Efficiency and Asymptotic Analysis (20 points)

1. [10 points] Rank the following functions of n by order of growth (starting with the slowest growing) using positive integers (1, 2, ...). Functions with the same order of growth should be ranked equal.

$\log n^3$	n	$n^2 \log n$	$2^{\lg n^2}$	$\log \sqrt{n}$	$n + \log n^4$	$2^{\log 16}$	n^{-1}	16	$n^{\log 4}$

Solution:

$\log n^3$	n	$n^2 \log n$	$4^{\lg n}$	$\log \sqrt{n}$	$n + \log n^4$	$2^{\lg 16}$	n^{-1}	16	$n^{\lg 4}$
3	4	6	5	3	4	2	1	2	5

2. [1 point] Express the function $n^3/1000 - 100n^2 - 100n + 3$ in terms of Θ-notation.

Solution: $n^3/1000 - 100n^2 - 100n + 3 \in \Theta\left(n^3\right)$

3. Are the following statements true or false. Briefly explain your answer.

   (a) [1 point] $n - 2\log n = \Omega(n)$

   Solution: True. $\log n$ is asymptotically smaller than n. Therefore, the left side is in fact $\Theta(n)$, and the equality is correct.

   (b) [1 point] $n^2 \log n = \Theta(n^2)$

   Solution: Wrong. $\log n$ is asymptotically larger than a constant. Therefore, the inequality is not correct.

   (c) [1 point] If $f(n) = O(g(n))$ and $g(n) = O(h(n))$, then $f(n) = O(h(n))$.

   Solution: True. $f(n)$ is asymptotically not larger than $g(n)$. It is also true for $g(n)$ and $h(n)$. Therefore, we can say the same thing for $f(n)$ and $h(n)$.

   (d) [1 point] $2n^2 + 4n - 17 = O(n^3)$.

   Solution: True. The left side can be simplified to n^2 and, therefore, the left side will be clearly smaller than the right side asymptotically.

4. Assume that n is a positive integer. For each of the following algorithm segments, compute the actual number of additions, subtractions, multiplications, and divisions that must be performed when the algorithm segment is executed (ignore the operations directly performed by the **for** statement). Then compute the asymptotic order of the sum.

(a) [2 points]

```
for k = 2 to n
    for j = 1 to 3n
        x = a[k] - b[j]
```

Solution: For each iteration of the inner loop there is one subtraction. There are $3n$ iterations of the inner loop for each iteration of the outer loop, and there are $n-2+1 = n-1$ iterations of the outer loop. Therefore, the number of iterations of the inner loop is $3n(n-1) = 3n^2 - 3n$. It follows that the total number of elementary operations that must be performed when the algorithm is executed is $3n^2 - 3n$.
$3n^2 - 3n$ is $\Theta\left(n^2\right)$, and so it has order n^2.

(b) [3 points]

```
r = 0
for i = 1 to n - 1
    p = 1
    q = 1
    for j = i + 1 to n
        p = p · c[j]
        q = q · (c[j])^2
    r = p + q
```

Solution: There are three multiplications for each iteration of the inner loop, and there is one additional addition for each iteration of the outer loop. The number of iterations of the inner loop can be deduced from the following table, which shows the values of i and j for which the inner loop is executed.

i	1				2				$\cdots$	$n-2$		$n-1$
j	2	3	$\cdots$	n	3	4	$\cdots$	n	$\cdots$	$n-1$	n	n

Hence, by Theorem 5.2.2, the total number of iterations of the inner loop is

$$(n-1) + (n-2) + \cdots + 2 + 1 = \frac{n(n-1)}{2}$$

Because three multiplications are performed for each iteration of the inner loop, the number of operations that are performed when the inner loop is executed is

$$3 \cdot \frac{n(n-1)}{2} = \frac{3}{2}\left(n^2 - n\right) = \frac{3}{2}n^2 - \frac{3}{2}n$$

Now an additional operation is performed each time the outer loop is executed, and because the outer loop is executed n times, this gives an additional n operations. Therefore, the total number of operations is

$$\left(\frac{3}{2}n^2 - \frac{3}{2}n\right) + n = \frac{3}{2}n^2 - \frac{1}{2}n.$$

$\frac{3}{2}n^2 - \frac{1}{2}n$ is $\Theta\left(n^2\right)$, and so it has order n^2.

Recursion (6 points)

5. [3 points] Using iteration to solve the following recursion:

$$T(n) = 2T(n/2) + n.$$

Solution:

$$\begin{aligned}
T(n) &= 2T(n/2) + n \\
&= 2(2T(n/4) + n/2) + n \\
&= 4T(n/4) + n + n \\
&= 4(2T(n/8) + n/4) + n + n \\
&= 8T(n/8) + n + n + n \\
&\vdots \\
&= nT(1) + n + n + \cdots + n \\
&\overset{\text{asymptotically}}{=} n + n\log n \\
&\overset{\text{asymptotically}}{=} n\log n
\end{aligned}$$

Therefore, $T(n) = \Theta(n \log n)$.

6. [3 points] Using iteration to solve the following recursion:

$$T(n) = 3T(n/2) + 1.$$

Solution:

$$
\begin{aligned}
T(n) &= 3T(n/2) + 1 \\
&= 3(3T(n/4) + 1) + 1 \\
&= 9T(n/4) + 3 + n \\
&= 9(3T(n/8) + 1) + 3 + 1 \\
&= 27T(n/8) + 9 + 3 + 1 \\
&= 3^3 T\left(n/\left(2^3\right)\right) + 3^2 + 3^1 + 3^0 \\
&\vdots \\
&= 3^{\lg n} T(1) + \sum_{i=0}^{\lg n - 1} 3^i \\
&= 3^{\lg n} T(1) + \frac{1 - 3^{\lg n}}{1 - 3} \\
&= 3^{\lg n} T(1) + 3^{\lg n}/2 - 1/2 \\
&\overset{\text{asymptotically}}{=} 3^{\lg n} \\
&= n^{\lg 3}
\end{aligned}
$$

Therefore, $T(n) = \Theta(n^{\lg 3})$.