Data Structure and Algorithms

Lab6

Submitted to:	**Ms. Sehar Waqar**
Submitted by:	**SABA**
Roll no:	**2019-CE-04**
Date:	**25th May,2021**

Department of Computer Engineering

UET,Lahore

Homework Question:

In this exercise, we will explore different types of randomized algorithms. We say that a randomized algorithm is a **Las Vegas algorithm** if it is always correct (that is, it returns the right answer with probability 1), but the running time is a random variable. We say that a randomized algorithm is a **Monte Carlo algorithm** if there is some probability that it is incorrect. For example, QuickSort (with a random pivot) is a Las Vegas algorithm, since it always returns a sorted array, but it might be slow if we get very unlucky.
We will revisit the Majority Element problem to get more insight on randomized algorithms.

Answer:

Algorithm	Monte Carlo or Las Vegas?	Expected Running Time	Worst-case running time	Probability of returning a majority element
Algorithm 1	Las Vegas	O(n)	infinite	1
Algorithm 2	Monte Carlo	O(n)	O(n)	$1-\frac{1}{2^{100}}$
Algorithm 3	Las Vegas	O(n)	$O(n^2)$	1

Algorithm 1 : It always gives correct result, so it is Las Vegas Algorithm.

Expected Running Time = Choosing a random p takes O(1) and isMajority(P,p) takes O(n). So, expected running time is O(n).
Worst-case running time = Infinite (for every choice of p € P ,p is not the majority element).
Probability of returning a majority element = P[failure] = 0. That's why Probability of returning a majority element is 1.

Algorithm 2 : It does not always gives you correct result, so it is Monte Carlo Algorithm.

Expected Running Time = Choosing a random p takes O(1) and isMajority(P,p) takes O(n). So, expected running time is O(n).
Worst-case running time = O(n) whether it is wrong or right.
Probability of returning a majority element:
Let P be the probability to find majority element

P[failure] = $\frac{1}{2^{100}}$

Probability of returning a majority element = 1 - $\frac{1}{2^{100}}$

Algorithm 3 : It always gives correct result, so it is Las Vegas Algorithm.
Expected Running Time = $O(n)$.
Worst-case running time = $O(n) + O(n^2) = O(n^2)$
Probability of returning a majority element = P[failure] = 0. That's why Probability of returning a majority element is 1.
