Say, we have algorithms for the same problem

Line by line analysis gives:

Algorithm A: T(n)= $10*n^2+5$

Algorithm B: T(n)= 100000*n+ 100000

	Algorithm A: T(n)= $10*n^2+5$	Algorithm B: T(n)= 100000*n+ 100000
n=1	T(1)=15	T(1)=200000
n=10	T(10)=1005	T(10)=1100000
.....		
n=1000	T(1000)=1000005	T(1000)= 100000000+100000

n=100000 (becoming larger than the second)

We can compare two functions based on order of growth

Asymptotic Analysis

Algorithms are compared based on *n* lends to infinity

Ignore constant terms from T(n)

Algorithm A: T(n) $\propto$ n^2 =O(n^2)

Algorithm B: T(n) $\propto$ n=$O(n$)

Big-oh notation: O

T(n)=f(n)= $5n^2$ + 10

g(n)= n^2

f(n) =O(g(n)) = O(n^2)?

f(n)>=0 and f(n)<=c*g(n)

Find n_0 , for which above condition holds for all n>n_0

f(n)= $5n^2$ + 10 <=$15n^2$

$5n^2$ + 10<$30n^2$ for all n>=1

n_0=1, c=30

f(n)=O(n^2)

$f(n) <= c.n^2$

$T(n)<c.g(n)$

Limit method