Divide-and-Conquer Formulas

Divide-and-conquer recursion leads to running times that can be expressed using recurrences like

$$f(n) = af(n/b) + g(n) \ ,$$

where $a \geq 1$ and $b > 1$. (Roughly, a is the number of divisions, b measures the size of the divisions, and $g(n)$ is the overhead from the parts of the algorithm not involved in the recursive calls.)

The tools needed to analyze this kind of recurrence are **unraveling** and **geometric sums**. We dealt with the cases where $g(n) = Cn^d = O(n^d)$ for constants C and d. This provided big O estimates whenever g is a polynomial. An earlier handout developed the following formulas to handle these situations, and you are responsible for the information there too, but the information in the box below will be provided on your exam:

$$\text{If } f(n) = af(n/b) + g(n), \text{ and } g(n) = Cn^d, \text{ then } f(n) = \begin{cases} O(\log n) & \text{if } d = 0 \text{ and } a = 1 \\ O(n^{\log_b(a)}) & \text{if } d = 0 \text{ and } a > 1 \\ O(n^d) & \text{if } d > 0 \text{ and } d > \log_b(a) \\ O(n^d \log n) & \text{if } d > 0 \text{ and } d = \log_b(a) \\ O(n^{\log_b(a)}) & \text{if } d > 0 \text{ and } d < \log_b(a) \end{cases}$$