Approximation Algorithms (236521) - HW4

Submission deadline: 25.1.19 (Friday)

Guidelines:
• The homework should be submitted individually.
• You are required to provide complete and formal proofs of your claims.
• The homework should be submitted to Ariel Kulik’s mailbox (on the 5th floor of Taub building).

Question 1 (20 pt)

Approximations based on linear programming – omitted proof:
• This question refers to the constraints in the LP relaxation of the problem of non-preemptive scheduling of jobs with release times on a single machine. Let \( C_j \) be a variable indicating the completion time of job \( j \in \{1,2,\ldots,n\} \). Recall that, given a solution \( \bar{C} \) for the LP, we number the jobs such that \( C_1 \leq C_2 \leq \ldots \leq C_n \).

Prove the following lemma from class:

**Lemma 1.** Given the variables \( C_j \), if the constraints in (2) are satisfied for the \( n \) subsets \( S_1, S_2, \ldots, S_n \), then the constraints are satisfied for any subset \( S \subseteq N \).

Question 2 (40 pt)

In the Min2SAT problem, we have variables \( X = \{x_1, x_2, \ldots, x_n\} \) and a set of clauses \( C = \{c_1, c_2, \ldots, c_m\} \). Each clause is of the form \( c_i = z_{i1} \lor z_{i2} \), where \( z_{i1}, z_{i2} \) are literals – a variable or negation of a variable. The goal is then to determine if there exists an assignment \( v : X \rightarrow \{T, F\} \), such that all clauses are satisfied, and if such exists, finding an assignment which sets the minimum number of variables to be \( T \).

1. (10 pt) Write an integer linear program for this problem.
2. (30 pt) Suggest an algorithm which uses the relaxation of the above problem, together with rounding technique of your choice. The algorithm should provide a 2-approximation.

Question 3 (40 pt)

Assume that we have a scheduling problem with precedence constraints between jobs imposed. We denote \( i \prec j \) if in any feasible schedule, job \( i \) must be completely processed before job \( j \) begins processing. Consider a variation of the single-machine scheduling problem, in which we have precedence constraints (all jobs are given at the beginning). That is, we are given \( n \) jobs with processing times \( p_j > 0 \) and weights \( w_j \geq 0 \). The goal is to find a non-preemptive schedule on a single machine, that is feasible with respect to the precedence constraints \( \prec \), and that minimizes the weighted sum of completion times \( \sum_{i=1}^n w_j C_j \). Give a 2-approximation algorithm for this problem.
Question 4 (bonus, 15 pt)

We presented in class an $f$-approximation algorithm for Set Cover that is based on solving a linear programming relaxation of the problem and adding a set $S$ to the solution if $x_S \geq 1/f$, where $f$ is the maximum frequency of any input element.

1. (10 pt) Suppose that we modify the algorithm such that the set $S$ is added to the solution iff $x_S > 0$. Show that the new algorithm also yields an $f$-approximation for the problem.
   
   Hint: Write the dual LP and use the complementary slackness conditions.

2. (5 pt) Show how an $f$-approximation algorithm for Set Cover can be used to obtain a 2-approximation for the Vertex Cover problem.

Reminder: In Vertex Cover we are given a graph $G = (V, E)$ and a non-negative weight function $w : V \rightarrow \mathbb{R}_+$. We say a subset $C \subseteq V$ is a cover of $G$ if for every $(u, v) \in E$ either $u \in C$ or $v \in C$. The objective is to find a cover $C$ of $G$ of minimal weight, $\sum_{v \in C} w(v)$. 