Etan Soclof

Greedy Drill

Explanation of how my greedy algorithm breaks through the O(nlogn):
Given a knapsack that can carry w pounds and n items of equal weight b, we can determine the amount of items that can fit into the bag - *a* - by calculating $\lfloor w/b \rfloor = a$. Once we know this value, instead of sorting all of the items, we only need to determine those that are less than the (a - 1)th least valuable item and those that are greater. This can be done in O(n) by using quick select.

Pseudocode

```
w = carrying capacity of knapsack
b = weight of a single item
a = ⌊w/b⌋

quick-select to partition the items between those that are more
valuable and then valuable than the (a - 1)th least valuable item

add all items more valuable than the (a - 1)th least valuable item to
the knapsack

if room in knapsack
     add fraction of (a - 1)th least valuable item to knapsack
```

Proof and Correctness

Because all items are of equal weight, the knapsack must contain the *w/b* most valuable items. By determining the (a - 1)th least valuable item by using quick select we know all items that are less valuable than the (a -1)th least valuable item and all items that are more valuable than the (a -1)th least valuable item. All the items that are more valuable than the (a-1)th least valuable item are the *w/b* most valuable items.
Quick select is O(n) and therefore the algorithm is O(n).