

a) $\log_2 n^2 + 1 \in O(n)$ $\xrightarrow{p(n)}$ $\xrightarrow{g(n)}$

Using limit method; $\log_2 n^2 + 1 = 2 \log_2 n + \log_2 2 = 2 \log_2 n$

$$\lim_{n \rightarrow \infty} \frac{2 \log_2 n}{n} = \frac{2 \cdot 2}{2n \cdot \ln 2} = \lim_{n \rightarrow \infty} \frac{2}{n \ln 2} = \frac{2}{\ln 2} \lim_{n \rightarrow \infty} \frac{1}{n} = 0$$

$\log_2 n^2 + 1 \in O(n) = f(n)$ smaller order of growth than $p(n)$

True

b) $\sqrt{n(n+1)} \in \Omega(n)$
 $f(n)$ $g(n)$

$$\lim_{n \rightarrow \infty} \frac{\sqrt{n(n+1)}}{n} = \frac{\sqrt{n^2 + n}}{n} = \frac{n}{n} = 1 = c, \text{ so}$$

$f(n)$ has same growth of order of $g(n)$, true

$\sqrt{n(n+1)} \in \Omega(n)$

c) $n^{n-1} \in \Theta(n^n)$

$$\lim_{n \rightarrow \infty} \frac{n^{n-1}}{n^n} \Rightarrow \frac{(n-1) \cdot n^{n-1} \cdot \ln(n)}{n \cdot n^n \cdot \ln(n)} = \frac{1 \cdot n^{n-1}}{1 \cdot n^n} = \frac{n^{n-1}}{n^n} = \frac{1}{n}$$

n^{n-1} 's growth order smaller than n^n , so false $n^{n-1} \in \Theta(n^n)$ statement

d) $O(2^n + n^3) \subset O(4^n)$

compare these $O(2^n)$ or $O(n^3)$ faster $\rightarrow \lim_{n \rightarrow \infty} \frac{2^n}{n^3} = \frac{1 \cdot 2^n \cdot \ln 2}{0 \cdot n^3 \cdot \ln n} = \infty$

2^n bigger than n^3

$O(2^n)$

$O(2^n) \subset O(4^n) \rightarrow \lim_{n \rightarrow \infty} \frac{4^n}{2^n} = \infty$ 4^n is faster than 2^n

Thus we can say that $O(2^n + n^3) \subset 2^n \rightarrow$ true

$$e) O(2 \log_3^3 n) \subset O(3 \log_2 n^2)$$

$$\lim_{n \rightarrow \infty} \frac{3 \log_2^3 n}{2 \log_3^3 n} = \frac{6 \log_2 n}{2 \log_3 n^{1/3}} \rightarrow \frac{6 \log_2 n}{\frac{2}{3} \log_3 n} = \frac{\frac{9}{n \ln 2}}{\frac{1}{n \ln 3}} = \frac{9 \ln 3}{\ln 2} = C$$

it means they have same growth rate

so, $O(2 \log_3^3 n) \subset O(3 \log_2 n^2)$, subset can be itself so statement is

True,

f) $\log_2 \sqrt{n}$ are same asymptotic order $(\log_2 n)^2$

$$\lim_{n \rightarrow \infty} \frac{\log_2 \sqrt{n}}{(\log_2 n)^2} = \frac{\log_2 n^{1/2}}{\log_2 n \log_2 n} = \frac{\frac{1}{2} \log_2 n}{\log_2 n \log_2 n} = \frac{\frac{1}{2}}{\log_2 n} \cdot \frac{1}{\log_2 n} \approx 0$$

it means $\log_2 \sqrt{n}$ has smaller growth rate than $(\log_2 n)^2$, so statement

is False

$$2) \quad n^2, n^3, n^2 \log n, n^{1/2}, \log n, 10^n, 2^n, 8^{\log_2 n}$$

$8^{\log_2 n} \rightarrow n^3$

$$n^{1/2} < n^2 < n^3 = n^3$$

$$n^2 \log n > \log n \rightarrow \text{compare } \log n, n^{1/2} \quad \lim_{n \rightarrow \infty} \frac{n^{1/2}}{\log n} = \frac{\frac{1}{2} n^{-1/2}}{\frac{1}{n} \log_2 e} = \frac{\frac{1}{2} \frac{1}{\sqrt{n}}}{\frac{\log_2 e}{n}} = \frac{1}{2} \frac{n}{\sqrt{n}} = \frac{1}{2} \sqrt{n} \rightarrow \infty$$

$$2^n 5^n > 2^n$$

$$\frac{1}{\sqrt{n}} \rightarrow 0$$

$$\text{so } \log n < \sqrt{n}$$

$$\log n < \sqrt{n} < n^2 < n^3 = 8^{\log_2 n}$$

$$\text{Compare } \lim_{n \rightarrow \infty} \frac{n^2 \log n}{n^3} = \frac{\log n}{n} = \frac{\frac{1}{n} \ln 2}{1} = \frac{1}{n} = 0$$

$$8 \log_2 n = n^3 > n^2 \log n > n^2 > \sqrt{n} > \log n$$

$$2^n \text{ or } n^3 \Rightarrow \lim_{n \rightarrow \infty} \frac{2^n}{n^3} = \frac{2^n \ln 2}{3n^2} = \frac{2^n}{n^2} \rightarrow \infty = \infty$$

New order

$$10^n > 2^n > 8 \log_2 n = n^3 > n^2 \log n > n^2 > \sqrt{n} > \log n$$

3) a)

for loop runs n times. Because of the length of the array size. In if else part doesn't affect the execution time, because they only run one times in for loop.

I did not accidentally assign values to variables first and second. But in question, they actually take the maximum value the integer can get.

In both cases, the algorithm will run in average of n -time.

$\Theta(n)$

b) Analysis loop

for (int i=2; i <= n; i++) \rightarrow input size n

if (i % 2) \rightarrow if i is even, then it executed one times.

else part \rightarrow if i is not even, executed that part next i value will be $i^2 - i$.

$$i_{n+1} = i_n(i_{n-1})$$

→ This part dominates that parts, because exponential.

→ If we say this function run k times

$$i^{2^k} \rightarrow k \text{ times run}$$

values is x^{2^k} , it increases until the value n .

$$x^{2^k} = n \quad \text{find } n \text{ take } \log$$

$$k = \log(\log n)$$

So, average time comp. $\Theta(\log(\log n))$

4) a) $\sum_{i=1}^n i^2 \log i$

$f(n) = i^2 \log i$ is non-decreasing function

$$\int_0^n i^2 \log i \, di \leq f(n) \leq \int_1^{n+1} i^2 \log i \, di$$

$$u = \log i$$

$$v = i^2 \quad v = \frac{i^3}{3}$$

$$\log i \cdot \frac{i^3}{3} - \frac{\log e}{3} \leq f(n) \leq \frac{(i+1)^3 \cdot (3 \log_e(i+1) - 1) + 1}{9 \log 2}$$

$$f(n) \in O(n^3 \cdot \log n) \quad \text{upper bound}$$

$$f(n) \in \Omega(n^3 \cdot \log n) \quad \text{lower bound}$$

so average time is equal to $\Theta(n^3 \cdot \log n)$

4) b)

$\sum_{i=1}^n i^3$ is it non decreasing?
Yes, non-decreasing fractions

$$\int_0^n i^3 di \leq f(n) \leq \int_1^{n+1} i^3 di$$

$$\frac{i^4}{4} \Big|_0^n \leq f(n) \leq \frac{i^4}{4} \Big|_1^{n+1}$$

$$\frac{n^4}{4} - 0 \leq f(n) \leq \frac{(n+1)^4}{4} - \frac{1}{4}$$

upper bound $O(n^4)$, lower bound $\Omega(n^4)$, Both upper and lower bound are same so we get $f(n) \in \Theta(n^4)$

4) c)

$\sum_{i=1}^n \frac{1}{2\sqrt{i}}$ it is non-increasing so:

$$\int_1^{n+1} \frac{1}{2\sqrt{i}} \leq f(n) \leq \int_0^n \frac{1}{2\sqrt{i}}$$

$$\sqrt{n+1} - 1 \leq f(n) \leq \sqrt{n}$$

$$\left. \begin{array}{l} f(n) = \Omega(\sqrt{n}) \\ f(n) = O(\sqrt{n}) \end{array} \right\} \Rightarrow f(n) = \Theta(\sqrt{n})$$

4) d) $\sum_{i=1}^n \frac{1}{i}$ non-increasing

$$\ln \frac{n+1}{1} \leq f(n) \leq \ln \frac{n+1}{1}$$

$$\ln(n+1) \leq f(n) \leq \ln(n+1) - \ln 0$$

underlined

\Rightarrow change

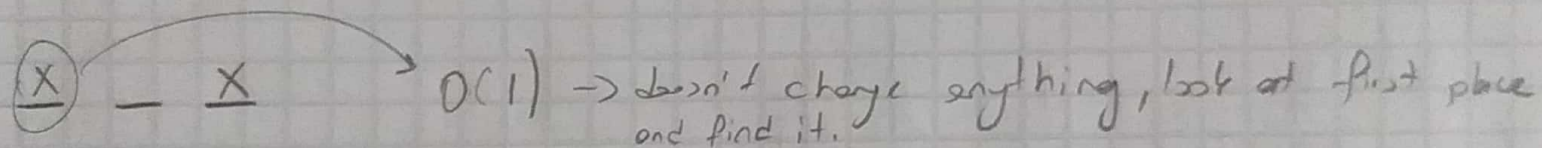
portion

$$\sum_{i=1}^n \frac{1}{i} = 1 + \sum_{i=1}^{n-1} \frac{1}{i}$$

5)

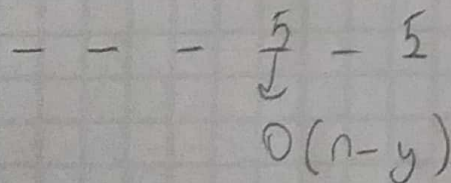
→ If we search first element of list, then our best case will be $O(1)$

→ If we search first element of list and that element is repeated



→ If we search last element of list, then our worst case will be $O(n)$, n refers to list size

→ If we search last element of list, and the last element repeated



↳ first position of last element, this time that wouldn't be worst case.

so worst case $O(n)$, best case $O(1)$