Quiz 5b

1. (2 point) Which would be faster, depth first search or breadth first search or either, for finding the way through a maze if a path is picked on random whenever there is a fork.
2. (2 point) Consider the following procedure for binary trees.

(define (mystery tree)

(if (equal? tree the-empty-tree)

the-empty-tree

(make-tree (max (entry tree)

(if (equal? (right-branch tree) the-empty-tree)

(entry tree)

(entry (right-branch tree)))

(if (equal? (left-branch tree) the-empty-tree)

(entry tree)

(entry (left-branch tree))))

(mystery (left-branch tree))

(mystery (right-branch tree)))))

What would the output tree look like if performed on the tree below?

0

/ \

1 2

/ \ / \

3 4 5 6

/ \

7 8

/ \

9 10

1. (1 point) Consider the binary tree that stores anything smaller than its nodes on the left side and anything larger than its nodes on its right side. As a number is added to it, that number moves along the correct side and becomes the leaf of the node closest to the root while following the rule of the tree. Here’s what the tree would look like if we inserted the numbers in a set in this order: 5, 2, 6, 15, 9, 10, 20

5

/ \

2 6

\

15

/ \

9 20

\

10

Now what would it look like in this order: 10, 6, 9, 2, 20, 5, 15

1. (2 points) Suppose we had a tree with nodes that kept a record of its depth from the root as its value. Design a procedure using filter, enumerate-tree, accumulate, and/or map that finds the depth of the leaf farthest from the root. You can also use recursion but it will take you longer to write. Hint: max might be useful depending on your implementation.

0

/ \

1 1

/

2

/ \

3 3

1. (3 points) Define a procedure, depth-of that searches a tree for a number and returns the depth of the node with that number. Root is at depth 0 and assume that there are no duplicates in the tree. Return #f if the number is not in the tree.

1 depth = 0

/ | \

3 7 5 depth = 1

/ | \

2 4 8 depth = 2

Let the variable **sample** be the tree above.

>(depth-of 3 sample)

1

>(depth-of 8 sample)

2