1. List the following Big-O time complexities from the fastest to the slowest:
O(nlogn), O(n2), O(1), O(2n), O(logn), O(n)

constant: O(1)
logarithmic: O(logn)
linear: O(n)
linearithmic: O(nlogn)
polynomial: O(n^2)
exponential: O(2^n)
factorial: O(N!)

2. Does Big-O complexity describe an upper bound, an average bound, or a lower bound

Big-O complexity describes an upper bound on the growth rate of an algorithm or function.

3. $n^3 + 16n^2 + 2n + 5$ = O(n^3)
4. 4logn+ 2n2+ nlogn = O(n^2)

5. O(n)
6. O(log n)
7. O(n^2)