

알고리즘 H/W #1

Due: 10/7(수) 3:00pm

제출: 302-313-2(최적화/금융공학 연구실)

조교: 이상엽(880-1851)

1. (각 5점) Asymptotic running time을 구하고 과정을 밝히시오. Master Thm을 이용할 수 있는 곳에는 이용해도 좋음. Master Thm을 이용하지 않는 경우에는 $O()$ 와 $\Omega()$ 모두 구하시오.

1.1 $T(n) = 4T(n/4) + b$ b : a constant

1.2 $T(n) = 3T(n/3) + n \log n$

1.3 $T(n) = 5T(n/5) + 3n$

1.4 $T(n) = T(n/4) + T(3n/4) + O(n)$

1.5 $T(n) = T(n/4) + T(3n/5) + O(n)$

1.6 $T(n) = T(2n/5) + T(4n/5) + O(n)$

1.7 $T(n) = 3T(n/3 + 9) + n$

2. (20점) 아래 알고리즘의 수행시간을 구하라.

```
sample(A[ ], p, r)
{
    if (r-p≤1) return 1;
    sum = 0;
    for i = p to r
        sum = sum + A[i];
    q = ⌊(r - p + 1)/3⌋;
    tmp = sum + sample(A, p, p+q-1) + sample(A, p+2q, r);
    return tmp;
}
```

3. (25점) 아래 알고리즘 test(n)의 수행 시간의 upper bound를 asymptotic notation으로 나타내어라. n은 양의 정수다. 문제를 풀면서 절대 $\frac{n}{3} + 5$ 과 $\frac{2n}{3} + 7$ 를 $\frac{n}{3}$ 으로 근사 취급해서는 안된다.

```
int test(n)
{
    if (n ≤ 50) then return n ;
    else return (test( $\frac{n}{3} + 5$ ) + test( $\frac{2n}{3} + 7$ )) ;
}
```

4. (20점) Quicksort의 worst-case running time은 $O(n^2)$ 이다. Quicksort의 worst-case running time을 $O(n \log n)$ 으로 만들려면 어떻게 하면 되는가? 이유도 밝혀야 한다. 여러분이 배운 내용들을 조합하면 된다.

5. (20점) Worst-case linear-time selection 알고리즘에서 5개로 나누는 대신 3개로 나누면 수행시간은 어떻게 되는가? 결과를 증명하라.