Wing Kuen Lee <cswkl@ust.hk>

1.2-1

Insertion sort: $f(n) = 8n^2$ steps. Merge sort: $g(n) = 64n\lg n$ steps

Solving the inequality $8n^2 \le 64n\lg n, \forall n \ge 0$ gives $n \le 8\lg n$. One way to find *n* is by an approximation method such as fixed-point iteration. By fixed-point iteration, when *n* is 43.98, 8lg *n* is 43.67. Now, substitute *n* = 43 and *n* = 44, we have:

n	*f(n)*	*g(n)*
43	14792	14934
44	15548	15374

Both *f(n)* and *g(n)* are monotonically increasing. However, when *n* = 1, *f*(1) = 8 > *g*(1) = 0, Thus, for 1 < *n* < 44, the insertion sort beats the merge sort.

1-1

First express each function in term of *t* (*i.e.* f(t)=n):

$1)\lg n = t \Rightarrow n = \lfloor 2^t \rfloor$

$2)\sqrt{n} = t \Rightarrow n = \lfloor t^2 \rfloor$

$3)n = t$

$4)n\lg n = t \Rightarrow \lg n = t/n \Rightarrow n = 2^{(t/n)}$

$5)n^2 = t \Rightarrow n = \lfloor \sqrt{t} \rfloor$

$6)n^3 = t \Rightarrow n = \lfloor \sqrt[3]{t} \rfloor$

$7)2^n = t \Rightarrow \lg 2^n = \lg t \Rightarrow n = \lfloor \lg t \rfloor$

$8)n! = t \Rightarrow$ by guessing!

Then express each time unit in term of microseconds, where $1s = 1 \times 10^6 \mu s$:

1 s	1 min	1 hr	1 day	1 mo	1 yr	1 century
1E6=10^6	6E7	3.6E9	8.64E10	2.592E12	3.1104E13	3.1104E15

Now find *n* by substituting each time unit from the second table into the functions in the first table. Only a portion of the table is given. The rest is left to the reader as exercise:

	1 s	1 min	1 hr	1 day	1 mo	1 yr	1 century
1)	2^{10^6}						
2)	1E12						
3)	1E6	6E7					
4)	62746	2801417					
5)	1000	7745					
6)	100	391					
7)	19	25					
8)	9	11					

2.2-1

$\Theta(n^3)$. [Reason: there exists a constant c1=1/10000, c2=1, n0=120000 such that $0 \le c1 \cdot n^3 \le f(n) \le c2 \cdot n^3, \forall n \ge n0$.]

2.3

a)

This problem is expected to be done using the way similar to p24 of the textbook. The analysis will be much easier if the while-loop is converted into a for-loop. Following the example, we have:

	Lines	Cost	Times
1	`y ← 0`	c1	1
2	`for i ← n to 0`	c2	(n+1)+1 *(the +1 is to check if i<0)*
3	`y ← a_i + x*y`	c3	$\sum_{i=0}^{n+1} 1$ =n+1

Thus, $T(n) = c1 + c2((n+1)+1) + c3(n+1) = \Theta(n)$, where ci>0.

b)

```
1   y ← 0
2   for i ← 0 to n
3      product ← 1
4      for j ← 1 to i
5         product ← product * x
6      y ← y + a[i] * product
7   return y
```

Step	Cost	Times
1	c1	1
2	c2	$1 + \sum_{i=0}^{n} 1$ = n+2
3	c3	$\sum_{i=0}^{n} 1$
4	c4	$\sum_{i=0}^{n} (\sum_{j=1}^{i} 1 + 1) = \sum_{i=0}^{n} (i+1) = 1 + \sum_{i=1}^{n} (i+1) = n(n-1)/2 + n + 1$
5	c5	$\sum_{i=0}^{n} \sum_{j=1}^{i} 1 = n(n-1)/2$
6	c6	n+1
7	c7	1

Thus,

$$T(n) = c1 + c2(n+2) + c3(n+1) + c4((n-1)/2 + n + 1) + c5(n(n-1)/2 + c6)n + 1) + c7 = \Theta(n^2)$$

3.1-1

Let $f(n)$ and $g(n)$ be any asymptotically nonnegative functions.
Then obviously, $\exists n_0$ such that $f(n)$ and $g(n)$ are nonnegative, $\forall n \geq n_0$. …(1)

By the definition of Θ-notation, $\Theta(f(n)+g(n))=\{h(n)$: there exists positive constants c_1, c_2, n_0 such that $\forall n \geq n_0$, $0 \leq c_1 (f(n)+g(n)) \leq h(n) \leq c_2 (f(n)+g(n))\}$. …(2)

By (1), $f(n) \leq f(n) + g(n)$ and $g(n) \leq f(n) + g(n)$.
Now let $c_2 = 1$. Then,

$$\begin{aligned} & \max(f(n), g(n)) \leq c_2(f(n)+g(n)) \\ & \Rightarrow 2\max(f(n), g(n)) \leq 2(f(n)+g(n)) \\ & \Rightarrow f(n)+g(n) \leq 2\max(f(n), g(n)) \leq 2(f(n)+g(n)) \\ & \Rightarrow \tfrac{1}{2}(f(n)+g(n)) \leq \max(f(n), g(n)) \leq (f(n)+g(n)) \end{aligned} \quad \text{…(3)}$$

By (2) and (3), $0 \leq c_1 (f(n)+g(n)) \leq \max(f(n),g(n)) \leq c_2 (f(n)+g(n))$, $\forall n \geq n_0$, where $c_1 = ½$ and $c_2 = 1$.
$\therefore \max(f(n), g(n)) = \Theta(f(n) + g(n))$.

3.1-2

Let $f(n)=(n+a)^b$, for real constants a, $b > 0$.

Case 1: $a \geq 0$.
Then $0 \leq n \leq n + a \leq 2n$ for all $a \leq n$. Also, $\log(n) \leq \log(n+a) \leq \log(2n)$
Since $b > 0$, a^b is monotonically increasing in b.

$$\begin{aligned} & b\log n \leq b\log(n+1) \leq b\log(2n) \\ & \Rightarrow \log n^b \leq \log(n+a)^b \leq \log(2n)^b \\ & \Rightarrow 0 \leq n^b \leq (n+a)^b \leq (2^b)n^b \\ & \therefore \exists c1 = 1, c2 = 2^b, n0 = a : 0 \leq c1n^b \leq (n+a)^b \leq c2n^b, \forall n \geq n0. \end{aligned}$$

$\therefore (n+a)^b = \Theta(n^b)$

Case 2: $a < 0$ implies $n + a = n - |a|$.
Then $0 \leq n/2 \leq n - |a| \leq n$, for all $2|a| \leq n$. Similar to case 1,

$$\begin{aligned} & 0 \leq n/2 \leq n-|a| \leq n \\ & \Rightarrow \log(n/2) \leq \log(n-|a|) \leq \log n \\ & \Rightarrow \log(n/2)^b \leq \log(n-|a|)^b \leq \log n^b \\ & \Rightarrow 0 \leq (1/2)^b n^b \leq (n-|a|)^b \leq n^b \\ & \therefore \exists c1 = (1/2)^b, c2 = 1. \end{aligned}$$

$\therefore (n - |a|)^b = \Theta(n^b)$

Combining Case 1 + case 2, $(n+a)^b = \Theta(n^b)$.