BU CLA CS 530: Analysis of Algorithms

Spring 1999

Problem Set 1

Due Thursday January 21, by 5:00 pm, in my mailbox in the main office of the CS department. Late submissions not accepted.

---

1. Exercise 6.3-2, page 114 in [CLR]. (You may need to read Section 6.3 first; even if you know the material already, it is useful to use the notation proposed in the book.)

2. (Review exercise.) Consider the following divide-and-conquer algorithm to find the minimum element in an array $A[1..n]$ of $n \geqslant 1$ integers. The main body of the algorithm makes a single call to the recursive procedure $\textsc{Minimum}(A, 1, n)$. The parameters p and r range over $\{1, \ldots, n\}$.

```
MINIMUM(A, p, r)
1.   if  r − p ⩽ 1
2.   then  return min{A[p], A[r]}
3.   else  x ← MINIMUM(A, p, ⌊(p + r)/2⌋)
4.         y ← MINIMUM(A, 1 + ⌊(p + r)/2⌋, r)
5.         return min{x, y}
```

   a. Write down a recurrence for each of the best-case, the average-case, and the worst-case, number of comparisons (between elements in the array A) used by $\textsc{Minimum}(A, 1, n)$.

   b. Solve the recurrences of part a. (You can assume that n is a power of 2.)

   c. Which is a better algorithm: the recursive (divide-and-conquer) algorithm given here, or the iterative algorithm at the top of page 186 in [CLR]?

3. Exercise 10.1-1, page 187 in [CLR].

4. Exercise 1 in Handout 2.

Assaf Kfoury
Created: 99.01.13
Modified: 99.01.14