CS2223 D Term 2020 Quiz 9

(1 point) Question 1: "My brain is open. . . ."
I pledge that I am taking this quiz on my own, with help from no one else and no notes:

$$T(n) = aT(n/b) + f(n) \quad \text{for } n = b^k, \quad k = 1, 2, \ldots$$
$$T(1) = c,$$

where $a \geq 1$, $b \geq 2$, $c > 0$. If $f(n) \in \Theta(n^d)$ where $d \geq 0$, then

$$T(n) \in \begin{cases} \Theta(n^d) & \text{if } a < b^d, \\ \Theta(n^d \log n) & \text{if } a = b^d, \\ \Theta(n^{\log_b a}) & \text{if } a > b^d. \end{cases}$$

(3 points) Question 2: Use the Master Theorem (above) to find the order of growth of

$$T(n) = 9T(n/3) + 4n^2.$$

a.) $\Theta(1)$
b.) $\Theta(n)$
c.) $\Theta(n \log n)$
d.) $\Theta(n^2)$
e.) $\Theta(n^2 \log n)$

(3 points) Question 3: Use the Master Theorem to find the running time of a Divide-and-Conquer algorithm that divides a problem into *four* subproblems of *one half* size in *constant* time.

a.) $\Theta(1)$
b.) $\Theta(n)$
c.) $\Theta(n \log n)$
d.) $\Theta(n^2)$
e.) $\Theta(n^2 \log n)$

(3 points) Question 4: Reason backwards from the Master Theorem on MERGESORT. We know that MERGESORT is a divide-and-conquer algorithm that creates two sublists–each of half size–and has $\Theta(n \log n)$ time complexity. What does that mean for the "housekeeping" operation $f(n)$ at each division/recombination stage?

a.) No work is required in "housekeeping", i.e. there is no d, hence $f(n) = 0$.
b.) The amount of "housekeeping" work does not depend on n, i.e. $d = 0$, hence $f(n) \in \Theta(1)$.
c.) Linear effort is required in "housekeeping", i.e. $d = 1$, hence $f(n) \in \Theta(n)$.
d.) The effort required in "housekeeping" is $f(n) \in \Theta(n \log n)$.
e.) Quadratic effort is required in "housekeeping", i.e. $d = 2$, hence $f(n) \in \Theta(n^2)$.

(1 point) Bonus Question: In the middle case above, $\Theta(n^d \log n)$, the base of the logarithm is omitted. Why?

a.) It is understood to be Euler's number, e, the base of the natural logarithm.
b.) It is understood to be 2—this is a computer science result, after all.
c.) It is understood to be 10, the base of the common logarithm.
d.) It is intentionally generic: $\log_{c_1} n = C \log_{c_2} n$ for positive n, $c_1 > 1$ and $c_2 > 1$, so $\Theta(n \log_{c_1} n) = \Theta(n \log_{c_2} n)$.
e.) It's a typographical error. It should be $\log_b$ as in the case below it.