1. $\log a_1 + \log a_2 + \cdots + \log a_n + \log S$

2. Suppose each a_i is $O(2^n)$ and S is also $O(2^n)$; the input size is $O(n^2)$
   When $S = O(n^9)$, the running time is polynomial in the input size

3. D-Fact-1 can be solved in polynomial time as it is same as primality testing

4. D-Fact-2 is at least as hard as D-Fact-1
   D-Fact-2 "captures" D-Fact-1 (by appropriately setting x, y)

5. Run D-Fact-2 on $(2, \sqrt{M})$. If it returns false, there is no factorization.
   Otherwise, let x and y equal 2 and $\sqrt{M}$ respectively.
   While $x < y$, run D-Fact-2 on both halves of the range $[x, y]$, then set x, y equal to whichever range returned true (arbitrarily if both returned true).
   Return x.

   We run D-Fact-2 $\log \sqrt{M}$ times, which is $O(n^2 \log \sqrt{M})$.
   Since M is $O(2^n)$ (since it's n-bit), this reduces to $O(n^2 * \frac{1}{2} \log 2^n) = O(n^2 * \frac{1}{2} * n * \log 2) = O(n^3)$.