5.0	
namananan manistra kata kananan kata kananan kata kanan kata kanan kata kanan kata kanan kata kanan kata kanan	Last week we learned about houps, priority-queves and so heapsortalg.
	Teday we'll prove an N(nlogn) lower bound for companion bused
MANIGE AND SEAL OF THE SEAL OF	Soit and then we'll show how to brook it!
	Time permitting we'll start discussing a very useful a typithate binary
and the same of th	search trees.
o, papara muunos suomas suoma su	
	Mid-term: Date is set to oct. 25th
	We will also see efficient algorithms for returning the i'th largest element in
a a pagaman ya a kanika wa katika ka katika wa ka katika wa Ka katika ta ka katika wa katika wa 18 katika wa k	A.
na kanada kaka ka	
an Sjanish ay a sa s	

11,11,1	
i sadina pagana ganggungangan anaminin "PP district" (PP)	

Lower bound for comparison-based conting.

usually proving lawer bounds is a difficult took as it requires studying all possible also.

We are going to prove a lower bound for a restricted class at algorithms for sorting. Specifically, we'll prove that algorithms that only make comparisons between array entries, cannot run foster than I (nlog 11). Horever, we'll prove they need to make a loost this number of comparisons

A lower bound is an agreement showing that any alg. for performing a contain took

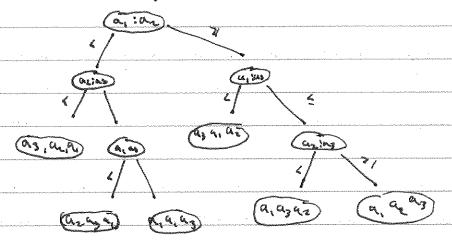
we all such algorithms comparison algorithms and they can be madeled as tollows. At each step the alg. picks two elements along compares them and according to the obtaine (along along a ray) proceeds to the next step. At the end the alg. has to come up with a permutation that sits the elements.

Picturially we can imagine the alg- as a decision-trees

arear arear

A last is the perutation the algorithms

Here's a concrete example to sorting 3 elements



Notice that we have 6 leaves corresponding to all passible 31=6 permut. Fuclead, when we have n elements there must be at least n1 leaves as every permutation is possible.

Now, what's the running time in the worst care? It is the height of the tree! Because this is a computation path that correspondes to some imput and that makes that many comparisons.

Her to obtain a lower band? Notice that a depth h true has < 2h leaves. Thus n! < 2h => h >, log n! = ~! (n/yn!

Corollary: Heapsort and Hergesort are assymptotically optimal alg. (up to a const. factor perhaps).

The lower board helps for comparison costs. Com we come up with a different kelled of uly that beats the low bound? Counting out is each an als ! but it is based on the assorption that all elements of A are integer in [0,4,-,0(a)=k? The way country outs works to that he every element ix it counts the number of elements less than x and then orthots the elements in the right order. E.g. If we know that, say 17 elements are smaller than x then we gut x in location 18. etc. The alg. will use two extra arrays. C will first count for such i the # of elements equal to i and then the * of elements smaller or early to i. & Thus, C will have 1241 elements C[0:12]. B will contain the serted array. Country - Sout (A.B. C. K) 1. For in the V L. CLilse 3. For it to Alleyth Cril whit & elements = i 4. CEACIJ = CEAEJJ +1 s. for int to K 6. Cris = Cris+Cri-13 7. For j= Alength to 1 B pots ACJI in the right location Q. B[C[ACj]] = ACj] c updated that AGIS is removed

9. CLACIJJ = CEACIJJ -1

```
5.4
                 EXMINE COMPANION CONTRACTOR CONTR
                      A=[153 0 2 303] K= 5
                     First C conts C= (202301)
                     Then it adds (= [2247 78]
                      Then we start writing B
                     First we go to CLACAT) = CLACATT = CL3]=7
                     We write in location 7 in B the value ALAJ=3
B= [A 2 3 u 5 6 7 8]
                       We update C[3] to ( C=[234678]
                       We then yo to C[A[7)] - C[0] = 2
                         8 = 1 0 3 1
                              C=[124678]
                        Thur to CEATES ] = CE37=6
                                                       C=[124578]
                       Etc
                       lunking time:
                        First For (ore = O(K)
                       Second for loop = O(n)
                       Third For loop = O(a) (assuming addition takes O(1))
                        Forth loop takes O(n)
                        Thus run time is O(n46) > O(4) when K=O(4).
                        HW! Prove counting sort is stable.
```

A took related to sorting is that at finding the i'th largest element in an array.

Show Important examples one the largest and smallest elements and the median.

Wereba

E.g. to return the Work largest demont we have the following simple als.

Maximum (A)

1. max = A[i]

2. For IFZ to Aleyth

3. If Alil > Max

4. Max = A[1]

s. adma wax

Bot what about the median? The i'th element?

I chea: similarly to quick sort we shall partition A to two parts. If the pirot is at location k then we will one rocurrien on the side corresponding to weather isk or isk (it isk we neturn the pirot).

We am we the same nardomised partition also of quicksort and get in expectation are 10(a) than also.

an to consequence of the second s	Randomited-Select (A, p. q, i) (We mad to poturn the ith element in
ر المراجعة	1 14 per
oyyysiadalla-iidydd-eir-Yhdalaethii i thaisi y saelid (2000) y saelidd (2000) y saelidd (2000) y saelidd (2000	· · · · · · · · · · · · · · · · · · ·
man market kilologia	3 q = Randomized-Purtition (A.P.s.) Same ulg. from quicksort
i a karangan mangan	4 ko 9-pai ks jocation of pint in AC1-r]
25.50 (25.00)	
	6. VELVI ANTA MATA
	2 Ele it ich
g 2000 million e selles professe for franke fra franke fra franke (1) for fifty beneve green in interest.	9. return Rundomital-Select (A, P, 34i) Semblest 9. Else return Randomital solart (A, 901, Y, i-k) He K elevents was
ness on any many space years of 14 parties of 1,4 ft	su elements were
e a manadamphana miliri dha magimmay i e pomiir me 11 e ini middid e philosophia dha dha dha dha dha dha dha dha dha dh	have analysts as in quicksout shows that who
and the second s	we have running time (O(a) (each step at partition is at example or 9 at
an ang ang ang ang ang ang ang ang ang a	size and besides this construct curt)
accommunities of an all and 30 of himselved free of millionistic definition of a failure feeling of 120 of	
nn e e e e e e e e e e e e e e e e e e	Next we will show how to get a deterministic algorithm.
manananin manana ji 2255424565656574444114295522245956444444	The idea is to find a pivot that was non elements smaller and larger
numanimezzanianian akefinian middialah kalingisikan ankan selam 1997 telim medilimban selam selam selam selam s	the it.
	What do wedo ?
mare emissed dimental anning a boundaries and a significant for a first of the control of the co	Partition in to groups of t elements. In each group tind the medials.
normanman etimakanan kangapapat (jyon kilon pilakan kilon kilon kilon kilon kilon kilon kilon kilon kilon kilon - -	So far time O(n). Now, keep only these elements and recurse.
ymm ynnomy phanomana a cand 57 (564) 553 (555) 553 (555) 544 (44) 545 (575) 553 (575) 554 (576) 554 (576) 554	Note that at each step at least half of the 2 medium are larger flow the
	median of medians (3) Thus hith of the groups contain 3 elements layer than
Marcel 01	

The steps of the ulg. Select we:

1. Partition to of gross of 5 and leap the method (Instead nest)

2 keep the median of each gray.

3. Using select, find the median of the a medians

4 Partition the input array around that was element.

5. depending on wheater izk (x is la. of that element) non Select by

Running fine: Steps 1,2 take (In) and so don step 4. Step 3 takes T(=) and step t takes at most T(==)

explanation: As we said, it X is the meeting then at least 3. I is a 30 elements are smaller than it and that number larger Hamit. Thus, in step 5 the size of the part we run on is s tin

⇒ T(a) s T(f) + T(言) + O(h)

this seems tricky but since = + = = < (this is why we picked () we get a lineartine alg !

Assume the O(1) is san and that T(1) & cn.

We will find c. that works.

T(u) & C(2)+c(2n)+an. \$ cn.

Thus, we need c(1-2-1) >a.

« C > 10.0

Observe that we run select for different i's in steps 3,5 1