Sorting by Counting

Class 18

Sorting Analysis

- sorting routine analyses so far

algorithm	best case	worst case
bubble	n^2	n^2
insertion	n^2	n^2
merge	$n \lg n$	$n \lg n$
quick	$n \lg n$	n^2
heap	$n \lg n$	$n \lg n$

- it is provably impossible to sort faster than mergesort or heapsort

Sorting Analysis

- sorting routine analyses so far

algorithm	best case	worst case
bubble	n^2	n^2
insertion	n^2	n^2
merge	$n \lg n$	$n \lg n$
quick	$n \lg n$	n^2
heap	$n \lg n$	$n \lg n$

- it is provably impossible to sort faster than mergesort or heapsort
- but what if we break the rules and cheat?

Basic Counting Sort

code countsort.cpp

trace with $\{62, 31, 84, 96, 19, 47\}$

Basic Counting Sort

code countsort.cpp

trace with $\{62, 31, 84, 96, 19, 47\}$

called counting sort because by counting we know where each element goes

- input size:
- basic operations:
- best & worst:
- analysis:

Basic Counting Sort

code countsort.cpp

trace with $\{62, 31, 84, 96, 19, 47\}$

called counting sort because by counting we know where each element goes

- input size: elements of array
- basic operations: for loops, element comparisons, assignment
- best & worst: no
- analysis: $T(n) \in \Theta(n^2)$

Counting Sort

- pros
  - no repeated swaps
  - moves each element exactly once: $\Theta(n)$ element moves
  - always interesting to see other ways to solve a problem
- cons
  - uses $\Theta(n)$ extra space
  - is in n^2 category, not nearly as good as heapsort and mergesort

Distribution Sorting

- now, assume that we know in advance that the items to be sorted are restricted to a small range of values
- here, only the values 1, 2, and 3 occur

2	2	3	2	1	1	1	2

- we can take advantage of this information about the input arrangement
- similar to count sort, but now we count how many of each value occurs
- i.e., we compute a frequency distribution

code distributionsort.cpp

Distribution Sort

- input size?
- arrangement matter?

- best & worst?
- analysis?

Distribution Sort

- input size? size of array n and size of range k
- arrangement matter? yes! depends on small range of input values
- best & worst? no!
- analysis? $T(n, k) \in \Theta(n + kn)$

- what does this mean in practice?