Name: CHAN King Yeung
SID: 1155119394
CSCI2100 Assignment 1

Question 1
Total frequency: $3n^2 + 8n + 5$
Time complexity: $O(n^2)$

Question 2

a) By definition, we have $g_i(n) \le c_i \cdot f_i(n)$ for any $n \ge n_i$
The product of $g_1(n)$ and $g_2(n)$ resulting an inequality as follow

$$g_1(n)g_2(n) \le c_1c_2 \cdot f_1(n)f_2(n) \text{ for any } n > max(n_1, n_2)$$

Then we have $g_1(n)g_2(n) = O\big(f_1(n)f_2(n)\big)$

b) $g(n) = \left(n^2 + \sqrt{n}\right) \cdot (n + log(n)) = O(n^2 \cdot n) = O(n^3)$

c) $g(n) = (n^3 + 3n^2 + 5) \cdot (n^2 + n^4) = \Theta(n^3 \cdot n^4) = \Theta(n^7)$

d) By definition, we have $c_1 \cdot f(n) \le g(n) \le c_2 \cdot f(n)$ for any n
The maximum of $f(n)$ and $g(n)$ resulting an inequality as follow

$$c_1\big(f(n) + g(n)\big) \le max\big(f(n), g(n)\big) \le c_2\big(f(n) + g(n)\big)$$

Given both $f(n)$ and $g(n)$ are nonnegative function, we have

$$max\big(f(n), g(n)\big) \le 1 \cdot \big(f(n) + g(n)\big) \text{ and } max\big(f(n), g(n)\big) \ge \tfrac{1}{2} \cdot \big(f(n) + g(n)\big)$$

Thus, $c_1 = \frac{1}{2}$ and $c_2 = 1$ holds the inequality for any n

---

Question 3

a)

6	a_h	c	9	d	a_t		b	$\emptyset$	4	a	d	3	c	b		$\emptyset$	a
	a			b			a_t			c			d			a_h	

b)

7	b	a_t	5	d	a		a	$\emptyset$	2	a_h	d	3	c	b		$\emptyset$	c
	a			b			a_t			c			d			a_h	

c)

6	c	d	$\cancel{8}$	$\cancel{d}$	$\cancel{a_t}$		d	$\emptyset$	3	a_h	a	5	a	a_t		$\emptyset$	c
	a			$\cancel{b}$			a_t			c			d			a_h	

Question 4

a) We have that $a = 4, b = 4, \lambda = 0$
Since $log_4(4) > 0$, we have that $g(n) = O\big(n^{log_b(a)}\big) = O(n)$

b) We have that $a = 2, b = 4, \lambda = \frac{1}{2}$
Since $log_4(2) = \frac{1}{2}$, we have that $g(n) = O\big(n^\lambda \cdot log(n)\big) = O\big(\sqrt{n}\, log(n)\big)$

c) We have that $a = 2, b = 4, \lambda = 2$
Since $log_4(2) < 2$, we have that $g(n) = O\big(n^\lambda\big) = O(n^2)$

Question 5

$f_1(n) = 2^{2^{100000}}$ runs in a constant time, thus, $f_1(n) = O(1)$
$f_2(n) = 2^{10000n}$ grows in 2^n, thus, $f_2(n) = O(2^n)$
$f_3(n) = \binom{n}{2}$ rewrite as $\frac{n(n-1)}{2}$, thus, $f_3(n) = O(n^2)$
$f_4(n) = n\sqrt{n}$ known as $n^{\frac{3}{2}}$, thus, $f_4(n) = O\left(n^{\frac{3}{2}}\right)$
The growth rate, therefore, is $f_2(n) \succ f_4(n) \succ f_3(n) \succ f_1(n)$

Question 6

a) Suppose $T(n) \leq c \cdot n^2$ holds for $n \leq k-1$
For $n = k$, we have $T(k) \leq T(k-1) + k \leq c \cdot (k-1)^2 + k = c \cdot k^2 - 2c \cdot k + c + k$
To make $T(k) \leq c \cdot k^2$, c must statisfy that $-2c \cdot k + c + k \leq 0 \Longrightarrow c \geq 1$
By induction, for any $k \geq 1$, we obtain $T(n) \leq 1 \cdot n^2$, therefore, $T(n) = O(n^2)$

b) We have that $a = 1$, $b = 2$, $\lambda = 0$
Since $log_2(1) = 0$, we have that $T(n) = O\big(n^\lambda \cdot log(n)\big) = O(log(n))$

Question 7

By the definition of Big-Omega, we obtain as follow

$$g(n) = \Omega\left(n^{log_b(a)} + \sum_{i=0}^{y} \left(\frac{a}{b^\lambda}\right)^i n^\lambda\right), \text{ where } y = log_b(n) - 1$$

a) Given $log_b(a) < \lambda$, we have $a < b^\lambda$ and $g(n) = \Omega\big(n^{log_b(a)} + c_0 \cdot n^\lambda\big)$ for some constant c_0
n^λ grow faster than $n^{log_b(a)}$, thus by the sum property we obtain $g(n) = \Omega\big(n^\lambda\big)$

b) Given $log_b(a) = \lambda$, we have $a = b^\lambda$ and $g(n) = \Omega\big(n^{log_b(a)} + n^\lambda \cdot log_b(n)\big)$
$n^\lambda \cdot log_b(n)$ grow faster than $n^{log_b(a)}$, thus by the sum property we obtain $g(n) = \Omega\big(n^\lambda \cdot log(n)\big)$

c) Given $log_b(a) > \lambda$, we have $a > b^\lambda$ and $g(n) = \Omega\big(n^{log_b(a)} + c_0 \cdot n^\lambda\big)$ for some constant c_0
$n^{log_b(a)}$ grow faster than n^λ, thus by the sum property we obtain $g(n) = \Omega\big(n^{log_b(a)}\big)$