

Lógica Computacional

Tarea 1

Rubí Rojas Tania Michelle

28 de febrero de 2019

1. Enuncia formalmente lo siguiente:

a) Sintaxis de la lógica proposicional.

Solución: Definamos un lenguaje para la lógica de proposiciones.

El alfabeto consta de:

- Símbolos o variables proposicionales (un número infinito) : p_1, \dots, p_n, \dots
- Constantes lógicas: \perp, \top
- Conectivos u operadores lógicos: $\neg, \wedge, \vee, \rightarrow, \leftrightarrow$
- Símbolos auxiliares: $(,)$

El conjunto de expresiones o fórmulas atómicas, denotado $ATOM$ consta de:

- Las variables proposicionales: p_1, \dots, p_n, \dots
- Las constantes \perp, \top

Las expresiones que formarán nuestro lenguaje $PROP$, llamadas usualmente fórmulas, se definen recursivamente como sigue:

- Si $\varphi \in ATOM$ entonces $\varphi \in PROP$. Es decir, toda fórmula atómica es una fórmula.
- Si $\varphi \in PROP$ entonces $(\neg\varphi) \in PROP$.
- φ, ψ entonces $(\varphi \wedge \psi), (\varphi \vee \psi), (\varphi \rightarrow \psi), (\varphi \leftrightarrow \psi) \in PROP$.
- Son todas.

b) Semántica de la lógica proposicional.

Solución: Definamos el significado de cada conectivo lógico:

- **La negación**

- La negación de la fórmula P es la fórmula $\neg P$.
- Símbolo utilizado: \neg
- Su significado en español es: No P , no es cierto que P , es falso que P , etc.
- Semántica (tabla de verdad):

P	$\neg P$
1	0
0	1

- **La conjunción**

- La conjunción de las fórmulas P, Q es la fórmula $P \wedge Q$. Las fórmulas P, Q se llaman conyuntos.
- Símbolo utilizado: \wedge
- Su significado en español es: P y Q , P además de Q , P pero Q , etc.
- Semántica (tabla de verdad):

P	Q	$P \wedge Q$
1	1	1
1	0	0
0	1	0
0	0	0

- **La disyunción**

- La disyunción de las fórmulas P, Q es la fórmula $P \vee Q$. Las fórmulas P, Q se llaman disyuntos.
- Símbolo utilizado: \vee
- Su significado en español es: P o bien Q , o P o Q , etc.
- Semántica (tabla de verdad):

P	Q	$P \vee Q$
1	1	1
1	0	1
0	1	1
0	0	0

- **La implicación**

- La implicación o condicional de las fórmulas P, Q es la fórmula $P \rightarrow Q$. La fórmula P es el antecedente y la fórmula Q es el consecuente de la implicación.
- Símbolo utilizado: \rightarrow
- Su significado en español es: si P entonces Q , P implica Q , P es condición suficiente para Q , Q siempre que P , P sólo si Q , etc.
- Semántica (tabla de verdad):

P	Q	$P \rightarrow Q$
1	1	1
1	0	0
0	1	1
0	0	1

• **La equivalencia**

- La equivalencia o bicondicional de las fórmulas P, Q es la fórmula $P \leftrightarrow Q$.
- Símbolo utilizado: \leftrightarrow
- Su significado en español es: P si y sólo si Q , P es equivalente a Q , P es condición necesaria y suficiente para Q , etc.
- Semántica (tabla de verdad):

P	Q	$P \leftrightarrow Q$
1	1	1
1	0	0
0	1	0
0	0	1

2. Dado el conjunto de proposiciones $\Gamma = \{\neg(p \wedge q), (t \leftrightarrow r), q, (\neg r)\}$. Verifica si el conjunto Γ es tautología, satisfacible o insatisfacible.

Solución:

3. Utilizando interpretaciones verifica si el siguiente argumento es verdadero o falso.

$$\{(p \wedge q), (q \vee r), (\neg s)\} \models p \wedge s$$

Solución:

4. Demuestra que los siguientes secuentes son válidos usando deducción natural.

a) $\{p \rightarrow (q \vee r)\} \vdash (p \rightarrow q) \vee (p \rightarrow r)$

Demostración.

□

b) $\{\} \vdash p \vee (q \wedge r) \rightarrow (p \wedge r) \vee q$

Demostración.

□

5. Realiza las siguientes sustituciones eliminando los paréntesis innecesarios en el resultado:

a) $((q \vee r)[q, p := \neg p, s] \rightarrow (r \wedge \neg(r \leftrightarrow p)))[p, r, q := r \vee q, q \wedge p, s]$

Solución:

$$((q \vee r)[q, p := \neg p, s] \rightarrow (r \wedge \neg(r \leftrightarrow p)))[p, r, q := r \vee q, q \wedge p, s]$$

$$\begin{aligned} &= (((\neg p) \vee r) \rightarrow (r \wedge \neg(r \leftrightarrow p)))[p, r, q := r \vee q, q \wedge p, s] \\ &= (((\neg(r \vee q) \vee (q \wedge p)) \rightarrow ((q \wedge p) \wedge \neg((q \wedge p) \leftrightarrow (r \vee q)))) \\ &= (((\neg(r \vee q) \vee q \wedge p) \rightarrow ((q \wedge p) \wedge \neg(q \wedge p \leftrightarrow r \vee q))) \end{aligned}$$

b) $(u \vee t) \rightarrow (\neg r \leftrightarrow (u \leftrightarrow s))[r, u, t := u, t, r]$

Solución:

$$\begin{aligned} (u \vee t) \rightarrow (\neg r \leftrightarrow (u \leftrightarrow s))[r, u, t := u, t, r] &= (u \vee t) \rightarrow (\neg(u) \leftrightarrow ((t) \leftrightarrow s)) \\ &= (u \vee t) \rightarrow (\neg u \leftrightarrow (t \leftrightarrow s)) \end{aligned}$$

6. Realizar el Tableaux de la siguiente fórmula en PL y da el modelo que satisfaga la fórmula en caso de que el Tableaux sea abierto.

$$\neg((q \vee \neg(p \rightarrow r)) \rightarrow (p \wedge (q \rightarrow r)))$$

Solución:

7. Obtener la forma normal conjuntiva de las siguientes fórmulas (mencionando la operación realizada en cada paso):

a) $((q \rightarrow r) \rightarrow q) \wedge (r \rightarrow q)$

Solución:

$$\begin{aligned} ((q \rightarrow r) \rightarrow q) \wedge (r \rightarrow q) &\equiv (\neg(\neg q \vee r) \vee q) \wedge (\neg r \vee q) && \text{ya que } P \rightarrow Q \equiv \neg P \vee Q \\ &\equiv ((q \wedge \neg r) \vee q) \wedge (\neg r \vee q) && \text{De Morgan} \\ &\equiv (\neg r \wedge (q \vee q)) \wedge (\neg r \vee q) && \text{asociatividad y conmutatividad} \\ &\equiv (\neg r \wedge q) \wedge (\neg r \vee q) && \text{idempotencia} \\ &\equiv (\neg r \wedge \neg r) \wedge (q \vee q) && \text{asociatividad y conmutatividad} \\ &\equiv \neg r \wedge q && \text{idempotencia} \end{aligned}$$

Como $\varphi = \neg r \wedge q$ es una conjunción de literales, entonces φ es de la Forma Normal Conjuntiva.

b) $\neg p \wedge q \rightarrow p \wedge (r \rightarrow q)$

Solución:

$$\begin{aligned} \neg p \wedge q \rightarrow p \wedge (r \rightarrow q) &\equiv (\neg p \wedge q) \rightarrow (p \wedge (r \rightarrow q)) && \text{precedencia y asociatividad de conectivos} \\ &\equiv \neg(\neg p \wedge q) \vee (p \wedge (\neg r \vee q)) && \text{ya que } P \rightarrow Q \equiv \neg P \vee Q \\ &\equiv (p \vee \neg q) \vee (p \wedge (\neg r \vee q)) && \text{De Morgan} \\ &\equiv (p \vee p) \vee (\neg q \wedge (\neg r \vee q)) && \text{asociatividad y conmutatividad} \\ &\equiv p \vee (\neg q \wedge (\neg r \vee q)) && \text{idempotencia} \\ &\equiv (p \vee \neg q) \wedge (\neg r \vee q) && \text{asociatividad} \end{aligned}$$

Como $\varphi = (p \vee \neg q) \wedge (\neg r \vee q)$ es una conjunción de disyunciones, entonces φ es de la Forma Normal Conjuntiva.

8. Obtener la Forma Normal Disyuntiva de $\neg(w \rightarrow \neg p) \vee \neg((\neg s \leftrightarrow w) \vee (p \wedge s))$.

Solución:

$$\begin{aligned} \neg(w \rightarrow \neg p) \vee \neg((\neg s \leftrightarrow w) \vee (p \wedge s)) &\equiv \neg(\neg w \vee \neg p) \vee \neg((\neg s \leftrightarrow w) \vee (p \wedge s)) \\ &\equiv \neg(\neg w \vee \neg p) \vee \neg(((s \vee w) \wedge (\neg s \vee \neg w)) \vee (p \wedge s)) \\ &\equiv (w \wedge p) \vee (((\neg s \wedge \neg w) \vee (s \wedge w)) \wedge (\neg p \vee \neg s)) \\ &\equiv (w \wedge p) \vee (\neg s \wedge \neg w) \vee (s \wedge w) \wedge (\neg p \vee \neg s) \\ &\equiv (w \wedge p) \vee (\neg s \wedge \neg w) \vee (s \wedge w \wedge \neg p) \vee \neg s \end{aligned}$$

Por falta de espacio, no pude colocar arriba la justificación de cada paso, pero lo explico en seguida:

- Eliminamos la implicación, ya que $P \rightarrow Q \equiv \neg P \vee Q$.
- Eliminamos la doble implicación, ya que $P \leftrightarrow Q \equiv (\neg P \vee Q) \wedge (P \vee \neg Q)$
- Hacemos que las negaciones figuren únicamente en las variables proposicionales.
- Eliminamos los paréntesis innecesarios, pues todos los operadores tienen la misma precedencia.
- Aplicamos asociatividad.

Como $\varphi = (w \wedge p) \vee (\neg s \wedge \neg w) \vee (s \wedge w \wedge \neg p) \vee \neg s$ es una disyunción de conjunciones o literales, entonces φ es de la Forma Normal Disyuntiva.

9. Obtener la Forma Normal Negativa de $(p \wedge (q \rightarrow r)) \rightarrow s$

Solución:

$$\begin{aligned}
 (p \wedge (q \rightarrow r)) \rightarrow s &\equiv \neg(p \wedge (\neg q \vee r)) \vee s && \text{ya que } P \rightarrow Q \equiv \neg P \vee Q \\
 &\equiv (\neg p \vee \neg(\neg q \vee r)) \vee s && \text{prop. de } \neg \\
 &\equiv (\neg p \vee (\neg\neg q \wedge \neg r)) \vee s && \text{prop. de } \neg \\
 &\equiv (\neg p \vee (q \wedge \neg r)) \vee s && \text{ya que } \neg\neg P \equiv P
 \end{aligned}$$

Como $\varphi = (\neg p \vee (q \wedge \neg r)) \vee s$ no contiene implicaciones ni equivalencias, y las negaciones que figuran en φ sólo afectan a fórmulas atómicas, entonces φ es de la Forma Normal Negativa.

10. a) Define una función recursiva **pa** que, dada una fórmula ϕ , devuelve el número de paréntesis abiertos ”(” que tiene ϕ .

Solución: Definimos recursivamente la función **pa** :: $PROP \rightarrow \mathbb{N}$ de la siguiente forma:

- $pa(\top) = 0$
- $pa(\perp) = 0$
- $pa(VarP) = 0$
- $pa(\neg\varphi) = pa(\varphi)$
- $pa((\varphi \star \psi)) = pa(\varphi) + pa(\psi) + 1$

- b) Define una función recursiva **pc** que dada una fórmula ϕ , devuelve el número de paréntesis cerrados ”)” que tiene ϕ .

Solución: Definimos recursivamente la función **pc** :: $PROP \rightarrow \mathbb{N}$ de la siguiente forma:

- $pc(\top) = 0$
- $pc(\perp) = 0$
- $pc(VarP) = 0$
- $pc(\neg\varphi) = pc(\varphi)$
- $pc((\varphi \star \psi)) = pc(\varphi) + pc(\psi) + 1$

- c) Sea $\phi = (((\neg p \wedge q) \vee \neg r) \rightarrow r)$. Prueba que **pa**(ϕ) - **pc**(ϕ) = 0.

Demostración. Aplicamos la definición de **pa** a ϕ :

$$\begin{aligned}
 pa(\phi) &= pa((((\neg p \wedge q) \vee \neg r) \rightarrow r)) && \text{def. de } \phi. \\
 &= pa(((\neg p \wedge q) \vee \neg r)) + pa(r) + 1 && \text{def. recursiva de } \mathbf{pa} \\
 &= pa((\neg p \wedge q)) + pa(\neg r) + 1 + 0 + 1 && \text{def. recursiva de } \mathbf{pa} \\
 &= pa(\neg p) + pa(q) + 1 + pa(r) + 1 + 0 + 1 && \text{def. recursiva de } \mathbf{pa} \\
 &= pa(p) + 0 + 1 + 0 + 1 + 0 + 1 && \text{def. recursiva de } \mathbf{pa} \\
 &= 0 + 0 + 1 + 0 + 1 + 0 + 1 && \text{def. recursiva de } \mathbf{pa} \\
 &= 3 && \text{aritmética}
 \end{aligned}$$

Análogamente, aplicamos la definición de **pc** a ϕ :

$$\begin{aligned}
pc(\phi) &= pc(((\neg p \wedge q) \vee \neg r) \rightarrow r)) && \text{def. de } \phi \\
&= pc(((\neg p \wedge q) \vee \neg r)) + pc(r) + 1 && \text{def. recursiva de } \mathbf{pc} \\
&= pc((\neg p \wedge q)) + pc(\neg r) + 1 + 0 + 1 && \text{def. recursiva de } \mathbf{pc} \\
&= pc(\neg p) + pc(q) + 1 + pa(r) + 1 + 0 + 1 && \text{def. recursiva de } \mathbf{pc} \\
&= pc(p) + 0 + 1 + 0 + 1 + 0 + 1 && \text{def. recursiva de } \mathbf{pc} \\
&= 0 + 0 + 1 + 0 + 1 + 0 + 1 && \text{def. recursiva de } \mathbf{pc} \\
&= 3 && \text{aritmética}
\end{aligned}$$

Por lo tanto, $\mathbf{pa}(\phi) - \mathbf{pc}(\phi) = 3 - 3 = 0$.

□

11. Define recursivamente una función **compress** que comprime los elementos consecutivos repetidos de una lista. Ejemplo: $\text{compress "mooloolaba"} = \text{"mololaba"}$. Prueba, usando tu definición, que:

$$\text{compress } [1, 2, 2, 3, 3, 3] = [1, 2, 3]$$

Solución:

```

compress :: (Eq a) => [a] -> [a]
compress [] = []
compress [a] = [a]
compress xs =
    if (head xs) == (head (tail xs)) then compress (drop 1 xs)
    else (head xs) : compress (tail xs)

```

Finalmente, probemos que $\text{compress } [1, 2, 2, 3, 3, 3] = [1, 2, 3]$

Demostración.

$$\begin{aligned}
\text{compress}[1, 2, 3, 3, 3] &= 1 : \text{compress}[2, 3, 3, 3] \\
&= 1 : 2 : \text{compress}[3, 3, 3] \\
&= 1 : 2 : \text{compress}[3, 3] \\
&= 1 : 2 : \text{compress}[3] \\
&= 1 : 2 : [3] \\
&= [1, 2, 3]
\end{aligned}$$

□