ECE 1125 Data Structures and Algorithms – Fall 2015

Sample Exam 1

Note the real exam will have 4-6 problems of this type.

1. Consider the following definition of a linked-list node.

```
struct list_node {
     double val; /* value of each element */
     struct list_node *next;
     struct list_node *prev;
};
struct list_node *head;
```

(a) Is this a singly-linked list or a doubly-linked list?

(b) Write a piece of code that iterates through the list starting at head and prints every element that is a multiple of 3.

(c) Write a piece of code that iterates through the list in reverse order starting from the last element, prints it, and then prints every other element as it proceeds backwards.

(d) Write a piece of code that iterates through the list and finds the mean value of all of the elements.

(e) What is the time complexity of your code in part b in Θ(..) notation? What is the spatial complexity?

2. Consider the following (unreasonable) definition of an array A, and assume that its elements have been filled in:

```
#define N (100000000000000)
double A[N] = {...};
```

(a) Write a piece of code to compute the mean of the elements of A. (You can ignore roundoff error, and assume that an int on this machine is capable of holding a number as big as N.)

(b) Assume that the machine your code runs on has a clock cycle of 1 ns, and that loading, comparing, incrementing and storing an integer loop counter takes 3 clock cycles, and adding two doubles and storing the result takes 5 clock cycles, and that any fixed setup code takes a negligible amount of time. How long does it take for your code to execute?

3. Consider the following 15 numbers to be inserted into a binary search tree: [67, 25, 59, 3, 48, 92, 75, 15, 8, 30, 69, 33, 17, 24, 77].

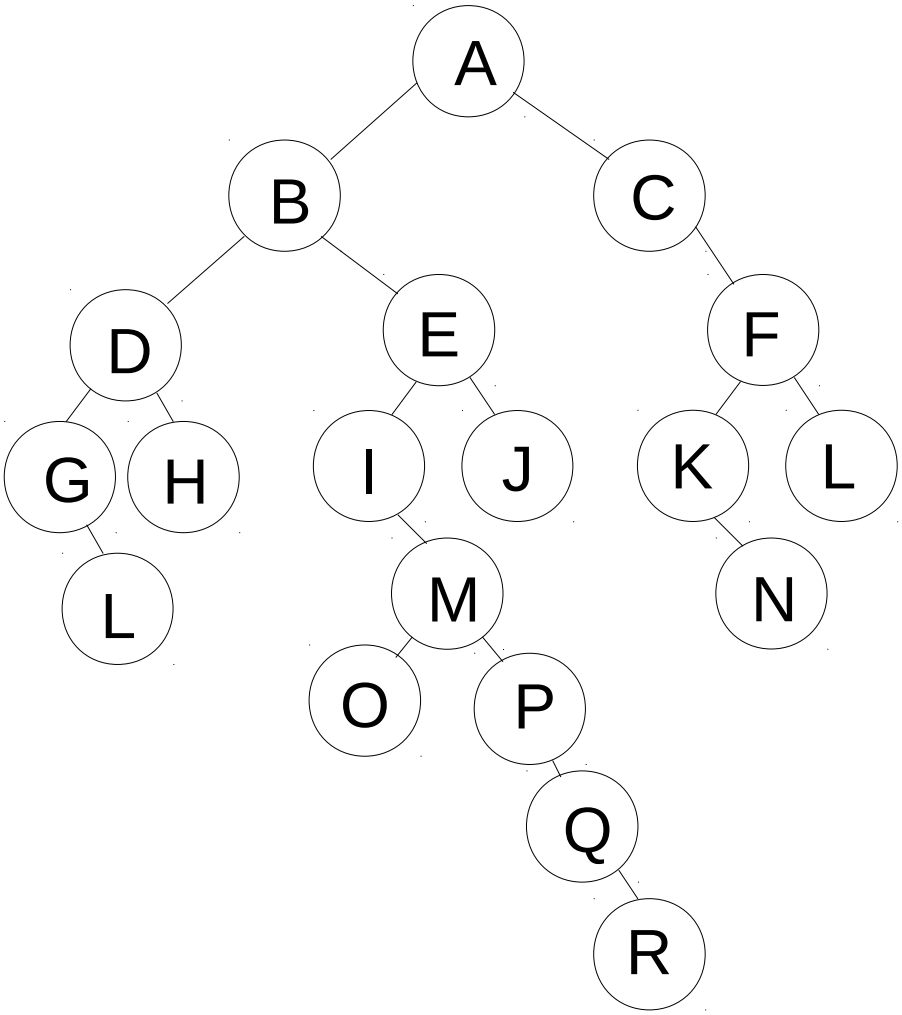
(a) Draw three valid, non-maximum height binary search trees containing these numbers.

(b) Draw the two worst-case (maximum height) binary search trees containing these numbers.

(c) What is the height of a full and complete (i.e. minimum height) binary search tree containing these numbers?

(d) Draw the minium height binary search tree containing these numbers.

10. Consider the following binary search tree:



(a) What is the element at the root?

(b) What is the height of the tree?

(c) What are the largest and smallest elements in the tree?

(f) If the elements were arranged to minimize the height of the binary search tree, what would that minimum height be?

(d) Draw the minium height binary search tree containing these numbers.

5. Consider a function that computes the sum c[k] = a[k] + b[k], of two input arrays a[n] and b[n].

```
int vector_sum(const double *a, const double *b, unsigned int n,
               double **p_c);
```

(a) Write C code that implements the function `vector_sum`, dynamically allocating the correct amount of storage for c (and checking the result of the allocation), and copying the result out. The function should and return 0 if successful and -1 if not successful.

(b) What is the time complexity of your code in part (a) in big-O notation? What is the spatial complexity?

6. Consider the following operations done on a stack: push 22, push 47, push 81, push 11, pop, push 32, push 3, pop, pop, pop, push 4.

(a) Draw the entire stack after every operation, clearly indicating the top of the stack with an arrow, and indicate the value returned by each call to pop.

7. Consider the following operations done on a queue: enq 22, enq 47, enq 81, enq 11, deq, enq 32, enq 3, deq, deq, deq, enq 4.

(a) Draw the entire queue after every operation, clearly indicating the head and tail of the queue, and indicate the value returned by each call to deq.

8. Consider a circular queue of integers holding a maximum of 6 integers, implemented in an array, initially empty, with a head and tail index.

(a) Show valid C declarations for the array, head and tail elements. Your declarations must explicitly set the initial values for head and tail.

(b) Consider the following operations to be done on this fixed-size circular buffer:

deq, enq 13, enq 36, enq 68, deq, deq, enq 20, enq 28, enq 69, enq 32, enq 81 enq 71

Show the value of the head and tail indices, the full contents of the array after the call to each operation, and show the returned value (including errors if applicable).

9. Recall that a full binary tree of depth k has 2^k – 1 nodes.

(a) What is the depth k needed to hold N=10^14 elements in a full binary search tree?

(b) How many comparisons are needed to finish searching a tree containing N=10^14 elements for an element that is not in the tree?

Suppose that comparing two elements on a machine took 10ns and ignore other time costs. Then,

(c) How long would it take to finish searching a binary search tree containing N=10^14 elements for an element that is not in the tree?

(d) How long would it take to search an unordered array containing N=10^14 elements for an element that is not in the array?