Assignment - 1

Total Points: 100

Instructions

Submission

You can either write your answer using Word/LaTex or take a picture/scan of your hand-written(neat and tidy) solution, then put your solution in one PDF file. Submit your PDF online using Canvas. All submission must be posted before deadline.

Problem 1 (Max Points:10)

Use mathematical induction to show that the solution to the recurrence:

$$T(n) = \begin{cases} 2 & \text{if } n = 2 \\ 2T(\frac{n}{2}) + n & n = 2^k, k > 1. \end{cases}$$

is $T(n) = n \lg n$.

Problem 2 (Max Points: 25)

The following statements are all true:

$$89n^2 + 20n + 5 = O(89n^2 + 20n + 5)$$
$$89n^2 + 20n + 5 = O(89n^2)$$
$$89n^2 + 20n + 5 = O(n^{10})$$
$$89n^2 + 20n + 5 = O(n^2)$$

However, the last one is the simplest, cleanest, and tightest of them, and we will refer to it as the minimal big-O form.

Part A (Max Points: 10)

Write the following expression in minimal big-O notation.
(1) $n^3 + 3^n$.
(2) $3nlog(5n)$.
(3) $100 * 2^n + 3^n$
(4) $80nlogn + 5n^3 + \sqrt{n}$
(5) $1^3 + 2^3 + ... + n^3$

Part B (Max Points: 15)

Give a minimal upper bound on $f(n) = 1+2+4+...+2^n$. Justify your answers using mathematical induction.

Problem 3 (Max Points:10)

Recall the definition of logarithm base two: saying $p = \log_2 m$ is the same as saying $m = 2^p$. In this class, we will typically write lg to mean $\log_2$.
(1) How many bits are needed to write down a positive integer n? Give your answer in big-O notation, as a function of n.
(2) How many times does the following piece of code print ?hello?? Assume n is an integer, and that division rounds down to the nearest integer. Give your answer in big-O form, as a function of n.

```
while n > 1:
    print "hello"
    n := n/2
```

Problem 4 (Max Points:15)

Use a recursion tree to determine a good asymptotic upper bound on the following recurrence:

$$T(n) = \begin{cases} 1 & \text{if } n = 1 \\ 4T(\frac{n}{2}) + n & n = 2^k, k \geq 1. \end{cases}$$

Use the substitution method to verify your answer.

Problem 5 (Max Points:10)

We can express insertion sort as a recursive procedure as follows. In order to sort A[1..n], we recursively sort A[1...n-1] and then insert A[n] into the sorted array A[1...n-1].Write a recurrence for the running time of this recursive version of insertion sort and also solve it.

Problem 6 (Max Points: 30)

For each of the following recurrences, give an expression for the run time T(n) if the recurrence can be solved with the Master Theorem. Otherwise, indicate that the Master Theorem does not apply. Justify your answer with reasoning.

(a) $T(n) = 2T(n/2) + n^4$

(b) $T(n) = T(7n/10) + n$

(c) $T(n) = 16T(n/4) + n^2$

(d) $T(n) = 2T(n/4) + \sqrt{n}$

(e) $T(n) = \sqrt{2}T(n/2) + \log n$

(f) $T(n) = 64T(n/8) - n^2 \log(n)$

(g) $T(n) = 2T(n/4) + n^{0.51}$

(h) $T(n) = 16T(n/4) + n!$

(i) $T(n) = 0.5T(n/2) + 1/n$

(j) $T(n) = 2^n T(n/2) + n^n$