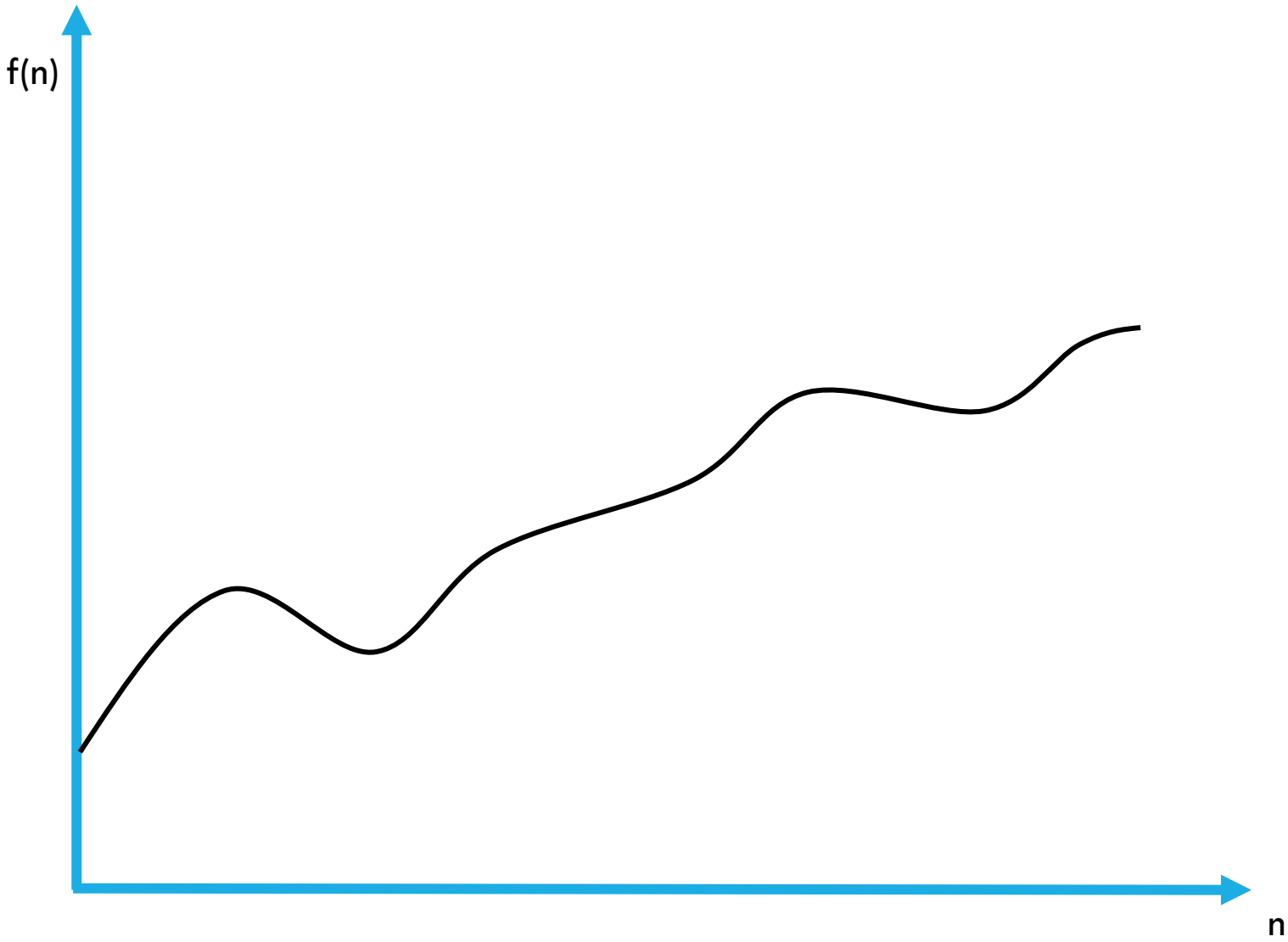
15CSE374
INTRODUCTION TO DATA STRUCTURES AND ALGORITHMS

Sarath tv

Last Lecture

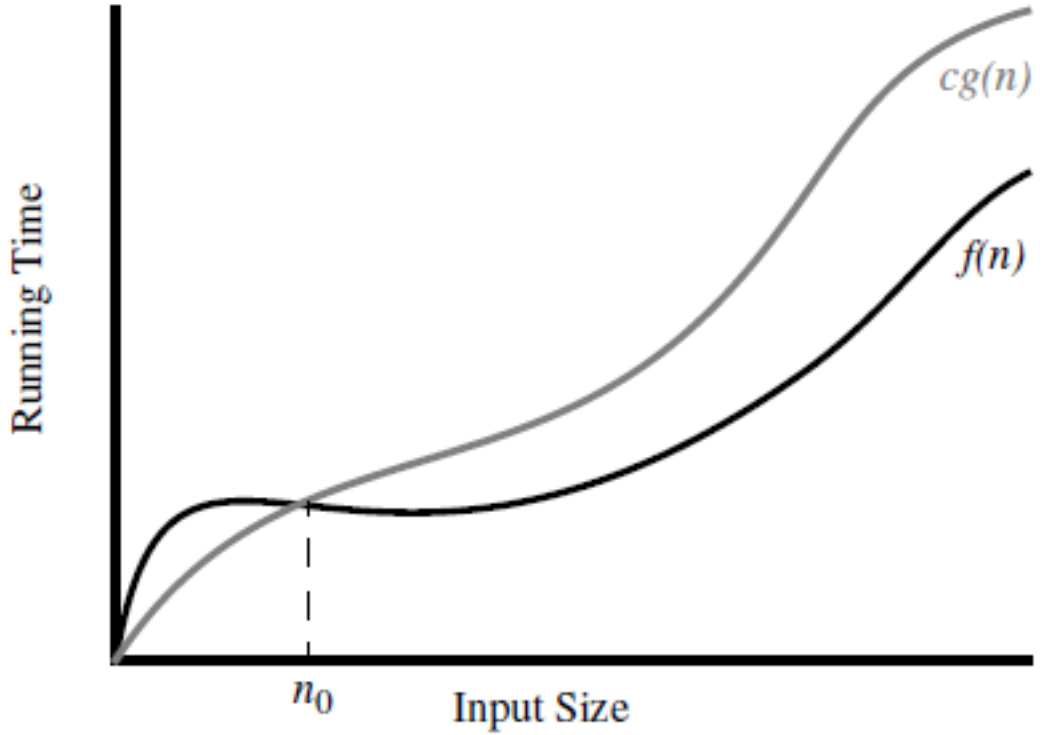
- Big O for Python codes.

Upper & Lower Bound



Asymptotic Analysis Formal Method

- Upper Bound
- Let $f(n)$ and $g(n)$ be functions mapping positive integers to positive real numbers.
- We say that $f(n)$ is $O(g(n))$ if there is a real constant $c > 0$ and an integer constant $n_0 \geq 1$ such that
  - $f(n) \leq cg(n)$, for $n \geq n_0$.


Running Time
cg(n)
f(n)
n_0
Input Size

Time Complexity

$$1 < \log n < n < n\log n < n^2 < n^3 < 2^n < n^n$$

Example Upper Bound

$1 < \log n < n < n\log n < n^2 < n^3 < 2^n < n^n$

- f(x) =2n+3
  - f(n) = O(g(n)) iff c and n0
  - f(n) ≤ c*g(n) for every n≥ n0
- First identify proper g(n).

Big Omega – Lower bound

- Lower Bound
- Let $f(n)$ and $g(n)$ be functions mapping positive integers to positive real numbers.
- We say that $f(n)$ is $\Omega(g(n))$ if there is a real constant $c > 0$ and an integer constant $n_0 \geq 1$ such that
  - $f(n) \geq c*g(n)$, for $n \geq n_0$.

Big Ω Example

$$1 < \log n < n < n\log n < n^2 < n^3 < 2^n < n^n$$

- f(x) =2n+3

Big – O (Substitution method)

How did we get those constant values…!!

Is those the tightest bound possible?

$O(g(n))$ is the set of functions with smaller or the same order of growth as $g(n)$. For example, $O(n^2)$ includes $O(1)$, $O(n)$, $O(n\log n)$, etc.

Example-1 Find upper bound for $f(n) = 3n + 8$

Big – O (Substitution method)

Example-1 Find upper bound for $f(n) = 3n + 8$

Solution: Let $g(n)=n$. If C =2, 3, 4, ...8,9,..

n	$f(n)$	$g_2(n)$	$g_3(n)$	$g_4(n)$	$g_5(n)$	$g_8(n)$	$g_{12}(n)$
1	11	2	3	4	5	8	12
2	14	4	6	8	10	16	24
3	17	6	9	12	15	24	36
4	20	8	12	16	20	32	48
5	23	10	15	20	25	40	60
6	25	12	18	24	30	48	72
7	29	14	21	28	35	56	84
8	32	16	24	32	40	64	96

Big – O (Substitution method)

Example-1 Find upper bound for $f(n) = 3n + 8$

Solution: $3n + 8 \leq 4n$, for all $n \geq 8$

$3n + 8 \leq 5n$, for all $n \geq 4$

...

$3n + 8 \leq 8n$, for all $n > 1$

$3n + 8 \leq 12n$, for all $n > 0$

Big – O (Substitution method)

Example-2 Find upper bound for $f(n) = n^2 + 1$

Solution: Let $g(n)=n^2$, let C = 1, 2, 3,...

n	$f(n)$	$g_1(n)$	$g_2(n)$	$g_3(n)$
1	2	1	2	3
2	5	4	8	12
3	10	9	18	27
4	17	16	32	48
5	26	25	50	75
6	37	36	72	108

Big – O (Substitution method)

Example-2 Find upper bound for $f(n) = n^2 + 1$

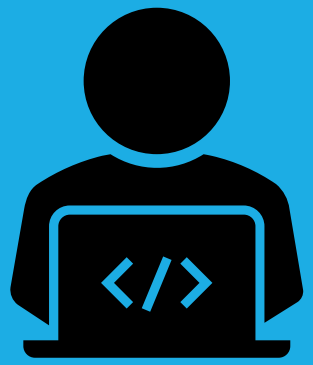
Solution: Let $g(n)=n^2$, let C = 1, 2, 3,...

n	$f(n)$	$g_1(n)$	$g_2(n)$	$g_3(n)$
1	2	1	2	3
2	5	4	8	12
3	10	9	18	27
4	17	16	32	48
5	26	25	50	75
6	37	36	72	108

$n^2 + 1 \leq 2n^2$, for all $n \geq 1$

$n^2 + 1 < 3n^2$, for all $n \geq 1$

- $f(n)=n^4+100n^2+50$



THANK YOU!!!!!