Q:1 A) merge sort

A: 3, 41, 52, 26, 38, 57, 9, 49

⇒ Divide 3, 41, 52 26 38 57 9 49

D:

3 41 51 26 38 57 9 49

3 41 51 26 58 57 9, 49

3 41 51 26 58 57 9 49

Merge Array

3 41 | 26 52 | 38 57 | 9 49

3 26 41 52 | ~~38 57 9~~ 9 38 49 57

3 9 26 38 41 49 52 57

final Array ⇒

3, 9, 26, 38, 41, 49, 52, 57

Q-1

B →

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

Q ≥ 2

A) $2^{n+1} = O(2^n)$

for

c and $n > 0$

$0 \le 2^{n+1} \le c2^n$ (for $n \geq n_0$)

$2^{n+1} = 2 \times 2^n$

$c = 2 \quad n \geq 1$

so $2^{n+1} = O(2^n)$

B

$$2^{2n} = O(2^n)$$

for c and $n > 0$

$$0 \le 2^{2n} \le c2^n$$

$$2^{2 1} < 2 2^1$$

$$2^n \times \cancel{2^n} \le c \times \cancel{2^n}$$

$$2^n \not\le c$$

$$2^{2n} \ne O(2n)$$

if $T(n)$ is $O(\lg n)$

then

$T(n) \le c \lg n$ for constant c and all $n > n_0$

so

$T(n/2) \le c \lg(n/2)$

$T(n) = T(\lceil n/2 \rceil) + 1$

$\le c \lg(n/2) + 1$

$\le c(\lg n - \lg 2) + 1$

$\le c \lg n - c + 1$

$\le c \lg n$

$c \ge 1$

~~so $(c \ge 1)$~~

$T(n) = 4T(n/2) + n^2$

When $a=4$ and $b=2$

$f(n) = n^2$

$= n^{\log_b a} = n^1 = n^2$

Case 2 =

$T(n) = \Theta(n^2 \log n)$

Q:m = B