

Pontificia Universidad Católica de Chile
Escuela de Ingeniería
Departamento de Ciencia de la Computación

IIC2213 — Lógica para Ciencia de la Computación —
IIC2214 — Teoría de la Computación —

TAREA 6

1er semestre 2026

Publicación: Lunes 15 de junio.
Entrega: **Jueves 25 de junio hasta las 23:59 horas, via buzón en Canvas**.

Indicaciones

- La tarea es **individual**. Los casos de copia serán sancionados con la reprobación del curso con nota 1,1.
- Cada una de las dos preguntas tiene una nota entre 1 y 7 (hay un punto base). La nota de la tarea es el promedio de ambas preguntas.
- La tarea **debe** ser escrita en LaTeX. No se aceptarán tareas escritas de otra forma. No olvide indicar su **nombre**, **número de alumno/a** y **sigla del curso (IIC2213 o IIC2214)**.
- No se aceptan atrasos (salvo que utilice su cupón `#problemaexcepcional`).

Pregunta 1

Definimos coNP como la clase de todos los lenguajes L cuyo complemento $\overline{L}$ está en NP. Decimos que un problema K es coNP-hard si para todo $L \in \text{coNP}$ se tiene $L \leq_p K$. Decimos que un problema K es coNP-completo si $K \in \text{coNP}$ y K es coNP-hard.

(a) Demuestre que si un lenguaje K es NP-completo, entonces su complemento $\overline{K}$ es coNP-completo.

(b) Demuestre que si existe un lenguaje coNP-completo que está en NP, entonces NP = coNP.

Solución

(a) Sea K un lenguaje NP-completo. Veamos que $\overline{K}$ es coNP-completo. El complemento de $\overline{K}$, que es K, está en NP, luego $\overline{K} \in \text{coNP}$. Sea $L \in \text{coNP}$ arbitrario. Tenemos que $\overline{L} \in \text{NP}$. Como K es NP-hard, sigue que $\overline{L} \leq_p K$. Notar que si $L_1 \leq_p L_2$, entonces $\overline{L}_1 \leq_p \overline{L}_2$. (basta tomar la misma reducción). Obtenemos que $L \leq_p \overline{K}$. Como L es arbitrario, se deduce que $\overline{K}$ es coNP-hard. Concluimos que $\overline{K}$ es coNP-completo.

(b) Supongamos que existe un lenguaje coNP-completo K tal que $K \in \text{NP}$. Veamos primero que NP $\subseteq$ coNP. Sea $L \in \text{NP}$ arbitrario. Tenemos que $\overline{L} \in \text{coNP}$ y como K es coNP-hard, se cumple que $\overline{L} \leq_p K$. Como $K \in \text{NP}$ y por la parte (a), deducimos que $\overline{L} \in \text{NP}$. Obtenemos que $L \in \text{coNP}$. Para ver que coNP $\subseteq$ NP, si $L \in \text{coNP}$, entonces $\overline{L} \in \text{NP}$, y como NP $\subseteq$ coNP, sigue que $\overline{L} \in \text{coNP}$. Lo último implica que $L \in \text{NP}$.

Pregunta 2

(a) (2.0 pts) Demuestre que el siguiente problema es NP-completo:

Double-SAT $= \{\langle\varphi\rangle \mid \varphi$ fórmula proposicional satisfacible por al menos 2 valuaciones$\}$.

(b) (4.0 pts) Recuerde que $\mathbb{Z}^n$ denota al conjunto de vectores enteros de dimensión n y $\mathbb{Z}^{n\times m}$ al conjunto de matrices enteras de dimensión $n \times m$. Demuestre que el siguiente problema es NP-completo:

$$\text{ILP} = \{\langle A, \bar{b}\rangle \mid A \in \mathbb{Z}^{n\times m},\ \bar{b} \in \mathbb{Z}^n, \text{ y existe } \bar{x} \in \mathbb{Z}^m \text{ tal que } A\bar{x} \leq \bar{b}\}.$$

Solución

(a) Veamos primero que Double-SAT está en NP. Podemos tomar una MTN que, dada fórmula φ, primero adivina 2 valuaciones σ_1 y σ_2, y luego verifica que sean distintas y que $\sigma_1(\varphi) = 1$ y $\sigma_2(\varphi) = 1$. Esta MTN toma tiempo polinomial y decide Double-SAT.
Veamos ahora que Double-SAT es NP-hard. Probaremos que SAT $\leq_p$ Double-SAT. Sea una fórmula φ. La reducción computa $\varphi' = \varphi \wedge (p \vee \neg p)$, donde p es una variable nueva que no

aparece en φ. Veamos que $\langle\varphi\rangle \in$ SAT $\iff$ $\langle\varphi'\rangle \in$ Double-SAT. Supongamos que φ es satisfacible, luego existe valuación σ tal que $\sigma(\varphi) = 1$. Podemos tomar σ_0 como la extensión de σ tal que $\sigma_1(p) = 0$, y σ_1 como la extensión de σ tal que $\sigma_1(p) = 1$. Ambas valuaciones hacen verdadera a φ' y son distintas. Obtenemos que φ' tiene al menos dos valuaciones que la hacen verdadera. Para el otro lado, supongamos que φ' tiene al menos dos valuaciones σ y σ' que la hacen verdadera. En particular, σ hace verdadera a φ (lo mismo para σ'). Podemos tomar la valuación σ restringida a las variables de φ, y esto nos da una valuación de φ que la hace verdadera.

(b) Veamos que ILP está en NP. Podemos tomar la MTN que, dado A y $\bar{b}$, adivina un vector de tamaño polinomial $\bar{x} \in \mathbb{Z}^m$, y verifica que $A\bar{x} \leq \bar{b}$.
Veamos que ILP es NP-hard. Probaremos que 3-SAT $\leq_p$ IPL. Sea $\varphi = C_1 \wedge \cdots \wedge C_m$, una fórmula en 3CNF con m cláusulas y variables $p_1, \ldots, p_n$. Definiremos una matriz A y un vector $\bar{b}$, a partir de un sistema de desigualdades lineales sobre las variables enteras $x_1, \ldots, x_n$. Para cada variable p_i, agregaremos al sistema las desigualdades $-x_i \leq 0$ y $x_i \leq 1$. Esto fuerza a que $0 \leq x_i \leq 1$, osea, x_i es 0 o 1, y luego, las soluciones al sistema corresponden a valuaciones para φ. Dado un literal ℓ, definimos

$$f(\ell) = \begin{cases} x_i & \text{si } \ell = p_i, \\ 1 - x_i & \text{si } \ell = \neg p_i. \end{cases}$$

Sin pérdida de generalidad, asumiremos que cada claúsula C_i tiene exactamente 3 literales (si tiene menos, repetimos algunos). Por cada cláusula $C_i = \ell_{i,1} \vee \ell_{i,2} \vee \ell_{i,3}$ de φ, agregamos al sistema la desigualdad lineal

$$f(\ell_{i,1}) + f(\ell_{i,2}) + f(\ell_{i,3}) \geq 1.$$

Esta desigualdad corresponde a la condición de que alguno de los literales de la cláusula sea verdadero, es decir, que la cláusula sea satisfecha.
Por último, notemos que podemos dejar el sistema de desigualdades en la forma requerida para las entradas de ILP, dejando las constantes a un solo lado y posiblemente multiplicando por -1 cuando necesitemos invertir la dirección de la desigualdad. Es decir, podemos calcular A y $\bar{b}$ asociado al sistema. Notemos que esta es una reducción de tiempo polinomial. Notar también que por construcción, una valuación que hace verdadera a φ, nos da una solución al sistema y viceversa. Concluimos que $\langle\varphi\rangle \in$ 3SAT $\iff$ $\langle A, \bar{b}\rangle \in$ ILP.

Ejemplo. Consideremos

$$\varphi = (p_1 \vee \neg p_2 \vee \neg p_3) \wedge (\neg p_1 \vee p_4) \wedge (p_2 \vee \neg p_3 \vee p_4 \vee p_5).$$

Para cada $i \in \{1, 2, 3, 4, 5\}$ agregamos las desigualdades $0 \leq x_i \leq 1$, que equivalen a $x_i \leq 1$ y $-x_i \leq 0$. Además, para las cláusulas agregamos las siguientes desigualdades:

$$x_1 + (1 - x_2) + (1 - x_3) \geq 1, \text{ que equivale a } -x_1 + x_2 + x_3 \leq 1.$$

También agregamos

$$(1 - x_1) + x_4 \geq 1, \text{ que equivale a } x_1 - x_4 \leq 0.$$

Finalmente, agregamos

$$x_2 + (1 - x_3) + x_4 + x_5 \geq 1, \text{ que equivale a } -x_2 + x_3 - x_4 - x_5 \leq 0.$$

Por lo tanto, el sistema de programación entera que resulta de la reducción es el siguiente:

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & -1 \\ -1 & 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & -1 & 0 \\ 0 & -1 & 1 & -1 & -1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{pmatrix} \leq \begin{pmatrix} 1 \\ 0 \\ 1 \\ 0 \\ 1 \\ 0 \\ 1 \\ 0 \\ 1 \\ 0 \\ 1 \\ 0 \\ 0 \end{pmatrix}.$$