Programming Assignment 2

1. You have a large text file containing words. Given any two words, find the shortest distance (in terms of number of words) between them in the file. Can you make the searching operation in O(1) time?

2. Given an NxN matrix of positive and negative integers, write code to find the sub- matrix with the largest possible sum.

3. Write a method to sort an array of strings so that all anagrams are next to each other.

4. Imagine a (literal) stack of plates. If the stack gets too high, it might topple. Therefore, in real life, we would likely start a new stack when the previous stack exceeds some threshold. Implement a data structure SetOfStacks that mimics a real set of stacks. This SetOfStacks should be composed of several stacks, and should create a new stack once the previous one exceeds capacity. SetOfStacks push() and SetOfStacks pop() should behave identically to a single stack (that is, pop() should return the same values as it would if there were just a single stack). Implement a function popAt(int index) which performs a pop operation on a specific sub-stack.

5. Design an algorithm and write code to find the first common ancestor of two nodes in a binary tree. Avoid storing additional nodes in a data structure.

6. Given an infinite number of quarters (25 cents), dimes (10 cents), nickels (5 cents) and pennies (1 cent), write a recursive code to calculate the number of ways of representing n cents.

7. Given a sorted array of n integers that has been rotated an unknown number of times, give an O(log n) algorithm that finds an element in the array. You may assume that the array was originally sorted in increasing order.

SOLVE ANY FOUR OF THE ABOVE QUESTIONS.

Allowed Programming Languages
You may use any programming language of the following: C, C++, JAVA, Python.

Submission procedure:
Please follow these instructions carefully:

- Comment your code as detailed as possible.
- Include a multiple line comment in the beginning of code to briefly explain your solution.
- Show the complexity of your solution in Big-O notation form.
- Create a folder and name it in the following format **yourname_yoursurname_matricola** for example **mario_rossi_1673378.** Put all the files inside this folder after you have named every problem solution **yourname_yoursurname_matricola_problemnumber,** for example problem one **mario_rossi_1673378_1.** Then you either submit the zipped version of the folder by emailing it at **terolli@di.uniroma1.it** or share it on Google Drive with **terolli@di.uniroma1.it.**

COPYING IS STRICTLY FORBIDDEN.

Deadline: 10 June 2016

Enjoy. :) :) :)