1. (20 points) Order the following functions by the most accurate asymptotic notation you learned in this course. \( n!, n^2 + \sqrt{n} \log^{10} n, n^{1/3}, \log^{100} n, n^3, 2^n, 10^{\sqrt{n}}, 2^{\log n}, 2^{2\log n}, 2^{\sqrt{\log n}}, 128, 128n \). Indicate which functions grow at the same rate (all logarithms are base 2). For example, if you are asked to order \( n, 2^n \) and \( 2^n 2 \), then you answer should be \("n = \Theta(2^n), 2^n = o(2^n 2)"\).

2. (20 points) Solve the following recurrence equations, expressing the answer in Big-Oh notation. Assume that \( T(n) \) is constant for sufficiently small \( n \).

   (a) \( T(n) = T(n/2) + 100 \)
   (b) \( T(n) = 8T(n/2) + n^2 \)
   (c) \( T(n) = 8T(n/2) + n^3 \)
   (d) \( T(n) = 8T(n/2) + n^4 \)
   (e) \( T(n) = T(n - 1) + \log n \)
   (f) \( T(n) = T(n - 3) + n \)

3. (20 points) Suppose there is a set \( S \) of \( d \)-dimensional vectors, \( |S| = n \). Suppose further that there are two vectors in \( S \) that are identical. If comparing any two \( d \)-dimensional vectors takes \( O(d) \) time, design an efficient algorithm that finds the two identical vectors.

4. (20 points) Suppose your team is working on a big project which involves analyzing two very large separate databases in the cloud. Each database contains \( n \) numerical values (so there are \( 2n \) numbers in total). Your job in the team is to determine the median of this set of \( 2n \) values. The only way you can access the databases is by sending queries to the databases. In each query you can specify an integer \( i \) to one of the two databases, and that database will return the \( i \)th smallest value in its record. Each query costs 1 dollar. If the size of each database is \( 10^{16} \), how much money would you request from your project manager? (Asking for more money than necessary would be treated as embezzlement and may result in firing from the job:-)

5. (20 points) You are given a set of \( n \) integers. Give an \( O(n^2) \) algorithm to decide if there exist three numbers \( a, b \) and \( c \) in the set such that \( a + b = c \) (Hint: sort the numbers first).