



NEXTCORE AINEXTCORE A

Guiding Principles

Design and Analysis
of Algorithms I

Nextcore AI - Gopal
Shangari

NEXTCORE AINEXTCORE A



GUIDING PRINCIPLE #1

“worst – case analysis” : our running time bound holds for every input of length n .

–Particularly appropriate for “general-purpose” routines

As Opposed to
–“average case” analysis
–benchmarks

} **REQUIRES DOMAIN
KNOWLEDGE**

BONUS : worst case usually easier to analyze.



Guiding Principle #2

Won't pay much attention to constant factors,
lower order terms

justification

1. Way easier
2. Constants depend on architecture / compiler / programmer anyways
3. Lose very little predictive power (as we'll see)

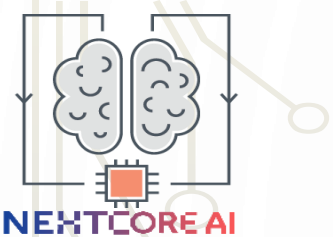


GUIDING PRINCIPLE #3

Asymptomatic Analysis : focus on running time for large input sizes n

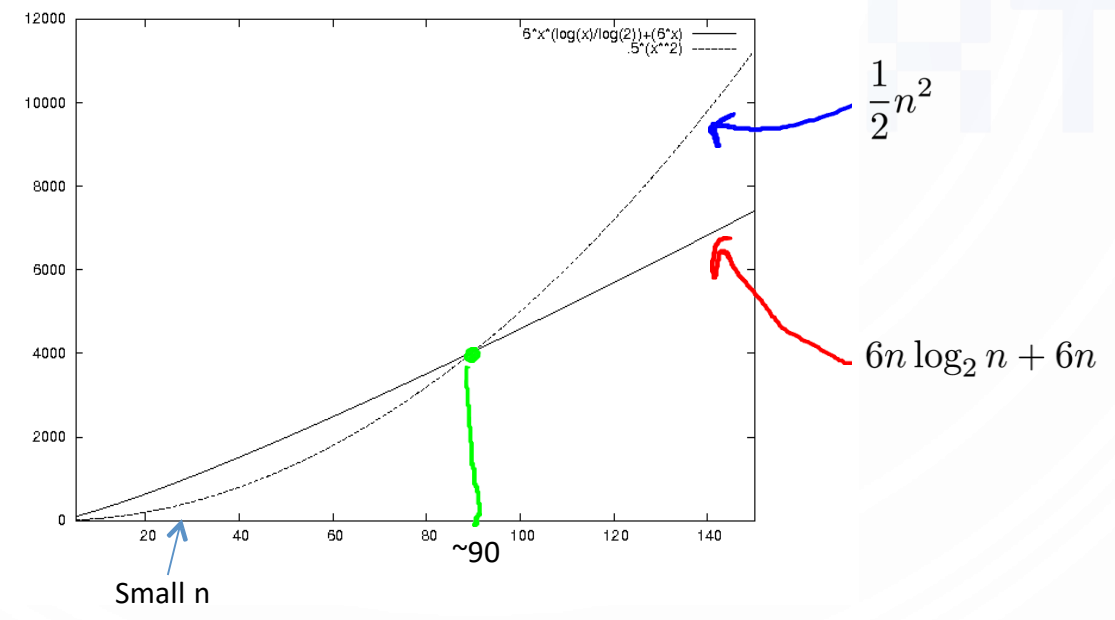
Eg : $6n \log_2 + 6n$ "better than" $\frac{1}{2}n^2$

justification: Only big problems are interesting!



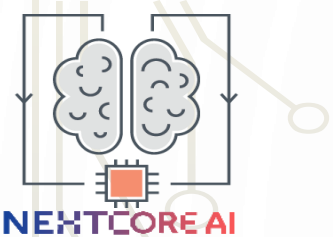
NEXTCORE AI

NEXTCORE AI



Nextcore AI -
Gopal
Shangari

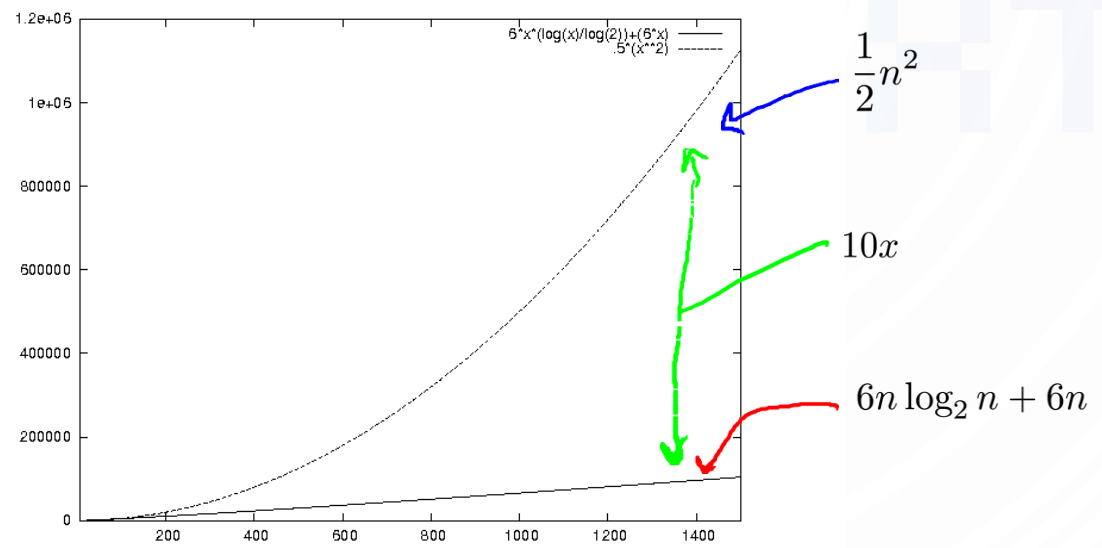
NEXTCORE AI



NEXTCORE AI

NEXTC

NEXTCORE AI



Nextcore AI -
Gopal
Shangari

NEXTCORE AI NEXTCORE AI



What Is a “Fast” Algorithm?

This Course : adopt these three biases as guiding principles

fast
algorithm

≈

worst-case running time
grows slowly with input size

Usually : want as close to linear ($O(n)$) as possible