# Assignment 1. Due April 15

CS180: Algorithms and Complexity
Spring 2015

Guidelines for submitting the solutions:

- It is strongly recommended to use LaTeX or other word processing software for submitting the homework. This is not mandatory but will be helpful both for you and for us. If submitting electronically, send your solutions to `sumeets@cs.ucla.edu`.
- Grades will take into account both the correctness and the clarity of the solutions. You are responsible for communicating your solution in a simple and understandable way. Sloppy answers will receive fewer or no points even if they are"correct". Unless otherwise specified, all answers will need to be throughly justified with complete proofs. However, you may use results proved in class without proofs as long as you state them clearly.
- Most importantly, make sure you adhere to the policies for academic honesty set out on the course webpage. The policies will be enforced strictly.

**Problems:**

1. Prove an upper bound for the function $T$ defined by the following relations: $T(1) = 1$, $T(n) \leq 34 \cdot T(n/17) + 17n$. To get full points, your answer should be tight up to $O(1)$ factors. [3 points]
2. Give an example of an instance of stable matching problem with at least three different stable matchings. That is, for some positive integer $n$, give ranking lists for each of $n$ doctors $L_1, ..., L_n$ and ranking lists for each of $n$ hospitals $L'_1, ..., L'_n$ such that there are at least three different stable matchings with these rankings. To get full points, it is enough to specify the lists and write down (or clearly draw) three stable matchings in your instance. [3 points]

3. You are given the information about the flight network between various cities as a graph $G = (V, E)$: the vertex set $V$ is the set of cities and there is an edge between two cities $u, v$ if there is a direct flight between the two cities. Your goal now is to determine the worst-case number of stops (or connecting flights) a person needs to make to travel between two different cities. For example, if $V = \{LA, Chicago, NYC, SF, London\}$ and

$$E = \{\{LA, Chicago\}, \{Chicago, NYC\}, \{LA, NYC\}, \{SF, LA\}, \{SF, Chicago\}, \{NYC, London\}\}.$$

Then, the answer is 2 as you need that many stops to go from $SF$ to $London$.

Give an algorithm which runs in time $O(|V| \cdot (|V| + |E|))$ to solve this problem. You should state the algorithm clearly, prove its correctness and prove its time complexity. [6 points]