Second Exam
CS 1102 Computer Science 2

Spring 2018

Thursday April 19, 2018
Instructor Muller

KEY

Before reading further, please arrange to have an empty seat on either side of you. Now that you are seated, please write your name **on the back** of this exam.

This is a closed-notes and closed-book exam. Computers, calculators, and books are prohibited.

- Partial credit will be given so be sure to show your work.
- Feel free to write helper functions if you need them.
- **Please write neatly.**

Problem	Points	Out Of
1		4
2		4
3		4
4		4
5		4
Total		20

Part 1: Short Answer

For true/false questions, please circle the correct answer.

1. True or false: Let t be a *binary tree* with k leaves. There are no paths in t of length greater than $\log_2 k$.

   Answer:

   `False`

2. True or false: Let t be a *full binary tree* with k leaves. There are no paths in t of length greater than $\log_2 k$.

   Answer:

   `False`

3. True or false: Let t be a *perfect binary tree* with k leaves. There are no paths in t of length greater than $\log_2 k$.

   Answer:

   `True`

4. True or false: Let t be a *Red/Black tree* with k leaves and consider paths in t without considering Red or Black link color. There are no paths in t of length greater than $\log_2 k$.

   Answer:

   `False`

5. True or false: Every Huffman coding tree is a full binary tree.

   Answer:

   `True`

6. True or false: Since the Huffman coding algorithm works for any kind of input (not just text), one can always compress a zip file to produce a still smaller file.

   Answer:

   `False`

7. True or false: Let $A = \{a, b, c\}$. $R = \{(b, b)\}$ is a transitive relation on A.

   Answer:

   `True`

8. Let $A = \{a, b, c\}$. Show any equivalence relation on A.

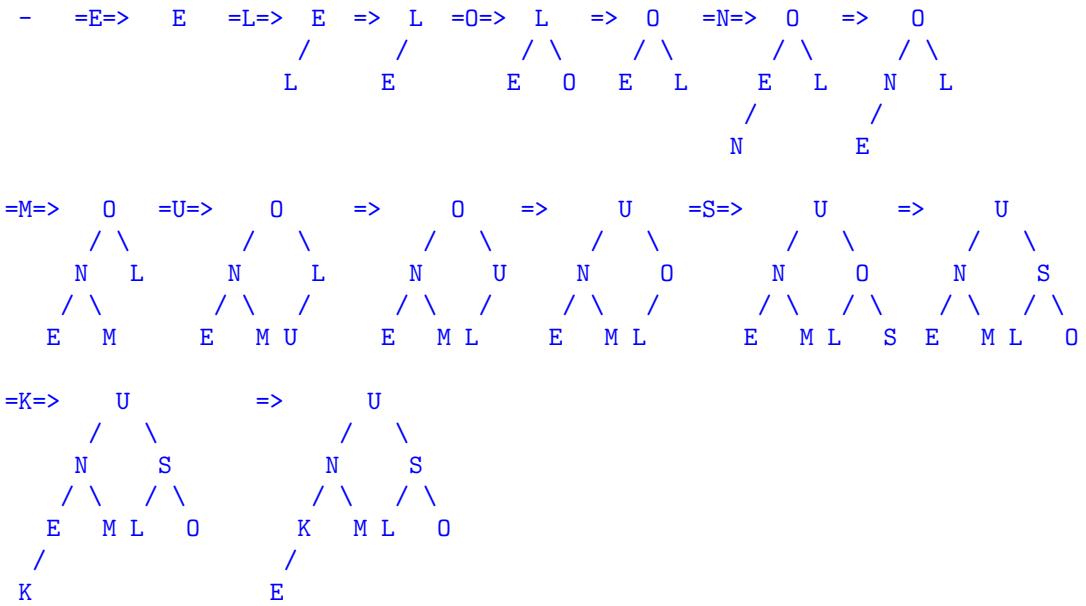
   Answer:

   `{(a, a), (b, b), (c, c)}`

Part 2: Binary Heaps

Show **all** of the successive binary trees that result from the left-to-right insertion of the letters `ELONMUSK` into an empty *max binary heap*. I.e., a binary heap in which the root contains the maximum value.

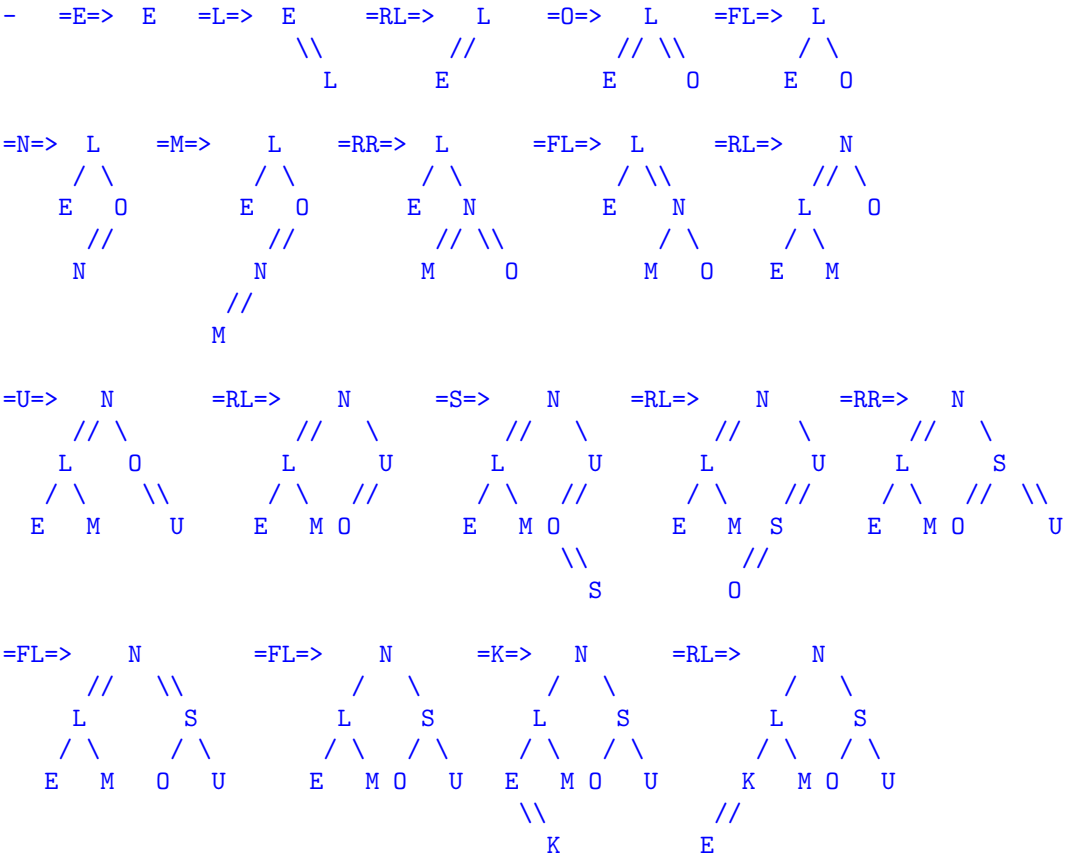
Answer:



Part 3: Red/Black Trees

Show **all** of the successive trees that result from the left-to-right insertion of the letters `ELONMUSK` into an empty *left-leaning Red/Black tree*.

Answer:



Part 4: Huffman Coding

A zip file contains the following frequency table and bit sequence.

A	M	N	O	R	T	H
2	1	1	1	1	1	1

```
01100110001110101011 00
```

The file was constructed with the same assumptions as in the problem set:

1. Letters are initially entered into the PQ in alphabetical order;
2. Ties are broken by placing the newly inserted entry *behind* all entries with the same priority;
3. In a Huffman Tree traversal, left means 0 and right means 1.

What is the uncompressed text? Please show all of your work.

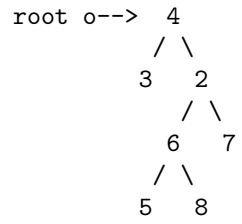
Answer:

MARATHON

Part 5: Traversing Trees

Consider a binary tree with integers in the nodes:

```
class Node {
  int info;
  Node left;
  Node right;
}
```



Write a function `int addPath(Node root, int path)` such that a call `addPath(root, path)` returns the sum of the integers on the binary path specified in `path`, with rightmost bit 0 meaning left and rightmost bit 1 meaning right. For example, with `root` as above and with `path = 5 = 0b101`, the call `addPath(root, path)` should return `4 + 2 + 6 + 8 = 20`. A call `addPath(root, path)` with `root == null` should return `0` for any path.

Answer:

```
public int addPath(Node root, int path) {
  if (root == null)
    return 0;
  Node next = path % 2 == 0 ? root.left : root.right;
  return root.info + addPath(next, path / 2);
}
```