Question 1

Let c be the cost of calling the function WORK. That is, the cost of the function is constant, regardless of the input value. Determine the respective closed-form $T(n)$ for the cost of calling WORK.

```
function A1(n : Z+)
    val ← 0
    for i from 1 to n by multiplying by 3 do
        for j from i to i^2 do
            val ← val + WORK(n)
        end for
    end for
    return val
end function
```

i: 1 3 9 .... n

$3^k = i$ 0 1 2 3 ...

$k = \log_3 i \to \log_3 n$

3^0 3^1 ...

Want: $T(n) = \sum_{k=1}^{\log_3 n} \sum_{j=3^k}^{(3^k)^2} c$

$$= c \sum_{k=0}^{\log_3 n} \sum_{j=3^k}^{(3^k)^2} 1 = c \sum_{k=1}^{\log_3 n} \left(\sum_{j=0}^{(3^k)^2} 1 - \sum_{j=0}^{3^k - 1} 1 \right)$$

$$\sum_{i=5}^{10} 1 = 6 = \sum_{i=0}^{10} 1 - \sum_{i=0}^{5-1} 1$$

$$= \sum_{i=1}^{10} 1 - \sum_{i=1}^{4} 1$$

$$\sum_{k=1}^{\log_3 n} \sum_{j=0}^{(3^k)^2} 1 = \sum_{k=1}^{\log_3 n} (3^k)^2 + 1$$

$$\sum_{k=0}^{n} r^k = \frac{1 - r^n}{1 - r}$$

$$= \boxed{\sum_{k=1}^{\log_3 n} (9)^k} + \sum_{k=1}^{\log_3 n} 1$$

$\frac{9}{8} c n^2 - \frac{3}{2} c n + \left(\frac{11}{8} + \log_3 n\right) c$

Question 2

Derive the closed-form $T(n)$ for the value returned by the following algorithm:

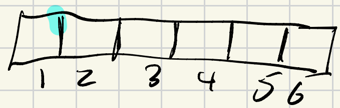
```
function A2(n : Z+)
    sum ← 0
    for i from 0 to n^4 − 1 do
        for j from i to n^3 − 1 do
            sum ← sum + 1
        end for
    end for
    return sum
end function
```

$$\sum_{i=0}^{\cancel{n^4-1}\; n^3-1} \sum_{j=i}^{n^3-1} 1 = \sum_{i=0}^{n^3-1} (n^3-1-i+1)$$

$i = n^3 + 1$

$$= \sum_{i=0}^{n^3-1} (n^3 - i)$$

$$\sum_{i=j}^{k} 1 = (k-j+1)$$



$$= \sum_{i=0}^{n^3-1} n^3 - \sum_{i=0}^{n^3-1} i$$

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

$$= n^6$$

$$\boxed{\sum_{i=0}^{n} i = \frac{n \times (n+1)}{2}}$$

$1 + 2 + \cancel{+99} + 100$

$50(101) = \frac{100(101)}{2}$

$||$

$0 + 1 + 2 + \ldots + 5050$

1

Question 3

(a) The following statements are true or false?

1. $n^2 = O(5^{\log n})$

$$n^2 \overset{?}{<<} 5^{\log n}$$

$$= n^{\log_2 5 \approx 2.1}$$

T because $\log_2 5 \geq 2$.

ex: base 4

$\log_4 n \rightarrow \log_2 n$

What if not base 2?

Change of base: $\log_a(b) = \frac{\log_c(b)}{\log_c(a)}$

$$\log_2 n = \frac{\log_4 n}{\log_4 2}$$

$.5$

2. $\frac{\log n}{\log \log n} \overset{?}{=} O(\sqrt{\log n})$ F

$>>$

$$\sqrt{\log n} = \frac{\log n}{\sqrt{\log n}}$$

3. $n^{\log n} = \Omega(n!)$ F

$$n^{\log n} \overset{?}{>>} n! = n \times (n-1) \times (n-2) \ldots 1$$

$$\geq \frac{n}{2} \times \frac{n}{2} \times \ldots \times \frac{n}{2} \cdot 1 \times 1 \ldots 1$$

$$n! \leq n^n \qquad = \left(\frac{n}{2}\right)^{n/2} \in \Theta(n^{n/2})$$

(b) Sort the following functions in increasing order of asymptotic (big-O) complexity:

$$f_1(n) = n^{\sqrt{n}}, \quad f_2(n) = 2^n, \quad f_3(n) = n^{10} \cdot 2^{n/2}, \quad f_4(n) = \binom{n}{2}$$

2^n $\quad n^{10} 2^{n/2} = \underline{(\sqrt{2})^n}$

$\binom{n}{2} \in O(n^2)$

$\binom{n}{k} \in O(n^k)$

$f_4 \quad f_1 \quad f_3 \quad f_2$

$O(n^2)$

$n^{\sqrt{n}} \overset{?}{<} 2^n$

$\sqrt{n} \log n \overset{?}{<} n \log 2$

$\log n \overset{?}{<} \sqrt{n} \rightarrow n^{\sqrt{n}} \leq 2^n$

Question 4

(a) Show that $\max\{f(n), g(n)\} = \Theta(f(n)+g(n))$ for any $f(n)$ and $g(n)$ that eventually become and stay positive.

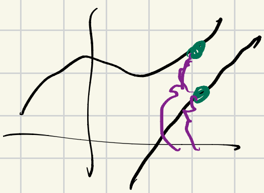
(b) Give an example of f and g such that f is not $O(g)$ and g is also not $O(f)$.

a) $\max\{f(n), g(n)\} = \Theta(f(n)+g(n))$

wts1: $\cdot \in O(f(n)+g(n))$

wts2: $\cdot \in \Omega(f(n)+g(n))$

wts: $\exists c, n_0 \in \mathbb{N}: \max\{f(n), g(n)\} \le c(f(n)+g(n))$ $(n \ge n_0)$



$c = 1$

n_0: first time both are positive

wts2: Ω $\exists c, n_0 \in \mathbb{N}: \max \ge c(f+g)$

n_0: same thing

[avg of $f(n), g(n)$ $\le \max(\cdot,\cdot)$]

$4^n + \cancel{\ast}\ 3^n$ $c = 1/2$

$\cancel{\ast} \le 4^n$

$g(x) \notin O(x)$

$f(x) \notin O(g(x))$

$f(x) = x$

$g(x) = \begin{cases} 1 & x \text{ is even} \\ x^2 & x \text{ is odd} \end{cases}$

$n \ge 1$

$$g \in \Theta(f) \rightarrow f \in \Theta(g) \quad \forall, f, g$$

$g \in O(f)$ $\quad$ $g \in \Omega(f)$ $\quad$ $f \in O(g)$ $\quad$ $f \in \Omega(g)$

$\exists c_1, n_1 \in \mathbb{N}:$ $\quad$ $\exists c_2, n_2 \in \mathbb{N}:$

$g \leq c_1 f$ $\quad$ $g \geq c_2 f$

$\frac{1}{c_2} g \geq f$

$\exists c_3 n_3 \in \mathbb{N}:$ $\quad$ $\exists c_4 n_4 \in \mathbb{N}$

$f \leq c_3 g$ $\quad$ $f \geq c_4 g$

$c_3 = \frac{1}{c_2}$

$n_3 = n_2$

$$h \in \Theta(g) \wedge g \in \Theta(f) \rightarrow h \in \Theta(f)$$

We know $h \in O(g)$, $g \in O(f)$

wts: $h \in O(f)$

$\exists c_0, n_0$ $\quad$ $\exists c_1, n_1$

$h \leq c_0 g$ $\quad$ $g \leq c_1 f$

$$h \le c_0 g \le c_0 c_1 f$$