1.1：

**Substitution Method:**

Hypothesis: T (n) = O (n2);

Show: T (n) ≤ cn2 for some c>0;

Assume: T (n/3) ≤ c(n/3)2;

T(n) ≤ 2c(n/3)2 + n2;

= (2c/9)n2 + n2;

= (2c/9 + 1)n2;

To accomplish T (n) ≤ cn2, we want 2c/9 + 1 ≤ c;

Thus, 2c + 9 ≤ 9c => c ≥ 9/7

T(n) ≤ cn2 for c ≥ 9/7;

Therefore T(n) = O (n2).

1.2:

**Recursion Method:**

T (n) = 2 T (n/3) + n2;

T (n) n2 => n2

/ \

/ \

(n/3)2 (n/3)2 => 2/9 n2

/ \ / \

(n/9)2 (n/9)2 (n/9)2 (n/9)2 => (2/9)2 n2

… … … …

T(1) T(1) … T(1) T(1) => Θ ()

Total layers.

= T(1)s Total: O(n2)

2.(a):

T(n) = 2T(n / 4) + n ;

a = 2 > 0; b = 4 > 0; f(n) = n ;

= n1/2 ;

f(n) = n1.5 = Ω (); Ԑ = 1 > 0;

af(n/b) = 2 (n/4)1.5 = 2 \* 4 -1.5 n1.5 = 0.25 n1.5 < cn1.5 for c = 0.5 <1

Therefore: T (n) = Θ (n3/2)

2.(b):

T (n) = 2T(n/3) +

a = 2 > 0; b = 3 > 0; f(n) = = = n2.32;

= n0.63;

So f(n) = Ω (); for Ԑ = log25 – log32 = 1.69 > 0;

af(n/b) = 2() = 2 \* = 0.156

< cf(n) for c = 0.3 < 1;

Therefore: T (n) = Θ ();

3.

Exercise 2.2-1

Express the function n3/1000 - 100n2 - 100n + 3 in terms of Θ-notation.

T (n) = n3/1000 - 100n2 - 100n + 3 = Θ (n3);

To prove that, there is a c1, c2 that:

c1n3 ≤ n3/1000 – 100n2 – 100n + 3 ≤ c2n3;

c1 ≤ 1/1000 – 100/n – 100/n2 + 3/n3 ≤ c2;

When n is very large:

– 100/n – 100/n2 + 3/n3 = 0;

Therefore:

c1 ≤ 1/1000 ≤ c2;

We choose n0 = **106**(**in order to keep c1>0, n0 must be large enough**), thus;

c1 ≤ 10000003 / 1000 – 100 \* 10000002 – 100 \* 1000000 + 3 ≤ c2;

c1 ≤ 1015 – 1014 – 108 +3 ≤ c2;

We combine these two inequalities together:

c1 ≤ 1/1000 ≤ 1015 – 1014 – 108 +3 ≤ c2;

So, when we choose **n0 = 106; c1 = 10-5; and c2 = 1015:**

c1n3 ≤ n3/1000 – 100n2 – 100n + 3 ≤ c2n3 is always true;

Therefore, by definition: T (n) = n3/1000 - 100n2 - 100n + 3 = Θ (n3);

4.

**Substitution Method:**

Hypothesis: T (n) = O(nlgn);

Show: T (n) ≤ cnlgn for some c>0;

Assume: T (n/3) ≤ c(n/3)lg(n/3);

T (2n/3) ≤ c(2n/3)lg(2n/3);

T(n) ≤ c(n/3)lg(n/3) + c(2n/3)lg(2n/3) + n

= cn/3(lgn - lg3) + 2cn/3(lgn - lg(3/2)) + n

= cn/3lgn - cn/3lg3 + 2cn/3lgn - 2cn/3lg(3/2) + n

= (cn/3lgn + 2cn/3lgn) - cn/3lg3 -2cn/3(lg3-lg2) + n

= cnlgn - cnlg3 + 2cn/3lg2 + n

= cnlgn + n (1 + 2c/3lg2 -clg3) ≤ cnlgn

To make this work, we need:

n(1 + 2c/3lg2 -clg3) ≤ 0

Because n is a positive integer, therefore:

1+2c/3lg2 - clg3 ≤ 0

c (2/3lg2 -lg3) ≤ -1

We can get 2/3lg2 - lg3 = - 0.92

So: -0.92c ≤ -1

c ≤ -1/-0.92

c ≤ 1.09

Now we find out, when c ≤ 1.09, T(n) ≤ cnlgn

Therefore T(n) = O (nlgn)