15-211: Recitation 2, Section M

Hui Han Chin, hchin@cmu.edu
PLAN FOR TODAY

- Quiz 1 Solutions
- Amortized Analysis
- Masters theorem
- Comparison Based model
- Sorting Algorithms
AMORTIZED ANALYSIS

- Averages over a sequence of operations
- Not all operations are equal!
- Bankers method is good for guess and check
  - $1, 2, 3, 4, 5, \log n, n, n^2$
- Aggregate method is good for quizzes and finals
  - Mean cost = total cost / no of ops
- Potential method is good for proving stuff
  - Amortized = True + $\Delta\Phi$
MASTER’S (CHIEF) THEOREM

Theorem 2.1 The recurrence

\[ T(n) = aT(n/b) + cn^k \]
\[ T(1) = c, \]

where \( a, b, c, \) and \( k \) are all constants, solves to:

\[ T(n) \in \Theta(n^k) \text{ if } a < b^k \]
\[ T(n) \in \Theta(n^k \log n) \text{ if } a = b^k \]
\[ T(n) \in \Theta(n^{\log_b a}) \text{ if } a > b^k \]
Comparison based model is where only 2 elements are compared to each other.

Think of it as a binary predicate or the compare operator.

Radix sort is not comparison based as you are not comparing 2 elements each time.
Which element have I picked?
How many questions should I ask?

{a,b,c}
{1,2,3}
{1,3,2}
{2,1,3}
{2,3,1}
{3,1,2}
{3,2,1}
<table>
<thead>
<tr>
<th>{a,b,c}</th>
<th>{1,2,3}</th>
</tr>
</thead>
<tbody>
<tr>
<td>{1,2,3}</td>
<td>{1,3,2}</td>
</tr>
<tr>
<td>{1,3,2}</td>
<td>{2,3,1}</td>
</tr>
<tr>
<td>{2,1,3}</td>
<td></td>
</tr>
<tr>
<td>{2,3,1}</td>
<td>{2,1,3}</td>
</tr>
<tr>
<td>{3,1,2}</td>
<td>{3,1,2}</td>
</tr>
<tr>
<td>{3,2,1}</td>
<td>{3,2,1}</td>
</tr>
</tbody>
</table>
**Which element have I picked?**

**How many questions should I ask?**

<table>
<thead>
<tr>
<th></th>
<th>$a &lt; b$ ?</th>
<th>$b &lt; c$ ?</th>
<th>$a &lt; c$ ?</th>
</tr>
</thead>
<tbody>
<tr>
<td>${a,b,c}$</td>
<td>${1,2,3}$</td>
<td>${1,2,3}$</td>
<td>${1,2,3}$</td>
</tr>
<tr>
<td>${1,2,3}$</td>
<td>${1,3,2}$</td>
<td>${2,3,1}$</td>
<td>${2,3,1}$</td>
</tr>
<tr>
<td>${1,3,2}$</td>
<td>${2,3,1}$</td>
<td>${2,3,1}$</td>
<td>${1,3,2}$</td>
</tr>
<tr>
<td>${2,1,3}$</td>
<td>${2,1,3}$</td>
<td>${3,2,1}$</td>
<td>${3,2,1}$</td>
</tr>
<tr>
<td>${2,3,1}$</td>
<td>${2,1,3}$</td>
<td>${3,2,1}$</td>
<td>${3,2,1}$</td>
</tr>
<tr>
<td>${3,1,2}$</td>
<td>${3,1,2}$</td>
<td>${2,1,3}$</td>
<td>${2,1,3}$</td>
</tr>
<tr>
<td>${3,2,1}$</td>
<td>${3,2,1}$</td>
<td>${2,1,3}$</td>
<td>${3,1,2}$</td>
</tr>
</tbody>
</table>
QUESTION

- Lower bound for sorting in a comparison based model is ** n \log n
- Alec invented a new comparison based heap that takes O(n) to build but has O(1) deleteMin()
- Is this possible?
SORTING ALGORITHMS

- HeapSort
- QuickSort
- MergeSort
- RadixSort
HEAPSORT

- BuildHeap in O(n)
- Often implemented as priority queue
- Useful if only a portion of the sorted list is required
QUICKSORT

- Java Arrays implements this
- Faster than most sorts on expected case
  - Due to partial sorted nature of data
- Worst case $O(n^2)$ for 15-211!
**MERGESORT**

- Highly Parallelizable sort
- Can break $n \log n$ bound in parallel version
- Useful for very large data set
- Java Collections implements this
RADIX SORT

- Non-comparison base sorting
- Useful for very small key space
- Think of a phone book