CS2205: Theory of Computation
Problem Class XII

Here are some laws and equations that you will use in some of these exercises.

$$\begin{aligned}
b^0 &= 1 & \log_b 1 &= 0\\
b^{-c} &= 1/b^c & \log_b b &= 1\\
b^{c+d} &= b^c \times b^d & \log_b cd &= \log_b c + \log_b d\\
b^{c-d} &= b^c/b^d & \log_b c/d &= \log_b c - \log_b d\\
b^{cd} &= (b^c)^d & \log_b a^c &= c \log_b a\\
b^{\log_c a} &= a^{\log_c b} & \log_b a &= (\log_c a)/\log_c b
\end{aligned}$$

The roots of a quadratic equation $an^2 + bn + c$ are given by

$$n = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

1. Show that the following identities hold.

   (a) $\log(2n \log n) = 1 + \log n + \log(\log n)$

   (b) $\log(n/2) = \log n - 1$

   (c) $\log \sqrt{n} = \frac{\log n}{2}$

   (d) $\log(\log \sqrt{n}) = \log(\log n) - 1$

   (e) $\log_4 n = \frac{\log n}{2}$

   (f) $\log_b b^n = n$

   (g) $b^{\log_b n} = n$

   (h) $2^{2 \log n} = n^2$

   (i) $4^n = 2^{2n}$

   (j) $n^2 2^{3 \log n} = n^5$

Observation. (f) and (g) are worth keeping in mind.
(f) says $\log_b b^n = n$, e.g. $\log 2^n = n$, $\log_{10} 10^n = n$, etc.
When solving equations where one side is b^n, it can be useful to take logs to the base b of both sides, which turns b^n into $\log_b b^n$, which simplifies to n.
(g) says $b^{\log_b n} = n$, e.g. $2^{\log n} = n$, $10^{\log_{10} n} = n$, etc.
When solving equations where one side is $\log_b n$, it can be useful to make both sides an exponent of b, which turns $\log_b n$ into $b^{\log_b n}$, which simplifies to n.

The *wrong* way of answering the remaining questions is by trial-and-error, i.e. by repeatedly guessing values for n. The *right* way is by solving equations.

---

2. Suppose algorithm A's worst-case running time is $20n + 3$ milliseconds, and algorithm B's is $18n + 5$ milliseconds. For what values of n, does algorithm A take

   (a) less time than,

   (b) the same time as, and

   (c) more time than

   algorithm B? Show your working.

3. Repeat Question 2 for algorithms C and D. C takes $n^2 + 3n$ milliseconds and D takes $7n + 5$ milliseconds.

4. Repeat Question 2 for algorithms E and F. E takes $1000n^4$ milliseconds and F takes n^{10} milliseconds.

5. How large a problem can algorithm A (above) solve in

   (a) 10^3 milliseconds;

   (b) 10^6 milliseconds; and

   (c) 10^9 milliseconds.

6. Repeat Question 5 for algorithm C.

7. Repeat Question 5 for algorithm E.