Pontificia Universidad Católica de Chile
Profesor: Sebastián Bugedo
Ayudante: Sofía Errázuriz

Lógica para ciencias de la computación - IIC2213
Ayudantía 12
23 de junio, 2023

El objetivo de esta ayudantía es repasar los conceptos del curso entre las cases 00 y 06, para esto se hará un resumen de la materia, acompañado de los siguientes ejercicios:

Lógica proposicional

Ejercicio 1.

1. Defina una función *largo* $|\cdot|: \mathcal{L}(P) \to \mathbb{N}$ que entregue el número de símbolos de una fórmula (contando paréntesis, variables y conectivos).
2. Defina una función $var : \mathcal{L}(P) \to \mathbb{N}$, que entregue el número de ocurrencias de variables proposicionales para cualquier $\varphi \in \mathcal{L}(P)$ (incluye repeticiones).
3. Demuestre que toda fórmula $\varphi \in \mathcal{L}(P)$ que no contiene el símbolo $\neg$ cumple que $|\varphi| \leq 4 \cdot var(\varphi)^2$.

Solución. Dado P, se define la función $|\cdot| : \mathcal{L}(P) \to \mathbb{N}$ para cada $\varphi \in \mathcal{L}(P)$ según

1. Si $\varphi = p$ con $p \in P$, entonces $|\varphi| = 1$, ya que el único símbolo de φ es p.
2. Si $\varphi = (\neg\psi)$ con $\psi \in \mathcal{L}(P)$, entonces $|\varphi| = |\psi| + 1$, ya que , ya que al negar ψ se agregan ambos paréntesis y el símbolo $\neg$.
3. Si $\varphi = (\psi_1 \circ \psi_2)$ con $\psi_1, \psi_2 \in \mathcal{L}(P)$ y $\circ \in \{\wedge, \vee, \to, \leftrightarrow\}$, entonces $|\varphi| = |\psi_1| + |\psi_2| + 3$ ya que a los símbolos de ψ_1 y ψ_2 se le agregarán los paréntesis y $\circ$.

Por otro lado, definimos $var : \mathcal{L}(P) \to \mathbb{N}$ para cada $\varphi \in \mathcal{L}(P)$ según

1. Si $\varphi = p$ con $p \in P$, entonces $var(\varphi) = 1$, ya que la única variable proposicional de φ es p.
2. Si $\varphi = (\neg\psi)$ con $\psi \in \mathcal{L}(P)$, entonces $var(\varphi) = var(\psi)$, ya que no se agrega ninguna variable
3. Si $\varphi = (\psi_1 \circ \psi_2)$ con $\psi_1, \psi_2 \in \mathcal{L}(P)$ y $\circ \in \{\wedge, \vee, \to, \leftrightarrow\}$, entonces $var(\varphi) = var(\psi_1) + var(\psi_2)$.

Finalmente, probemos que para toda fórmula $\varphi \in \mathcal{L}(P)$ tal que no contiene negaciones, se tiene que $|\varphi| \leq 4 \cdot var(\varphi)^2$. Para esto, nuevamente usaremos inducción.

1. Si $\varphi = p$ con $p \in P$, entonces $|\varphi| = 1 < 4 = 4 \cdot var(\varphi)^2$

2. Si $\varphi = (\psi_1 \circ \psi_2)$ con $\psi_1, \psi_2 \in \mathcal{L}(P)$ y $\circ \in \{\wedge, \vee, \rightarrow, \leftrightarrow\}$, si tenemos que $|\psi_i| \leq 4 \cdot var(\psi_i)^2$ para $i = 1, 2$, entonces $|\varphi| = |\psi_1| + |\psi_2| + 3 < 4 \cdot var(\psi_i)^2 + 4 \cdot var(\psi_i)^2 + 3$. Por otro lado notemos que $var(\varphi)^2 = (var(\psi_1) + var(\psi_2))^2$ y $3 < 8var(\psi_1)var(\psi_2)$. Juntando todas las desigualdades se concluye lo pedido.

$\square$

Ejercicio 2. Sea Σ un conjunto satisfacible de fórmulas proposicionales, y φ una fórmula proposicional que no es una tautología. Además, suponga que Σ y φ no tienen variables proposicionales en común. ¿Es cierto que $\Sigma \not\models \varphi$? Demuestre o de un contraejemplo.

Solución. Llamemos $p_1, \ldots, p_n$ a las variables proposicionales de Σ y $q_1, \ldots, q_m$ a las variables proposicionales de φ. Si Σ es satisfacible, entonces existe una valuación σ_1 tal que $\sigma_1(\Sigma) = 1$, además como φ no es tautología, existe valuación σ_2 tal que $\sigma_2(\varphi) = 0$. Con esto, definimos una nueva valuación $\sigma_3 : P \to \{0, 1\}$ tal que

$$\sigma_3(r) = \begin{cases} \sigma_1(r) & si \quad r \in \{p_1, \ldots, p_n\} \\ \\ \sigma_2(r) & si \quad r \in \{q_1, \ldots, q_m\} \\ \\ 0 & \text{en otro caso} \end{cases}$$

Entonces tenemos que $\sigma_3(\Sigma) = 1$, pero $\sigma_3(\varphi) = 0$, es decir $\Sigma \not\models \varphi$. $\square$

Ejercicio 3. Dado un grafo $G = (N, A)$, una secuencia de nodos $(a_1, \ldots, a_n)$ es un *camino en* G si para todo $i \in [1, n-1]$ se tiene que $(a_i, a_{i+1}) \in A$. Decimos que G contiene un *circuito Hamiltoniano* si existe un camino $(a_1, \ldots, a_n)$ en G tal que:

- n es el número de nodos de N,
- $a_i \neq a_j$ para cada $i, j \in \{1, \ldots, n\}$ tales que $i \neq j$,
- $(a_n, a_1) \in A$.

Construya una fórmula proposicional φ tal que G contiene un circuito Hamiltoniano si y sólo si φ es satisfacible.

Solución. Dado un grafo $G = (V, E)$ definimos variables proposicionales $p_{i,j}$ con $i, j \in V$ y las oraciones

$$\varphi_1 = \bigwedge_{(i,j) \in E} p_{i,j} \wedge \bigwedge_{(i,j) \notin E} \neg p_{i,j}$$

$$\varphi_2 = \bigvee_{\{k_1, \ldots, k_n\} \in V} \left(p_{k_1, k_n} \wedge \bigwedge_{i \in [n-1]} p_{k_i, k_{i+1}} \right)$$

Entonces definimos $\varphi = \varphi_1 \wedge \varphi_2$. Demostraremos que φ es satisfacible si y solo si hay un circuito Hamiltoniano.

($\Rightarrow$) Notemos que si φ es satisfacible, entonces existe una valuación σ tal que $\sigma(\varphi) = 1$. En particular, esto quiere decir que $\sigma(\varphi_1) = 1$ entonces la valuación debe ser

$$\sigma(p_{i,j}) = \begin{cases} 1 & si \quad (i,j) \in E \\ \\ 0 & si \quad (i,j) \notin E \end{cases}$$

Además, debe existir un ordenamiento de los vértices $k_1, \ldots, k_n$ tal que $\sigma(p_{k_i,k_{i+1}}) = 1$ y $\sigma(p_{k_1,k_n})$ es decir, existen todas las aristas (k_i, k_{i+1}) y (k_1, k_n). Es decir, existe un circuito Hamiltoniano.

($\Leftarrow$) Si hay un circuito Hamiltoniano, entonces tomamos la valuación

$$\sigma(p_{i,j}) = \begin{cases} 1 & si \quad (i,j) \in E \\ \\ 0 & si \quad (i,j) \notin E \end{cases}$$

Esta valuación claramente satisface φ_1 y para φ_2, notemos que si tomamos $k_i = a_i$ del circuito Hamiltoniano, entonces $\sigma(p_{k_i,k_{i+1}}) = 1$ y $\sigma(p_{k_1,k_n})$, por lo tanto $\sigma(\varphi_2) = 1$. En conclusión, $\sigma(\varphi) = 1$. □

Máquinas de Turing y complejidad

Ejercicio 4. Demuestre que si tenemos dos lenguajes $\mathcal{L}_1$ y $\mathcal{L}_2$ tales que existe una reducción de $\mathcal{L}_1$ a $\mathcal{L}_2$, entonces:

- Si $\mathcal{L}_2$ es decidible, entonces $\mathcal{L}_1$ es decidible.
- Si $\mathcal{L}_1$ es indecidible, entonces $\mathcal{L}_2$ es indecidible.

Solución. Notemos que la segunda afirmación es contrapositiva de la primera, entonces basta probar que si $\mathcal{L}_2$ es decidible, entonces $\mathcal{L}_1$ es decidible. Para esto, supongamos que $\mathcal{L}_2$ es decidible, entonces existe una máquina de Turing $\mathcal{M}$ que decide $\mathcal{L}_2$. Si f es la reducción de $\mathcal{L}_1$ a $\mathcal{L}_2$, entonces podemos definir una máquina $\mathcal{M}'$ tal que aplica f y después $\mathcal{M}$, claramente esto decide $\mathcal{L}_1$ por definición de reducción, entonces el lenguaje es decidible. □

Ejercicio 5. Construya una reducción de 3-COL a 4-COL, utilice esto para probar inductivamente que k-coloreabilidad es un problema NP-completo

Solución. En esta demostración solo se dará una idea y se espera que quienes quieran completen los detalles. La reducción que se construye es agregar un vértice al grafo y unirlo a todos los otros vértices. Si el grafo original es 3-coloreable, al agregarle un vértice podemos pintarlo de un color diferente, entonces es 4-coloreable. Si el grafo generado es 4-coloreable, entonces el vértice extra debe ser de un color diferente a todos los otros, por lo tanto, el grafo original es 3-coloreable. □