Worth: 7.5% (*best four of the five assignments*) **Due:** *before* 6:00pm on Wed. 20 March

Required filename for MarkUs submission: `a4.pdf` **Remember to write the *full name* and *MarkUs username* of *every* group member (up to three) prominently on your submission.**

---

*Please read and understand the policy on Collaboration given on the Course Information Sheet. Then, to protect yourself, list on the front of your submission **every** source of information you used to complete this homework (other than your own lecture and tutorial notes). For example, indicate clearly the **name** of every student from another group with whom you had discussions, the **title and sections** of every textbook you consulted (including the course textbook), the **source** of every web document you used (including documents from the course webpage), etc.*

*For each question, please write up detailed answers carefully. Make sure that you use notation and terminology correctly, and that you explain and justify what you are doing. Marks **will** be deducted for incorrect or ambiguous use of notation and terminology, and for making incorrect, unjustified, ambiguous, or vague claims in your solutions.*

---

1. **Tasks and Tools**

   You have m different tasks to complete and to help you, n different software tools you could purchase, each with a positive integer cost c_i. You have no choice about which tasks to complete (you must complete them all), but you get to choose which tools you will purchase.

   Tools are not necessary to complete tasks; however, certain tasks have additional costs if they are completed without the use of specific tools. Information about these additional costs is provided through non-negative integer *dependencies*: for all pairs i, j, $d_{i,j}$ is the additional cost of completing task i without tool j—a dependency can be equal to zero to indicate that there is no additional cost.

   Finally, there are known *incompatibilities* between certain tools: for each tool i, you have a list of all the other tools with which tool i is incompatible. Obviously, it is not possible to install incompatible tools at the same time.

   Give a Linear Programming (or Integer Programming) solution to determine which tools to purchase in order to minimize your total cost (from the purchase of tools and the dependencies). Please write your algorithm in pseudo-code and include a high-level English description (either separately or as comments throughout your algorithm). Also, remember to justify the correctness of your solution—this is important!

2. ***P*, *NP*, and *coNP***

   For each decision problem below, state whether it belongs to *P*, *NP*, or *coNP*—make the strongest claim that you can. Justify your answer by giving an algorithm of the appropriate type together with a brief argument that it satisfies the required properties.

   (a) ALLSMALLCYCLES ("ASC" for short)

   Input: Graph $G = (V, E)$, edge weights $w : E \to \mathbb{Z}^+$, vertex $s \in V$, bound $B \in \mathbb{Z}^+$.

   Question: Does *every* cycle in G that starts at vertex s have total weight *no more than* B?

   (b) ALLLARGECYCLES ("ALC" for short)

   Input: Graph $G = (V, E)$, edge weights $w : E \to \mathbb{Z}^+$, vertex $s \in V$, bound $B \in \mathbb{Z}^+$.

   Question: Does *every* cycle in G that starts at vertex s have total weight *at least* B?

(c) SomeLargeCycles ("SLC" for short)

Input: Graph $G = (V, E)$, edge weights $w : E \to \mathbb{Z}^+$, vertex $s \in V$, bound $B \in \mathbb{Z}^+$.

Question: Does *some* cycle in G start at vertex s and have total weight *at least* B?

(d) SomeSmallCycles ("SSC" for short)

Input: Graph $G = (V, E)$, edge weights $w : E \to \mathbb{Z}^+$, vertex $s \in V$, bound $B \in \mathbb{Z}^+$.

Question: Does *some* cycle in G start at vertex s and have total weight *no more than* B?

3. Properties of $\to_p$

(a) Give a detailed argument that for all decision problems D_1, D_2, D_3, $D_1 \to_p D_2$ and $D_2 \to_p D_3$ implies $D_1 \to_p D_3$.

(b) Suppose D' is some *NP*-complete problem and D is any decision problem. What can we conclude about D if we know that $D' \to_p D$ and $D \to_p D'$? Make the strongest claim you can and prove it.

(c) If D is *NP*-complete and $D \in coNP$, what can we conclude? Make the strongest claim you can and prove it.