Design and Analysis of Algorithms

Exercises 5/8

1. The Dining Philosophers Problem can be solved by Las Vegas algorithm using two steps resource reserving principle: (**1**) Try to reserve randomly one fork either from left or right. If two philosophers try to reserve the same work, they will wait for the next round. (**2**) If successful, then he tries to reserve the other fork. If again successful, he will eat. Those who fail, will try again next round. Analyse **(a)** the expected and **(b)** worst case time complexities for this algorithm when *N* is the number of philosophers.

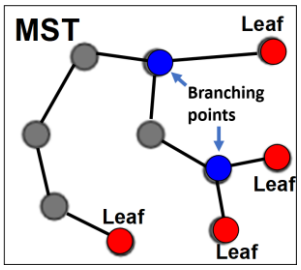
2. Design a randomized algorithm for sorting. Is your algorithm Las Vegas or Monte Carlo method?

3. The clustering animator is given here: https://cs.uef.fi/ml/software/animator/. Test the k-means in three different ways: (**a**) restarting k-means, (**b**) make the recommended drag-and-drop, (**c**) simulate random swap. Solve out empirically how many iterations are needed (on average) to solve the *last swap* (CI=1 → CI=0). Derive the corresponding probability of success for these three cases.

4. Implement Monte Carlo algorithm for calculating the area of a circle. Report the accuracy of the algorithm using k=10, 100, 1000 and 10,000 repeats.

5. Simulating dynamic linked list using array, we can achieve data structure where the search problem can be solved by O($\sqrt{N}$) time. Is this "closer" to O(log *N*) or O(*N*)? Give both mathematical and empirical argument for your answer.

6. Christofides algorithm adds links to MST to make TSP tour possible. Another idea is to remove *branching points* by re-connecting them to the selected *leaf nodes*. Is TSP allowed to have any branching points or leaf nodes? Design algorithm based on this idea. Design an algorithm to select which links to remove and where to add them. You can use the small example to demonstrate the behaviour.



Large example: