Subject: Concept Review: Sorting  
Posted by davidmccoy on Tue, 07 Feb 2017 20:57:59 GMT  
View Forum Message <> Reply to Message

Starting this thread per the proposed schedule for study for the first exam. Discuss the sorting practice problems here.  
View questions here: http://beastie.cs.ua.edu/concepts/cs/ds/sorting.html

Work together on the proposed answers to questions on this shared Google Doc (comment reasoning/arguments behind answers)

Upcoming topics (threads start two days in advance): Recurrences 1-35, Feb 10 | Recurrences 36-70, Feb 11 | SB trees 1-26, Feb 12 | SB trees 26-52, Feb 13  
Full schedule

Subject: Re: Concept Review: Sorting  
Posted by davidmccoy on Thu, 09 Feb 2017 05:05:26 GMT  
View Forum Message <> Reply to Message

edit: Got this answered in class today. Selection sort optimizes bubble sort's behavior by committing many less writes (from swaps), instead just reading and comparing values and then inserting into the sorted portion.

Quote:5. Which sort optimizes the worst case behavior of bubble sort?  
A. selection sort  
B. heapsort  
C. insertion sort  
D. mergesort  
E. stooge sort  
F. quicksort

Anyone have insight into the answer for this? I'm sure we already brought it up in class a while ago.

Subject: Re: Concept Review: Sorting  
Posted by dtturner2 on Tue, 14 Feb 2017 13:29:28 GMT  
View Forum Message <> Reply to Message

I'm still not quite sure why this one is selection sort. Did anyone ever remember what the explanation was for this?
Worst case behavior for bubble sort is quadratic, best case behavior for selection sort is quadratic so I don't really see how that would optimize anything.

I say either insertion sort or mergesort is the answer here. I want to say mergesort since the algorithm compares adjacent values and swaps them accordingly, as in bubblesort. However, the wording of the question makes me think the answer is an algorithm that ONLY optimizes worst case, which would be insertionsort.

It's selection sort, since bubblesort can make n-1 swaps (swaps are very expensive), while selection sort accomplishes the same thing but with a single swap.