Lower Bound on Comparison-Based Sorting

https://cs.pomona.edu/classes/cs140/

Outline

Topics and Learning Objectives

- Discuss a lower bound for the running time of all comparison-based sorting algorithms

Exercise

- Lower bound

Extra Resources

- Introduction to Algorithms, 3rd, Chapter 8

Comparison-Based Sorting

Claim: The worst-case, lower bound on comparison-based sorting is $\Omega(n \lg n)$

Comparison-based sorting methods:

- Merge sort, quicksort, heapsort, insertion sort, bubble sort, …
- General purpose routines

Non-comparison-based sorting methods:

- Bucket sort, counting sort, radix sort, …
- These methods look at the values (not just at the relative ordering)
- They assume something about the distribution of the data
- They can operate in linear time

Proof

- Consider an array of the values 1..n

How many different orderings?

- The array has n! different orderings (permutations)
- We can only use the results of comparisons to reorder elements
- Suppose an algorithm makes k comparisons
- We don’t know what k is just yet
- How many possible distinct comparisons sequences do we have?

2^k

We need an equation based on **k**

- What is a reasonable upper bound on k?

n^2

- What is the lower bound on k?

$n \lg n$

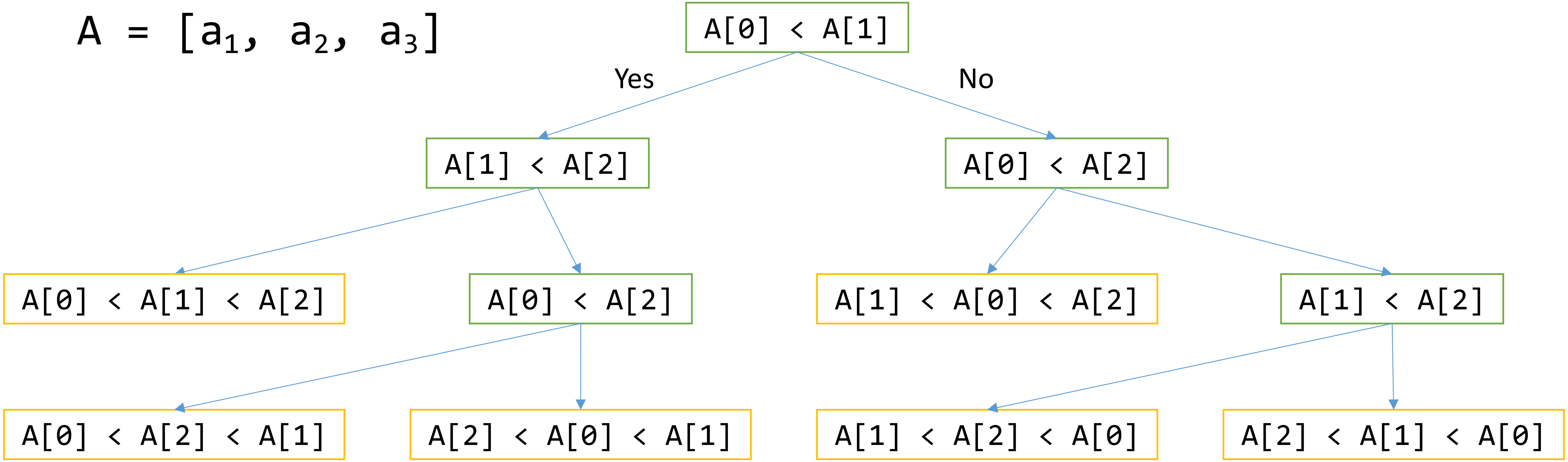
Proof

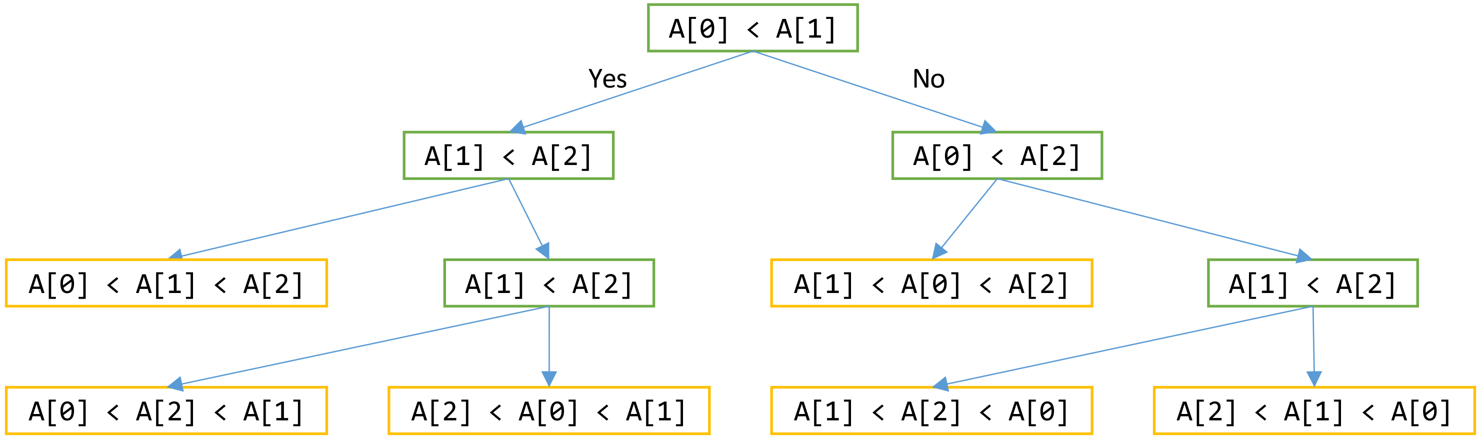
k is the **maximum** depth of the tree

Given each of the n! inputs and the k comparisons:

- We have 2^k distinct comparison sequences
- For each of the k comparison we can return value a or value b
- You can think of these comparisons as a decision tree

$A = [a_1, a_2, a_3]$





How many leaves as a function of n? $n!$

What is the height of the tree as a function of k? k

What is the **maximum** number of leaves in a depth k **binary** tree? 2^k

What is the minimum height of a **binary** tree with n! leaves? $\lg(n!)$

Let's find a bound on k

What is bigger?

- The number of leaves with n! numbers OR
- The maximum number of leaves for a tree of height k?

Let's find a bound on k

What is bigger?

- The number of leaves with n! numbers OR
- The maximum number of leaves for a tree of height k?

Might not have a "full" tree

Number of leaves with n numbers

$$n! \leq 2^k$$

Maximum number of leaves with depth k (k Comparisons)

$$\ln(n!) \leq \ln(2^k)$$

$$\ln(n!) \leq k \cdot \ln(2)$$

$$\ln(n!) \leq k \cdot c_1$$

Lower Bound!

Number of comparisons k is at least…

Let's find a bound on k

Stirling's approximation:
$$\ln(n!) = n \cdot \ln(n) - n + O(\ln(n))$$

$$n \cdot \ln(n) - n + O(\ln(n)) \leq k \cdot c_1$$

$$n \cdot \ln(n) - n + O(\ln(n)) \leq n \cdot \ln(n) + O(\ln(n)) \leq k \cdot c_1$$

$$n \cdot \ln(n) + O(\ln(n)) \leq k \cdot c_1$$

$$n \cdot \ln(n) + O(\ln(n)) \leq n \cdot \ln(n) + c_2 n \ln(n) \leq k \cdot c_1$$

$$c_3 n \ln(n) \leq k \cdot c_1$$

$$\frac{c_3}{c_1} n \ln(n) \leq k$$

$$c_4 n \ln(n) \leq k$$

$$k = \Omega(n \ln(n))$$