Lab W1D3

Question 1. Write an algorithm

beautiful(A, n)

Input : An integer array with n elements

such that the best case running time is equal to the worst case running time. Write the algorithm and give your analysis to justify your claim.

Question 2. Order them based on their complexity.

2^n , 2^(2n), 2^(n + 1), 2^(2^n) (Note: ^ stands for exponent operation. Example: 2^n = 2^n)

Question 3. Mention one algorithm you know for each of the time complexities listed.

O(1), O(log n), O(n), O(n log n), O(n^2), O(n^3), O(2^n)

Example. O(n log n) : Quicksort

Question 4. Apply Master Theorem and determine the time complexity of

fib(n) shown in slide 48.

Question 5

Solve the recurrence

T(1) = 1

T(n) = 2T(n/2) + c

without using the Master Theorem.

Hint: Assume n = 2^k.

Follow the steps I did just before lunch break. Only difference is I did for 32. You are doing it for any perfect power of 2.

Question 5. Practice Master theorem. It is a very important result in Analysis of algorithms. There are many resources on the internet. Show three different examples covering three possible cases. Show your detailed work.

Have fun!