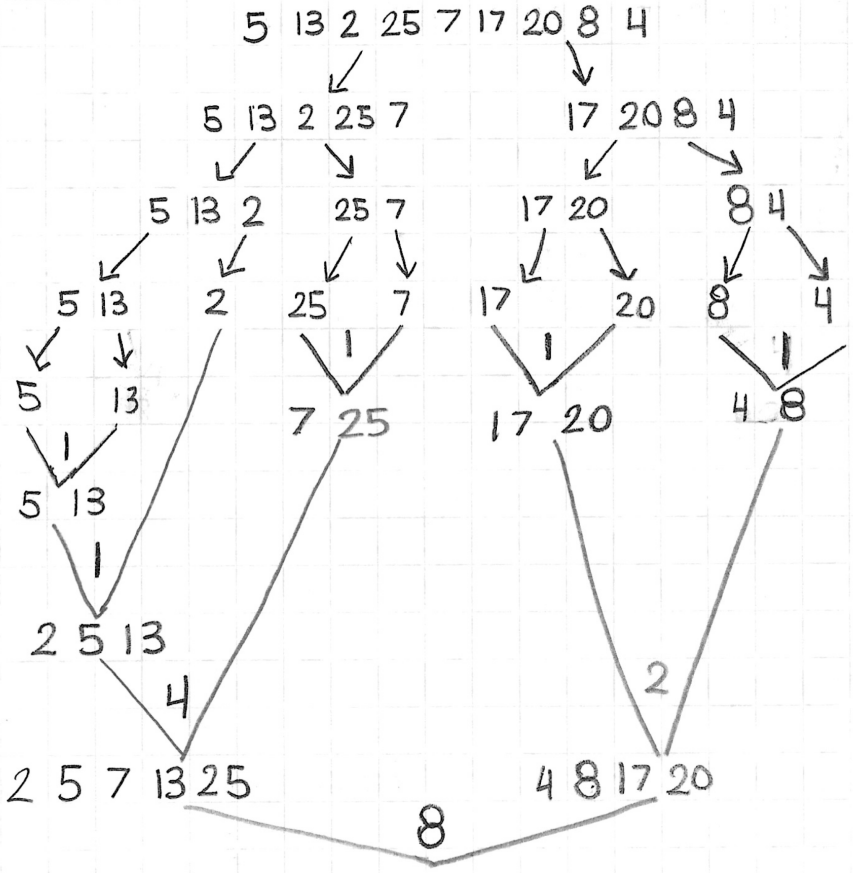
CS340 => Assignment 5

Maksim Sharoika

Problem 1)

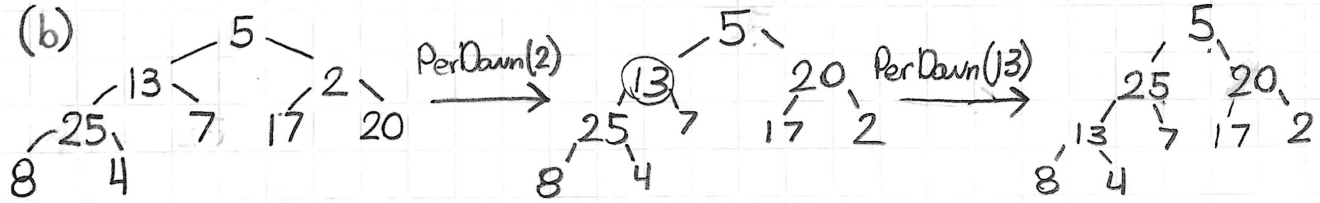
(a)



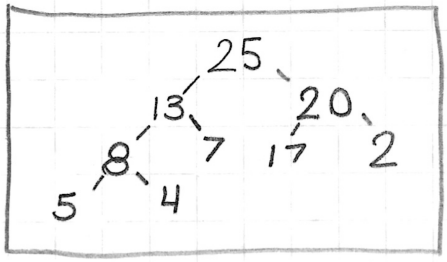
5
+ 6
+ 8
19

19 total Comparsons

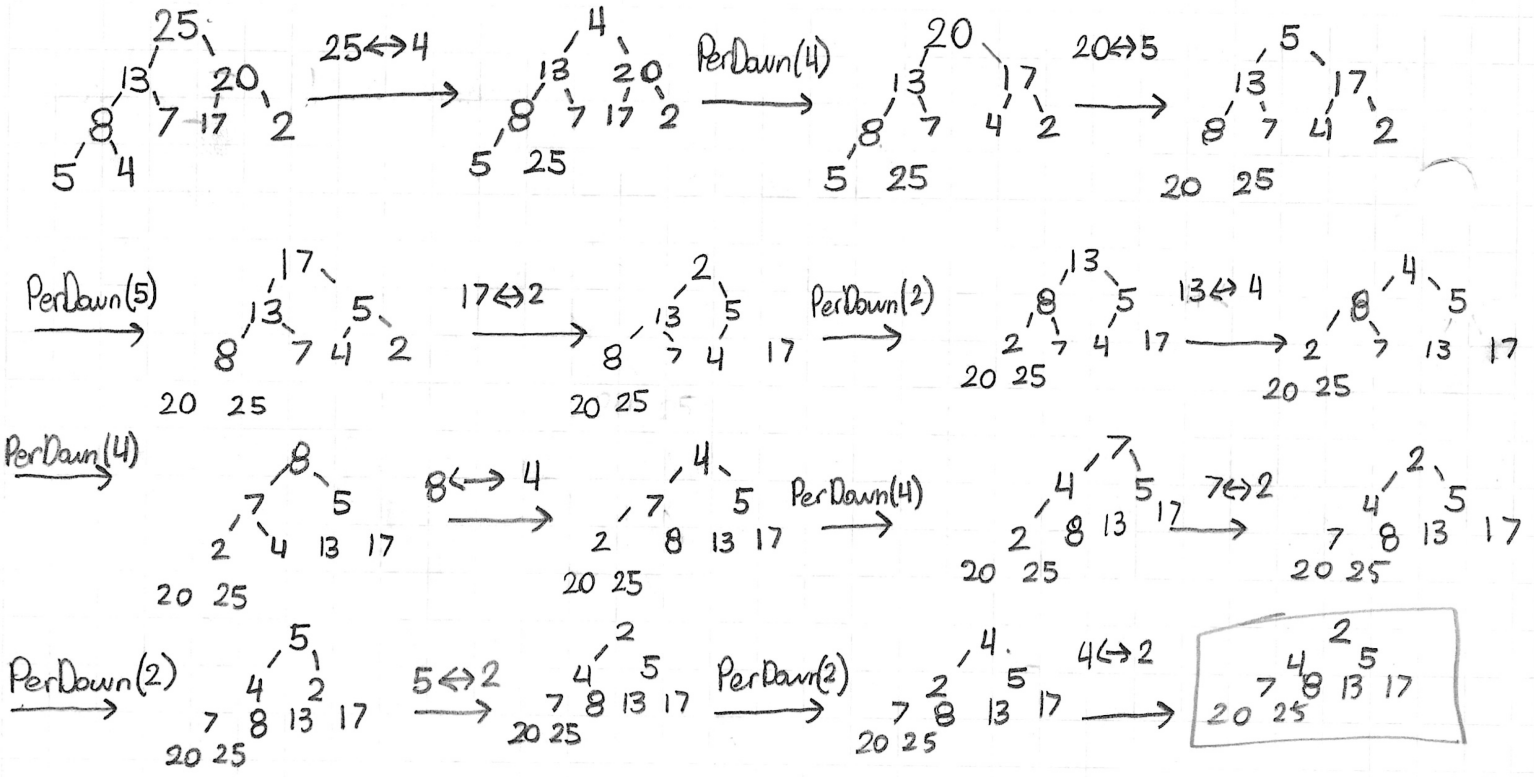
(b)



PerDown(5)

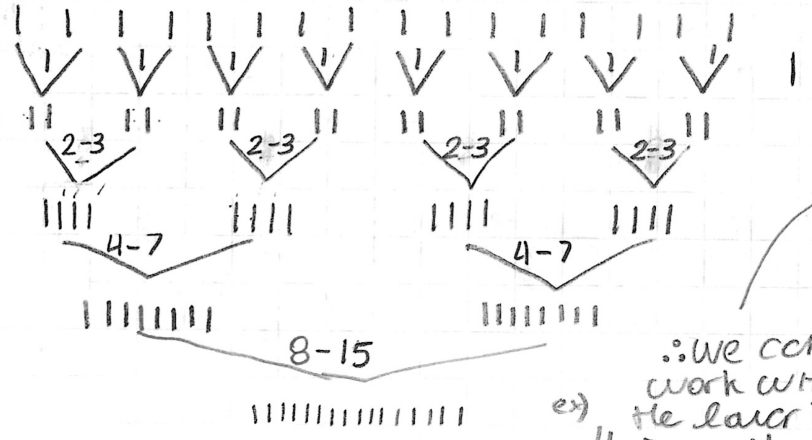


* The heap is now following the heap property.



If heap is array implementation: [2, 4, 5, 7, 8, 13, 17, 20, 25]

Problem 2) $N = 2^4 = 16$ ← Example.



∴ we can work with the layer # e.g. 4-7 → 4

* We can assume the algorithm always picks to the left or right, but doesn't swap sides
∴ we can pick the layer comparison 1, 2, 4, 8... → $\approx \frac{N}{2}$ for the step.
∴ each merging step we are looking at $\Theta(N)$.

Merge sort gives a height / number of merges of $\Theta(\log(N))$

Therefore if we take the running time / amount of steps and then the comparisons per step we can get a final worstcase running time for input of N.

$$\Theta(N) \cdot \Theta(\log(N)) = \boxed{\Theta(N \cdot \log(N))}$$

Problem 3)

(a) $a=2 \quad b=4 \quad f(N)=1=N^0 \quad z=\log_4 2=0.5$

$0.5>0 \therefore T(N)=\Theta(N^z)=\Theta(N^{1/2})=\Theta(\sqrt{N})$

(b) $a=2 \quad b=4 \quad f(N)=N^{0.5} \quad z=\log_4 2=0.5$

$0.5=0.5 \therefore T(N)=\Theta(\sqrt{N}\log(N))$

(c) $a=2 \quad b=4 \quad f(N)=N^2 \quad z=\log_4 2=0.5$

$0.5<2 \therefore$ there is a $n_0 \in \mathbb{N}$, where $c<1$ so that $2 \cdot N^2/4 \leq c N^2$ for all $N \geq N_0$.

$N_0=1$ & $C=1/2$

$T(N)=\Theta(N^2)$

(d) $a=9 \quad b=3 \quad f(N)=N^1 \quad z=\log_3(9)=2$

$2>1 \therefore T(N)=\Theta(N^z)=\Theta(N^2)$

Problem 4) $T(N)=2T(N/2)+N\log(N)$

$a=2 \quad b=2 \quad f(N)=N\log(N) \quad z=\log_2 2=1$

$f(N)=\Omega(N^1)=\Omega(N^z)$

$f(N)$ needs to be $\Omega(N^x)$ for some x that is greater than z. That value does not exist/ is not possible. Therefore we cannot apply the master theorem to this example.

↑ this is following the given rules / statement from our lecture.

Problem 5) [77, 17, 66, 19, 30, 24, 64, 14, 23]

(a) N = 9 $2^3 - 1 = 7 < 9$ Gap : 7, 3, 2.

									gap	Comparisons
(77)	17	66	19	30	24	64	(14)	23	7	1
14	(17)	66	19	30	24	64	77	(23)	7	1
(14)	17	66	(19)	30	24	64	77	23	3	1
14	(17)	66	19	(30)	24	64	77	23		1
14	17	(66)	19	30	(24)	64	77	23		1
14	17	24	(19)	30	66	(64)	77	23		1
14	17	24	19	(30)	66	64	(77)	23		1
14	17	(24)	19	30	(66)	64	77	(23)		2
(14)	(17)	23	19	30	24	64	77	66	1	1
14	(17)	(23)	19	30	24	64	77	66		1
14	(17)	(23)	(19)	30	24	64	77	66		2
14	17	19	(23)	(30)	24	64	77	66		1
14	17	19	(23)	(30)	(24)	64	77	66		2
14	17	19	23	24	(30)	(64)	77	66		1
14	17	19	23	24	30	(64)	(77)	66		1
14	17	19	23	24	30	(64)	(77)	(66)		2
14	17	19	23	24	30	64	66	77		

2 | 7 | 11

Total Comparisons : 20

(b) My code have Shellsort (N/2) Maksimsort (N/4) and Hibbard sort ($2^k - 1$), also a bunch of array manipulation functions to simplify the work.

→ The file is attached as a main.app with the assignment submission.

A random 10, 100, 1000 array was created and sorted with each (Screenshot attached)

Comparisons

	10	100	1000 ← array size
ShellGap Sort (N/2)	35	802	14757
HibbardGapSort ($2^k - 1$)	(26)	761	(14569)
MaksimGapSort (N/4)	31	(501)	14950