Pontificia Universidad Católica de Chile
Departamento de Ciencia de la Computación
IIC1253 - Matemáticas Discretas

Pauta Tarea 1

2º semestre 2025 - Profesores M. Arenas - A. Kozachinskiy - M. Romero

---

Pregunta 1

(a) (2.0 pts) Encuentre una fórmula en CNF que sea equivalente a:

$$(p \to (q \to (r \to s))).$$

Debe explicar claramente su desarrollo para obtener la fórmula.

(b) (2.0 pts) Definimos el conectivo binario XOR según la siguiente tabla de verdad:

p	q	XOR
0	0	0
0	1	1
1	0	1
1	1	0

Demuestre que el conjunto $\{\mathsf{XOR}, \to\}$ es funcionalmente completo.
(*Hint:* Recuerde que el conjunto $\{\neg, \to\}$ es funcionalmente completo.)

(c) (2.0 pts) Demuestre que el conjunto $\{\mathsf{XOR}\}$ **no** es funcionalmente completo.
(*Hint:* Demuestre que la fórmula $\neg p$ no se puede expresar utilizando sólo XOR.)

Solución

(a) Podemos aplicar la regla de la implicancia ($\varphi \to \psi \equiv \neg\varphi \vee \psi$) tres veces y obtener lo siguiente:

$$\begin{aligned}
(p \to (q \to (r \to s))) &\equiv \neg p \vee (q \to (r \to s)) \\
&\equiv \neg p \vee (\neg q \vee (r \to s)) \\
&\equiv \neg p \vee (\neg q \vee (\neg r \vee s)) \\
&\equiv \neg p \vee \neg q \vee \neg r \vee s
\end{aligned}$$

La última fórmula está en CNF, ya que es la conjunción de disyunción de literales (variable o negación de variable). Recordar que la conjunción puede ser de 1 sólo término, como en este caso.

Una forma alternativa, es aplicar la demostración vista en clases de que cada tabla de verdad tiene una CNF equivalente. La demostración que vimos es como sigue. Tomamos la negación de la fórmula, en este caso, $\neg(p \to (q \to (r \to s)))$ y construimos su tabla de verdad. Aplicamos el método que vimos en clases para obtener una DNF a partir de una tabla de verdad: miramos todas las filas que se hacen verdaderas, para cada una de ellas se construye una conjunción de literales que sólo se hace verdadera en esa fila, y luego se toma la disyunción de todas esas filas. Si se aplica el método en este caso se obtiene $(p \wedge q \wedge r \wedge \neg s)$ (hay sólo una fila en la tabla que se hace verdadera, la fila $p = 1, q = 1, r = 1, s = 0$). Obtenemos que:

$$\neg(p \to (q \to (r \to s))) \equiv (p \wedge q \wedge r \wedge \neg s)$$

Finalmente hay que aplicar la ley de doble negación y la ley de De Morgan para obtener la CNF equivalente final:

$$\begin{aligned} (p \to (q \to (r \to s))) &\equiv \neg(\neg(p \to (q \to (r \to s)))) \\ &\equiv \neg(p \wedge q \wedge r \wedge \neg s) \\ &\equiv \neg p \vee \neg q \vee \neg r \vee s \end{aligned}$$

(b) Siguiendo el hint, basta escribir $\neg$ en términos de XOR y $\to$. Notar que $\neg\varphi \equiv \varphi \to \alpha$ donde α es alguna fórmula que siempre se haga falsa. Una posible formula que siempre se hace falsa es $\varphi \ \mathsf{XOR}\ \varphi$. Obtenemos que:

$$\neg\varphi \equiv \varphi \to (\varphi \ \mathsf{XOR}\ \varphi)$$

Verifiquemos lo de arriba. Tomemos una valuación arbitraria. Si $\neg\varphi$ es verdadero, entonces φ es falso, y lo de la derecha también es verdadero (ya que el antecedente del implica es falso). Si $\neg\varphi$ es falso, entonces φ es verdadero, y lo de la derecha también es falso (ya que el antecedente del implica es verdadero y la consecuencia es falsa). Otra forma de verificar la equivalencia es hacer una "tabla de verdad" con todos los posibles valores de verdad que podría tener φ.

(c) Siguiendo el hint, hay que demostrar que $\neg p$ no se puede escribir sólo usando XOR. Por contradicción, supongamos que existe una fórmula φ que sólo usa XOR tal que $\neg p \equiv \varphi$. Podemos tomar la valuación que hace falso a la variable p. Con esta valuación $\neg p$ es verdadero, pero φ es falso, ya que todas las subfórmulas de φ deben evaluar a falso. Recordar que φ sólo usa la variable p y XOR, y que $0 \ \mathsf{XOR}\ 0 = 0$. Esto contradice el hecho de que son equivalentes.

Distribución de puntajes:

(a) 1.0 pts por dar una fórmula correcta. 1.0 pts por explicar claramente el desarrollo.

(b) 1.0 pts por dar una forma correcta de escribir $\neg$ usando XOR y $\rightarrow$. 1.0 pts por demostrar que la forma es efectivamente correcta.

(c) 2.0 pts por demostrar claramente que $\neg p$ no se puede expresar con XOR.

Descuentos y puntajes parciales a criterio del corrector.

Pregunta 2

(a) (4.0 pts) El problema del *cuadrado latino* de 4×4 se define como sigue. Tenemos un tablero de 4×4 casillas. Algunas de las casillas están *ocupadas*, vale decir, tienen un número entre $\{1, 2, 3, 4\}$. El resto de las casillas están *libres*. El objetivo es verificar si existe una *solución*, esto es, una forma de asignarle números entre $\{1, 2, 3, 4\}$ a las casillas libres, tal que en cada una de las 4 filas y en cada una de las 4 columnas, los números que aparecen sean distintos. Por ejemplo, una posible instancia al problema puede ser el siguiente tablero:

		1	
	3		
			4
2		**3**	

Una posible solución es la siguiente:

4	2	**1**	3
1	**3**	4	2
3	1	2	**4**
2	4	**3**	1

Por otra parte, el siguiente tablero **no** tiene solución (verifíquelo):

		1	
	3		**1**
			4
2		**3**	

Para describir el conjunto de casillas ocupadas usaremos triples de la siguiente forma: un triple (i, j, k), donde $1 \leq i, j, k \leq 4$, indica que la casilla en la fila i y columna j está ocupada con el número k. Por ejemplo, las casillas ocupadas del primer ejemplo quedan descritas por:

$$\{(1, 3, 1), (2, 2, 3), (3, 4, 4), (4, 1, 2), (4, 3, 3)\},$$

mientras que en el segundo ejemplo quedan descritas por:

$$\{(1,3,1),(2,2,3),(2,4,1),(3,4,4),(4,1,2),(4,3,3)\}.$$

Dado un tablero con casillas ocupadas $\{(i_1,j_1,k_1),\ldots,(i_m,j_m,k_m)\}$, escriba una fórmula φ en la lógica proposicional tal que:

el tablero tiene solución si y sólo si φ es satisfacible.

Para esto SOLO debe utilizar variables proposicionales $x_{i,j,k}$, donde $1 \leq i,j,k \leq 4$, que indican que la casilla de la fila i y columna j recibe el número k.

(b) (2.0 pts) Utilizando su fórmula proposicional de la parte anterior y el solver **Z3**, encuentre una solución para el siguiente tablero:

2			
		1	
	4		**3**
3			

Debe pegar su código de Python junto con la solución obtenida en su documento LaTeX. Recuerde también subir su archivo `.py` como fue indicado en las instrucciones.

Solución

(a) Supongamos que tenemos un tablero del cuadrado latino de 4×4, con casillas ocupadas $\{(i_1,j_1,k_1),\ldots,(i_m,j_m,k_m)\}$. Para definir la fórmula proposicional que representa este problema usamos las siguientes variables:

$x_{i,j,k}$: indica que la casilla de la fila i y columna j recibe el número k, donde $1 \leq i,j,k \leq 4$.

Debemos imponer las siguientes restricciones:

1. Para cada fila i y cada columna j, la casilla en esa fila y columna recibe un único número.

$$\bigwedge_{i=1}^{4}\bigwedge_{j=1}^{4}\bigvee_{k=1}^{4}(x_{i,j,k} \wedge \bigwedge_{\ell \neq k} \neg x_{i,j,\ell}).$$

Otra solución acá es separar la restricción en dos: "recibe al menos un número" y "no puede recibir más de un número":

$$\bigwedge_{i=1}^{4}\bigwedge_{j=1}^{4}\bigvee_{k=1}^{4} x_{i,j,k} \qquad \wedge \qquad \bigwedge_{i=1}^{4}\bigwedge_{j=1}^{4}\bigwedge_{k<k'} \neg(x_{i,j,k} \wedge x_{i,j,k'}).$$

2. Para cada fila i, no pueden haber dos números iguales en casillas distintas de esa fila:

$$\bigwedge_{i=1}^{4} \bigwedge_{j_1<j_2} \bigwedge_{k=1}^{4} \neg(x_{i,j_1,k} \wedge x_{i,j_2,k}).$$

3. Para cada columna j, no pueden haber dos números iguales en casillas distintas de esa columna:

$$\bigwedge_{j=1}^{4} \bigwedge_{i_1<i_2} \bigwedge_{k=1}^{4} \neg(x_{i_1,j,k} \wedge x_{i_2,j,k}).$$

4. Incorporamos las casillas ocupadas $\{(i_1, j_1, k_1), \ldots, (i_m, j_m, k_m)\}$:

$$\bigwedge_{\ell=1}^{m} x_{i_\ell,j_\ell,k_\ell}.$$

La fórmula final es la conjunción de las fórmulas anteriores.

Pueden haber otras formas de expresar las mismas restricciones. Por ejemplo, las dos últimas se pueden escribir como cláusulas usando la ley de De Morgan:

$$\bigwedge_{i=1}^{4} \bigwedge_{j_1<j_2} \bigwedge_{k=1}^{4} (\neg x_{i,j_1,k} \vee \neg x_{i,j_2,k}) \qquad \bigwedge_{j=1}^{4} \bigwedge_{i_1<i_2} \bigwedge_{k=1}^{4} (\neg x_{i_1,j,k} \vee \neg x_{i_2,j,k}).$$

Si en la primera restricción, sólo se impone que cada casilla reciba al menos un número, la solución también es correcta. En estricto rigor, no es necesario imponer que sea un único numero. Si la fórmula es satisfacible y una casilla tiene más de un número, basta escoger uno de esos números para cada casilla, y esto será una solución.

(b) Ver código Python de ejemplo en nuestro repositorio.

Distribución de puntajes:

(a) 1.0 pts por dar una fórmula correcta para cada una de las 4 restricciones.

(b) 2.0 pts por entregar el código Python y una solución válida al tablero.

Descuentos y puntajes parciales a criterio del corrector.