COMP 4030/6030 Spring 2018. Assignment 7
Due: April 5, 2018 (before class starts)

1. (30 points) Given a list L consisting of n **unordered** objects, an item is a majority element if its frequency is greater than n/2. For example, in the list [C, C, T, T, C, T, C, C], C is the majority item, because the frequency of C is 5, which is greater than 4. Another example, the list [C, C, T, C, C, S, T, T] has no majority item. These objects are not ordered. You can only compare them using equality as follows: L[i] == L[j].

   Write an iterative program, which counts the frequency of each object, to determine the majority element of a list of objects. Your program should return an object or None, in case there is no majority element.

   def iterative_majority_finding(L):
       # your code goes here

2. (10 points) Specify the running time function of your program.

3. (30 points) Write a Python program to find the majority element of a list of unordered objects, based on the following strategy and observations:

   • Divide the list in two halves (say, Left and Right) and find the majority element in each half.
   • If the original list, L, has a majority element, it must be the majority element of either the list Left or the list Right.
   • Take care of the base case.

4. (10 points) Specify the running time function of this program in Problem 3 and describe its running time using Theta.

5. (20 points) In this "investment" problem, you are given $**T**, which is the total amount you can invest on **n** different products. Product **x** (x is a number between 0 and n-1) has a cost, **c[x]** and a profit **p[x]**. The goal is to maximize the profit without spending more than $**T**.

   Example:

Product	A	B	C	D
Cost	24	10	10	7
Profit	24	18	18	10

   If T = 24, the best investment is to spend $20 (out of $24) to buy products B and C and make $36 in profit.

The API of your program looks like this:

def invest(T, Costs, Profits):
        # return a number, which is the best profit you can make.

You will have to do two things:
- Explain your strategy cleanly and neatly in English.
- Write a Python program to implement your strategy.

Points will be given based on the soundness of your strategy and a faithful implementation of that strategy. If your strategy always gives the best investment, that's great. You will get all points. In case your strategy does not always give the best investment, you still get 90% of the points if you describe a sound strategy and your implementation reflects that strategy.

Plagiarism Policy:
You can discuss how to solve the problems with your classmates, but the solution must be your own. Using other people's solution will result in a zero for the assignment and possible additional penalties.

Submission:
Put your name as part of the file name and upload your submission to eCourseware Dropbox.