CS566 A2 HW2 | Jiankun Dong.

1) a). When it's $O(n^2)$.

$$\frac{T_1(10{,}000)}{T(1{,}000)} = 10^2 \quad \therefore T(10000) = \boxed{100 \text{ sec}}$$

b). When it's $O(n \log n)$

$$T_2(10.000) = (10 \cdot \log 10) \cdot 1 \text{ sec} = \boxed{10 \text{ sec.}}$$

~~2~~. Find $\Theta(f(x))$ for the following:

2) $a=2$, $b=2$, $f(n)=n^3$

$\therefore f(n)=O(n^3) \therefore c=3$

$\log_b a = \log_2 2 = 1 < 3$

$\therefore af(\frac{n}{b}) = 2f(\frac{n}{2}) \leq kn^3$.

$2\frac{n^3}{2^3} \leq kn^3$

pick $k=\frac{1}{4}$

$\therefore \boxed{T(n)=\Theta(n^3)}$

3) $T(n)=T(\frac{9}{10}n)+n$.

~~$a=1$ $b=\frac{10}{9}$ $f(n)=\Theta(n)$~~

~~$C_{crit}=\log_{\frac{10}{9}} 1 = \frac{\log 1}{\log \frac{10}{9}} = 0 < 1$~~

$\therefore T(n) = T((\frac{9}{10})^2 n)+n+\frac{9}{10}n$

$= n+\frac{9}{10}n+(\frac{9}{10})^2 n+(\frac{9}{10})^3 n+\cdots$

$= n\underbrace{(1+\frac{9}{10}+\frac{9^2}{10^2}+\cdots)}_{\text{constant}}$

$\therefore T(n) = \boxed{\Theta(n).}$

4) $T(n)=16T(\frac{n}{4})+n^2$

$a=16$ $b=4$ $C_{crit}=\log_4 16=2$

$f(n)=n^2=O(n^2) \therefore c=2=C_{crit}$

$\therefore T(n)=\boxed{\Theta(n^2\log n)}$

5) $T(n)=7T(\frac{n}{3})+n^2$

$a=7$ $b=3$ $C_{crit}=\log_3 7$

$f(n)=n^2=O(n^2)$ $c=2>\log_3 7$ $\therefore c>C_{crit}$.

$af(\frac{n}{b}) \leq kf(n)$

$7f(\frac{n}{3}) \leq kn^2$

$\frac{7}{9}n^2 \leq kn^2$

$\therefore k=\frac{7}{9}$

$\therefore T(n)=\Theta(f(n))=\boxed{\Theta(n^2)}$

6) $T(n)=2T(\frac{n}{4})+\sqrt{n}$

$a=2\ \ b=4\ \ Crit=\log_4 2=\frac{1}{2}$

$C=\frac{1}{2}=Crit.$

$f(n)=n^{\frac{1}{2}}=O(n^{\frac{1}{2}})$

$\therefore \boxed{T(n)=O(n^{\frac{1}{2}}\log n)}$

7) $T(n)=T(n-1)+n$

$T(n-1)=T(n-2)+(n-1)$

$\vdots$

$T(1)=T(0)+1$

$\therefore T(n)=T(0)+1+\cdots+n=T(0)+\frac{n(n+1)}{2}.$

$\because T(0)$ is constant

$\therefore \boxed{T(n)=O(\frac{n(n+1)}{2})=O(n^2)}$

~~a=1 b~~

8) $T(n)=2T(n-1)+1,\ T(1)=1$

$a=2\ \ b=1\ \ f(n)=O(1)$

$T(n-1)=2T(n-2)+1$

$T(2)=2T(1)+1$

~~$T(n)=O(2^{n-1}(T(1)))$~~

$\therefore O(T(n))=O(2T(n-1)+1)=O(2T(n-1))$

$=O(2^2T(n-2))$

$\vdots$

$=O(2^{n-1}T(1))$

$O(T(n))=\boxed{O(2^{n-1})}=\boxed{O(2^n)}$

9) Quick Sort w/ pivot using the middle entry

a) $(5,2,3,4,1)$

Δ pivot.

$\boxed{1}$ swap of 2, 4

when picking 3 as pivot. (middle index)

b) $(1,2,3,4,5)$

1 swap (1,5)

$\boxed{2}$ swap (2,4)

Done.

c) $(5,1,2,3,4)$

~~1 swap (1,3)~~

1 swap (2,4), pivot $\to$ 4

2 swap (1,4), pivot $\to$ 1

3 swap (1,3), pivot $\to$ 3

$\boxed{4}$ swap (2,1) ~~pivot=1~~

Done.

(a, b). Worst case : ~~Sorted Array~~

\# of comparisons $= (n-1) + (n-2) + \cdots + 1$

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

$$= \boxed{\frac{n^2-n}{2}} \quad (n \geq 2)$$

Average :

~~Total cases = n!~~

~~Total checks = $\frac{n(n-1)}{2}$ +~~

This algorithm always goes through the entire array (unsorted) to find the max of the unsorted part.

$\therefore$ average case = worst case = $\boxed{\frac{n^2-n}{2}}$