Chapter 5 Exercises

Brady Field

25 May 2016

Partners in crime

Thom Allen - helped with max algorithm and classifying growths

Exercise 1 on page 174

a. Write a pseudocode for a divide-and-conquer algorithm for finding a position of the largest element in an array of n numbers.

b. What will be your algorithm's output for arrays with several elements of the largest value?

c. Set up and solve a recurrence relation for the number of key comparisons made by your algorithm.

d. How does this algorithm compare with the brute-force algorithm for this problem?

Solution

a.

```
maxPos(list, l, r)
IF l == r
   RETURN l
SET mid to (l + r) / 2
SET left to maxPos(list,l,mid)
SET right to maxPos(list,mid+1,r)
IF list of left > list of right
   RETURN left
ELSE
   RETURN right
```

b. The largest array index of all possible max values
c. assuming not ints...
$T(n) = 2T(n/2) + 2^0, T(1) = 0$
$2T(n/2) = 2^2T(n/2^2) + 2^1$
$2^2T(n/2^2) = 2^3T(n/2^3) + 2^2$
...
$2^{k-1}T(n/2^{k-1}) = 2^kT(n/2^k) + 2^{k-1}$
$T(n) = 2^kT(1) + \sum_{n=0}^{k-1} 2^i$
$T(n) = 2^k(0) + 2^k - 1, 2^k = n$
$T(n) = n - 1$
d. They are the same. Both need to check every position exactly once

Exercise 5 on page 175

Find the order of growth for solutions of the following recurrences.
a. $T(n) = 4T(n/2) + n, T(1) = 1$
b. $T(n) = 4T(n/2) + n^2, T(1) = 1$
c. $T(n) = 4T(n/2) + n^3, T(1) = 1$
NOTE: No exercises are assigned for the remaining sections 5.2-5.5.

Solution

$\Theta(n^d)$ if a $< b^d$
$\Theta(n^d logn)$ if a $= b^d$
$\Theta(n^{\log_b a})$ if a $> b^d$
a. a = 4, b = 2, d = 1, $4 > 2^1$, $\Theta(n^{\log_b a})$
b. a = 4, b = 2, d = 2, $4 = 2^2$, $\Theta(n^d logn)$
c. a = 4, b = 2, d = 3, $4 < 2^3$, $\Theta(n^d)$