1. Accroding to regularity condition, we have

$$af(n/b) \leq cf(n) \qquad a \geq 1,\ b \geq 1,\ 0 < c < 1$$

then we can imply $\frac{a}{c} f(n/b) \leq f(n)$

$$\frac{a}{c} f(n) \leq f(bn)$$

Because we need to prove $f(n) = \Omega(n^{\log_b a + \varepsilon})$ stands.

therefore we have to find $n^{\log_b a}$.

$$a/c\, f(1) \leq f(b)$$

assume $n = b^k \Rightarrow k = \log_b n$

$$(a/c)^k f(1^k) \leq f(b^k)$$

$$f(n) \geq (a/c)^{\log_b n} f(1)$$

$$\because (a/c)^{\log_b n} = n^{\log_b a/c} > n^{\log_b a}$$

$\therefore n^{\log_b a + \varepsilon}$ stands. ε exist.

2. If list A is sorted decreasing order, then everytime we chose a pivot from the end. it will always devide the list in two parts and one is empty and another is $n-1$.

This is exactly the worst case of quikesort algorithm.

Therefore. $T(n) = T(n-1) + \theta(n) \in \theta(n^2)$.

3.(a) Because $a(n) \in \theta(g(n))$. we got $0 \leq C_1 g(n) \leq a(n) \leq C_2 g(n)$ where $C_1 . C_2 > 0$ and $b(n) \in O(g(n))$. we got $0 \leq b(n) \leq C_3 g(n)$. where $C_3 > 0$

Since both $a(n)$ and $b(n)$ are positive non-decreasing. $T(n) = a(n) + b(n)$ should be positive and non-decreasing as well. Also, since $a(n) \leq C_1 g(n)$ and $b(n) \geq 0 \Rightarrow T(n) \geq C_1 g(n)$. Since $a(n) \leq C_2 g(n)$ and $b(n) \leq C_3 g(n) \Rightarrow T(n) \leq (C_2 + C_3) g(n)$

Therefore. $0 \leq C_1 g(n) \leq T(n) \leq (C_2 + C_3) g(n)$. $C_1, C_2 + C_3 > 0$

$T(n) \in \theta(g(n))$.

(b) if $T(n) = \theta(n^{\log_b a}) + O(n^{\log_b a})$, then we have

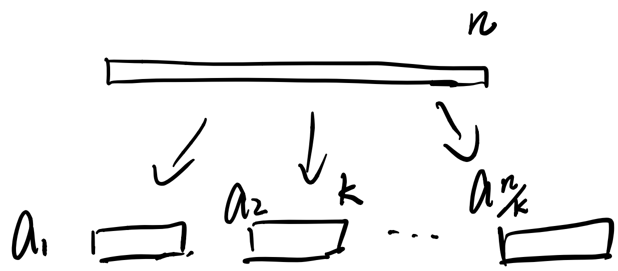
$$0 \le C_1 \cdot n^{\log_b a} \le T(n) \le (C_2 + C_3) \cdot n^{\log_b a} \quad \text{where } C_1, C_2, C_3 > 0$$

we can consider $C_2 + C_3$ as an positive constant.

Therefore. $0 \le C_1 n^{\log_b a} \le T(n) \le C_4 n^{\log_b a}$ where $C_4 = C_2 + C_3 > 0$

$T(n) \in \theta(n^{\log_b a})$.

4.



$a_1 < a_2 < \cdots < a_{n/k}$

(a) when elements in every sub-sequence is sorted from small to large, the comparison will be the lease.

$T(n) = \theta(n) + kT(\frac{n}{k})$

$T(n) = \frac{n}{k} T(k)\, k \lg k = n \lg k$.

$T(n) \in \Omega(n \lg k)$.

(b) Compare each element in A_n. The upper bound should be the worst case in which in each sub-sequence A_n, comparison was done $k!$ times.

$2^h \ge (k!)^{n/k}$

$h \ge \lg (k!)^{n/k}$

$= n/k \lg(k!)$

$\ge n/k \cdot \frac{k}{2} \lg \frac{k}{2}$

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

5. (1) Because A contains either 0 or 1, therefore no matter how situation is bad, it will always be a linear time. $T(n) \in \theta(n)$

(2) If the elements in A are 0 to 5 which are six different elements, the worst case would be completely sorted reversely.

$$T(n) = \theta(n) + n\lg 5 \in O(n).$$