Assignment 1

Some Practice Questions on Complexity and Recurrence Relation

1. Is $2^{n+1} = O(2^n)$? Is $2^{2n} = O(2^n)$?
2. Solve the following recurrences. Assume that T (n) is constant for $n \leq 2$. Make your bounds as tight as possible, and justify your answers:
   a) $T(n) = 2T(n/2) + n^3$
   b) $T(n) = T(9n/10) + n$
   c) $T(n) = 16T(n/4) + n^2$
   d) $T(n) = 2T(n/4) + \sqrt{n}$
   e) $T(n) = T(\sqrt{n}) + 1$
   f) $T(n) = 7T(n/2) + n^2$
   g) $T(n) = 4T(n/2) + n^2 \sqrt{n}$
   h) $T(n) = T(n-2) + 2 \lg n$
   i) $T(n) = 3T(n/2) + n \lg n$
   j) $T(n) = 3T(n/3 + 5) + n/2$
   k) $T(n) = 2T(n/2) + n/\lg n$
   l) $T(n) = T(n-1) + 1/n$
   m) $T(n) = \sqrt{n}T(\sqrt{n}) + n$

3. An array A[1........n] contains all the integers from 0 to n except 1. It would be easy to determine the missing integer in O(n) time by passing an auxiliary array B[0..........n] to record which numbers appear in A. In this problem, however, we cannot access an entire integer in A with a single operation. The elements of A are represented in binary, and the only operation we can use to access them is "fetch the jth bit of A[i]," which takes constant time. Show that if we use only this operation, we can still determine the missing integer in O(n) time.