ECSE 210L
Design and analysis of algorithm

Dr. Shakti Sharma

Assistant Professor,

School of Engineering and Applied Sciences

Bennett University, Greater Noida

About the instructor

- M.Tech. and Ph.D. from IIT Bombay

“ Ant Colony Optimization and applications to satellite image analysis”

- Research Areas
  - Evolutionary Algorithm (GA, ACO, PSO)
  - Satellite Image Analysis (Classification, Super-resolution, Domain Adaptation)
  - Development of new algorithm
  - GIS

Insertion Sort

- Time Complexity of insertion sort

```
insertion_sort(A)
    For i = 2 to A.length
            Key = A[i]
            j = i-1
            while j > 0 and A[j] > Key
                    A[j+1] = A[j]
                    j = j-1
            A[j+1] = Key
```

Asymptotic Notation

$n = 10^6$

- Big 'O'
- Θ
- Little 'o'
- Ω
- Little 'ω'

$f(n) = 3n^2 + 5n + 6$

$\Rightarrow 3 \times 10^{12} + 5 \times 10^6 + 6$

$T(n)$

for i = 1 to n^2
{ State ; 3 unit of time
}

for i = 1 to n
{ statement ; 5 unit of time
}

6 unit of time

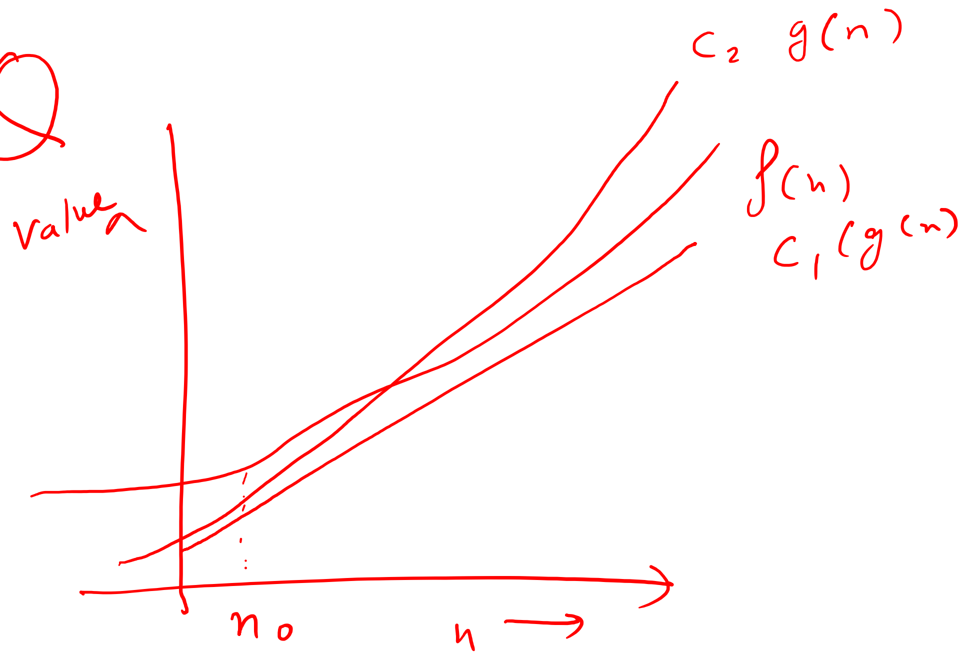
Average case complexity

Θ

- $0 \le c_1 g(n) \le f(n) \le c_2 g(n)$ for all $n \ge n_0$

$$\boxed{f(n) = 3n^2+5n+3000 = \Theta(n^2)}$$

$$f(n) \xrightarrow[=]{} \Theta(g(n))$$



$$c_1 n^2 \le 3n^2 + 5n + 3000 \le c_2 n^2$$

$$\Rightarrow \quad c_1 \le 3 + \cancel{\frac{5}{n}}^{\,0} + \cancel{\frac{3000}{n^2}}^{\,0} \le c_2$$

$$\Rightarrow \quad c_1 \le 3 \le c_2$$

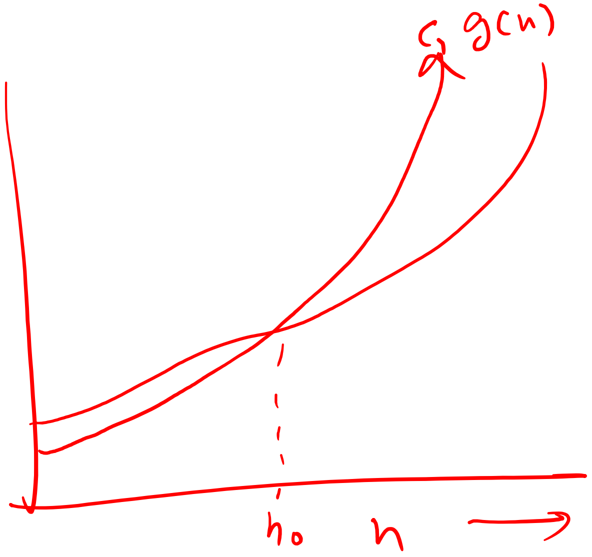
divided by n^2

$2n^2$

$4n^2$

$$f(n) = 3n^2 + 5n + 3000$$

$$0 \le f(n) \le c_1 g(n) \text{ for all } n \ge n_0$$



$$f(n) = O(g(n))$$

$$= O(n^2)$$

$$\Rightarrow O(n^3) \Rightarrow O(n^4) \ldots$$

$O(2^n)$

$O(n \log n)$

$$\boxed{f(n) = n \log n + n + n^2 + c + 2^n + \log n}$$

$$f(n) = \begin{cases} \log(n) & K < 0.5 \\ n \log n & K > 0.5 \end{cases}$$

$\Theta = 0 \le c_1 g(n) \le f(n) \le c_2 (g(n)) = O(n^3)$

$\ne \Theta(n^3)$

$\Rightarrow \boxed{\cancel{c_1 n^3 \le 3n^2 + 5n + 3000}} \le \underline{c_2 n^3} = \Theta(n^2)$

$\Omega \Rightarrow 0 \le c_1 g(n) \le f(n)$ for all $n \le n_0$]

There exist some non-negative c_1 such as:

$f(n) = \Omega(g(n))$

little 'o' $\Rightarrow$ $0 \leq f(n) < c \cdot g(n)$ for all $n \geq n_0$.

or

$$\lim_{n \to \infty} \frac{f(n)}{g(n)} = 0$$

little 'ω' $\Rightarrow$ $0 \leq c\, g(n) < f(n)$ for all $n \geq n_0$

for $i := 1$ to n

Statement1;

for $j = 1$ to $\lfloor \log n \rfloor$

Statement2;

$$f(n) = n + \log n$$

Time Complexity of this Algorithm can be written as

$\Rightarrow O(n), \Theta(n), \Omega(n), O(n^2), \Theta(n^2), \Omega(n^2)$

$O(\log n), \Theta(\log n), \Omega(\log n), O(1), \Theta(1), \Omega(1)$

Big 'O'.

there exist some c_1 such that

$0 \leq f(n) \leq c_1 g(n)$, for all $n \geq n_0$

$f(n) = O(g(n))$

Ex#

$= O(n)$ ✓

$= O(n^2)$ ✓

$= O(n \log n)$ ✓

$\neq O(\log n)$

$f(n) = d_1 n + d_2$ { d_1 & d_2 are positive constants)

for i = 1 to n
Statement;

$0 \leq d_1 n + d_2 \leq c_1 \log n$

'Ω'

$0 \leq c_1 (g(n)) \leq f(n)$

$c_1 \log n \leq d_1 n + d_2$

$f(n) = \Omega(n)$

$\neq \Omega(n^2)$

$= \Omega(\log n)$

$\Theta(g(n))$

$$0 \leq c_1 g(n) \leq f(n) \leq c_2 g(n)$$

$$f(n) = d_1 n + d_2$$

$$g(n) = \overset{\times}{1}, \ \overset{\times}{\log n}, \ \underline{\overset{\checkmark}{n}}, \ \overset{\times}{n^2}$$

$$T(n) = \begin{cases} c_1 \log n + c_2 & k < 1 \\ c_3 n^2 + c_4 n + c_5 & k > 1 \\ n + 5 & k = 1 \end{cases}$$

Select the correct options:

Correct options

$f_1(n) = n + n^{0.99999} \log n$

$f_2(n) = 1^n + 2^n + n^n$

$f_3(n) = n^{2009} + 2^n$

$O(n)$, $O(n^{0.999999} \log n)$, $O(2^n)$, $O(n^n)$, $O(n^{2009})$

1 2 3 4 5

Θ 6 7 8 9 10

Ω 11 12 13 14 15

Formal Definitions of different asymptotic Notations

- Big 'O'

$O(g(n))$ = {f(n), such that **there exist** positive constants c and n_0 such that
$0 \le f(n) \le cg(n)$ for all $n \ge n_0$}

- Θ

$\Theta(g(n))$ = {f(n), such that **there exist** positive constants c_1, c_2, and n_0 such that
$0 \le c_1 g(n) \le f(n) \le c_2 g(n)$ for all $n \ge n_0$}

- Ω

$\Omega(g(n))$ = {f(n), such that **there exist** positive constants c and n_0 such that
$0 \le cg(n) \le f(n)$ for all $n \ge n_0$}

Formal Definitions of different asymptotic Notations

- Little 'o'

o(g(n)) = {f(n), such that **for any** positive constants c, there exist a constant n_0 such that

$0 \leq f(n) < cg(n)$ for all $n \geq n_0$}

$\lim_{n \to \infty} \frac{f(n)}{g(n)} = 0$

- Little 'ω'

ω(g(n)) = {f(n), such that **for any** positive constants c, there exist a constant n_0 such that

$0 \leq cg(n) < f(n)$ for all $n \geq n_0$}

$$\lim_{n \to \infty} \frac{f(n)}{g(n)} = \infty$$