Algorithm Analysis and Design

Exercises, Ch 2

(Mathematical Analysis)

Solutions are based on book's solution manual

1. For each of the following algorithms, indicate (i) a natural size metric for its inputs, (ii) its basic operation, and (iii) whether the basic operation count can be different for inputs of the same size:
   a. computing the sum of n numbers
   b. computing n!
   c. finding the largest element in a list of n numbers
   d. Euclid's algorithm
   e. pen-and-pencil algorithm for multiplying two n-digit decimal integers

2.
   a. Consider the definition-based algorithm for adding two $n \times n$ matrices. What is its basic operation? How many times is it performed as a function of the matrix order n? As a function of the total number of elements in the input matrices?
   b. Answer the same questions for the definition-based algorithm for matrix multiplication.

3. For each of the following functions, indicate how much the function's value will change if its argument is increased fourfold.
   a. $\log_2 n$ b. $\sqrt{n}$ c. n d. n^2 e. n^3 f. 2^n

4. For each of the following pairs of functions, indicate whether the first function of each of the following pairs has a lower, same, or higher order of growth (to within a constant multiple) than the second function.
   a. $n(n + 1)$ and $2000n^2$
   b. $100n^2$ and $0.01n^3$
   c. $log_2^2\ n$ and $\log_2 n^2$
   d. 2^{n-1} and 2^n
   e. $(n-1)!$ and $n!$

5. Use the informal definitions of O, Θ and Ω to determine whether the following assertions are true or false.
   a. $n(n + 1)/2 \in O(n^3)$
   b. $n(n + 1)/2 \in O(n^2)$
   c. $n(n + 1)/2 \in \Theta(n^3)$
   d. $n(n + 1)/2 \in \Omega(n)$

6. For each of the following functions, indicate the class $\Theta(g(n))$ the function belongs to. (Use the simplest $g(n)$ possible in your answers.) Prove your assertions.
   a. $(n^2 + 1)^{10}$
   b. $\sqrt{(10 n^2+7n +3)}$
   c. $2n \log(n + 2)^2 + (n + 2)^2 \log n^2$
   d. $2^{n+1} + 3^{n-1}$
   e. floor ($\log_2 n$)

7. Prove that the following are listed in increasing order of their order of growth.
   $\log n$, n, $n \log n$, n^2, n^3, 2^n

8. Prove that every polynomial of degree k, $p(n) = a_k n^k + a_{k-1} n^{k-1} + \ldots + a_0$ with $a_k > 0$, belongs to $\Theta(n^k)$.
   Answer

9. Compute the following sums.
   a. $1 + 3 + 5 + 7 + \cdots + 999$
   b. $2 + 4 + 8 + 16 + \cdots + 1024$
   c. $\sum_{i=3}^{n+1} 1$ **d.** $\sum_{i=3}^{n+1} i$ **e.** $\sum_{i=0}^{n-1} i(i + 1)$

10. Find the order of growth of the following sums. Use the Θ(g(n)) notation with the simplest function g(n) possible.
   a. $\sum_{i=0}^{n-1} (i^2+1)^2$ **b.** $\sum_{i=2}^{n-1} \lg i^2$

11. Consider the following algorithm.

```
ALGORITHM Mystery(n)
    //Input: A nonnegative integer n
```

```
S ← 0
for i ← 1 to n do
    S ← S + i * i
return S
```

a. What does this algorithm compute?
b. What is its basic operation?
c. How many times is the basic operation executed?
d. What is the efficiency class of this algorithm?
e. Suggest an improvement, or a better algorithm altogether, and indicate its efficiency class. If you cannot do it, try to prove that, in fact, it cannot be done.

12. Solve the following recurrence relations.
   a. $x(n) = x(n-1) + 5$ for $n > 1$, $x(1) = 0$
   b. $x(\mathrm{n}) = 3x(n-1)$ for $n > 1$, $x(1) = 4$
   c. $x(n) = x(n-1) + n$ for $n > 0$, $x(0) = 0$
   d. $x(n) = x(n/2) + n$ for $n > 1$, $x(1) = 1$ (solve for $n = 2^k$)

13. Consider the following recursive algorithm for computing the sum of the first n cubes: $\mathrm{S(n)} = 1^3 + 2^3 + \ldots + n^3$.

```
ALGORITHM S(n)
    //Input: A positive integer n
    //Output: The sum of the first n cubes
        if n = 1 return 1
        else return S(n − 1) + n * n * n
```

   a. Set up and solve a recurrence relation for the number of times the algorithm's basic operation is executed.
   b. How does this algorithm compare with the straightforward nonrecursive algorithm for computing this sum?