

CSE780: Analysis of Algorithms – Homework 1

Due: Oct. 11, 2011 (Tuesday) in class

(1) [Asymptotic notation] (15 pts): Prove or disprove the following statements. Note that the functions that we are interested in in this course should be positive function when n is large enough. So be careful when you construct counter-examples.

- (a). $f(n) = \Theta(g(n))$ implies $h(f(n)) = O(h(g(n)))$ if $h(n)$ is an increasing function (i.e., $h(n_1) > h(n_2)$) for $\forall n_1 > n_2$;
- (b). $f(n) \neq o(g(n))$ and $g(n) \neq o(f(n))$ implies $f(n) = \Theta(g(n))$;
- (c). $f(n) + g(n) = \Theta(\max\{f(n), g(n)\})$

(2) [Analyze algorithms] (25 pts): Provide the asymptotic time complexity for each algorithm below. Describe what the output of function $G(n)$ in (c) is.

```
FUNCTION T(n)
  x = 0; i = 1;
  while (i ≤ n) do
    j = 1;
    while j ≤ i do
      x = x + i - j;
      j = j * 2;
    end while
    i = i + 1;
  end while
  return (x)
```

(a)

```
FUNCTION F(n)
  x = 0;
  for i = 1 to n do
    j = 0;
    while (j ≤ n) do
      j = j + i;
    end while
  end for
  return (x)
```

(b)

```
FUNCTION G(A, n)
  /* A is an array of size n. */
  if (n = 1) then return A[1];
  else
    for i = 1 to ⌊n/2⌋ do
      B[i] = max{A[2i], A[2i - 1]};
    end for
    x = G(B, ⌊n/2⌋);
    x = max{x, A[n]};
  return (x)
```

(c)

(3) [Growth of functions] (12 pts): Order the following functions by order of growth. Make your answer as tight as possible. For example, if you are given n , $2n$, and n^2 , you should give $n = \Theta(2n) = O(n^2)$. (ln is \log_e where e is the natural number, and \log stands for \log_2 .)

$\ln n, \log(n^3), \log(n!), n \log n, (\log n)!, 4^{\log n}$.

(4) [Recurrences] (56 pts): Give asymptotic bounds for $T(n)$ in each of the following recurrences. Assume that $T(n)$ is constant for sufficiently small n . Make your bounds as tight as possible, and justify your answer.

- (a). $T(n) = T(n - 5) + T(2) + n^2$
- (b). $T(n) = T(\alpha n) + T((1 - \alpha)n) + \beta n$, where $0 < \alpha < 1$ and $\beta > 0$ are constants
- (c). $T(n) = T(n/3) + T(2n/3) + \log n$
- (d). $T(n) = T(n/3) + T(n/2) + n$

(e). $T(n) = \max\{2T(n/2) + n, T(n/5) + T(4n/5) + n\}$

(f). $T(n) = 3T(n/3 + 10) + n/2$

(g). $T(n) = 2T(n/2) + n/\log n$

(h). $T(n) = T(n - 1) + 1/n$