BRONX COMMUNITY COLLEGE
of the City University of New York
DEPARTMENT OF MATHEMATICS AND COMPUTER SCIENCE

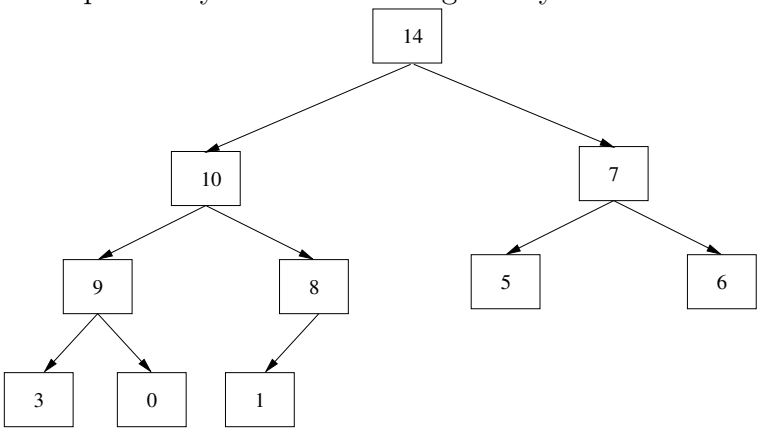
CSI33 **Sample Final Exam**

NAME_____

Directions: Solve problems 1 through 5 of Part I and choose 5 of the 6 problems in Part II. Use the separate answer booklet to write your answers. You may refer to the textbook, handouts, or notes taken in class, but no calculators, computers or conversation will be permitted.
Part I: Solve problems 1 through 5.

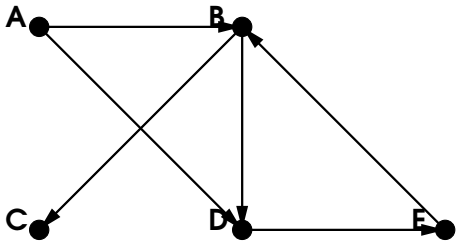
1. (10) Indicate whether the following statements are TRUE or FALSE:
   i. Binary search is never as slow as linear search.
   ii. A heap structure is not a natural choice to implement a priority queue.
   iii. Linked lists are useful because insertion and deletion time is faster than it is for an array.
   iv. An item in a binary search tree can always be found in $\Theta(\log(n))$ time.
   v. A recursive function does not always need a base case.
   vi. A Python list is a homogeneous container.
   vii. In languages like Python, the call stack is used to keep track of nested function calls.
   viii. In a C++ implementation of an Abstract Data Type, a method can be private.
   ix. The stage of the C++ build process in which `#include` directives are performed is called the preprocessor stage.
   x. For a binary search tree to satisfy the AVL property, both subtrees of any given node must have exactly the same height.

2. A heap initially has the following binary tree structure:



A. (5) Draw the array-based representation of this binary tree.
B. (5) Show each change made to this tree in the process of adding the value 12, by drawing the tree as it appears after each stage of the insertion.

3. (10) Represent this directed, unweighted graph as a Python List two ways: as an adjacency matrix and as an adjacency list. Which representation is most efficient for a graph which has very few edges between nodes?



4. (10) The integers 11,2,7,4,5,3 and 8 are inserted in that order into a container object. Give the order in which these values are retrieved, if the container is
A. a stack,
B. a queue,
C. a priority queue.

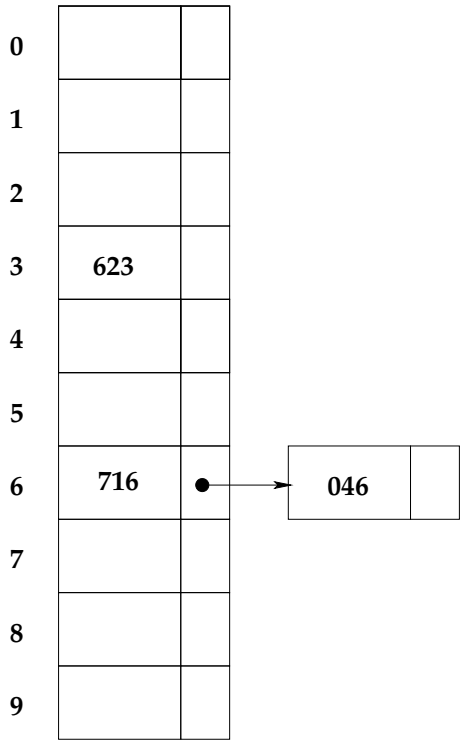
5. a. (5) State the AVL property for binary search trees.

   b. (5) The sequence of values from the previous problem, in that order, is inserted into an empty AVL tree. (After any insertion, rebalance the tree by performing the appropriate rotation if the tree is unbalanced according to the AVL property.) Draw the tree as it appears after every insertion.

Part II: Solve 5 of the 6 problems (6 through 11) in this part.

6. (10) Below is an array with 10 positions which is being used as a hash table to keep customer information. The key to each record is the last 3 digits of the customer's id number. The hash function f is given by f(k) = k%10, that is, the index slot in the array for key k is the remainder of dividing k by 10. The method of collision resolution will be separate chaining. Draw boxes and arrows in the following diagram to give the state of the hash table after the following keys are inserted in this order: (The first three insertions have already been illustrated as a hint.)

   `716, 623, 046, 770, 879, 862, 764, 445, 787, 357, 154, 944, 843, 261, 238.`



7. (10) Give box-and-arrow diagrams that illustrate the state of the C++ memory model immediately after each of these statements, including the last. What problem will occur if the statement with the `delete` operation is removed from the program?

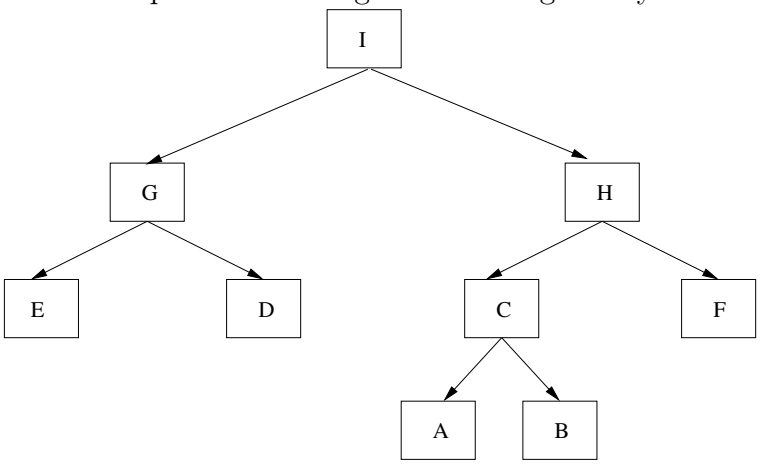
```
{
  int *a, *b, c;
  a = &c;
  b = new int;
  c = 17;
  *b = *a + 5;
  delete b;
}
```

8. (10) The following C++ function finds the sum of squares of integers in an array:

```
int sum_of_squares(int a[], int size)
{
  int i;
  int sum = 0;
  for (i = 0; i<size:  i++)
    sum += a[i]*a[i];
  return sum;
}
```

Rewrite this function as a template function which could be called as `sum_of_squares<int>`, `sum_of_squares<double>`, or as an instance for any other class which overloads the `+` and `*` operators.

9. (10) Write the output of preorder, inorder, and postorder traversal when each of them is performed using the following binary tree:



10. (10) Name three methods (member functions) which must be specially coded for any C++ class which allocates and manages dynamic memory. Explain in detail what each method does in terms of allocating and deallocating memory, as well as copying data in existing memory. What memory problems can occur if the default versions of these C++ methods are used instead?

11. Sorting

    a. (5) Perform the Selection Sort algorithm to sort the following list of integers, explaining how each value reaches its final position:

    ```
    [3,7,2,5,1,6,4,8]
    ```

    b. (5) Give the Θ-running time of Selection Sort using n, the number of items being sorted. Then give a clear explanation of why it takes that long to sort n items.