

Andrew Hunkeler

12.9 2,

12.13) there would be an inf loop. When two elements are the same, it keeps swapping them indefinitely

12.17) NO, to do it a lot of buckets have to be made

12.19) Merge-sort is  $O(n \log n)$  where Quicksort is  $O(n)$

12.39) a radix sort could be done

It is  $O(dn)$  where  $d$  is max number of digits  $d \leq n$  as well as a constant  
this implies  $O(n)$

70  
10x4  
10x4  
10x4

13.8.] 
$$\begin{matrix} A & B & C & D & E & F \\ 10 \times 5 & 5 \times 2 & 2 \times 20 & 20 \times 12 & 12 \times 4 & 4 \times 60 \end{matrix}$$

$$(A \cdot B) = 10 \cdot 5 \cdot 2 = 100$$
  

$$(B \cdot C) = 5 \cdot 2 \cdot 20 = 200$$
  

$$(C \cdot D) = 2 \cdot 20 \cdot 12 = 480$$
  

$$(D \cdot E) = 20 \cdot 12 \cdot 4 = 960$$
  

$$(E \cdot F) = 12 \cdot 4 \cdot 60 = 2880$$

$$(A \cdot B) = 10 \cdot 2 \cdot 20 = 400$$
  

$$(C \cdot D) = 480$$
  

$$(D \cdot E) = 960$$
  

$$(E \cdot F) = 2880$$

$$\begin{matrix} A'' & D & E & F \\ 10 \times 20 & 20 \times 12 & 12 \times 4 & 4 \times 60 \end{matrix}$$

$$(A'' \cdot D) = 2400$$
  

$$(D \cdot E) = 960$$
  

$$(E \cdot F) = 2880$$

$$\begin{matrix} A'' & D' & F \\ 10 \times 20 & 20 \times 4 & 4 \times 60 \end{matrix}$$

$$(A'' \cdot D') = 900$$
  

$$(D' \cdot F) = 960$$

$$\begin{matrix} A''' & F \\ 10 \times 4 & 4 \times 60 \end{matrix}$$

$$(A''' \cdot F) = 2400$$

$$(((A \cdot B) \cdot C) (D \cdot E)) \cdot F$$
 total = 46000 applications

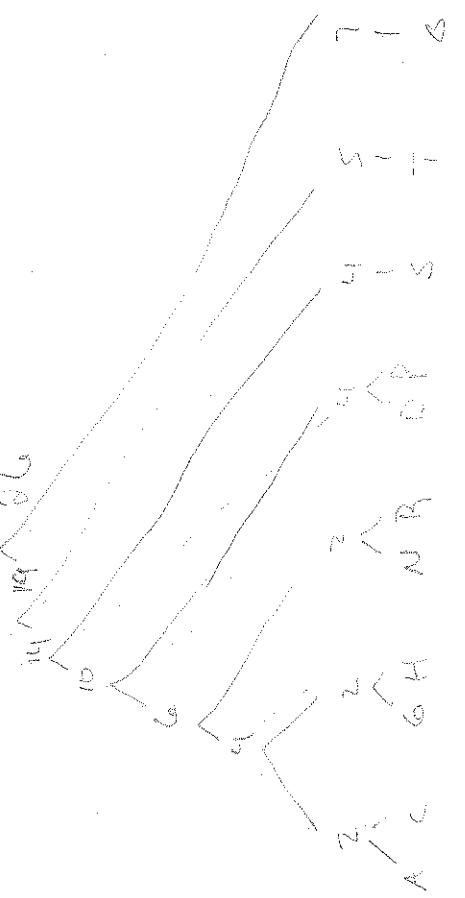
$$(((10 \cdot 5 \cdot 5 \cdot 2) 2 \times 20) (20 \times 12 \cdot 12 \times 4)) 4 \times 60$$

Did A)

13.11 } A C D G H N O P R S T  
 1 1 2 1 1 1 7 2 1 4 5

14 14 20

B.)



13.10  
app

13.10) Villages

skull and bones  
101146, babies

(13, 29) you want to pick the biggest  
value of a coin and recursively pick again  
this will assure the minimal amount of  
coins. say we want 50¢  $\left| \begin{array}{l} 2 \text{ quarters} = 50¢ \\ \text{less} \end{array} \right|$  where as 5 dimes  
= 50¢

13.5) 9 cyber dollars

S.Ce)

S.8) We tried to test the books code  
however we kept getting 0.0 as a time  
so we could not test

However list.pop() would be  $O(1)$   
and list.pop(n) would be  $O(n)$

9,3)  $\{(1, D), (4, B), (5, A), (7, F)\}$   
 $\{(3, G), (4, B), (5, A), (6, D), (7, F)\}$   
 $\{(5, A), (6, L), (7, F), (8, G)\}$   
 $\{(2, H), (6, F), (7, F), (8, G)\}$   
 $\{(7, F), (8, G)\}$



9.13.) (2 5 6 4 10 23 39 18 26 15)

23 39 18 26 15

23 26 15 10 2

18 6 15 10

4 2 5

23 26 10 18 15 16 2 5 4 39

23 16 18 15 10 2 5 4 26 39

2 4 5 10 15 16 18 23 26 39

9.15) When you make a heap  
via model!n should have a right  
child but not a left child

9.18)  $\exists n_0 \forall n \geq n_0 \exists c \ 0 \leq f(n) \leq cn$

$n_0 = 0$

$c = 1$

$$\sum_{i=1}^{\log n} i \leq n \log n$$

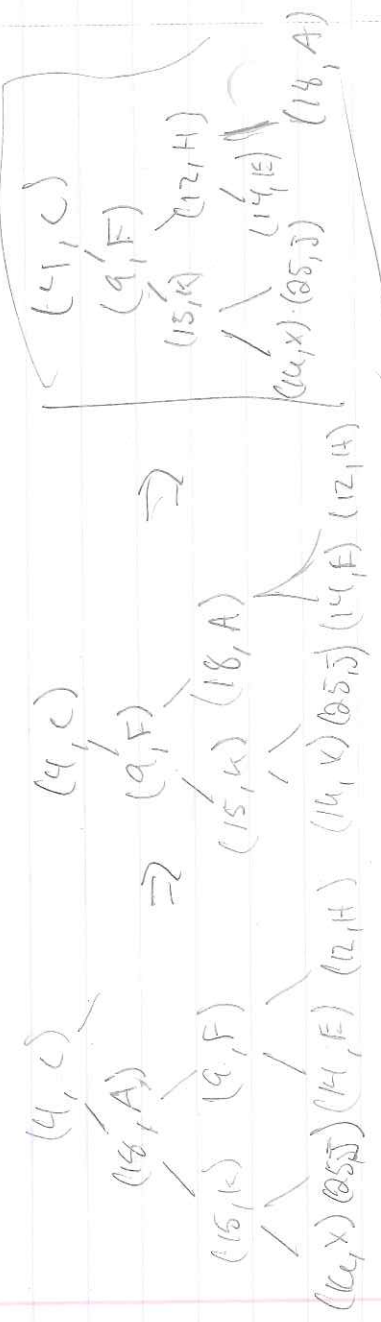
$$\log(1) + \log(2) + \dots + \log(n) \leq n \log n$$

$$\log(n) \leq n \log(n)$$

for a  $n \geq n_0$

so  $\sum n \log n$

9.22) Right side does not change



9.35) start node =  $n$ . If the start has  
key greater than stop. else record  $n$   
and set start node to left side of  $n$ . Recur  
 $O(2n+1) = O(n)$