Erli Cai

Question 1: 10a

Problem	Input	Output	Algorithm	Runtime	Other Questions
Sorting	Unsorted Array	Sorted Array	Insertion Sort	$\theta(n^2)$	-
			Bubble Sort	$\theta(n^2)$	-
			Mergesort	$\theta(nlogn)$	T(n) = 2T(n/2) + $\theta(n)$
			Quicksort	$\theta(nlogn)$	Expect runtime: $\theta(nlogn)$, Worst runtime: $\theta(n^2)$
			Heapsort	$\theta(nlogn)$	It is in-place (no extra memory needed)
Find Minimum	Unsorted Array	Minimum Value	-	$\theta(n)$	-
	Min-heap			$\theta(1)$	-
	Max-heap			$\theta(n)$	-
	BST			$\theta(height)$	worst case: $\theta(n)$
	AVL Tree			$\theta(logn)$	-
Find kth Smallest	Unsorted Array	Element	Select	$\theta(n)$	worst case: $\theta(n)$
			Randomized Selection	$\theta(n^2)$	expected runtime: $\theta(n)$ worst case runtime:$\theta(n^2)$
	Min-heap		-	$\theta(klogn)$	
	BST			$\theta(h+k)$	-
	AVL Tree			$\theta(h+k)$	-
	AVL Tree Augmented with number of nodes in left subtree			$\theta(logn)$	-

Problem	Input	Output	Algorithm	Runtime	Other Questions
Find rank of element	Unsorted Array	Integer between 1 and n	Count the number of elements less than target element in the array	$\theta(n)$	-
	Min-heap		-	$\theta(n)$	-
	BST			$\theta(n)$	-
	AVL Tree			$\theta(n)$	-
	AVL Tree Augmented with number of nodes in left subtree			$\theta(1)$	-
Sorting Cont'd	Unsorted array of integers in range $\{1 \dots k\}$	Sorted Array	Counting sort	n+k	-
	Unsorted array of integers of length l using d digits		Radix sort	l(n+d)	-
Enumerate how many numbers are in a given interval	answer28	Integer	Range-Counting	answer29	-
MST	Undirected graph	Tree	Prim's Algorithm	$\theta(n + mlogn)$	-
	Undirected graph		Kruskal's Algorithm	$\theta(n + m)$	-
SSSP	Unweighted graph + source s	Tree	BFS	$\theta(mlogm + n)$	
	Weighted graph + source s	Tree	Bellman-Ford	$\theta(mn)$	
	Weighted graph + source s	Tree and True/False	Dijkstra	$\theta(mlogm + n)$	-
Finding cut-vertices	A graph	cut edges	Hopcroft-Tarjan's algorithm	$\theta(m + n)$	-

Data Structures Comparsion - Fill out entire table with runtimes

	Insert	**Delete (pointer known)**	**Search**	**Preprocessing (Build structure from unsorted array)**
Unsorted array	$\theta(1)$	$\theta(n)$	$\theta(n)$	$\theta(1)$
Sorted array	$\theta(n)$	$\theta(n)$	$\theta(logn)$	$\theta(nlogn)$
BST	$\theta(height)$	$\theta(height)$	$\theta(height)$	$\theta(nlogn)$
AVL Tree	$\theta(logn)$	$\theta(logn)$	$\theta(logn)$	$\theta(nlogn)$
Hash table w/ chaining, array size m	$\theta(1)$	$\theta(1+\alpha)$	$\theta(1+\alpha)$	$\theta(n)$
Hash table w/ uniform open addressing, array size m	$\theta((1-\alpha)^{-1})$	$\theta((1-\alpha)^{-1})$	$\theta((1-\alpha)^{-1})$	$\theta(n)$

Question 2: 10b

3 . A stable sorting algorithm will preserve the original order of elements with equal value.
Merge sort, counting sort, radix sort are stable.

4. It will still work if we group elements into groups of 3.
But the runtime will no longer be linear. (It will probably have O(nlogn) runtime)

5. Let X = # times we see four 1's in a row.
$X_i = 1$ if we ith to (i+3)-th digits are 1.

$X = \Sigma_{i=1}^{i=13} X_i$

$E[X_i] = P(X_i = 1) = \frac{1}{2} \times \frac{1}{2} \times \frac{1}{2} \times \frac{1}{2} = \frac{1}{16}$
$E[X] = 13 \times \frac{1}{16} = \frac{13}{16}$

6. His strategy/statement about runtime is incorrect.
Consider a decision tree for this problem.
Each node will have 2 children. And there are 2^12 cases in total.
Therefore, the decision tree will have height = 12.
The (worst-case) runtime of his algorithm is at least the height of decision tree, which is 12.
So the algorithm will have a low bound 12. So his strategy can't guarantee to find the number with at most 10 questions.