COMPUTER SCIENCE 3P03 (FALL 2020)

Algorithms

Instructor: Ke Qui

Assignment #2

Student Name: Sawyer Fenwick

Student Number: 6005011

1. A: $t(n) = 7t(n/2) + n^2$

B: $t(n) = a\,t(n/4) + n^2$

A

$a = 7$ $\quad a > b^K$

$b = 2$ $\quad \therefore t(n) = \Theta(n^{\log_b a}) = \Theta(n^{\log_2 7})$

$K = 2$

B

$a = ?$ $\quad$ if $a > b^K$, then $a > 4^2$, $a > 16$

$b = 4$ $\quad \therefore t(n) = \Theta(n^{\log_4 a})$

$K = 2$

$\log_4 a < \log_2 7$

$\frac{\log_2 a}{\log_2 4} < \log_2 7$

$\log_2 a < 2\log_2 7$

$\log_2 a < \log_2 7^2$

$a < 7^2$

$a < 49$

$\boxed{\therefore\ a = 48}$

2. Median (List, K)

```
    if $K = n/2$
        return List[$n/2$]
    else
        Split(List, List[$n/2$])  →  split the list into two lists List1 and List2.
                                     List1 contains elements smaller than List[$n/2$],
                                     List2 contains elements larger than List[$n/2$].
        if $K < n/2$
            Median(List1, K)
        else
            Median(List2, K)
```

$t(n) = t(n/2) + cn$

$a = 1, \quad b = 2, \quad K = 1, \quad p = 0$

$a < b^K$ and [illegible] $p \geq 0$

$\therefore t(n) = \Theta(n^1 \log^0 n) = \Theta(n)$

3. $t(n) = kt(n/3) + cn^2$

$= O(n^{\log_3 k})$

$\log_3 k < \log_2 7$

$\frac{\log_2 k}{\log_2 3} < \log_2 7$

$\log_2 k < \log_2 7 \cdot \log_2 3$

$\log_2 k < \log 21$

$\boxed{k < 21}$

$t(n) = kt(n/4) + cn^2$

$= O(n^{\log_4 k})$

$\log_4 k < \log_2 7$

$\frac{\log_2 k}{\log_2 4} < \log_2 7$

$\log_2 k < \log_2 7 \cdot \log_2 4$

$\log_2 k < \log_2 7^2$

$\boxed{k < 49}$

4. Since $A[1] < A[2] < \ldots < A[n]$ we can choose the middle of the array : $A[m]$.

If $A[m] = i$, stop.

Else if $A[m] < i$, [illegible] repeat this process on the right half of the array.
Else if $A[m] > i$, repeat this process on the left half of the array.

This is basically binary search which has recurrence $t(n) = t(n/2) + c$ which is $O(\log n)$ which is $o(n)$.

5. a) Weigh two of the coins. If the scale shows one is heavier, thats the counterfeit. If they weigh the same, the one not weighed must be the counterfeit.

b) $n = 3^K$, $K > 0$

Split the n coins into 3 groups of size $n/3$. Weigh two of the groups. If they are equal repeat the steps with the unweighed group. If they are not equal repeat the steps with the heavier group.

$t(3) = 1$

$t(n) = t(n/3) + t(3)$

$= t(n/3) + 1$

$t(n) = t(n/3) + 1$

$= t(n/3^2) + 1 + 1$

$= t(n/3^3) + 1 + 1 + 1$

$= t(3) + (K-1)$

$= 1 + K - 1$

$= K$

$= \log_3 n$

$3^K = n$

$\log_3 n = K$

6. n nuts, n bolts.

When sorted they will look like:

$$\begin{matrix} \text{nut}_1 \\ \text{bolt}_1 \end{matrix} < \begin{matrix} \text{nut}_2 \\ \text{bolt}_2 \end{matrix} < \dots < \begin{matrix} \text{nut}_n \\ \text{bolt}_n \end{matrix}$$

choose any nut.
use that nut to partition the other nuts into two groups, one that is smaller than the pivot nut and one that is bigger. Do the same thing ~~[illegible]~~ with the bolts.

Now we have 3 groups:

nuts < pivot	pivot nut	nuts > pivot
bolts < pivot	pivot bolt	bolts > pivot
g1	g2	g3

This takes $O(n)$ time.
Repeat the process on g1, g3. which acts as quicksort: $O(n \log n)$