Student Name: ______________________________ J#: ______________

CSC 323 Algorithm Design and Analysis, Spring 2017
Instructor: Dr. Natarajan Meghanathan

Quiz 1 (January 31, 2017) Max. Points: 30 Max. Time: 20 min.

1) (9 pts) Derive the asymptotic relationship between the two functions: $n^2\log(n)$ and $n\log(n^{100})$

$f(n) = n^2 \log n$; $g(n) = n \times \log(n^{100}) = 100n \times \log n$

$$\lim_{n\to\infty} \frac{f(n)}{g(n)} = \lim_{n\to\infty} \frac{n^2 \log n}{100 n \log n} = \lim_{n\to\infty} \frac{n}{100} = \infty.$$

as $n \to \infty$, $f(n)$ grows faster than $g(n)$

Hence, $g(n) = O(f(n))$

$$\boxed{n \log(n^{100}) = O(n^2 \log(n))}$$

2) (9 pts) Let $f(n) = 5n^3 + 6n + 2$. Find a function g(n) such that f(n) = O(g(n)) and f(n) ≠ Θ(g(n)). Show that your choice for g(n) is correct using the Limits approach.

We need to choose a function g(n) whose degree is strictly greater than 3, the degree of f(n).

Let $g(n) = n^4$.

$$\lim_{n\to\infty} \frac{f(n)}{g(n)} = \lim_{n\to\infty} \frac{5n^3 + 6n + 2}{n^4} = \lim_{n\to\infty} \frac{5}{n} + \frac{6}{n^3} + \frac{2}{n^4} = \underline{\underline{0}}$$

Hence, $g(n)$ grows much faster than $f(n)$ as $n \to \infty$.

$f(n) = O(g(n))$

$\underline{5n^3 + 6n + 2 = O(n^4)}$

Student Name: ______________________________ J#: ________________

3) (12 pts) Consider an array of n elements that <u>needs to be sorted in the descending order</u>. Given here is the code snippet of an algorithm for this purpose. It uses a ***Swap function*** to exchange arguments. If the input array to this code is A[i] = i, for i = 0, 1, 2, ..., n-1, <u>Show that</u> the exact number of times the Swap function is called is **n(n-1)/2**.

```
for (int j = 0; j ≤ n-2; j++)
    for (int k = 0; k ≤ n - j - 2; k++)
        if (A[k] < A[k+1] )
            Swap(A[k], A[k+1])
```

Given an array $A = 0, 1, 2, \ldots, n-1$, we need to sort it in descending order $n-1, n-2, \ldots, 2, 1, 0$.

$0 \quad 1 \quad 2 \quad 3 \quad 4, \ldots, n-2, n-1$

$j=0$ ($k = 0 \ldots n-2$): $1 \quad 2 \quad 3 \quad 4 \quad 5, \ldots\ldots n-1, \boxed{0}$

$j=1$ ($k = 0, n-3$): $2 \quad 3 \quad 4 \quad 5 \quad 6 \ldots\ldots n-1, \boxed{1}, 0$

$j=2$: $3 \quad 4 \quad 5 \quad 6 \quad 7 \ldots\ldots \boxed{2} \quad 1, \quad 0.$

$\vdots$

$j=n-2$: $n-1 \quad \boxed{n-2} \ldots\ldots 3, 2, 1, 0$

We would need to execute the <u>swap</u> function for each value of j and k.

times the swap function is executed is $\sum_{j=0}^{n-2} \sum_{k=0}^{n-j-2}$ (1)

$$= \sum_{j=0}^{n-2} \left[(n-j-2) - (0) + 1\right]$$

$$= \sum_{j=0}^{n-2} \left[n-j-1\right]$$

$$= (n-0-1) + (n-1-1) + (n-2-1) + \ldots + (n-(n-2)-1)$$

$$= (n-1) + (n-2) + (n-3) + \ldots + (1)$$

$$= \sum_{i=1}^{n-1} i = \frac{(n-1)(n-1+1)}{2} = \frac{n(n-1)}{2} = \underline{\Theta(n^2)}$$