$f(x) = 17x^6 - 45x^3 + 2x + 8$

3. It is a fact that, for all real numbers $x > 2$,

$$10|x^6| \leqslant |17x^6 - 45x^3 + 2x + 8| \leqslant 30|x^6|.$$

Given this, what sort of Big-O, Big-Ω and/or Big-Θ statements are possible here?

f is $O(x^6)$ w/ witnesses $k = 20$, $C = 30$

Also f is $\Omega(x^6)$ w/ witnesses $k = 32$, $C = 10$

The two taken together show that f is of order x^6.

Key tool in finding witnesses: Δ-ineq.

$|17x^6 - 45x^3 + 2x + 8|$

$\leq |17x^6| + |45x^3| + |2x| + |8|$

Some Facts:

1. If $m \geqslant n$ and f is a polynomial of degree n, then $f(x)$ is $O(x^m)$.

$\log_b(n!) \leq \log_b(n^n) = n\log_b n$

2. $n!$ is $O(n^n)$ and, as a consequence, $\log_b n!$ is $O(n \log_b n)$, for any $b > 1$.

$n! = \underbrace{n(n-1)(n-2)\cdots(1)}_{n \text{ factors}} \leq \underbrace{n \cdot n \cdot n \cdots n}_{n \text{ factors}}$

Conversely n^n is $\Omega(n!)$

3. It can be shown that $n < 2^n$ for $n \geqslant 1$ and, as a consequence, $\log_b n$ is $O(n)$ for all $b > 1$.

n is $O(2^n)$ taking logs $\log_b n$ is $O(\log 2^n)$

$(n \cdot \log 2)$

$O(n)$

4. If $f_1(x)$ is $O(g_1(x))$ and $f_2(x)$ is $O(g_2(x))$, then $(f_1 + f_2)(x)$ is $O(\max(|g_1(x)|, |g_2(x)|))$.

Take $f(n) = \log(n!) + n^2$ is $O(\max(|n\log n|, n^2))$

is $O(n \log n)$

5. If $f_1(x)$ is $O(g_1(x))$ and $f_2(x)$ is $O(g_2(x))$, then $(f_1 f_2)(x)$ is $O(g_1(x) g_2(x))$.

So $f_1(n) = 3n^2 \log_2 n + 2n^3$ $\quad f_2(n) = n + 2$

f_1 is $O(n^3)$ $\quad f_2$ is $O(n)$

So product $f_1 \cdot f_2 = (n+2)(3n^2 \log_2 n + 2n^3)$ is $O(n^4)$

6. As a result of Facts 3 and 5, we have

$$n \log_b n \text{ is } O(n^2), \qquad x^p (\log_b x)^q \text{ is } O(x^{p+q}), \qquad \text{etc.}$$

$f(x) = x^3 (\log_2 x)^5$ is $O(x^8)$

By #9, f is $O(x^4)$.

7. If $f(x)$ is $O(g(x))$ and $g(x)$ is $O(h(x))$, the $f(x)$ is $O(h(x))$.

Transitivity of orders

8. Let $c > b > 1$, and $d > 0$. For comparing of a power function x^d with an exponential growth runction b^x, we have

$$x^d \text{ is } O(b^x), \qquad \text{but not vice versa.}$$

For comparing the two exponential growth functions c^x, b^x we have

$$b^x \text{ is } O(c^x), \qquad \text{but not vice versa.}$$

9. It requires calculus, but it can be shown that for any $b > 0$, $c > 0$, $(\log_b x)^c$ is $O(x)$.

$(\log_2 x)^8$ and $(\log_2 x)^{80}$ both $O(x)$.

There is, therefore, this increasing sequence of orders: 1, $\log_b n$, $(\log_b n)^2$, $(\log_b n)^3$, $\ldots$, n, $n \log_b n$, $n(\log_b n)^2$, $\ldots$, n^2, $n^2 \log_b n$, n^3, $\ldots$, 2^n, 3^n, $\ldots$, $n!$.

Theorem 1: Let $f(x)$ be a polynomial of degree n—that is,

$$f(x) = a_n x^n + a_{n-1} x^{n-1} + \cdots + a_1 x + a_0,$$

with $a_n \neq 0$. Then

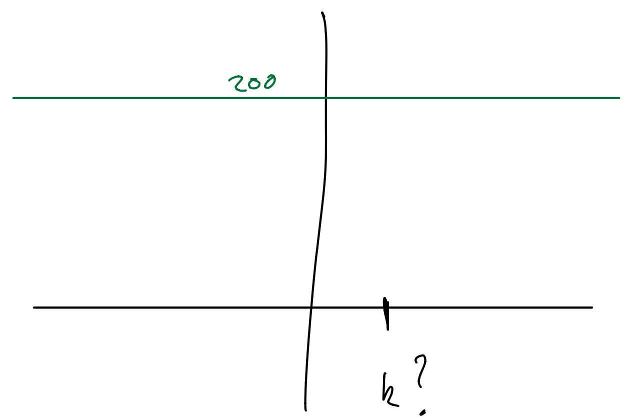
- $f(x)$ is $O(x^s)$ for all integers $s \geqslant n$.
- $f(x)$ is not $O(x^r)$ for all integers $r < n$.
- $f(x)$ is $\Omega(x^r)$ for all integers $r \leqslant n$.
- $f(x)$ is not $\Omega(x^s)$ for all integers $s > n$.
- $f(x)$ is $\Theta(x^n)$.

$f(x) = 200$ is $O(1)$.

witnesses

$k = 0$

$C = 201$



Algorithmic Complexity

Basic idea: relate size n of input to, for instance

- time complexity (often assessed by number of steps)
  - worst-case analysis
  - average-case analysis
- space complexity
- terms like
  - linear complexity
  - quadratic complexity
  - polynomial complexity
  - exponential complexity

Algorithm:

1. Seek divisor of $n \in \mathbb{Z}^+$

   Find a divisor of n

   Alg.

   start at $j = 2$

   Does j divide n

   – i.e. compare

   $\lfloor n/j \rfloor$ w/ n/j

   add 1 to j

   loop until $j \cancel{= n}$ $> \sqrt{n}$

   $O(n)$ $O(\sqrt{n})$

   linear complexity

   Similar to analysis of linear search algorithm
2. binary search
3. bubble sort
4. matrix multiplication