2-10 FURTHER READING MATERIALS

Lower bound theories have continuously attracted researchers. Some recently published papers on this subject include: Dobkin and Lipton (1979); Edwards and Elphick (1983); Frederickson (1984); Fredman (1981); Grandjean (1988); Hasham and Sack (1987); Hayward (1987); John (1988); Karp (1972); McDiarmid (1988); Mehlhorn, Naher and Alt (1988); Moran, Snir and Manber (1985); Nakayama, Nishizeki and Saito (1985); Rangan (1983); Traub and Wozniakowski (1984); Yao (1981); and Yao (1985).

For some very interesting newly published papers, consult Berman, Karpinski, Larmore, Plandowski and Rytter (2002); Blazewicz and Kasprzak (2003); Bodlaender, Downey, Fellows and Wareham (1995); Boldi and Vigna (1999); Bonizzoni and Vedova (2001); Bryant (1999); Cole (1994); Cole and Hariharan (1997); Cole, Farach-Colton, Hariharan, Przytycka and Thorup (2000); Cole, Hariharan, Paterson and Zwick (1995); Crescenzi, Goldman, Papadimitriou, Piccolboni and Yannakakis (1998); Darve (2000); Day (1987); Decatur, Goldreich and Ron (1999); Demri and Schnoebelen (2002); Downey, Fellows, Vardy and Whittle (1999); Hasegawa and Horai (1991); Hoang and Thierauf (2003); Jerrum (1985); Juedes and Lutz (1995); Kannan, Lawler and Warnow (1996); Kaplan and Shamir (1994); Kontogiannis (2002); Leoncini, Manzini and Margara (1999); Maes (1990); Maier (1978); Marion (2003); Martinez and Roura (2001); Matousek (1991); Naor and Ruah (2001); Novak and Wozniakowski (2000); Owolabi and McGregor (1988); Pacholski, Szwast and Tendera (2000); and Peleg and Rubinovich (2000); and Ponzio, Radhakrishnan and Venkatesh (2001).

Exercises

2.1 Give the best, worst and average numbers of exchanges needed in bubble sort, whose definition can be found in almost any textbook on algorithms. The best case and worst case analyses are trivial. The average case analysis can be done by the following process:

(1) Define inversion of a permutation. Let $a_1, a_2, \ldots, a_n$ be a permutation of the set $(1, 2, \ldots, n)$. If $i < j$ and $a_j < a_i$, then

(a_i, a_j) is called an inversion of this permutation. For instance, (3, 2) (3, 1) (2, 1) (4, 1) are all inversions of the permutation (3, 2, 4, 1).

(2) Find out the relationship between the probability that a given permutation has exactly k inversions and the number of permutations of n elements having exactly k inversions.

(3) Using induction, prove that the average number of exchanges needed for bubble sort is $n(n-1)/4$.

2.2 Write a program for bubble sort. Run an experiment to convince yourself that the average performance of it is indeed $O(n^2)$.

2.3 Find the algorithm of Ford-Johnson algorithm for sorting, which is reproduced in many textbooks in algorithms. It was shown that this algorithm is optimal for $n \leq 12$. Implement this algorithm on a computer and compare it with any other sorting algorithm. Do you like this algorithm? If not, try to determine what is wrong with the analysis.

2.4 Show that to sort five numbers, we need at least seven comparisons. Then demonstrate that the Ford–Johnson algorithm does achieve this lower bound.

2.5 Show that to find the largest number in a list of n numbers requires at least $n - 1$ comparisons.

2.6 Show that to find the second largest one of a list of n numbers, we need at least $n - 2 + \lceil \log n \rceil$ comparisons.

Hint: We cannot determine the second largest element without having determined the largest element. Thus,the analysis can be done by the following:

(1) Show that at least $n - 1$ comparisons are necessary to find the largest element.

(2) Show that there is always some sequence of comparisons which forces the second largest one to be found in $\lceil \log n \rceil - 1$ additional comparisons.

2.7 Show that if $T(n) = aT\left(\frac{n}{b}\right) + n^c$, then for n a power of b and

$$T(1) = k,\ T(n) = ka^{\log_b n} + n^c\left(\frac{b^c}{a - b^c}\right)\left(\left(\frac{a}{b^c}\right)^{\log_b n} - 1\right).$$

2.8 Show that if $T(n) = \sqrt{n}T(\sqrt{n}) + n$, $T(m) = k$ and $m = n^{1/2^i}$, then $T(n) = kn^{(2^i-1)/2^i} + in$.

2.9 Read Theorem 10.5 of Horowitz and Sahni 1978. The proof of this theorem gives a good method to find a lower bound.

2.10 Show that binary search is optimal for all searching algorithm performing comparisons only.

2.11 Given the following pairs of functions, what is the smallest value of n such that the first function is larger than the second one.

(a) 2^n, $2n^2$.

(b) $n^{1.5}$, $2n \log_2 n$.

(c) n^3, $5n^{2.81}$.

2.12 Is $\Omega(n \log n)$ time a lower bound for the problem of sorting n integers ranging from 1 to C, where C is a constant? Why?