UNIVERSIDAD DE CHILE
ESCUELA DE INGENIERIA
DEPARTAMENTO DE CIENCIAS DE LA COMPUTACION

Matemáticas Discretas para la Computación - CC3101
Guía 1

1. Dado oraciones α, β de la lógica proposicional, demuestre que $\alpha \models \beta$ y $\beta \models \alpha$ si y sólo si $\models (\alpha \leftrightarrow \beta)$ (es decir, $\alpha \models \beta$ y $\beta \models \alpha$ si y sólo si $(\alpha \leftrightarrow \beta)$ es una tautología).

2. Dado conjunto de oraciones Σ de la lógica proposicional, además de oraciones α y β, demuestre que $\Sigma \models \alpha \to \beta$ si y sólo si $\Sigma \cup \{\alpha\} \models \beta$.

3. Dado conjunto de oraciones Σ de la lógica proposicional, además de oraciones α y β, demuestre que si α es una tautología, entonces $\Sigma \cup \{\alpha\} \models \beta$ si y sólo si $\Sigma \models \beta$.

4. Dado conjunto de oraciones Σ de la lógica proposicional, además de oraciones φ, ψ y θ, demuestre que si $\varphi \to \psi$ es una tautología, entonces $\Sigma \cup \{\varphi, \psi\} \models \theta$ si y sólo si $\Sigma \cup \{\varphi\} \models \theta$.

5. Dado conjunto de oraciones Σ de la lógica proposicional, además de oraciones α y β tal que α y $\Sigma \cup \{\beta\}$ no tienen variables proposicionales en común. ¿Es cierto que $\Sigma \models \beta$ si y sólo si $\Sigma \cup \{\alpha\} \models \beta$?

6. Sea p una variable proposicional. Para oraciones proposicionales α, β defina recursivamente la *oración obtenida desde α reemplazando p por β*, denotado por $\alpha[\beta/p]$, como sigue:

   - $\alpha[\beta/p] \ = \ \begin{cases} \alpha & \text{si } \alpha \text{ es proposición atómica y } \alpha \neq p \\ \beta & \text{si } \alpha = p \end{cases}$
   - $(\alpha_1 \vee \alpha_2)[\beta/p] \ = \ \alpha_1[\beta/p] \vee \alpha_2[\beta/p]$
   - $(\neg\alpha)[\beta/p] \ = \ \neg\alpha[\beta/p]$

   Demuestre que si α_1 y α_2 son equivalentes entonces para todo oración β se tiene que $\beta[\alpha_1/p]$ y $\beta[\alpha_2/p]$ también son equivalentes.

7. Defina recursivamente el *dual* de una oración proposicional ϕ, denotado por ϕ^*, como sigue:

   - $p^* = \neg p$, si $p \in P$;
   - $(\alpha \vee \beta)^* = \alpha^* \wedge \beta^*$;
   - $(\alpha \wedge \beta)^* = \alpha^* \vee \beta^*$;
   - $(\neg\alpha)^* = \neg\alpha^*$.

   Demuestre que para toda oración proposicional ϕ se tiene que ϕ^* es equivalente a $\neg\phi$.

8. Sea $\oplus$ el conectivo lógico binario definido como sigue: El valor de verdad de $\alpha \oplus \beta$ es 1 si y sólo si el valor de verdad de α es distinto del valor de verdad de β.

   ¿Es el conjunto $\{\oplus, \leftrightarrow\}$ de conectivos lógicos funcionalmente completo?

9. El conectivo lógico NOR es definido de la siguiente forma:

p	q	p NOR q
0	0	1
0	1	0
1	0	0
1	1	0

Demuestre que el conectivo NOR basta para expresar todas las oraciones proposicionales.

Formalmente, demuestre que para cada oración proposicional α que utiliza los conectivos $\{\neg, \vee, \wedge\}$ es posible encontrar una oración proposicional α^*, que sólo utiliza el conectivo NOR, y tal que $\alpha \equiv \alpha^*$.

10. El conectivo ternario MAYORIA es definido de la siguiente forma:

p	q	r	MAYORIA(p, q, r)
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

Demuestre que el conectivo MAYORIA por si solo no puede expresar todas las oraciones proposicionales que utilizan las variables proposicionales p, q, r.

11. ¿Es $\{\neg, \texttt{MAYORIA}\}$ funcionalmente completo?

12. El conectivo unario $\perp$ es definido de la siguiente forma:

p	$\perp p$
0	0
1	0

Este conectivo usualmente se denota sin la letra proposicional porque su valor de verdad es siempre 0 (por ejemplo, denotamos $p \wedge (\perp q)$ como $p \wedge \perp$).

Demuestre que con los conectivos $\{\neg, \texttt{MAYORIA}, \perp\}$ es posible expresar todas las oraciones proposicionales (es decir, todas las oraciones que pueden ser escritas con los conectivos usuales $\{\neg, \vee, \wedge\}$).

13. Sea $P = \{p, q, \dots\}$ un conjunto de proposiciones y sea f una fila de la tabla de verdad para las proposiciones en P. Defina Σ_f como el conjunto de todas las oraciones de la lógica proposicional que utilizan proposiciones en P y cuyo valor de verdad es 1 en la fila f.

Demuestre que para cualquier conjunto Σ de oraciones que utilizan proposiciones en P, si $\Sigma_f \subseteq \Sigma$ y Σ es satisfacible, entonces $\Sigma_f = \Sigma$.

14. Recuerde que un conjunto de fórmulas Θ en la lógica proposicional es satisfacible si y solo si existe una fila de la tabla de la verdad que hace verdadera a cada fórmula θ en Θ.

    Sea Σ un conjunto de fórmulas en la lógica proposicional. Asuma que el conjunto de variables proposicionales que son mencionadas en Σ es infinito (y, por tanto, Σ es también infinito).

    Decimos que Σ es *finitamente* satisfacible si y solo si todo subconjunto finito de Σ es satisfacible.

    Sea p una variable proposicional cualquiera. Demuestre que si Σ es finitamente satisfacible entonces $\Sigma \cup \{p\}$ o $\Sigma \cup \{\neg p\}$ es finitamente satisfacible (note que es posible que ambos lo sean).

15. Un *literal* es una variable proposicional p o su negación $\neg p$. Decimos que p es un literal *positivo* y que $\neg p$ es un literal *negativo*.

    Una *cláusula* es una disyunción de literales. Una cláusula es de *Horn* si y solo si contiene a lo más un literal positivo. Una fórmula es de *Horn* si y solo si es una conjunción de cláusulas de Horn.

    Demuestre que existe una cláusula que no es equivalente a ninguna fórmula de Horn.

16. Una *cláusula* es una fórmula de la lógica proposicional de la forma $\ell_1 \vee \cdots \vee \ell_n$, donde cada ℓ_i es un *literal*, es decir, una proposición p o su negación $\neg p$. Por ejemplo, $p \vee \neg q \vee r$ es una cláusula.

    La *regla de resolución* de la lógica proposicional establece lo siguiente: Si C_1 y C_2 son cláusulas y p es una variable proposicional, entonces desde las cláusulas $(C_1 \vee p)$ y $(C_2 \vee \neg p)$ es posible deducir la cláusula $C_1 \vee C_2$. Por ejemplo, desde las cláusulas $(p \vee \neg q \vee r)$ y $(\neg r \vee s)$ es posible deducir la cláusula $p \vee \neg q \vee s$.

    Demuestre que la regla de resolución es correcta. Esto es, si C_1 y C_2 son cláusulas y p es una variable proposicional, entonces

    $$\{(C_1 \vee p),\ (C_2 \vee \neg p)\} \models C_1 \vee C_2.$$

17. Formalice el siguiente argumento en el cálculo proposicional:

    > "Si Superman fuera capaz y deseara prevenir el mal, entonces lo haría. Si Superman fuera incapaz de prevenir el mal, entonces sería impotente, y si no deseara prevenir el mal, entonces sería malévolo. Si Superman existe, no es ni impotente ni malévolo. Superman no previene el mal. Entonces, Superman no existe."

    Demuestre que 'Superman no existe' es consecuencia lógica de esta formalización.

18. Un grafo G es una tupla (N, A), donde N es un conjunto de nodos y $A \subseteq N \times N$ es un conjunto de arcos. Un grafo es no dirigido si cada vez que $(a, b) \in A$ se tiene que $(b, a) \in A$.

    Un grafo no dirigido $G = (N, A)$ es 3-coloreable si existe una asignación de colores para los nodos tal que nodos adyacentes reciben colores distintos. Formalmente, G es 3-coloreable si existe una función $f : N \to \{\mathit{blanco}, \mathit{azul}, \mathit{rojo}\}$ tal que para cada $(a, b) \in A$ se tiene que $f(a) \neq f(b)$.

Demuestre que el problema de 3-coloración puede ser reducido al problema de satisfacibilidad. Vale decir, encuentre un algoritmo que dado un grafo G construye una oración proposicional φ_G tal que G es 3-coloreable si y sólo si φ_G es satisfacible. Estime el número de pasos que realiza su algoritmo cuando el grafo G tiene n nodos y m arcos.

19. Una oración de la lógica proposicional está en *CNF* si es de la forma

$$\bigwedge_{1 \leq i \leq n} (l_i^1 \vee l_i^2 \vee \cdots \vee l_i^{m_i}),$$

donde $m_i > 0$ $(1 \leq i \leq n)$ y para cada $1 \leq i \leq n$ y $1 \leq j \leq m_i$, l_i^j es una variable proposicional o su negación. Decimos que el *rango* de esta oración es $(m_1, \ldots, m_n)$.

Un *grafo* G es una tupla (N, A), donde N es un conjunto de nodos y $A \subseteq N \times N$ es un conjunto de arcos. Un grafo es *no dirigido* si cada vez que $(a, b) \in A$ se tiene que $(b, a) \in A$. El grafo G es *simple* si para todo $a \in N$ se tiene que $(a, a) \notin A$. Por último, G tiene un *clique* de tamaño k, para $k > 0$, si existen nodos distintos $a_1, \ldots, a_k$ en N tal que para cada $1 \leq i, j \leq k$, $(a_i, a_j) \in A$.

Demuestre que para toda oración ϕ en CNF de rango $(m_1, \ldots, m_n)$ es posible *construir* un grafo simple y no dirigido G_ϕ, tal que

- G_ϕ tiene a lo más $\sum_{1 \leq i \leq n} m_i$ nodos; y
- ϕ es satisfacible si y solo si G_ϕ tiene un clique de tamaño n.

Además, la construcción de G_ϕ solo puede estar basada en la forma de ϕ (es decir, en su sintaxis), y no en su semántica (es decir, la construcción no puede estar basada en si ϕ es satisfacible o no).

20. Decimos que dos grafos $G_1 = (N_1, A_1)$ y $G_2 = (N_2, A_2)$ son *isomorfos* si existe una biyección $f : N_1 \to N_2$ tal que para todo a y b en A_1 se tiene que $(a, b) \in A_1$ si y sólo si $(f(a), f(b)) \in A_2$.

Encuentre un algoritmo que dados dos grafos G_1 y G_2 construye una oración proposicional φ tal que G_1 y G_2 son isomorfos si y sólo si φ es satisfacible. Estime el número de pasos de su algoritmo cuando G_1 tiene n_1 nodos y m_1 arcos, y G_2 tiene n_2 nodos y m_2 arcos.

21. Dada una matriz C de 3×3 que contiene números entre 0 y 3, decimos que C es *completable* si es que existe una manera de reemplazar los números 0 por números entre 1 y 3 de tal forma que la suma de cada fila y de cada columna es la misma. Por ejemplo, la siguiente matriz es completable:

2	0	0
0	2	0
0	0	3

puesto que podemos reemplazar los valores 0 por los siguientes valores:

2	2	1
2	2	1
1	1	3

de manera tal que la suma de cada fila y de cada columna es 5. En cambio, la siguiente matriz no es completable:

1	1	1
0	0	0
3	0	0

Dada una matriz C de 3×3, construya una oración φ en lógica proposicional tal que C es completable si y sólo si φ es satisfacible. En particular, φ tiene que ser construida de tal forma que cada valuación σ que satisface a φ represente una forma de completar C.

22. El *principio de los cajones* establece que si $n+1$ objetos son distribuidos en n cajones, entonces al menos habrá un cajón con más de un objeto.

    Demuestre el principio para $n = 2$ usando cálculo proposicional y la noción de consecuencia lógica.

23. Sea $E(x, y)$ un predicado binario utilizado para representar la noción de adyacencia en grafos. En cada una de las siguientes preguntas escriba una oración de la lógica de primer orden que represente la propiedad mencionada.

    (a) El grafo es un clique.
    (b) El grafo contiene un clique con 4 nodos.
    (c) El grafo tiene un ciclo con 4 nodos.
    (d) Existen elementos en el grafo cuya distancia es 4.
    (e) La distancia máxima entre dos nodos del grafo es 3.
    (f) El grafo contiene exactamente 6 nodos.

24. ¿Cuáles de las siguientes afirmaciones son válidas? Justifique su respuesta.

    (a) $\forall x\, \varphi \;\equiv\; \neg\exists x \neg\varphi$.
    (b) $\exists x\, \varphi \;\equiv\; \neg\forall x \neg\varphi$.
    (c) $\exists x\, (\varphi \vee \psi) \;\equiv\; (\exists x\, \varphi) \vee (\exists x\, \psi)$.
    (d) $\forall x\, (\varphi \vee \psi) \;\equiv\; (\forall x\, \varphi) \vee (\forall x\, \psi)$.
    (e) $\exists x\, (\varphi \wedge \psi) \;\equiv\; (\exists x\, \varphi) \wedge (\exists x\, \psi)$.
    (f) $\forall x\, (\varphi \wedge \psi) \;\equiv\; (\forall x\, \varphi) \wedge (\forall x\, \psi)$.

25. Demuestre que las siguientes oraciones de la lógica de primer orden son equivalentes: $\forall x P(x) \wedge \exists x Q(x)$ y $\forall x \exists y (P(x) \wedge Q(y))$.

26. Asuma que el dominio de discurso son los números naturales, y que contamos con (a) un predicado binario $<$ que es intepretado como el orden lineal estándar en $\mathbb{N}$, y (b) dos predicados ternarios $\cdot$ y $+$ que definen a la multiplicación y suma en $\mathbb{N}$, respectivamente.

    Exprese en lógica de primer orden las siguientes propiedades de los números naturales usando solo los predicados mencionados en el párrafo anterior:

- Todo número natural positivo es par o impar, pero no ambos.
- El sucesor de todo número par es impar.
- Existe un número infinito de números primos.
- Para todo par (n, n') de números naturales positivos, existe un único par (p, c) tal que $p \geq 0$, $0 \leq c \leq n - 1$ y $n' = pn + c$.

27. Sea $x \subseteq y$ un predicado binario. Intuitivamente $\subseteq$ representa la relación de subconjunto; es decir, $x \subseteq y$ representa que x es un subconjunto de y.

    Exprese en lógica de primer orden las siguientes propiedades del predicado $x \subseteq y$:

    - La relación $x \subseteq y$ es un orden parcial; i.e. es refleja, antisimétrica y transitiva.
    - Existe un único elemento $\perp$ que está contenido en cualquier otro conjunto (i.e. el conjunto vacío).
    - Existe un único conjunto $\top$ que contiene a todo otro conjunto (es decir, el conjunto universo).
    - La unión de dos conjuntos siempre existe, y además es única (note que $x \cup y = z$ si y sólo si z es el "menor" conjunto con respecto al orden parcial $\subseteq$ que contiene tanto a x como a y).
    - La intersección de dos conjuntos siempre existe, y además es única (note que $x \cap y = z$ si y sólo si z es el "mayor" conjunto con respecto al orden parcial $\subseteq$ que está contenida tanto en x como en y).
    - Todo conjunto tiene un *complemento*; es decir, para todo conjunto x existe un conjunto $\bar{x}$ tal que $x \cap \bar{x} = \perp$ y $x \cup \bar{x} = \top$.

28. Estudie en detalle toda la Sección 1 del libro del curso, y resuelva los ejercicios (primero los fáciles para soltar la mano, y luego los difíciles que son los más interesantes).

29. Estudie en detalle los 4 primeros capítulos de la Sección 8 del libro del curso. Realice también los ejercicios de esos capítulos.