CSCI 532 – Algorithm Design
Assignment 4

Name: Badarika Namineni CWID: 50233279

Question 1: Using Figure 8.2 as a model, illustrate the operation of COUNTING-SORT on the array A = (6, 0, 2, 0, 1, 3, 4, 6, 1, 3, 2).

Solution:

1) Array A = (6, 0, 2, 0, 1, 3, 4, 6, 1, 3, 2)

There are 11 elements in the array.

min = 0 max = 6

Lets create a index from 0 to 6

index	0	1	2	3	4	5	6
Array C (Counts)	2	2	2	2	1	0	2

Array C has the count of each element of the index.

Now, lets sort them by summing the count with Previous index.

	0	1	2	3	4	5	6
SumCount C =	2	4	6	8	9	9	11

Lets write an array with the indices of no. of elements.

In our case, it is 11. So, we have 11 indices.

	1	2	3	4	5	6	7	8	9	10	11
B:	0	0	1	1	2	2	3	3	4	6	6

Step1: After 6 is added under 11, the count is reduced by 1 which makes 10.

Step2: By adding 0 at 2, the value is 1 in 0.

Step3: The 6 becomes 5 in index 2.

Step4: Since 0 has 1 it is placed under 1 in A.

Step5: The count 4 becomes 3 in 1.

Step4: 8 becomes 7 after adding 3.

Step5: Since 2 has 5 in C it is added in 5 in A.

Array C :

	0	1	2	3	4	5	6	
	0	2	4	6	8	9	9	⇒ Sum count

⇒ index

Final sorted array

B:

	1	2	3	4	5	6	7	8	9	10	11	
	0	0	1	1	2	2	3	3	4	6	6	⇒ Sorted array

⇒ index

Question 2: Suppose that we were to rewrite the for loop header in line 10 of the COUNTING-SORT as for j = 1 to A.length
Show that the algorithm still works properly. Is the modified algorithm stable?

Solution:
The algorithm works properly even with the "for j = 1 to A.length". The placement of the elements does not affect with the change in the for loop but it is NOT STABLE because the equal elements will be placed in the reverse order.

Question 3: Using Figure 8.3 as a model, illustrate the operation of RADIX-SORT on the following list of English words: COW, DOG, SEA, RUG, ROW, MOB, BOX, TAB, BAR, EAR, TAR, DIG, BIG, TEA, NOW, FOX.

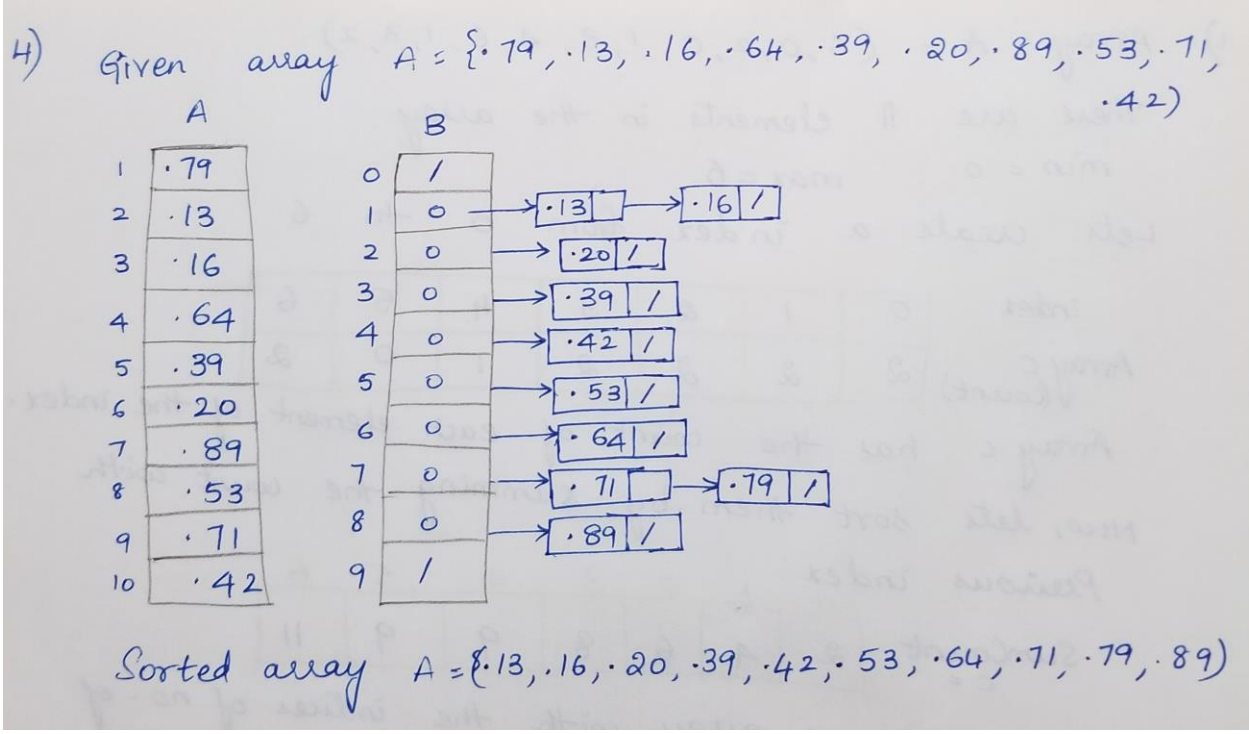
Solution:

Given words	**Sorting by right most letter**	**Sorting by middle letter**	**Sorting by left most letter**
COW	SEA	TAB	BAR
DOG	TEA	BAR	BIG
SEA	MOB	EAR	BOX
RUG	TAB	TAR	COW
ROW	DOG	SEA	DIG
MOB	RUG	TEA	DOG
BOX	DIG	DIG	EAR
TAB	BIG	BIG	FOX
BAR	BAR	MOB	MOB
EAR	EAR	DOG	NOW
TAR	TAR	COW	ROW
DIG	COW	ROW	RUG
BIG	ROW	NOW	SEA
TEA	NOW	BOX	TAB
NOW	BOX	FOX	TAR
FOX	FOX	RUG	TEA

It is sorted using Radix-sort

Question 4: Using Figure 8.4 as a model, illustrate the operation of BUCKET-SORT on the array A = [.79, .13, .16, .64, .39, .20, .89, .53, .71, .42]

Solution:



Question 5: Suppose we use a hash function h to hash n distinct keys into an array T of length m. Assuming simple uniform hashing, what is the expected number of collisions? More precisely, what is the expected cardinality of {{k, l}: k ≠l and h(k) = h(l) (k and l where k is not equal to l and h(k) = h(l)})?

Solution:

Let's use linearity of expectation to solve this. Suppose all the keys are ordered. Let X_i be the number of $l > k_i$ so that $h(l) = h(k_i)$ which is same as $\sum_{j>i} Pr(h(k_j) = h(k_i)) = \sum_{j>i} 1/m = (n-i)/m$.

Now define the random variable Y to be the total number of collisions, so that $Y = \sum_{k=l} X_{kl}$. The expected number of collisions is

$E[Y] = E[\sum_{k=l} X_{kl}]$

$= \sum_{k=l} E[X_{kl}]$

$=(n/2)*1/m$

$=n(n-1)/2*1/m$

$=\mathbf{n^2-n/2m}$

Question 6: For the set of keys {1, 4, 5, 10, 16, 17, 21}, draw binary search trees of height 2, 3, 4, 5, and 6.

Solution: