③ 4.5 - 1

ⓐ $T(n) = 2T(n/4) + 1$

Here $a = 2$, $b = 4$, $n^{\log_b a} = \sqrt{n}$

$f(n) = 1 = O(n^{1/2 - \epsilon})$ for $\epsilon = 1/2$

$\therefore T(n) = \theta(\sqrt{n})$ (case I)

ⓑ $T(n) = 2T(n/4) + \sqrt{n}$

Here $a = 2$, $b = 4$, $n^{\log_b a} = \sqrt{n}$

$f(n) = \sqrt{n} = \theta(n^{\log_b a}) = \theta(\sqrt{n})$

$\therefore T(n) = \theta(\sqrt{n} \log n)$ (case II)

ⓒ $T(n) = 2T(n/4) + n$

Here $a = 2$, $b = 4$, $n^{\log_b a} = \sqrt{n}$

$f(n) = O(n) = \Omega(n^{1/2 + \epsilon})$ for $\epsilon = 1/2$

Now $af(n/b) \leq cf(n)$ has to be proved for some $c < 1$

$\Rightarrow 2 \cdot \frac{n}{4} \leq cn$

$\Rightarrow c \geq \frac{1}{2}$

so according to case III of master method

$T(n) = \theta(f(n)) = \theta(n)$

ⓓ $T(n) = 2T(n/4) + n^2$

Here $a = 2$, $b = 4$, $n^{\log_b a} = \sqrt{n}$

$f(n) = O(n^2) = \Omega(n^{1/2 + \epsilon})$ for $\epsilon = 1.5$

Now $2f\left(\frac{n}{4}\right) \leq cf(n)$

$\Rightarrow 2 \cdot \frac{n^2}{16} \leq cn^2$

$\Rightarrow c \geq \frac{1}{8}$

$\therefore T(n) = \theta(n^2)$ (case III).

$\boxed{4.5\text{-}4}$ $T(n) = 4T(n/2) + n^2 \log n$

Here $a = 4$, $b = 2$, $n^{\log_b a} = n^2$

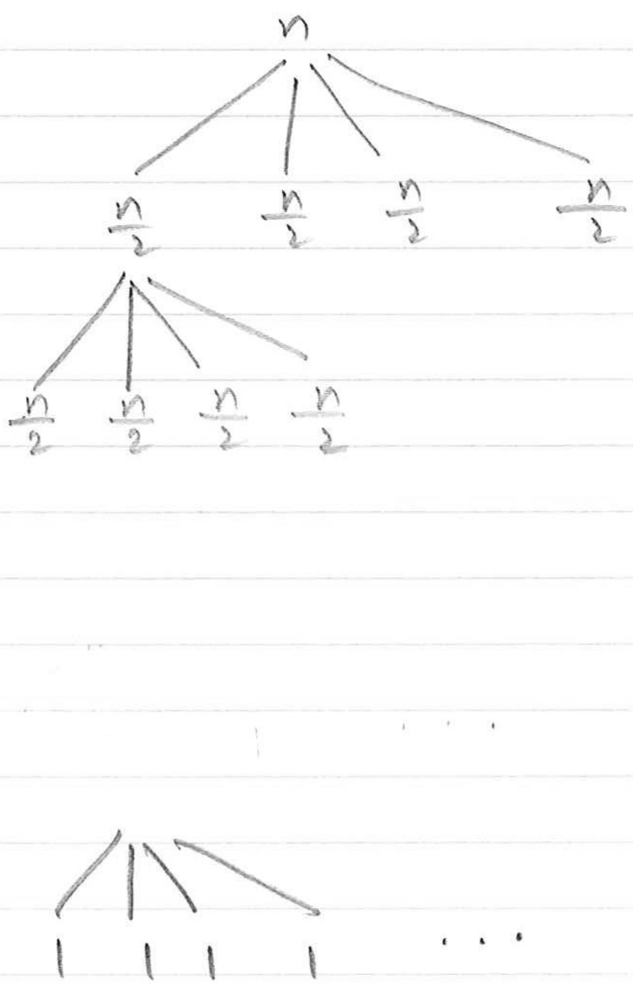
$f(n) = n^2 \log n = \Omega(n^{\log_b a + \epsilon})$

$\Rightarrow n^2 \log n \geq c \cdot n^2 \cdot n^\epsilon$

$\Rightarrow \log n \geq c n^\epsilon$

which is false for any $\epsilon > 0$ and $c > 0$ because logarithmic growth can't be greater than polynomial growth. so we can't use master method here.

for i	Running Time	cost
0		$n^2 \log n$
1		$4\left(\frac{n}{2}\right)^2 \log \frac{n}{2}$
2		$4^2\left(\frac{n}{2^2}\right)^2 \log \frac{n}{2^2}$
⋮		
i		$4^i\left(\frac{n}{2^i}\right)^2 \log \frac{n}{2^i}$
⋮		
$\log n$		



From the recursion tree

$$\text{cost} = \sum_{i=0}^{\log n} 4^i \left(\frac{n}{2^i}\right)^2 \log \frac{n}{2^i}$$

$$= \sum_{i=0}^{\log n} n^2 \log \frac{n}{2^i}$$

$$= \sum_{i=0}^{\log n} \left(n^2 \log n - n^2 \log 2^i\right)$$

$$= n^2 \log n \cdot \log n - \sum_{i=0}^{\log n} n^2 \log 2^i$$

$$= n^2 (\log n)^2 - n^2 \sum_{i=0}^{\log n} i$$

$$= n^2 (\log n)^2 - n^2 \cdot \frac{\log n \, (\log n + 1)}{2}$$

$$= O(n^2 (\log n)^2).$$

(4) We want to find the number of subproblem x such that running time

$$T(n) = xT\left(\frac{n}{3}\right) + O(\log n)$$

Here $a = x$, $b = 3$ $f(n) = \log n$

As $T(n) = O(n^2)$, $f(n) = \log n$ is not going to dominate $T(n)$. Its bound is going to be determined by $O(n^2)$.

So $n^{\log_b a} = O(n^2)$ (master method applied)

$$\therefore n^{\log_3 x} = O(n^2)$$

$$\Rightarrow \log_3 x < 2$$

$$\Rightarrow x < 9$$

So maximum 8 subproblems of size $n/3$ can be taken.

① [a] 4 marks

Let X be the outcome of a single roll of a fair 6 face die.

so, $X \in \{1, 2, 3, 4, 5, 6\}$

and $P(X=1) = P(X=2) = P(X=3) = P(x=4) = P(x=5)$
$= P(x=6) = \frac{1}{6}$

Now $E[X] = \sum_{x_i} x_i \, P(x_i)$

$= 1 \cdot \frac{1}{6} + 2 \cdot \frac{1}{6} + 3 \cdot \frac{1}{6} + 4 \cdot \frac{1}{6} + 5 \cdot \frac{1}{6} + 6 \cdot \frac{1}{6}$

$= \frac{7}{2}$

[b] 4 marks

Let X_i be the indicator random variable counting when face i is up, where $i = 1$ to 6

and let X be the random variable for sum of n die rolls.

$$\text{Now } E[X] = \sum_{i=1}^{6} E[X_i] \cdot i$$

$$= \sum_{i=1}^{6} \frac{n}{6} \cdot i \quad \left[E[X_i] = \frac{n}{6} \right]$$

$$= \frac{n}{6} \cdot \sum_{i=1}^{6} i$$

$$= \frac{n}{6} \cdot \frac{6(6+1)}{2}$$

$$= \frac{7}{2} n$$

(2) 5 marks

let x be the random variable which denotes our gain

so $X(11) = 10$

$X(\alpha 1) = X(1\alpha) = 1$ [where α is any value between 2-6]

$X(\alpha\alpha) = -0.5$

Again $P(X=10) = \frac{1}{6} \times \frac{1}{6} = \frac{1}{36}$

$P(X=1) = \frac{1}{6}\cdot\frac{5}{6} + \frac{1}{6}\cdot\frac{5}{6} = \frac{10}{36}$

$P(X=-0.5) = 1 - \left(\frac{1}{36} + \frac{10}{36}\right) = \frac{25}{36}$

Expected value, $E[X] = P(X=10) \times 10 +$
$P(X=1) \times 1 +$
$P(X=-0.5) \times -0.5$

$= \frac{1}{36} \times 10 + \frac{10}{36} \times 1 + \frac{25}{36}(-0.5)$

$= \frac{5}{24}$