CSE321
Introduction to the Algorithm
Homework 1

Ahmet Tugkan AYHAN
1901042692

1

a) $(2^n + n^3) \in O(4^n)$ ★ Statement is TRUE

* $f(n) = O(g(n))$ iff there are two positive constants $\overset{\nearrow 1}{c}$ and $\overset{\nearrow 10}{n_0}$ such that $f(n) \leq c.g(n)$ for all $n \geq n_0$

* First, $n^3 \leq 2^n \leq 4^n$ for all $n \geq 10$, so the equation becomes $\Rightarrow 2^n + n^3 \leq 2^n + 2^n \leq \underbrace{4^n + 4^n}_{2.4^n}$

* Finally from the notation rules $4^n.2 \in O(4^n)$, thus $2^n + 3^n \in O(4^n)$

b) $\sqrt{10n^2 + 7n + 3} \in \Omega(n)$ ★ Statement is TRUE

* $f(n) = \Omega(g(n))$ iff there are two positive constants c and n_0 such that $f(n) \geq c.g(n)$ for all $n \geq n_0$

* $f(n) = \sqrt{10n^2 + 7n + 3}$, $g(n) = n$, $n_0 = 1$, $c = \sqrt{10}$, $\frac{f(n)}{g(n)} (max) = \sqrt{20}$, $(min) = \sqrt{10}$, $\sqrt{10n^2+7n+3} \geq \sqrt{10}n \Rightarrow \sqrt{10 + \frac{7}{n} + \frac{3}{n^2}} \geq \sqrt{10}$ for every $n \geq 1$

* After we took limit, while n goes to infinity for every c value that is less than or equal to $\sqrt{10}$ satisfies the equation for all $n \geq 1$

c) $(n^2 + n) \in o(n^2)$ ★ Statement is FALSE

* There must be a n_0 value for all $c > 0$ which satisfies $f(n) < c.g(n)$ for all $n \geq n_0$

* $f(n) = n^2 + n$, $g(n) = n^2$, $1 + \frac{1}{n} < c$, for $c = 1$; $\frac{1}{n} < 0$, $n_0 = ?$, but there is no n_0 value which satisfies this equation (n_0 must be positive $\mathbb{N}$)

d) $3\log_2^2 n \in \Theta(\log_2 n^2)$ ★ Statement is FALSE

* $f(n) = \Theta(g(n))$ iff there are three positive constants c_1, c_2 and n_0 such that $c_1.g(n) \leq f(n) < c_2.g(n)$ for all $n \geq n_0$

* $f(n) = 3\log_2^2 n$, $g(n)$ $2\log_2 n$, Also $f(n)$ should be equal to $O(g(n))$ and $\Omega(g(n))$

* For $f(n) = O(g(n)) \Rightarrow f(n) \leq c.g(n) \Rightarrow \frac{f(n)}{g(n)} \leq c \Rightarrow \frac{3\log_2 n . \log_2 n}{2\log_2 n} = \frac{3}{2}\log_2 n \leq c$ There is no constant value which satisfies that.

e) $(n^3+1)^6 \in O(n^3)$ ★ Statement is FALSE

* $f(n) \leq c.g(n) \Rightarrow \frac{f(n)}{g(n)} \leq c \Rightarrow \frac{(n^3+1)^6}{n^3} \leq c \Rightarrow \underbrace{n^{18} + \dots + 1}_{} \leq c$

ex/ $c = 2$, if $n_0 = 1$, $n = 1$ ✓, $n = 2$ ✗

* $f(n) = (n^3+1)^6 = n^{18} + a.n^{17} + \dots + 1$, $g(n) = (n^3)$

for any $c > 0$ there is no n_0 value that satisfies condition For all $n \geq n_0$ / impossible

②

a) $2n\log(n+2)^2 + (n+2)^2 \log\frac{n}{2}$

* $4n\log(n+2) + (n+2)^2\log(n-2)$

* $\Theta(4n\log(n+2)) + \Theta((n+2)^2\log(n-2))$ Next Getting rid of constants

* $\Theta(n\log n) + \Theta(n^2\log n)$ Next Getting only biggest one

* $\Theta(n^2\log n)$

b) $0.001n^4 + 3n^3 + 1$

* $\lim\limits_{n\to\infty} \frac{0.001n^4}{3n^3+1} = \infty$, thus writing the constant value and $3n^3$ is unnecessary since their impact is not that much as $0.001n^4$.

* $\Theta(0.001n^4) + \Theta(3n^3) + \Theta(1)$

* $\Theta(0.001n^4)$

* $\Theta(n^4)$

---

③

a) $\log n$, $n^{\log n}$, $n^{1.5}$

* $\lim\limits_{n\to\infty} \frac{n^{\log n}}{\log n} = \frac{\infty}{\infty} \xrightarrow{\text{L'Hopital}} \lim\limits_{n\to\infty} \frac{2\log n \cdot n^{\log n} \cdot \cancel{\frac{1}{n}}}{\cancel{\frac{1}{n}} \cdot \frac{1}{\ln 10}} = \infty$, thus $\log n < n^{\log n}$

* $\lim\limits_{n\to\infty} \frac{\log n}{n^{1.5}} = \frac{\infty}{\infty} \xrightarrow{\text{L'Hopital}} \lim\limits_{n\to\infty} \frac{\frac{1}{n}\cdot\frac{1}{\ln 10}}{1.5(n)^{0.5}} = 0$, thus $\log n < n^{1.5}$

* $\lim\limits_{n\to\infty} \frac{n^{1.5}}{n^{\log n}} = \frac{1}{n^{\log n - 1.5}} = \frac{1}{\infty} = 0$, thus $n^{1.5} < n^{\log n}$

★ $\log n < n^{1.5} < n^{\log n}$

b) $n!$, 2^n , n^2

! Stirling's Formula : $n! \sim \sqrt{2\pi n}\left(\frac{n}{e}\right)^n$

* $\lim_{n\to\infty} \frac{2^n}{n^2} = \frac{\infty}{\infty} \xrightarrow{\text{L'Hopital}} \frac{2^n \ln 2}{2n} \xrightarrow{\text{L'Hopital}} \lim_{n\to\infty} \frac{2^n \ln^2 2}{2} = \infty$, thus $2^n > n^2$

* $\lim_{n\to\infty} \frac{2^n}{n!} = \frac{2^n}{\sqrt{2\pi n}\left(\frac{n}{e}\right)^n} = \frac{2^n . e^n}{\sqrt{2\pi n} . n^n} = \overset{0}{\cancel{\frac{1}{\sqrt{2\pi n}}}} \cdot \overset{0}{\left(\cancel{\frac{2e}{n}}\right)^n} = 0$, thus $n! > 2^n$

★ $n^2 < 2^n < n!$

c) $n\log n$, $\sqrt{n}$

* $\lim_{n\to\infty} \frac{n\log n}{\sqrt{n}} = \frac{n \log n}{n^{\frac{1}{2}}} = n^{\frac{1}{2}} . \log n = \infty$, thus $n\log n > \sqrt{n}$

d) $n2^n$, 3^n

* $\lim_{n\to\infty} \frac{n2^n}{3^n} = n.\left(\frac{2}{3}\right)^n = \infty . 0 \rightarrow \lim_{n\to\infty} \frac{n}{\left(\frac{3}{2}\right)^n} = \frac{\infty}{\infty} \xrightarrow{\text{L'Hopital}} \lim_{n\to\infty} \frac{1}{\left(\frac{3}{2}\right)^n \cdot \ln\frac{3}{2}} = \frac{1}{\infty} = 0$, thus $3^n > n.2^n$

e) $\sqrt{n+10}$, n^3

* $\lim_{n\to\infty} \frac{n^3}{\sqrt{n+10}} = \frac{\infty}{\infty} \xrightarrow{\text{L'Hopital}} \frac{3n^2}{\frac{1}{2\sqrt{n+10}}} = \frac{6n^2\sqrt{n+10}}{1} = \infty$, thus $n^3 > \sqrt{n+10}$

4

a) Comparison of two matrix elements of Matrix B → Basic Operation

b) Worst Case : $W(n) = \sum_{i=0}^{n-2} . \left(\sum_{J=i+1}^{n-1} 1\right) = \sum_{i=0}^{n-2} . (n-1-i-1+1) = \sum_{i=0}^{n-2} (n-i-1) = \sum_{i=1}^{n-1} (n-i)$

$= (n-1) + (n-2) + \dots + \underbrace{(n-(n-1))}_{1}$

$= 1 + 2 + \dots + (n-1)$

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

$n-(n-1)$, $-n+1$

c) $W(n) = \frac{(n-1).n}{2} = \frac{n^2-n}{2}$

$W(n) \in \Theta\left(\frac{n^2-n}{2}\right) = \Theta(n^2)$

⑤

a) Incrementing the matrix element by the product of the other two matrix elements

Basic Operation $(C[i,j] \mathrel{+}= A[i,k] * B[k,j])$

b) $A(n) = \sum_{i=0}^{n-1} . \sum_{j=0}^{n-1} . \left(\sum_{k=0}^{n-1} 1\right)$

$= \sum_{i=0}^{n-1} . \sum_{j=0}^{n-1} . (n-1+1) = \sum_{i=0}^{n-1} . \left(\sum_{j=0}^{n-1} n\right) = \sum_{i=0}^{n-1} . (\underbrace{n+n+n+\dots+n}_{n \text{ times}}) = \sum_{i=0}^{n-1} n^2$

$= \sum_{i=0}^{n-1} n^2 = (\overbrace{n^2+n^2+\dots+n^2}^{n \text{ times}}) = n^2 . n = n^3$ times basic operation is executed.

c) $A(n) = n^3 \Rightarrow A(n) \in \Theta(n^3)$

⑥ Algorithm

```
arr[n] = {x, y, z, ....}   // n elements
k = desired number
for i=0 to n-1 do
    for j=i+1 to n-1 do
        if arr[i]*arr[j] == k
            print (i,j)
```

* Basic operation is comparison

* $A(n) = \sum_{i=0}^{n-1} . \sum_{j=i+1}^{n-1} 1 = \sum_{i=0}^{n-1} (n-1-(i+1)+1) = \sum_{i=0}^{n-1} (n-i-1) = \sum_{i=1}^{n} (n-i)$

* $\sum_{i=1}^{n} (n-i) = (n-1)+(n-2)+ \dots + 0 = \frac{(n-1).n}{2} = \frac{n^2-n}{2}$

* $A(n) = \frac{n^2-n}{2}$, $A(n) \in \Theta\left(\frac{n^2-n}{2}\right) = \Theta(n^2)$