

Introducción a los Algoritmos

Cálculo Proposicional

Pedro Sánchez Terraf

CIEM-FaMAF — Universidad Nacional de Córdoba

FaMAF

UNC

20 de abril de 2016



- 1 Validez y Satisfactibilidad
 - Tablas de Verdad
- 2 Repaso de demostraciones
- 3 Demostraciones con expresiones booleanas
 - Variantes en demostraciones con booleanos
 - Asociatividad de \equiv
 - Conmutatividad de \equiv
 - Axiomas y Teoremas
 - Reglas para demostrar teoremas
- 4 Tarea.

Expresiones Booleanas Válidas y Satisfactibles

Una expresión de tipo *Bool* puede ser:

- 1 válida** si es *True* para todos los valores de sus variables (puedo **demostrar** que es equivalente a *True*);
- 2 satisfactible** si hay al menos un valor de las variables que las hace *True* (hay un **ejemplo**);
- 3 no válida** si es *False* para algún valor de sus variables; (hay un **contraejemplo**);
- 4 no satisfactible** si es *False* para todos los valores de sus variables (puedo **demostrar** que es equivalente a *False*);

Expresiones Booleanas Válidas y Satisfactibles

Una expresión de tipo *Bool* puede ser:

- 1 **válida** si es *True* para todos los valores de sus variables (puedo **demostrar** que es equivalente a *True*);
- 2 **satisfactible** si hay al menos un valor de las variables que las hace *True* (hay un **ejemplo**);
- 3 **no válida** si es *False* para algún valor de sus variables; (hay un **contraejemplo**);
- 4 **no satisfactible** si es *False* para todos los valores de sus variables (puedo **demostrar** que es equivalente a *False*);

Expresiones Booleanas Válidas y Satisfactibles

Una expresión de tipo *Bool* puede ser:

- 1 **válida** si es *True* para todos los valores de sus variables (puedo **demostrar** que es equivalente a *True*); P. ej.: $2 * x = x + x$.
- 2 **satisfactible** si hay al menos un valor de las variables que las hace *True* (hay un **ejemplo**);
- 3 **no válida** si es *False* para algún valor de sus variables; (hay un **contraejemplo**);
- 4 **no satisfactible** si es *False* para todos los valores de sus variables (puedo **demostrar** que es equivalente a *False*);

Expresiones Booleanas Válidas y Satisfactibles

Una expresión de tipo *Bool* puede ser:

- 1 **válida** si es *True* para todos los valores de sus variables (puedo **demostrar** que es equivalente a *True*); P. ej.: $2 * x = x + x$.
- 2 **satisfactible** si hay al menos un valor de las variables que las hace *True* (hay un **ejemplo**);
- 3 **no válida** si es *False* para algún valor de sus variables; (hay un **contraejemplo**);
- 4 **no satisfactible** si es *False* para todos los valores de sus variables (puedo **demostrar** que es equivalente a *False*);

Expresiones Booleanas Válidas y Satisfactibles

Una expresión de tipo *Bool* puede ser:

- 1 válida** si es *True* para todos los valores de sus variables (puedo **demostrar** que es equivalente a *True*); P. ej.: $2 * x = x + x$.
- 2 satisfactible** si hay al menos un valor de las variables que las hace *True* (hay un **ejemplo**); P. ej.: $x < 5$.
- 3 no válida** si es *False* para algún valor de sus variables; (hay un **contraejemplo**);
- 4 no satisfactible** si es *False* para todos los valores de sus variables (puedo **demostrar** que es equivalente a *False*);

Expresiones Booleanas Válidas y Satisfactibles

Una expresión de tipo *Bool* puede ser:

- 1 **válida** si es *True* para todos los valores de sus variables (puedo **demostrar** que es equivalente a *True*); P. ej.: $2 * x = x + x$.
- 2 **satisfactible** si hay al menos un valor de las variables que las hace *True* (hay un **ejemplo**); P. ej.: $x < 5$.
- 3 **no válida** si es *False* para algún valor de sus variables; (hay un **contraejemplo**);
- 4 **no satisfactible** si es *False* para todos los valores de sus variables (puedo **demostrar** que es equivalente a *False*);

Expresiones Booleanas Válidas y Satisfactibles

Una expresión de tipo *Bool* puede ser:

- 1 válida** si es *True* para todos los valores de sus variables (puedo **demostrar** que es equivalente a *True*); P. ej.: $2 * x = x + x$.
- 2 satisfactible** si hay al menos un valor de las variables que las hace *True* (hay un **ejemplo**); P. ej.: $x < 5$.
- 3 no válida** si es *False* para algún valor de sus variables; (hay un **contraejemplo**); P. ej.: $2 * x = 0$.
- 4 no satisfactible** si es *False* para todos los valores de sus variables (puedo **demostrar** que es equivalente a *False*);

Expresiones Booleanas Válidas y Satisfactibles

Una expresión de tipo *Bool* puede ser:

- 1 válida** si es *True* para todos los valores de sus variables (puedo **demostrar** que es equivalente a *True*); P. ej.: $2 * x = x + x$.
- 2 satisfactible** si hay al menos un valor de las variables que las hace *True* (hay un **ejemplo**); P. ej.: $x < 5$.
- 3 no válida** si es *False* para algún valor de sus variables; (hay un **contraejemplo**); P. ej.: $2 * x = 0$.
- 4 no satisfactible** si es *False* para todos los valores de sus variables (puedo **demostrar** que es equivalente a *False*);

Expresiones Booleanas Válidas y Satisfactibles

Una expresión de tipo *Bool* puede ser:

- 1 válida** si es *True* para todos los valores de sus variables (puedo **demostrar** que es equivalente a *True*); P. ej.: $2 * x = x + x$.
- 2 satisfactible** si hay al menos un valor de las variables que las hace *True* (hay un **ejemplo**); P. ej.: $x < 5$.
- 3 no válida** si es *False* para algún valor de sus variables; (hay un **contraejemplo**); P. ej.: $2 * x = 0$.
- 4 no satisfactible** si es *False* para todos los valores de sus variables (puedo **demostrar** que es equivalente a *False*); P. ej.: $x + 1 = x$.

Tablas de Verdad

Temporariamente, vamos a utilizar tablas de verdad para decidir satisfactibilidad y validez de fórmulas proposicionales. Luego usaremos demostraciones.

p	q	$p \vee q$	$p \vee q \equiv q$
<i>True</i>	<i>True</i>	<i>True</i>	<i>True</i>
<i>True</i>	<i>False</i>	<i>True</i>	<i>False</i>
<i>False</i>	<i>True</i>	<i>True</i>	<i>True</i>
<i>False</i>	<i>False</i>	<i>False</i>	<i>True</i>

Temporariamente, vamos a utilizar tablas de verdad para decidir satisfactibilidad y validez de fórmulas proposicionales. Luego usaremos demostraciones.

p	q	$p \vee q$	$p \vee q \equiv q$
<i>True</i>	<i>True</i>	<i>True</i>	<i>True</i>
<i>True</i>	<i>False</i>	<i>True</i>	<i>False</i>
<i>False</i>	<i>True</i>	<i>True</i>	<i>True</i>
<i>False</i>	<i>False</i>	<i>False</i>	<i>True</i>

Justificación

$$\begin{aligned} & 5 * (x + 3) = 20 \\ \equiv & \{ \text{Conmutativa } + \} \\ & 5 * (3 + x) = 20 \end{aligned}$$

- Debo justificar usando propiedades **válidas** o **definiciones**.

$$\begin{aligned} & 5 * (x + 3) = 20 \\ \equiv \{ & \text{Conmutativa } + \} \\ & 5 * (3 + x) = 20 \end{aligned}$$

Justificación

Conmutativa de +:

$$a + b = b + a$$

Sustituyo a por x

Sustituyo b por 3

para obtener

$$x + 3 = 3 + x.$$

- Debo justificar usando propiedades **válidas** o **definiciones**.

$$\begin{aligned} & 5 * (x + 3) = 20 \\ \equiv \{ & \text{Conmutativa } + \} \\ & 5 * (3 + x) = 20 \end{aligned}$$

Justificación

Conmutativa de +:

$$a + b = b + a$$

Sustituyo a por x

Sustituyo b por 3
para obtener

$$x + 3 = 3 + x.$$

- Debo justificar usando propiedades **válidas** o **definiciones**.



Justificaremos ahora usando expresiones booleanas válidas. Un ejemplo es la *Conmutatividad de \equiv* .

Justificación

Conmutatividad de \equiv :

$$(P \equiv Q) \equiv (Q \equiv P)$$

$$\begin{aligned} & r \equiv p \wedge q \equiv r \\ \equiv \{ & \text{Conmutatividad } \equiv \} \\ & p \wedge q \equiv r \equiv r \end{aligned}$$

Justificaremos ahora usando expresiones booleanas válidas. Un ejemplo es la *Conmutatividad de \equiv* .

$$\begin{aligned} & r \equiv p \wedge q \equiv r \\ \equiv \{ & \text{Conmutatividad } \equiv \} \\ & p \wedge q \equiv r \equiv r \end{aligned}$$

Justificación

Conmutatividad de \equiv :

$$(P \equiv Q) \equiv (Q \equiv P)$$

Sustituyo P por r

Sustituyo Q por $p \wedge q$; obtengo

$$(r \equiv p \wedge q) \equiv (p \wedge q \equiv r).$$

Otra expresión booleana válida es la *Definición* \Rightarrow .

Justificación

Definición \Rightarrow :

$$(P \Rightarrow Q) \equiv (P \vee Q \equiv Q)$$

$$\begin{aligned} & q \equiv r \Rightarrow q \equiv r \\ \equiv \{ & \text{Definición } \Rightarrow \} \\ & q \equiv r \vee q \equiv q \equiv r \end{aligned}$$

Otra expresión booleana válida es la *Definición* \Rightarrow .

Justificación

Definición \Rightarrow :

$$(P \Rightarrow Q) \equiv (P \vee Q \equiv Q)$$

Sustituyo P por r

Sustituyo Q por q ; obtengo

$$(r \Rightarrow q) \equiv (r \vee q \equiv q).$$

$$\begin{aligned} & q \equiv r \Rightarrow q \equiv r \\ \equiv \{ & \text{Definición } \Rightarrow \} \\ & q \equiv r \vee q \equiv q \equiv r \end{aligned}$$

Asociando y Conmutando con \equiv

Hasta ahora todo es igual. Pero las demostraciones con fórmulas proposicionales (las expresiones booleanas hechas con \equiv , \wedge , \vee , ... y variables de tipo *Bool*) permiten mucha más flexibilidad.

Asociando con \equiv

Puedo **agrupar** las expresiones separadas por \equiv de la manera que quiera, poniendo paréntesis.

Ejemplo

Para la *Conmutatividad* de \equiv , tenemos

$$(P \equiv Q) \equiv (Q \equiv P)$$

$$P \equiv (Q \equiv Q \equiv P)$$

$$(P \equiv Q \equiv Q) \equiv P$$

...



UNC

Universidad
Nacional
de Córdoba



1613-2013
400
AÑOS

Asociando y Conmutando con \equiv

Hasta ahora todo es igual. Pero las demostraciones con fórmulas proposicionales (las expresiones booleanas hechas con \equiv , \wedge , \vee , ... y variables de tipo *Bool*) permiten mucha más flexibilidad.

Asociando con \equiv

Puedo **agrupar** las expresiones separadas por \equiv de la manera que quiera, poniendo paréntesis.

Ejemplo

Para la *Conmutatividad de \equiv* , tenemos

$$(P \equiv Q) \equiv (Q \equiv P)$$

$$P \equiv (Q \equiv Q \equiv P)$$

$$(P \equiv Q \equiv Q) \equiv P$$

...

Asociando y Conmutando con \equiv

Hasta ahora todo es igual. Pero las demostraciones con fórmulas proposicionales (las expresiones booleanas hechas con \equiv , \wedge , \vee , \dots y variables de tipo *Bool*) permiten mucha más flexibilidad.

Conmutando con \equiv

Puedo **ordenar** las expresiones separadas por \equiv de la manera que quiera (y luego agrupar a placer).

Ejemplo

Para la *Definición* de \Rightarrow , tenemos

$$P \Rightarrow Q \equiv P \vee Q \equiv Q$$

$$P \vee Q \equiv P \Rightarrow Q \equiv Q$$

$$Q \equiv P \Rightarrow Q \equiv P \vee Q$$

...

$$\begin{aligned} & r \equiv p \wedge q \equiv r \\ \equiv & \{ \text{Conmutatividad} \equiv \} \\ & p \wedge q \equiv r \equiv r \\ \equiv & \{ \text{Conmutatividad} \equiv \} \\ & p \wedge q \end{aligned}$$

Justificación

Conmutatividad de \equiv :

$$(P \equiv Q) \equiv (Q \equiv P)$$

Sustituyo P por r

Sustituyo Q por $p \wedge q$; obtengo

$$(r \equiv p \wedge q) \equiv (p \wedge q \equiv r).$$

$$\begin{aligned} & r \equiv p \wedge q \equiv r \\ \equiv \{ & \text{Conmutatividad } \equiv \} \\ & p \wedge q \equiv r \equiv r \\ \equiv \{ & \text{Conmutatividad } \equiv \} \\ & p \wedge q \end{aligned}$$

Justificación

Conmutatividad de \equiv :

$$(P \equiv Q \equiv Q) \equiv P$$

Sustituyo P por $p \wedge q$

Sustituyo Q por r ; obtengo

$$(p \wedge q \equiv r \equiv r) \equiv (p \wedge q).$$

Justificación

Definición \Rightarrow :

$$(P \Rightarrow Q) \equiv (P \vee Q \equiv Q)$$

$$\equiv \left\{ \begin{array}{l} \underline{q \equiv r \Rightarrow q \equiv r} \\ \text{Definición } \Rightarrow \\ r \vee q \equiv r \end{array} \right\}$$



UNC

Universidad
Nacional
de Córdoba



Justificación

Definición \Rightarrow :

$$P \Rightarrow Q \equiv P \vee Q \equiv Q$$

$$\equiv \left\{ \begin{array}{l} \underline{q \equiv r \Rightarrow q \equiv r} \\ \text{Definición } \Rightarrow \\ r \vee q \equiv r \end{array} \right\}$$



UNC

Universidad
Nacional
de Córdoba



Justificación

Definición \Rightarrow :

$$\equiv \left\{ \begin{array}{l} \underline{q \equiv r \Rightarrow q \equiv r} \\ \text{Definición } \Rightarrow \\ r \vee q \equiv r \end{array} \right\}$$

$$Q \equiv P \Rightarrow Q \equiv P \vee Q$$



UNC

Universidad
Nacional
de Córdoba



$$\equiv \left\{ \begin{array}{l} \underline{q \equiv r \Rightarrow q \equiv r} \\ \text{Definición } \Rightarrow \\ r \vee q \equiv r \end{array} \right\}$$

Justificación

Definición \Rightarrow :

$$Q \equiv P \Rightarrow Q \equiv P \vee Q$$

Sustituyo P por r

Sustituyo Q por q ; obtengo

$$q \equiv r \Rightarrow q \equiv r \vee q.$$

Fijaremos un conjunto de fórmulas proposicionales válidas y justificaremos nuestras pruebas usando al principio sólo ellas.

Los axiomas que usaremos están en la Lista Completa (“Digesto”) en mi web. Definimos intuitivamente (de manera “recursiva”) la noción de *teorema*.

Definición

Un **teorema** es:

- 1 un axioma; o sino
- 2 una fórmula proposicional equivalente a un axioma o a un teorema ya demostrado.

Una versión mucho más detallada y formal de esta definición la pueden encontrar en el libro “Cálculo de Programas”, Capítulo 3, pp.15-27.

Fijaremos un conjunto de fórmulas proposicionales válidas y justificaremos nuestras pruebas usando al principio sólo ellas.

Los axiomas que usaremos están en la Lista Completa (“Digesto”) en mi web. Definimos intuitivamente (de manera “recursiva”) la noción de *teorema*.

Definición

Un **teorema** es:

- 1 un axioma; o sino
- 2 una fórmula proposicional equivalente a un axioma o a un teorema ya demostrado.

Una versión mucho más detallada y formal de esta definición la pueden encontrar en el libro “Cálculo de Programas”, Capítulo 3, pp.15-27.

Fijaremos un conjunto de fórmulas proposicionales válidas y justificaremos nuestras pruebas usando al principio sólo ellas.

Los axiomas que usaremos están en la Lista Completa (“Digesto”) en mi web. Definimos intuitivamente (de manera “recursiva”) la noción de *teorema*.

Definición

Un **teorema** es:

- 1 un axioma; o sino
- 2 una fórmula proposicional equivalente a un axioma o a un teorema ya demostrado.

Una versión mucho más detallada y formal de esta definición la pueden encontrar en el libro “Cálculo de Programas”, Capítulo 3, pp.15-27.

Teoremas: Cómo demostrarlos

En la práctica, admitiremos un par de reglas extra (que, de todos modos, son correctas desde el punto de vista del libro).

- 1 Todo teorema es equivalente a *True*.
- 2 Si pruebo que algo es equivalente a *True*, entonces es un teorema.
- 3 Si salgo de una fórmula E y llego a otra F justificando con teoremas, entonces $E \equiv F$ es un teorema.

Por ejemplo, en último ejemplo probamos que

$$q \equiv r \Rightarrow q \equiv r \equiv r \vee q \equiv r$$

es un teorema.

Teoremas: Cómo demostrarlos

En la práctica, admitiremos un par de reglas extra (que, de todos modos, son correctas desde el punto de vista del libro).

- 1 Todo teorema es equivalente a *True*.
- 2 Si pruebo que algo es equivalente a *True*, entonces es un teorema.
- 3 Si salgo de una fórmula E y llego a otra F justificando con teoremas, entonces $E \equiv F$ es un teorema.

Por ejemplo, en último ejemplo probamos que

$$q \equiv r \Rightarrow q \equiv r \equiv r \vee q \equiv r$$

es un teorema.

Teoremas: Cómo demostrarlos

En la práctica, admitiremos un par de reglas extra (que, de todos modos, son correctas desde el punto de vista del libro).

- 1 Todo teorema es equivalente a *True*.
- 2 Si pruebo que algo es equivalente a *True*, entonces es un teorema.
- 3 Si salgo de una fórmula E y llego a otra F justificando con teoremas, entonces $E \equiv F$ es un teorema.

Por ejemplo, en último ejemplo probamos que

$$q \equiv r \Rightarrow q \equiv r \equiv r \vee q \equiv r$$

es un teorema.

$$\equiv \left\{ \begin{array}{l} \underline{q \equiv r \Rightarrow q \equiv r} \\ \text{Definición } \Rightarrow \\ r \vee q \equiv r \end{array} \right\}$$

Justificación

Definición \Rightarrow :

$$Q \equiv P \Rightarrow Q \equiv P \vee Q$$

Sustituyo P por r

Sustituyo Q por q ; obtengo

$$q \equiv r \Rightarrow q \equiv r \vee q.$$

Tareas para ahora mismo

- 1 Hacer Ejercicios 4a, 4b y 4c del Práctico 3 usando tablas de verdad.
- 2 Leer los axiomas, poniendoles paréntesis de acuerdo a la precedencia.
- 3 ¿De cuántas maneras se puede leer la *Definición de \neg* ? (es el axioma **A4**).
- 4 ¿Qué teorema demostramos en la filmina 10 (página 24)?
- 5 Completar los espacios en blanco, justificando con los axiomas:

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

- 6 Entender el ejemplo en el ejercicio 6 del Práctico 4.