Name: _____

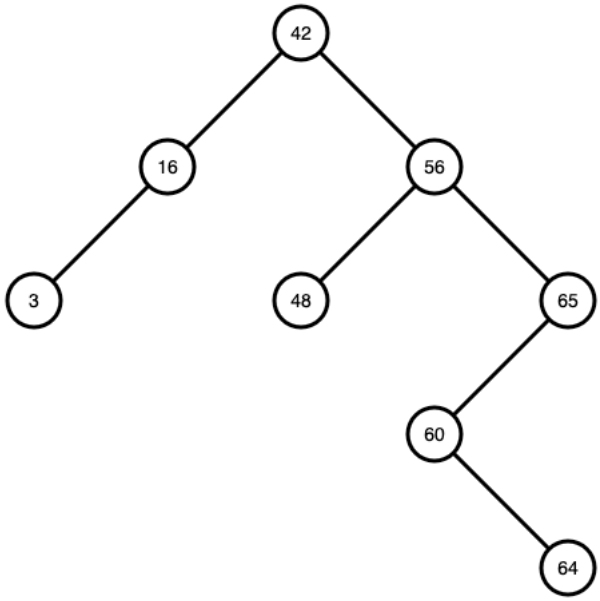
Section: _____

CST 370 – Practice Midterm 2

Question:	1	2	3	4	5	Total
Points:	20	20	20	20	20	100
Score:						

1. (20 points) A Binary Balancing Act

   (a) (points) Label the balances of the following binary tree.



   (b) (points) What is the Big-O time complexity for the insert, lookup, and delete operations on a balanced binary search tree? Remember to label your input size.

   (c) (points) Is the tree in part 1 an AVL tree? If so, justify why. If not, draw an AVL tree that contains the same numbers as the tree above.

(d) (points) In any language of your choosing, write two functions `rotateLeft` and `rotateRight`. `rotateLeft` should take a node and rotate it to the left, update the heights accordingly, and finally return the new root. `rotateRight` should take a node and rotate it right, update the heights accordingly, and finally return the new root. In C++, the node structure would look something like this:

```
struct Node {
    int data;
    int height;
    Node* left;
    Node* right;
};
```

2. (20 points) Hepta-Hashing.

   (a) (points) Hank wants to insert these numbers into a hash map with 7 buckets. This hash map has a hash function that is defined as:*$h(x) = x \bmod 7$

   $$14, 3, 28, 5, 9, 16, 32$$

   After inserting all the numbers, which numbers would be stored in which buckets?

   * $a \bmod b$ returns the remainder when dividing a by b. (For example: $12 \bmod 7 = 5$ $0 \bmod 7 = 0$, $3 \bmod 7 = 3$)

0	
1	
2	
3	
4	
5	
6	

   (b) (points) Describe the hash function used to bucket numbers in radix sort.

   (c) (points) Draw out the partially sorted list after each iteration of radix sort on the following numbers.

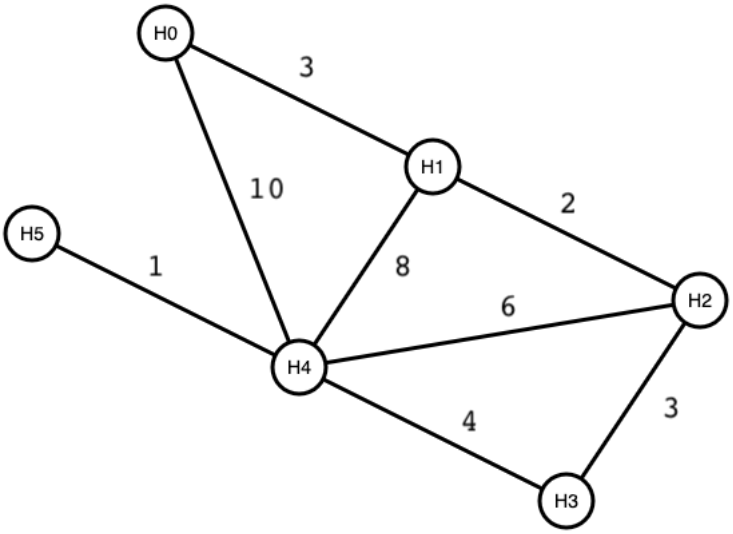
   [234, 52, 9, 44, 137, 653, 36, 169, 200]

3. (20 points) Super Special Search Problem

   (a) (points) When would you use the bellman-ford algorithm over dijkstras or a*?

   (b) (points) What is a heuristic function? What does it mean for one to be admissable?

   (c) (points) Given the weighted graph below, use dijkstra's algorithms to find the length of the shortest path between H0 and H5. Be sure to draw the distance map showing the distances.



4. (20 points) Tricky Tries

   (a) (points) Tiffany inserts the words below into an empty trie. Draw what the trie looks like after inserting all the words.

```
pig
pie
pit
spit
pal
sort
spoon
pat
```

   (b) (points) How many nodes are left in the trie after deleting the word `spoon`?

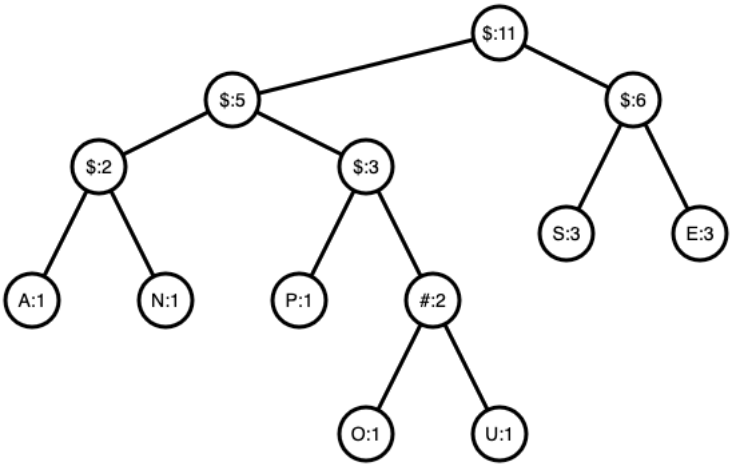
(c) (points) In any language of your choosing, write a function that prints out all words in a trie in alphabetical order. You may assume there is a helper function `char getChar(int index)` that will return the character represented by that index.

5. (20 points) Huffman Huffing

   (a) (points) Draw a huffman tree that encodes "mississippi monsoon". A character frequency map is given below.

Space	M	N	P	O	I	S
1	2	2	2	3	4	5

   (b) (points) Given the huffman tree below, decode 1000100001011



(c) (points) In the programming language of your chouce, Implement a function for producing an encoding map given the root of a huffman tree.