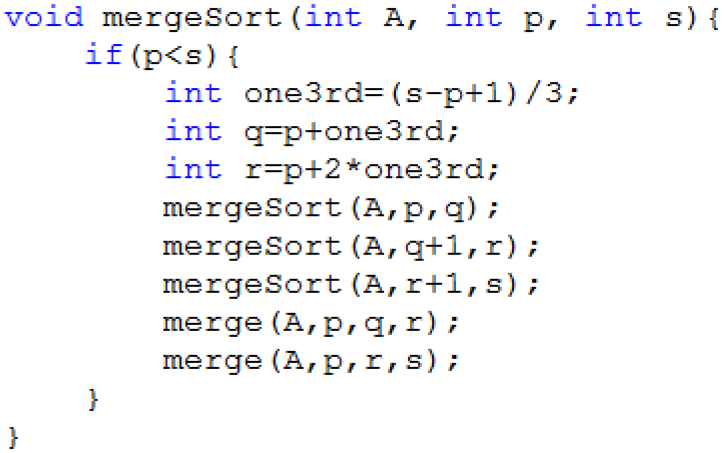
Roll Number: _______________ Section: _____________

CS 302 DESIGN AND ANALYSIS OF ALGORITHMS
MID 1 FALL 2016

Date: 19th Sept, 2016. **Total Marks: 40** **Time: 60 min**

NOTE: Answer in the provided space. You can get extra sheets only for rough work. They will NOT be collected.

QUESTION 1 [4*5 = 20 marks] Following is a modified version of merge sort. It uses the same merge function as studied in class.



```
void mergeSort(int A, int p, int s){
    if(p<s){
        int one3rd=(s-p+1)/3;
        int q=p+one3rd;
        int r=p+2*one3rd;
        mergeSort(A,p,q);
        mergeSort(A,q+1,r);
        mergeSort(A,r+1,s);
        merge(A,p,q,r);
        merge(A,p,r,s);
    }
}
```

a) Draw a tree to show the division of the problem by this algorithm. Show the values of parameters p and s at each call. The first call in the main is: mergeSort(A, 0, 26)

b) Write down a recurrence relation T(n), with base case T(1) to describe the running time of this modified version of merge sort.

c) Solve the recurrence in (b) to get an asymptotic upper bound (big-Oh bound).

d) Based on the result in (c), if you were given a choice to use either regular merge sort or this version, which one will you choose and why? An answer without a reason will not get any points.

QUESTION 2 [20 marks] Write an algorithm to take union of two sets of numbers, i.e. two arrays A, B of integers. Since these are sets, the numbers are distinct in each array, however, they could contain common elements. Make your algorithm as asymptotically efficient as possible.

Algorithm (in English):

C++ code