



MIDTERM SAMPLE TEST

CSE 2011 – Fundamentals of Data Structures

Summer 2013

1- Growth Rate

Order the following running time θ bounds by asymptotic growth rate in non-descending order. Indicate which functions grow at the same rate, if any. Explanations are **NOT** required.

$N \log(10N)$, $N!$, $10 / \log N$, 100^N , $N \log^2 N$, $5 / N$

2- Running Time Calculations

Describe the worst case running time of the following java style pseudocode functions in Big-Oh notation in terms of **the variable n**. No proof/description is required.

I.

```
public static int myFunction1 (int n)
{
    int x = 0;
    for (int i = 0; i < n * n; i = i + 2)
        x++
    return x;
}
```

Runtime:

II.

```
public static int myFunction2 (int n)
{
    int x = 0;
    for (int i = 1; i < n * n; i = i * 2)
        x++
    return x;
}
```

Runtime:

III.

```
public static int myFunction3 (int n)
{
    int x = 0;
    for ( int i = 1; i <= n * n * n; i++ )
        for ( int j = 1; j <= n; j = j * 2 )
            x++
    return x;
}
```

Runtime:

3- Solving a Recurrence Relation

Solve the following recurrence by finding a Big-Oh bound for $T(N)$, given that $T(1) = 1$. The calculation **must** be shown for full marks.

$$T(N) = T(N/2) + 10$$

4- Recursion

Write a recursive method that finds the minimum value in an array *A* of int values without using any loops.

5- Trees

Part I.

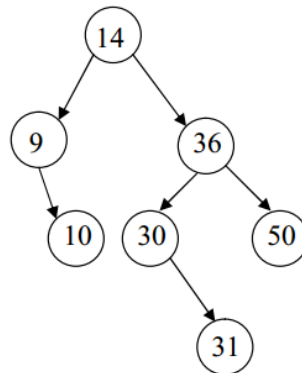
What is the height and the depth of the tree shown below:

Height = _____

Depth = _____

Part II.

Give the Preorder traversals of the tree shown below:



6- Stacks and Queues

Describe the contents of stack *s* after the method `convert` executes. That is, describe the contents in a general manner based on what is in *s* before the code executes.

```
public void convert(Stack<Object> s) {
    ArrayList<Object> list = new ArrayList<Object>();
    while (s.size() > 0) {
        list.add(s.pop());
    }
    for (Object o : list) {
        s.push(o);
    }
}
```

What happens if a queue is used instead of a stack in the code above, e.g.,

```
public void convert(Queue<Object> q) {
    ArrayList<Object> list = new ArrayList<Object>();
    while (q.size() > 0) {
        list.add(q.dequeue());
    }
    for (Object o : list) {
        q.enqueue(o);
    }
}
```