

Data Structures and Algorithms Homework 10

Due Wednesday Nov 6; Joseph Sepich (jps6444)

1 Problem 1

1.1 Part 1

1.2 Part 2

1.3 Part 3

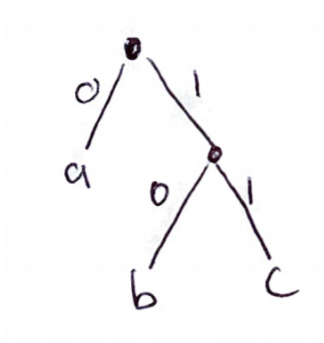
2 Problem 2

3 Problem 3

4 Problem 4

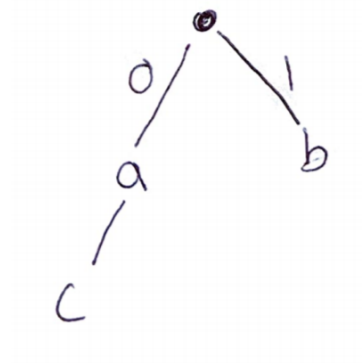
4.1 Part a

“b” and “c” in the given code would be 2 levels below the root node of the full binary tree and “a” would be 1 level below. This is the tree obtained in the greedy algorithm where the heuristic was to add the lowest frequencies to the tree first. This implies that $f_a > f_b; f_z > f_c$. The full tree is illustrated below.



4.2 Part b

This code cannot possibly be obtained. This code contains prefixes meaning that the code for a is the prefix for the code for c. For example, if you had “00” in an encoded string of bits, then the code code either mean “aa” or “c”, which means the code is ambiguous and not possibly obtained from Huffman encoding. This is also not possible in the Huffman encoding algorithm, because it implies a letter was used where an inner node is and not as a leaf, which is not possible in the algorithm, which combines two letters to make an inner node. An illustration is below.



4.3 Part c

This code cannot possibly be obtained with the Huffman algorithm. Although it is a valid code, it does not follow the way the greedy approach is setup. The herutisitc that the algorithm runs on is to add the two lowest frequencies to the tree next by combing those two nodes with a parent node. The fact that “a” is listed as 10 implies there would be another encoded letter at 11, but there is not so “a” would have to be just 1 to change this to a code obtainable from Huffman encoding. An illustration is below.

