**Increasing order of complexity:**

1. **Constant time complexity – O(1) -- BEST**
2. **Logarithmic time complexity – O(log n) – Binary Search Algorithm**
3. **Linear time complexity – O(n) – Linear searching**
4. **O(n logn)**
5. **Quadratic time complexity – O(n^2) – sorting**
6. **Cubic time complexity – O(n^3) – Matrix Multiplication**
7. **Polynomial Time Complexity – O(n^c) where c > 0**
8. **Exponential Time Complexity – O(c^n) where c > 1 -- WORST**

**n! < n^n**

**Relationship between 2^n < n! < n^n**

**2^n < n^n**

**2^n – O(n!)**

**n! – O(n^n)**

**[1] O(n) < O(n^3)**

**[2] O(2^n) < O(3^n)**

**[3] O(log n) < O(n)**

**[4] O(log n)^2 < O(n) ------ O(log n) ^ 100 < O(n)**

**[5] O((log n)^log n) > O(n)**

**[6] Taking log on both sides**

**[7] Log n \* log \* log n > log n**

**[8] Log2n > log3n -- Log3n = log2n/log2 3 = log2n/1.5**