes to the left; break ties in the initial sort alphabetically).

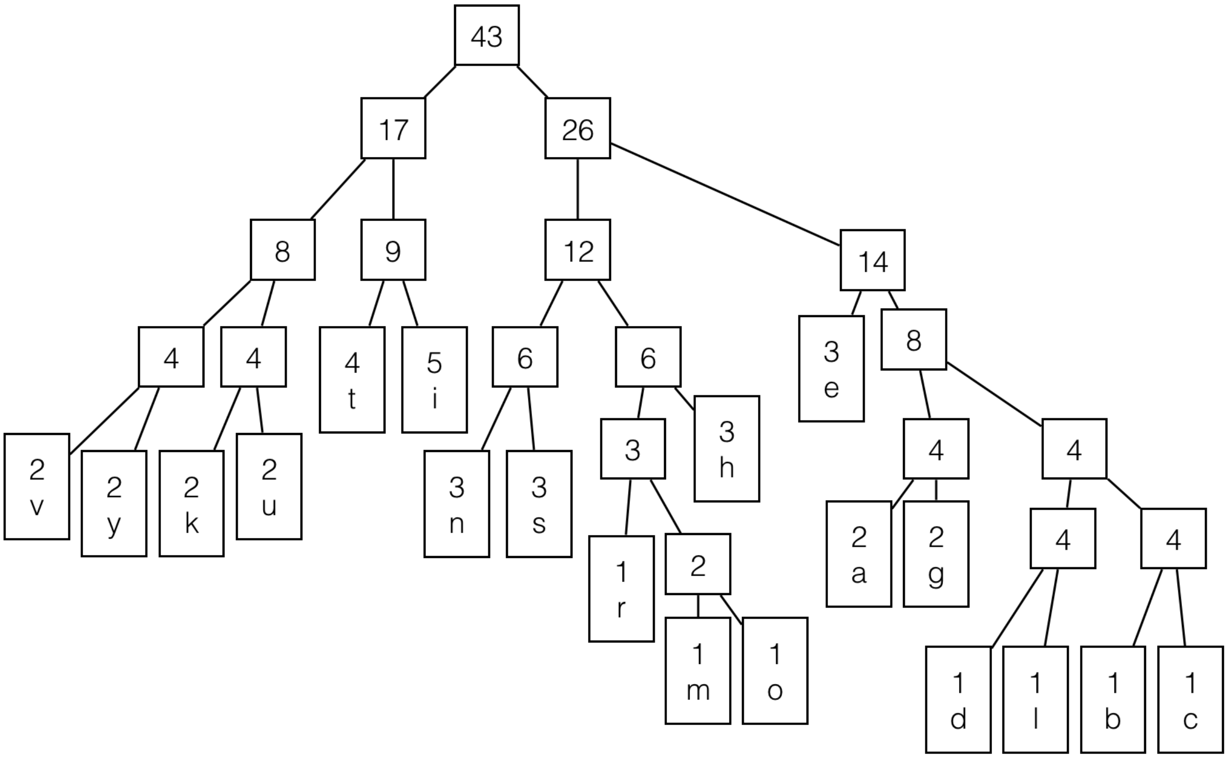
Encode the string using your Huffman tree:

How many bits did you use?

What was the compression ratio (uncompressed bit count/ compressed bit count)?

## Exercise 2

Using the following Huffman tree decode the binary strings below.



01000111010000101100001

11111111100100000011010110011

1111101101111010111100000110

0101011011100101111001

010101111100100000101001111010110000011100011101

0101011110101010110111100

There are 19 characters in the strings above. Assuming we use the same number of bits for each character, how many bits would be required to encode each string? What is the compression ratio for each string?