CSC 2420 Fall 2017, Assignment 2
Due date: November 22, by 11am

It is certainly preferable for you to solve the questions without consulting a published source. However, if you are using a published source then you must specify the source and you should try to improve upon the presentation of the result.

If you would like to discuss any questions with someone else that is fine BUT at the end of any collaboration you must spend at least one hour playing video games or watching two periods of Maple Leaf hockey or maybe even start reading a good novel before writing anything down.

If you do not know how to answer a question, state "I do not know how to answer this (sub) question" and you will receive 20% (i.e. 2 of 10 points) for doing so. You can receive partial credit for any reasonable attempt to answer a question BUT no credit or arguments that make no sense.

In class I can clarify any questions you may have about this assignment.

1. Consider the following makespan problem in the restricted machines model. The input is a set of unit processing time jobs $\{J_1, \ldots, J_n\}$ where each job J_i can be scheduled on some subset $S_j \subseteq \{M_1, \ldots, M_m\}$ of the m identical machines.

   - Show that the online greedy algorithm will not be better than a $\Omega(\log m)$ approximation. The online greedy algorithm schedules jobs on the least loaded machine and breaks ties in favour of the machine with the smallest index. Can you achieve this inapproximation for arbitrarily large $n >> m$?
   - Show how to optimally solve this makespan problem by reducing the problem to optimal flows.

2. Suppose we have a maximum flow f with $val(f) \geq 1$ in an flow network $\mathcal{F} = (G, s, t, c)$ with integral capacities.

   - Does there always exist an edge e such that by decreasing the capacity $c(e)$ of e by one unit to $c(e)-1$, the value of the maximum flow is decreased by exactly one unit? Does your answer depend on the network having integral capacities? Briefly justify your answer.
   - Assuming again integral capacities, we want to increase the flow value by two units and want to do so by increasing the capacity of some edges by one unit. Can this always be done? When it can be done, explain how you could efficiently determinine the fewest number of edges needed to do this.

3. Recall the Max-2-SAT problem. Given a CNF formula with exactly 2 literals per clause, the goal is to find an assignment satisfying the maximum number of clauses. Consider an oblivious local search algorithm that starts from an arbitrary truth assignment, and at every iteration, looks at the local neighborhood $N(\tau_i)$ of the current truth assignment τ_i, and chooses the truth assignment in the neighborhood satisfying the maximum number of clauses. The algorithm stops when τ_i satisfies the maximum number of clauses among all truth assignments in $N(\tau_i)$. In this question, you will analyze the approximation ratio obtained by the algorithm upon termination. In the lecture, we analyzed the 1-flip

neighborhood, i.e., when $N(\tau)$ contains all truth assignments that differ from τ on at most one variable. We showed that upon termination, the algorithm provides a 2/3 approximation. We now improve upon this by enlarging the neighborhood.

- Consider the larger neighborhood that allows flipping all variables. That is, $N(\tau)$ contains all truth assignments that differ from τ on at most one variable, or on all variables. Show that upon termination, the algorithm provides a 3/4 approximation.

4. Consider the weighted max-di-cut problem. That is, we are given an edge weighted directed graph $G = (V, E)$ with weight function $w : E \to \mathbb{R}^{\geq 0}$, and the goal is to find a subset of vertices $S \subseteq V$ so as to maximize the total weight of edges from S to $V \setminus S$, i.e., maximize $f(S) = \sum_{(u,v)\in E, u\in S, v\in V\setminus S} w(u, v)$.

   - Show that f is a non-monotone submodular set function.
   - What is the expected approximation ratio of the naive randomized algorithm which assigns each vertex to S independently with probability 1/2? Prove both the upper and the lower bound.
   - What do you get when you de-randomize this algorithm by the method of conditional expectation? Express the resulting deterministic algorithm in the form of a greedy algorithm. What information did you need for each vertex when deciding whether to assign it to S?

5. Consider the following weighted partial vertex cover problem. We are given a graph $G = (V, E)$ with node costs $c : V \to \mathbb{Q}^+$ and edge costs $d : E \to \mathbb{Q}^+$. The goal is to find a partial cover $V' \subseteq V$ so as to minimize the sum of the costs of nodes in V' plus the costs of edges "not covered" by V' (i.e., edges whose neither endpoints are in V').

   - Provide a $\{0, 1\}$ integer linear program (ILP) for this problem.
   - Using LP relaxation and deterministic rounding, design a polynomial-time constant approximation algorithm. What is the approximation ratio that you obtain?

6. Consider the following water flow problem. There is a *directed* circular ring network $G = (V, E)$ with n nodes in which each node is connected to the previous and the next node on the circle. That is, $V = \{0, 1, \ldots, n-1\}$, and E consists of edges $i \to (i+1) \mod n$ and $i \to (i-1) \mod n$ for each $i \in V$. There is a set of requests $\{C_1, \ldots, C_t\}$, all of which must be fulfilled. Satisfying request C_j requires sending p_j units of flow from node s_j to node f_j. For each request, the flow can be sent either in the clockwise direction or in the counterclockwise direction. The load L_e on a *directed edge* e is the total flow passing through the edge. Your goal is to minimize $\max_{e \in E} L_e$.

   - Formulate this as a $\{0, 1\}$ integer linear program (ILP). Indicate the intended meaning of each variable in the ILP.
   - Using an LP relaxation followed by rounding, show you how can derive a polynomial-time constant approximation algorithm. What is the constant that you obtain?