CSC373— Algorithm Design, Analysis, and Complexity — Spring 2018

Assignment 3: NP and FILP — Due: 9am, Thurs, Mar 29, 2018

Submit answers to the course MarkUs page, `https://markus.teach.cs.toronto.edu/csc373-2018-01`. Your solutions must be in one PDF file named A3.pdf. Your solutions can be neatly hand-written and scanned. You do not need to repeat the questions themselves, and you can make use of any material in the class's lecture notes or tutorials simply by citing them.

You are encouraged to work in groups of size at most five. If you do work in a group then you must, of course, submit this work to MarkUs as a group.

1. [**10pts**] **Interval Scheduling.** Suppose we are given a set of jobs $J(k)$ which start at time s_k and end at time f_k, where $s_k < f_k$ are non-negative integers for $k = 1, 2, \ldots, K$. We assume all the finish times are distinct. We wish to schedule a maximum size subset of these jobs such that no two scheduled jobs overlap in time (i.e., the open time intervals (s_j, f_j) and (s_k, f_k) do not intersect for any pair of jobs $J(j)$ and $J(k)$ that are scheduled). We represent such a maximal subset of jobs by the set of their indices, $S \subseteq \{1, 2, \ldots, K\}$.

   (a) [**5pts**] Consider the intersection graph for this problem $G = (V, E)$, where the vertices, say v_n, are in one to one correspondence with the jobs, $J(n)$ and there is an edge $(v_n, v_m) \in E$ iff $n \neq m$ and jobs $J(n)$ and $J(m)$ intersect (i.e., $(s_n, f_n) \cap (s_m, f_m) \neq \emptyset$). Show that this leads to a straight forward poly-time reduction of the interval scheduling problem using the independent set problem on G (i.e, IntervalSched$(\{J(n)\}_{n=1}^{N}) \leq_p$ searchIndepSet(G).

   (b) [**5pts**] Here we briefly consider what might be special about these intersection graphs to allow for a poly-time solution. Presumably the fact that each job runs in single time interval embedded in a common time axis is important for the existence of a polynomial time solution.

   Perhaps this is due to the fact that the jobs can be ordered, say by finish time. To investigate this, suppose we consider changing the above intersection graph G to a directed graph, where there is a directed edge $(v_j, v_k) \in E$ iff jobs $J(j)$ and $J(k)$ intersect and $f_j < f_k$. Show that the resulting graph G is a DAG (directed acyclic graph). Is there a simple greedy algorithm for the maximum independent set problem on a DAG, or is this problem also NP-hard?

2. [**10pts**] **Degree12Tree.** Using a polynomial reduction with one of the problems on slide 18 of the NP-complete lecture notes, show that the following problem is NP-complete:

   Degree12Tree(G): Given any (undirected) graph $G = (V, E)$, does there exist a subgraph $T = (V, F)$ of G (i.e., with $F \subseteq E$) such that, T is a spanning tree of G and the maximum degree of any vertex of T is 12?

   Note, the **degree** of a vertex v in $T = (V, F)$ is defined to be the number of edges in F that include v as one of their endpoints.

3. [**20pts**] **FILP.** Consider the feasibility problem for integer linear programming:

   FILP$(A, \vec{b})$: Given a $m \times n$ matrix A and a $m \times 1$ vector $\vec{b}$, where all elements of A and $\vec{b}$ are integers, consider the feasible set $F \subset \mathbb{Z}^n$ where $\vec{x}$ must satisfy

$$\begin{aligned} A\vec{x} &\leq \vec{b}, \\ \vec{x} &\geq \vec{0}. \end{aligned}$$

   That is, the decision problem FILP$(A, \vec{b})$ is to determine if there exist an integer-valued feasible solution $\vec{x} \in \mathbb{Z}^n$ of the above inequalities (i.e., is the set F defined above non-empty)?

   (a) [**2pts**] Show that FILP is in NP.

(b) **[5pts]** Show that FILP is NP-complete by showing a poly-time reduction 3-Sat $\leq_p$ FILP. In this reduction you must have a one to one correspondence between the logical variables x_i^S in the 3-SAT formula and integer variables x_i^F in FILP, where x_i^S is true (false) iff $x_i^F = 1$ (0, respectively). Given the simplicity of this representation you can drop the superscripts S and F from these two sets of variables.

(c) **[5pts]** Show that Set-Cover $\leq_p$ FILP with a similarly direct reduction. In this reduction you must use the variables $\vec{x}$ in FILP which are either 0 or 1, and $x_k = 1$ iff the k^{th} set in the Set-Cover problem has been selected.

(d) **[3pts]** Use the reduction in part (c) to show that

$$\text{Set-Cover} \notin \text{co-NP} \implies \text{FILP} \notin \text{co-NP}.$$

(e) **[5pts]** Consider the EqualThirds problem considered in Assignment 2, Question 4. That is, given a list of positive integers $P = (p_1, p_2, \ldots, p_n)$ we want to determine if it is possible to partition $I(n) = \{1, 2, \ldots, n\}$ into three mutually disjoint subsets, say $S_i \subset I(n)$ for $i = 1, 2, 3$, such that: a) $S_i \cap S_j = \emptyset$ for $i \neq j$; b) $I(n) = \cup_{i=1}^{3} S_i$; and c) the following equation holds:

$$\sum_{k \in S_1} p_k = \sum_{k \in S_2} p_k = \sum_{k \in S_2} p_k = \left[\sum_{k \in I(n)} p_k \right] /3. \tag{1}$$

You must use a reduction for which $\vec{x}$ is a $2n \times 1$ vector with values in $\{0, 1\}$, where $x_i = 1$ for $1 \leq i \leq n$ iff $i \in S_1$, and $x_{i+n} = 1$ for $1 \leq i \leq n$ iff $i \in S_2$.

(f) **[0pts] Hard and Unmarked.** Show a polytime reduction of the Hamiltonian Cycle problem using FILP (i.e, show HamCycle $\leq_p$ FILP without going through another NP-complete problem).