Assignment2
279/377
2020 spring

1.

Let

$$p(n) = \sum_{i=0}^{d} a_i n^i ,$$

where $a_d > 0$, be a degree-d polynomial in n, and let k be a constant. Use the definitions of the big O, Ω, Θ notations to prove the following properties:

a. If $k \geq d$, then $p(n) = O(n^k)$.

b. If $k \leq d$, then $p(n) = \Omega(n^k)$.

c. If $k = d$, then $p(n) = \Theta(n^k)$.

2.

Show that for any real constants a and b, where $b > 0$,

$$(n + a)^b = \Theta(n^b) .$$

3. Implement the brute force algorithm of the max subarray problem. You algorithm should have running time of $O(n^2)$

4. Implement the linear algorithm for max subarray problem explained in lecture.

5. Use divide-and-conquer technique to find the max value of an input integer array. Assuming we divide at the middle of the array to create two subproblems each time.
   - Write the pseudocode of the algorithm.
   - Write the recursive running time equation.
   - Find out the running time from this recursive equation.
   - What is a better way to divide so that the running time is lower?