CS-202
Assignment 1
Theory

Anukool Dwivedi
B19071
CSE

(1)

Soln (a) Yes, there is no requirement for the function in the big-Oh to be tight. Also, the big-Oh bound refers to the worst-case input and some inputs may not elicit the worst case.

Soln (b) Yes, there is no requirement for the function in the big-Oh to be tight. So, we might say that $O(n^2)$ but its possible that all inputs take ~~time~~ $O(n)$ time.

Soln (c) Yes, because an algorithm takes $\Theta(n^2)$ in worst case but not on all cases. So it might take $O(n)$ time on some inputs.
For eg, worst case of insertion sort is $\Theta(n^2)$, but if the array is already sorted then it will take $O(n)$ time.

② In my algorithm, I used the fact that GCD(a, 0) or GCD(0, a) is a [i.e. a number itself].

⇒ So, let us suppose we have 2 positive integers a and b.

⇒ If a and b are both even, then they will definitely has 2 as a common factor. So, we store this 2 and further call for GCD(a/2, b/2).

$\therefore$ GCD(a, b) = 2 * GCD(a/2, b/2).

⇒ If a is odd and b is even, then, they do not has 2 as a common factor. So: GCD(a,b) = GCD(a, b/2).

⇒ If a is even, and b is odd, then GCD(a,b) = GCD(a/2, b).

⇒ If a and b are both odd; then, if x is a common factor of a and b.

So, x divides a,
x divides b,

So: $a = \lambda_1 x$ (for some $\lambda_1 \in Z^+$)
$b = \lambda_2 x$ (for some $\lambda_2 \in Z^+$)

Also: $a - b = \lambda_1 x - \lambda_2 x$
$\Rightarrow x(\lambda_1 - \lambda_2)$

So: x divides $|a-b|$ also.

$\therefore$ GCD(a, b) = GCD(a, |a-b|),
where a is minimum of a and b.

DATE / /

$\Rightarrow$ Also, at each case, the numbers are reducing. So, at last, the algorithm converges.

$\Rightarrow$ The algorithm stops when any of the number becomes zero.

(3)

(a) $T(n^2) = 7T(n^2/4) + cn^2$, $T(1) = 1$

Let us replace n^2 with x.

So, $T(x) = 7T(x/4) + cx$ —— (I)

Now, by Substitution:

$T(x/4) = 7T\left(\frac{x}{4^2}\right) + \frac{cx}{4}$ —— (II)

Replacing (II) in (I)

$\Rightarrow T(x) = 7\left[7T(x/4^2) + \frac{cx}{4}\right] + cx$

$\Rightarrow T(x) = 7^2T(x/4^2) + \frac{7}{4}cx + cx$

Now, for some integer k,

$\Rightarrow T(x) = 7^kT\left(\frac{x}{4^k}\right) + cx\left[\left(\frac{7}{4}\right)^0 + \left(\frac{7}{4}\right)^1 + \left(\frac{7}{4}\right)^{k-1}\right]$

$\Rightarrow T(x) = 7^kT\left(\frac{x}{4^k}\right) + cx\left[\frac{(7/4)^k - 1}{7/4 - 1}\right]$

$\Rightarrow$ Let $x/4^k = 1 \Rightarrow k = \log x/2$

$\Rightarrow T(x) = 7^{\log x/2}T(1) + cx \times \left[\frac{4}{3}\left[\left(\frac{7}{4}\right)^{\frac{\log x}{2}} - 1\right]\right]$

On simplification;

$T(x) \approx x^{1.4} + \frac{4}{3} c x^{1.4}$

So, $T(n) = n^{1.4} + \frac{4}{3} c n^{1.4}$

So, it is $\boxed{O(n^{1.5})}$

(b) $T(n) = nT(\sqrt{n})$, $T(2) = 4$

$\Rightarrow T(\sqrt{n}) = \sqrt{n}\, T(n^{1/2^2})$

$\Rightarrow T(n) = n\sqrt{n}\, T(n^{1/2^2})$

$\Rightarrow T(n) = n^{\left(\frac{1}{2^0}+\frac{1}{2^1}\right)} T(n^{1/2^2})$

for some k (integer)

$\Rightarrow T(n) = n^{\left(\frac{1}{2^0}+\frac{1}{2^1}+\cdots\frac{1}{2^{k-1}}\right)} T(n^{1/2^k})$

let $n^{1/2^k} = 2$

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

Now; on simplification;

$\Rightarrow T(n) = 4\left[n^{2-\frac{2}{2^k}}\right]$

$\Rightarrow 4\left[n^{2-\frac{2}{\log n}}\right]$

So, it is $\underline{\underline{O(n^2)}}$.

(c) $T(n) = T(n/2) + 2T(n/4) + \frac{3n}{2}$,

$T(1) = 0$, $T(2) = 2$

$\Rightarrow T(n/2) + 2T(n/4) \leq T(n/2) + 2T(n/2)$

$\Rightarrow T(n/2) + 2T(n/4) + 3n/2 \leq 3T(n/2) + \frac{3n}{2}$

So, I will use the RHS to get upper bound.

$\Rightarrow T(n) = 3T(n/2) + 3n/2$

$\Rightarrow$ By substitution;

$\Rightarrow T(n/2) = 3T\left(\frac{n}{2^2}\right) + \frac{3n}{2^2}$

$\Rightarrow T(n) = 3^2 T\left(\frac{n}{2^2}\right) + \left(\frac{3}{2}\right)^2 n + \frac{3}{2} n$

$\Rightarrow$ for some k,

$\Rightarrow T(n) = 3^k T\left(\frac{n}{2^k}\right) + \frac{3}{2} n \left[\left(\frac{3}{2}\right)^0 + \left(\frac{3}{2}\right)^1 + \cdots + \left(\frac{3}{2}\right)^{k-1}\right]$

$\Rightarrow$ let $n/2^k = 1$ $\boxed{k = \log n}$

$\Rightarrow 3^{\log n} T(1) + \frac{3}{2} n \left[\frac{\left(\frac{3}{2}\right)^{\log n} - 1}{1/2}\right]$

$\Rightarrow 0 + 3n\left[\cancel{\log}\, n^{\log(3/2)} - 1\right]$

$\Rightarrow 3n\left[n^{\log 3/2} - 1\right] \leq 3n\left[n^{\log 4/2} - 1\right]$

$\Rightarrow 3n(n^{\log 3/2} - 1) \leq 3n(n-1)$

So, it is $\boxed{O(n^2)}$

(d) $T(n) = 4T(n/2) + n^3$, $T(1) = 1$

$\Rightarrow T(n/2) = 4\,T\left(\frac{n}{2^2}\right) + \left(\frac{n}{2}\right)^3$ [by subsitution]

$\Rightarrow T(n) = 4^2\,T\left(\frac{n}{2^2}\right) + 4\left(\frac{n}{2}\right)^3 + n^3$

$\Rightarrow$ for some integer k;

$\Rightarrow$

$T(n) = 4^k\,T\left(\frac{n}{2^k}\right) + n^3\left[\frac{1-\frac{1}{2^k}}{1-\frac{1}{2}}\right]$

$\Rightarrow$ let $n/2^k = 1 \Rightarrow k = \log n$

$\Rightarrow T(n) = 4^{\log n} \times T(1) + n^3 \times 2 \times \left[1 - \frac{1}{2^{\log n}}\right]$

$\Rightarrow n^{\log 4} + 2n^3 - \frac{2n^3}{n^{\log 2}}$

$\Rightarrow n^2 + 2n^3 - 2n^2 = 2n^3 - n^2$

So; It is $\boxed{O(n^3)}$

(c) $T(n) = T(n/2) + c\log n$

$\Rightarrow T(n/2) = T\left(\frac{n}{2^2}\right) + c\log\left(\frac{n}{2}\right)$

$\Rightarrow T(n) = T\left(\frac{n}{2^2}\right) + c\left[\log\left(\frac{n}{2^0}\right) + \log\left(\frac{n}{2^1}\right)\right]$

$\Rightarrow$ for some integer k;

$\Rightarrow T(n) = T\left(\frac{n}{2^k}\right) + c\left[\log\left(\frac{n}{2^0}\right) + \log\left(\frac{n}{2^1}\right) + \ldots + \ldots \log\left(\frac{n}{2^{k-1}}\right)\right]$

$\Rightarrow T(n) = T\left(\frac{n}{2^k}\right) + c\left[\log\left[\frac{n^k}{2^{\frac{k(k-1)}{2}}}\right]\right]$

$\Rightarrow$ let $n/2^k = 1 \Rightarrow \boxed{k = \log n}$

and let $T(1) = 1$

$\Rightarrow T(n) = T(1) + c\left[\log\left[\frac{n^{\log n}}{2^{\log n(\log n - 1)/2}}\right]\right]$

$\Rightarrow T(n) \approx 1 + c\left[(\log n)^2 - \frac{(\log n)^2}{2}\right]$

$\Rightarrow T(n) \approx 1 + \frac{c}{2}(\log n)^2$

$\boxed{\text{So, it is } O((\log n)^2)}$

(4) In this we have to find ~~such~~ g such that $\lim_{n\to\infty} \frac{f(n)}{g(n)} \to 0$

(a) $f(n) = n^{1.2}/\log n$

$n^{1.2}/\log n < \frac{n^3}{\log n}$ for $n > 1$

So, $\lim_{n\to\infty} \frac{(n^{1.2}/\log n)}{(n^3/\log n)} = \lim_{n\to\infty} \frac{1}{n^{1.8}} \to 0$

So it is $o(n^3/\log n)$

(b) $f(n) = n^2$

$n^2 < n^3$ for $n>1$

So, $\lim_{n\to\infty} \frac{n^2}{n^3} = \lim_{n\to\infty} \frac{1}{n} \to 0$

So, it is $o(n^3)$

(c) $f(n) = n \log n$

$n \log n < n^2 \log n$ for $n>1$

So, $\lim_{n\to\infty} \frac{n \log n}{n^2 \log n} = \lim_{n\to\infty} \frac{1}{n} \to 0$

So, it is $o(n^2 \log n)$

(d) $f(n) = (1.1)^n$

$(1.1)^n < 3^n$ for $n \geq 1$

So $\lim_{n \to \infty} \frac{(1.1)^n}{3^n} = \lim_{n \to \infty} \left(\frac{11}{30}\right)^n \to 0$

$\boxed{\text{So, it is } o(3^n)}$

(e) $f(n) = (0.9)^n$

$(0.9)^n < n$ for $n \geq 1$

So, $\lim_{n \to \infty} \frac{(0.9)^n}{n} = \lim_{n \to \infty} \frac{9^n}{10^n \times n} \to 0$

So, it is $o(n)$

(f) $f(n) = (\log n)^3$

$(\log n)^3 < n(\log n)^3$ for $n > 1$

$\Rightarrow$ So, $\lim_{n \to \infty} \frac{(\log n)^3}{n(\log n)^3} = \lim_{n \to \infty} \frac{1}{n} \to 0$

So, it is $o(n \log^3 n)$

Soln (5) In this problem; We could express :-

$$r = \sum_{i=1}^{n} \sum_{j=1}^{i} \sum_{k=j}^{j+i} \underbrace{\sum_{l=1}^{i+j-k} 1}_{A}$$

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

$$\text{Now; } r = \sum_{i=1}^{n} \sum_{j=1}^{i} \underbrace{\sum_{k=j}^{j+i} (i+j-k)}_{B}$$

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

$$\Rightarrow [(i+j)-j] + [i+j-(j+1)] + \text{---} + [i+j-(j+i)]$$

$$\Rightarrow i + i-1 + i-2 + \text{---} + i-i$$

$$\Rightarrow i(i+1) - \frac{i(i+1)}{2} = \frac{i(i+1)}{2}$$

$$\Rightarrow r = \sum_{i=1}^{n} \sum_{j=1}^{i} \frac{i^2+i}{2} = \sum_{i=1}^{n} \frac{i^3+i^2}{2}$$

$$\Rightarrow \frac{1}{2}\left[\sum_{i=1}^{n} i^3 + \sum_{i=1}^{n} i^2\right]$$

$$\Rightarrow \frac{1}{2}\left[\left[\frac{n(n+1)}{2}\right]^2 + \frac{n(n+1)(2n+1)}{6}\right]$$

$$\Rightarrow \boxed{r = \frac{n^2(n+1)^2}{8} + \frac{n(n+1)(2n+1)}{12}}$$

So, this is the value of r returned by the function XYZ.

And worst case time complexity is $O(n^4)$ as the value of r depends on the 4 degree polynomial.