Homework 2 Solutions

2.1 \( 2/N, 37, \sqrt{N}, N, N \log \log N, N \log N, N \log(\sqrt{N}), N \log^2 N, N^{1.5}, N^2, N^3, \log N, N^4, 2^{N^2}, 2^N. \)

\( N \log N \) and \( N \log(\sqrt{N}) \) grow at the same rate.

2.2 (a) True.
(b) False. A counterexample is \( T_1(N) = 2N, T_2(N) = N, \) and \( f(N) = N. \)
(c) False. A counterexample is \( T_1(N) = N^2, T_2(N) = N, \) and \( f(N) = N^2. \)
(d) False. The same counterexample as in part (c) applies.

2.5 Let \( f(N) = 1 \) when \( N \) is even, and \( N \) when \( N \) is odd. Likewise, let \( g(N) = 1 \) when \( N \) is odd, and \( N \) when \( N \) is even. Then the ratio \( f(N)/g(N) \) oscillates between 0 and \( \infty. \)

2.6 (a) \( 2^{2^N} \)
(b) \( O(\log \log D) \)

2.7 For all these programs, the following analysis will agree with a simulation:

(I) The running time is \( O(N). \)
(II) The running time is \( O(N^2). \)
(III) The running time is \( O(N^3). \)
(IV) The running time is \( O(N^4). \)
(V) \( j \) can be as large as \( i^2, \) which could be as large as \( N^2. k \) can be as large as \( j, \) which is \( N^3. \) The running time is thus proportional to \( N \cdot N^2 \cdot N^2, \) which is \( O(N^5). \)
(VI) The if statement is executed at most \( N^2 \) times, by previous arguments, but it is true only \( O(N^2) \) times (because it is true exactly \( i \) times for each \( i. \) Thus the innermost loop is only executed \( O(N^2) \) times. Each time through, it takes \( O(j^2) = O(N^2) \) time, for a total of \( O(N^4). \) This is an example where multiplying loop sizes can occasionally give an overestimate.

2.12 (a) 12000 times as large a problem, or input size 1,200,000.
(b) input size of approximately 425,000.
(c) \( \sqrt{12000} \) times as large a problem, or input size 10,954.
(d) 12000\(^{1/2} \) times as large a problem, or input size 2,289.
2.12.

Input size 100, running time 0.5 ms

1 min = 60 sec = 60 \times 1000 = 60,000 ms

(a) Linear time algorithm

\[ \frac{N}{100} = \frac{60,000}{0.5} = 120,000 \]

\[ N = 12,000,000 = 12 \times 10^6 \]

(b) \(O(N \log N)\) time algorithm

\[ \frac{N \log N}{100 \log 100} = \frac{60,000}{0.5} = 120,000 \]

Find \(N\) s.t. \(N \log N = 12 \times 10^6 \times \log 100\)

Since \(\log 100 \approx 7\)

\[ N \log N = 12 \times 10^6 \times 7 = 84 \times 10^6 \]

Resume \(N = 2^{22} \approx 4,000,000\)

\[ \log N = 22 \]

Then \(N \log N = 22 \times 4 \times 10^6 = 88 \times 10^6\)

Thus, \(N \approx 4,000,000\)
(c) \[ \frac{N^2}{100^2} = \frac{60,000}{0.5} = 120,000 \]

\[ N^2 = 12 \times 10^8 \]

\[ N = (12 \times 10^8)^{\frac{1}{2}} \approx 3,4641 \]

\( \sqrt{120,000} \) times as large as a problem

\[ \sqrt[3]{120,000} \] times as large as a problem

(a) \[ \frac{N^3}{(100)^3} = \frac{60,000}{0.5} = 120,000 \]

\[ N^3 = 12 \times 10^{10} \]

\[ N = (12 \times 10^{10})^{\frac{1}{3}} = (120)^{\frac{1}{3}} \times 10^3 \approx 5 \times 10^3 = 5,000 \]

\( \sqrt[3]{120,000} \) times as large as a problem
2.22 Compute $x^2, x^4, x^8, x^{10}, x^{20}, x^{40}, x^{60}$, and $x^{62}$.

2.31 No. If $low = 1$, $high = 2$, then $mid = 1$, and the recursive call does not make progress.