Due Date:11-August-2024
20% penalty for 1 day late
40% penalty for 2 days late
Submission not allowed afterwards

CS2009: Design and Analysis of Algorithms (Fall 2024)
Assignment 1
Total Marks: 100

1. Show the steps merge sort uses to sort the following list of integers in the descending order (from the highest to the lowest / biggest to the smallest): [7.5 points]

9, 19, 14, 29, 13, 4, 23, 10, 37

Consider the following variation on Merge Sort, that instead of dividing input in half at each step of Merge Sort, you divide into **four** part, sort each part, and finally combine all of them using a **four-way merge** subroutine. What is the overall asymptotic running time of this algorithm? [7.5 points].

2. Prove $3n^3 + 5n^2 + 25n = Ω(n^3)$. Determine the values of constant c and n_0. [5 Points]

3. Prove $5 \log_2 \log_2 n + 4\log^2 n = O(?)$. Determine the values of constant c and n_0. [5 Points].

4. By using the frequency count method, find the time complexity of the following.[10 Points]

```
int p = 0;
for (i = n / 2; i > 1; i /= 6) {
    for (j = 2; j <= n; j *= 4) {
        for (k = 0; k <= j; k *= 3) {
            p = p + n / 2;
        }
    }
}
```

```
int p = 0;
for (i = n; i > 1; i = i / 2) {
    for (j = 2; j <= n; j += 1) {
        p = p + n;
        if (p > (n ^ i)) break;
    }
}
```

5. List the six functions below in non-decreasing asymptotic order of growth.Please give mathematical justification for the ordering you specify[5 Points]

$\lg \lg n^2$	$\log^2 n$	n^{-2}	$\lg(2^{\lg(n^2)})$	$n^{(1/2)}$	n^2

6. Watch the video lecture on Big O, Big Ω and Big Θ notation from http://www.youtube.com/watch?v=6Ol2JbwoJp0. Write the summary of the lecture in your words. [10 Points].

7. For each of the following questions, indicate whether it is T (True) or F (False) and justify using some examples e.g. assuming a function? [20 Points]

- For all positive $n = \omega(\sqrt{n})$
- For all positive $f(n)$; if $f(n)$=O(g(n)), then g(n)=Ω($f(n)$)
- Disprove that f(n) = O($f(n)^2$) by showing a counterexample, show a function $f(n)$ for which this does not hold.
- Prove that the running time of an algorithm is Θ($f(n)$) if and only if its worst-case running time is O($f(n)$) and its best-case running time is Ω($f(n)$).

Due Date:11-August-2024
20% penalty for 1 day late
40% penalty for 2 days late
Submission not allowed afterwards

CS2009: Design and Analysis of Algorithms (Fall 2024)
Assignment 1
Total Marks: 100

8.You have a computer and n processes with the processing time t_1 ,t_2 ,…t_n . You have to pick the order in which to run the processes. Let p_j denote the jth process you run. Then, the completion time C_i for the process p_j is defined as (the sum of times for all the processes up till this one ends).

$$C_i = \sum_{j=1}^{i} t_{pj}$$

Design 2 algorithms that give its time complexity **in polynomial time and logarithmic time respectively** that minimizes the average completion time $\frac{1}{n}\sum_{i=1}^{n} C_i$