Assignment : Design Analysis of Algorithms

Submission due date April 21,2017

Instructions

- All the code files need to be zipped into a single file named as [id1]_[id2]_[id3].tar.gz or [id1]_[id2]_[id3].zip and submitted into the assignment folder by the due date.
- Only the code files submitted to the assignment folder will be considered for demo purposes.
- Marks will be assigned individually. Each member is expected to be able to demo the code he/she worked on and explain the approach as well as various parts of the code if necessary.
- It is best for each member to focus mainly on one problem. However as a group you are expected to cover all problems.
- In case of plagiarism, you will be given a 0 for the assignment. If you use pieces of code that you found online, clearly cite the source. However this assignment is expected to substantially reflect your effort.

1. **Nice rectangles**
   There is a grid that consists of 2 rows and $n$columns. A non-empty rectangle of cells is called nice if and only if the sum of numbers in its cells equals 0. What is the maximum possible number of nice rectangles that can be chosen such that no two chosen rectangles should share a cell?

   Input
   The first line of the input contains an integer n $(1 \leq n \leq 1000)$which represents the number of columns in the grid. The next two lines contain numbers in the grid.

   Output
   Print one integer, denoting the maximum possible number of cell-disjoint nice rectangles.

   Example
   Input
   6
   70 70 70 70 70 -15
   90 -60 -30 30 -30 15

   Output
   3

2. Given two sequences of integers, write a program to determine a common increasing subsequence of maximal possible length. Sequence $S_1, S_2, \ldots, S_N$ of length N is called an increasing subsequence of a sequence $A_1, A_2, \ldots, A_M$ of length M if there exist $1 \leq i_1 < i_2 < \ldots < i_N \leq M$ such that $S_j = A_{i_j}$ for all $1 \leq j \leq N$, and $S_j < S_{j+1}$ for all $1 \leq j < N$.

   Input
   Each sequence is described by its length M $(1 \leq M \leq 500)$ and M integer numbers A_i $(-231 \leq A_i < 231)$ the sequence itself.

   Output
   On the first line of the output file print L, the length of the greatest common increasing subsequence of both sequences. On the second line print the subsequence itself. If there are several possible answers, output any of them.

   Sample Input
   5
   1 4 2 5 -12
   4
   -12 1 2 4

   Sample Output
   2
   1 4

Graph algorithms

Write a program to read a graph from a text file and do the following. The graph to be used for this question is stored in a file with .gml extension, it contains a listing of nodes and edges along with edge weights. It may be useful to write scripts to transform data to a more convenient format. You may write your own code for the tasks specified, however it may be preferable to use existing libraries since these are usually written with scalability in mind. Note that the given graph contains about 16000 vertices and about 120K edges.

3. For the given graph get the largest connected component. Report the number of nodes in the largest connected component as well as output the corresponding node ids.

4. Compute the diameter of the largest connected component. The diameter of a graph is the longest shortest path i.e. the maximum of minimum weight paths between any pair of nodes. While shortest path algorithms could be used, this is infeasible for large graphs. Instead, randomly select pairs of vertices and evaluate minimum weight path (also referred to as shortest path) between them.