Samed Nolbant
180104 2614

1) cutting(n, count) → T(n)

if n ≤ 1 $\Theta(1)$

return count $\Theta(1)$

else $\Theta(1)$

return cutting (n/2 + cutting +1) T(n/2)

$T(n) = \begin{cases} T(n) = T(n/2) \\ T(1) = 1 \end{cases}$ → according to master theorem

$a = 1$ $\quad n^{\log_b a} = n^{\log_2 1} = 1 = f(n)$

$b = 2$

$f(n) = 1$ $\quad T(n) = O(n^{\log_b a}, \log n) = O(\log n)$

Approach: After the first cutting done, new parts puts into machine together to get low price until the size equal to 1.

2)

Approach: Using divide and conquer sorting algorithm, we can find best and worst results.

Best case = $T(n) = 2T(\frac{n}{2}) + n$ → according to master theorem :

$a = 2$, $b = 2$ $\quad f(n) = n$ $\quad n^{\log_b a} = n$ so $T(n) = O(n \log n)$

they are equal $n^{\log_b a} \in \Theta(f(n))$

worst case = $T(n) = T(n-1) + n$, $T(1) = 0$

$$\left.\begin{aligned} T(n) &= \cancel{T(n-1)} + n \\ \cancel{T(n-1)} &= \cancel{T(n-2)} + n-1 \\ &\vdots \\ \cancel{T(2)} &= \underbrace{T(1)}_{0} + 2 \end{aligned}\right\} \quad \frac{n \cdot n+1}{2} - 1 = \frac{n^2-1}{2} = T(n) = O(n^2)$$

3) approach; using quick select algorithm problem can be solved easily to find k/th element.

worst case = $T(n) = T(n-1) + n$

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

$\vdots$

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

$T(1) = 1$

$T(n) = \frac{n \cdot n+1}{2}$

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

best case = best case occur when the pivot equal to k/th element. After a linear partition we found the result.

$T(n) = O(n)$

4) Approach: to find the reverse ordered pair, merge sort algorithm can be used. After the divides array like merge sort simple helper function used to counts reverse ordered elements.

$T(n) = 2T(\frac{n}{2}) + n$ acording to master theorem

$a = 2, \; b = 2, \; f(n) = n \qquad n^{\log_b a} = n \Rightarrow n \qquad n^{\log_b a} \in \Theta(f(n))$

$\underline{T(n) = (n \cdot \log n)}$

5) Approach: to find the exponentiation operation results, using brute force, in each iteration on power result calculated by multiplying base.

Analyze of brute force algorithm is $O(n)$ $n \Rightarrow$ power of base

$$\sum_{i=0}^{n} 1 = n = T(n) = \Theta(n)$$

using divide and conquer approach, subproblem is created by dividing the power by two until the power to zero.

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

according to master theorem: $a = 1$, $b = 2$, $f(n) = 1$

$$n^{\log_b a} = n^0 = 1 = f(n) \Rightarrow T(n) = O(\log n)$$