CAPÍTULO 1

Relaciones

1. El concepto de relación

Según la Real Academia Española, el término relación remite a:

1. Exposición que se hace de un hecho.
2. Conexión, correspondencia de algo con otra cosa.
3. Conexión, correspondencia, trato, comunicación de alguien con otra persona.
4. Trato de carácter amoroso. U. m. en pl. Tienen relaciones desde hace tiempo
5. Lista de nombres o elementos de cualquier clase.
6. Informe que generalmente se hace por escrito, y se presenta ante una autoridad.
7. En el poema dramático, trozo largo que dice un personaje para contar o narrar algo.
8. Gram. Conexión o enlace entre dos términos de una misma oración.
9. Mat. Resultado de comparar dos cantidades expresadas en números.
10. En diversos bailes tradicionales, copla que se dicen los integrantes de las parejas.
11. Conocidos o amigos influyentes. Sin relaciones no se puede triunfar en esa profesión.

Probablemente esta cita no aporte mucho a las ideas que el lector tiene sobre lo que es una relación, pero en realidad aquí estamos interesados en la *determinación* que la matemática logra para sus propósitos de dicho concepto. El punto 9, que refiere a la matemática, nos da una pista: el resultado de comparar dos cantidades expresadas en números no es otra cosa que un "si" o un "no", si es que la relación esta perfectamente determinada. Por ejemplo, podemos determinar completamente la relación "es menor que", siempre que logremos responder "si o "no" cada vez que se hace la pregunta ¿es x menor que y?, cualquiera sean los elementos x e y que se tomen.

Vemos entonces que una relación entre individuos del universo X e individuos del universo Y, determina un conjunto: el conjunto de todos los pares (ordenados) de individuos para los cuales la pregunta

¿está x relacionado con y?

se contesta afirmativamente. Ese conjunto será nuestra determinaciόm matemática del concepto de relación.

Por ejemplo, si

C representa la relación "es la capital de"

P representa la relación "es subconjunto de"

entonces decimos que el par (El Cairo, Egipto) pertenece al conjunto determinado por la relación C, mientras que $(\{1,2\},\{1,3,4\})$ no pertenece al conjunto determinado por la relación P. Nótese que es importante el *orden* de los objetos en cuestión, ya que por un lado porque pueden pertenecer a universos distintos, y por otro lado porque al alterar el orden puede cambiar el valor de verdad de la proposición. Esto ocurre con el predicado P.

Las relaciones "denotan" conjuntos de pares ordenados de la misma manera que una proposición Q (que se refiere a individuos en un universo X) denota el conjunto formado por los individuos que la satisfacen:

$$Q \quad \text{denota} \quad \{x \in X : Q(x)\}$$

DEFINICIÓN 1.1. Sean A y B conjuntos. Una *relación* entre A y B es un subconjunto del producto cartesiano $A \times B$.

Si R es una relación entre A y B decimos que x está relacionado con y (y lo denotamos $x \sim_R y$) si $(x,y) \in R$. Notaciones alternativas son $R(x,y)$, xRy o simplemente $x \sim y$. Si $A = B$ solemos decir que R es una relación sobre A.

EJEMPLO 1.1. Sea $A = \{2,\ 3\}$ y $B = \{3,\ 4,\ 5,\ 6\}$, y consideremos la relación "divide", que vincula elementos de A con elementos de b. Podemos definir R mediante:

$$R = \{(a,b) \mid a \in A,\ b \in B \text{ y } a \text{ divide a } b\}$$

Luego $R = \{(2,4),\ (2,6),\ (3,3),\ (3,6)\}$ y decimos que $2 \sim 4$, $2 \sim 6$, $3 \sim 3$ y $3 \sim 6$. También decimos 2 no está relacionado con 5 y que 3 no está relacionado con 4 y se escribe $2 \not\sim 5$ y $3 \not\sim 4$.

EJEMPLO 1.2. Sea $A = B = \mathbb{Z}$, $R = \{(x,y) \mid y = x^2\}$. R es un subconjunto de $\mathbb{Z} \times \mathbb{Z}$ y con una cantidad infinita de elementos:

$$\begin{array}{ll} -1 \sim 1 & 1 \sim 1 \\ -2 \sim 4 & 2 \sim 4 \\ -3 \sim 9 & 3 \sim 9 \\ \vdots & \vdots \end{array}$$

Tres tipos de relaciones son las más relevantes dentro de la matemática: las *funciones*, las *relaciones de orden* y las de *equivalencia*. Las funciones son relaciones entre dos conjuntos que pueden ser distintos y no se incluirán en este texto debido a que es usual verlas en profundidad en otras materias. Las relaciones de orden y de equivalencia son relaciones sobre un mismo conjunto y veremos su definición y algunas propiedades en este capítulo, concentrándonos sobre todo en las relaciones de orden.

Consideraremos de ahora en más relaciones sobre un mismo conjunto.

2. Propiedades de las relaciones

La forma natural de distinguir tipos de relaciones es considerar sus propiedades mas relavantes. Cuando hablamos de *propiedades de las relaciones*, nos estamos refiriendo a aquellas características que no tengan que ver con un universo particular, sino que refieran a situaciones factibles de ser observadas (afirmadas o refutadas) en cualquier relación, independientemente del universo particular en donde cada una este definida. El uso de distintos tipos de relaciones en diversas áreas de la matemática ha arrojado variados tipos de propiedades, de las cuales vamos a mencionar las que son relevantes para nuestros objetivos.

DEFINICIÓN 2.1. Sea R una relación sobre un conjunto A. Decimos que R es

(a) *reflexiva* si y sólo si para todo a en A, $a \sim a$, en símbolos:

$$\forall a \;\; a \sim a;$$

(b) *simétrica* si y sólo si para todo $a, b \in A$, $a \sim b$ implica que $b \sim a$, en símbolos:

$$\forall a \, \forall b \;\; a \sim b \;\Rightarrow\; b \sim a;$$

(c) *antisimétrica* si y sólo si para todo $a, b \in A$ $a \sim b$ y $b \sim a$ implican que $a = b$, en símbolos:

$$\forall a \, \forall b \;\; (a \sim b) \wedge (b \sim a) \;\Rightarrow\; a = b;$$

(d) *transitiva* si y sólo si para todo a, b y c, $a \sim b$ y $b \sim c$ implican que $a \sim c$, en símbolos:

$$\forall a \, \forall b \, \forall c \; (a \sim b) \wedge (b \sim c) \;\Rightarrow\; a \sim c.$$

Antes de continuar, para poner en juego las propiedades definidas, sugerimos responder las siguientes cuestiones.

(1) Para cada una de las siguientes relaciones responda si es válida cada una de las cuatro propiedades anteriores.
   (a) Sobre las ciudades de argentina: la distancia de x a Buenos Aires es menor o igual que la distancia de y a Buenos Aires.

(b) Sobre las ciudades de argentina: la distancia de x a Buenos Aires es igual a la distancia de y a Buenos Aires.

(2) Sea G un grafo dirigido con vértices V. Considere sobre V la relación

"existe un camino dirigido que lleva desde x hasta y".

Considere además las siguientes propiedades sobre los grafos:
(a) G es **no dirigido** (si existe una arista de a a b, también existe una de b a a)
(b) G es **completo**
(c) G es **acíclico**

Determine cuales de estas propiedades son **suficientes**, y cuales **necesarias** para asegurar que:
(a) La relación es **reflexiva**
(b) La relación es **simétrica**
(c) La relación es **antisimétrica**

(3) Responda si la relación sobre $A = \mathbb{R}$ dada por $a \sim b$ si y sólo si $a \leq b$ es reflexiva, simétrica, antisimétrica y/o transitiva.

(4) Responda si la relación sobre $A = \mathcal{P}(X)$ dada por $A \sim B$ si y sólo si $A \subseteq B$ es reflexiva, simétrica, antisimétrica y/o transitiva.

3. Relación "divide" y "módulo"

Estas dos relaciones serán ejemplos paradigmáticos de las categorías de relaciones que nos interesa estudiar. Analicemos ahora las propiedades que cada una de estas relaciones tiene.

Sea R la relación sobre $\mathbb{N}$ dada por:

$$a \sim b \text{ si y sólo si } a \text{ divide a } b.$$

(1) Es reflexiva:
$a = a.1$, luego a divide a a para todo $a \in \mathbb{N}$;

(2) no es simétrica, pues
2 divide a 4 pero 4 no divide a 2;

(3) es antisimétrica, pues
si a divide a b entonces $b = a.m$ para algún $m \in \mathbb{Z}$, $m > 0$,
si b divide a a entonces $a = b.k$, para algún $k \in \mathbb{Z}$, $k > 0$,
por lo tanto $a = b.k = (a.m).k = a.(m.k)$, de donde se deduce que $m = k = 1$ y entonces $a = b$.

(4) es transitiva, pues
si a divide a b entonces $b = a.k$, para algún $k \in \mathbb{Z}$,
si b divide a c entonces $c = b.j$, para algún $j \in \mathbb{Z}$,
pero entonces $c = b.j = a(k.j)$, es decir que a divide a c.

Resumiendo, la relación "**divide**" es **reflexiva**, **antisimétrica** y **transitiva**.

Considere ahora $A = \mathbb{Z}$, y sea R la relación dada por

$$a \sim b \text{ si y sólo si } a \text{ y } b \text{ tienen la misma paridad.}$$

Observemos que si a y b son ambos pares o ambos impares entonces $a-b$ es par y que si tienen distinta paridad entonces $a-b$ es impar. Luego

$$R = \{(m,n) : \ 2|m-n\}$$

A esta relación se la conoce como "congruencia módulo 2". De manera similar se puede definir la "congruencia módulo k", en la que dos números resultarán congruentes si al dividirlos por k tienen elmismo resto.

Vamos ahora a justificar las propiedades de la relación R.

(1) es reflexiva,
$m-m=0$, y 0 es par;
(2) es simétrica
si $m-n$ es par, entonces $n-m=-(m-n)$ es par;
(3) no es antisimétrica
$2 \sim 4$ y $4 \sim 2$ pero $2 \neq 4$;
(4) es transitiva
si $m-n$ es par y $n-p$ es par entonces $m-p=(m-n)+(n-p)$ que es par.

Resumiendo, la relación "**módulo** 2" es **reflexiva**, **simétrica** y **transitiva**.

4. Relaciones de equivalencia

El ejemplo de la relación "módulo" nos acerca a un tipo de relaciones muy relevante en matemática, que están presentes fuertemente en los desarrollos de las estructuras algebraicas y la topología.

DEFINICIÓN 4.1. Una relación $\simeq$ sobre un conjunto A es de *equivalencia* si es reflexiva, simétrica y transitiva.

Sobre cualquier conjunto, la relación "igual" es trivialmente una relación de equivalencia. Además hemos justificado detalladamente que la relación "tienen la misma paridad" también lo es. De la misma manera se puede demostrar que la relación "módulo k" es una relación de equivalencia.

Las tres propiedades que definen este tipo de relaciones permiten que emerja la noción de *clase de equivalencia* de un elemento x, refiriéndose ésta al conjunto de todos los elementos que están relacionados con x.

DEFINICIÓN 4.2. Sea $\simeq$ una relación de equivalencia sobre un conjunto A y sea x un elemento de A. La *clase de equivalencia de* x se denota por $[x]$ y es el conjunto

$$[x] = \{y \mid y \in A \text{ e } y \simeq x\}.$$

Note que la simetría hace que la propiedad $y \simeq x$ pueda ser reemplazada por $x \simeq y$, obteniendo el mismo conjunto. Por ejemplo, en la relación "tienen la misma paridad", los números pares están en la clase de equivalencia del 2, mientras que los impares están en la clase de equivalencia del 1.

Preguntas:

(1) ¿Que resulta la clase de x, entendida como el conjunto $[x] = \{y \mid y \in A \text{ e } y \sim x\}$, en la relación "divide"?
(2) ¿Cuáles de las siguientes propiedades son ciertas en las clases de la relación "misma paridad", y cuáles en las clases de la relación "divide"?
   (a) $x \sim y \Rightarrow [x] = [y]$
   (b) $[x] = [y] \Rightarrow x = y$
   (c) $x \in [x]$
   (d) $x \not\sim y \Rightarrow [x] \cap [y] = \emptyset$

La acción conjunta de la transitividad y la simetría provocan que las clases de equivalencia no se superpongan (son disjuntas), salvo que se trate de las clases de dos elementos relacionados (por ejemplo el 2 y el 10, en la relación anterior). En este caso las clases coinciden. El lector habrá podido observar este fenómeno analizando las propiedades (a) y (d) de arriba para la relación "misma paridad".

TEOREMA 4.1. *Sea $\simeq$ una relación de equivalencia en un conjunto A y sean x, y elementos de A. Entonces*

(1) $[x] = [y]$ *si y sólo si* $x \simeq y$.
(2) *si* $x \not\simeq y$, *entonces* $[x]$ *e* $[y]$ *son disjuntas.*

Prueba: (1) Sean $[x]$ e $[y]$ dos clases de equivalencia, tales que $[x] = [y]$. Eso significa que

$$\{a \mid a \simeq x\} = \{a \mid a \simeq y\}.$$

Puesto que $x \simeq x$ eso significa que $x \in [x]$ y por lo tanto $x \in [y]$. Luego $x \simeq y$.

Recíprocamente, si $x \simeq y$ queremos ver que $[x] = [y]$. Probaremos entonces que $[y] \subseteq [x]$ y que $[x] \subseteq [y]$. Ahora bien, $a \in [x]$ si y sólo si $a \simeq x$. Como $x \simeq y$ por transitividad se sigue que $a \simeq y$ y por lo tanto $a \in [y]$.

Análogamente, $a \in [y]$ si y sólo si $a \simeq y$. Pero entonces $a \simeq y$ e $y \simeq x$ de donde se sigue que $a \simeq x$ y por lo tanto $a \in [x]$.

(2) Supongamos que $x \not\simeq y$, y tomemos $a \in [x] \cap [y]$, para arribar a una contradicción. Como $a \in [x]$, entonces $a \simeq x$, y por simetría, $x \simeq a$. Por otro lado, como $a \in [y]$, entonces $a \simeq y$. Por transitividad $x \simeq y$, lo que es una contradicción. □

De esta manera, las relaciones de equivalencia están ligadas a la noción de partición de un conjunto. Recordemos la definición:

DEFINICIÓN 4.3. Una *partición* de un conjunto A es una familia de subconjuntos no vacíos de A que son disjuntos entre sí y cuya unión es todo A.

Por ejemplo, las siguientes son particiones de $A = \{a,\ b,\ c\}$:
P_1 : $\{a\}, \{b\}, \{c\}$;
P_2 : $\{a\}, \{b,c\}$;
P_3 : $\{a,\ b,\ c\}$.

Si $\simeq$ es una relación de equivalencia sobre A, entonces podemos partir A de manera que cada parte agrupe a todos los elementos que son equivalentes entre sí.

Por ejemplo, considere $A = \{0,\ 1,\ 2,\ 3,\ 4,\ 5,\ 6\}$, y sea $\simeq$ la relación dada por:

$$a \simeq b \text{ si y sólo si 3 divide a } (a-b).$$

Podemos buscar las "partes" en las que se parte A comenzando a rastrear los equivalentes a 0, a saber, $0,3,6$. Luego, una parte de la partición está dada por $\{0,3,6\}$.

Para encontrar otra de las partes, buscamos los que son equivalentes a alguno de los elementos que no estén en la primera parte, por ejemplo 1. Podemos repetir este procedimiento hasta agotar el conjunto.

La relación $\simeq$ conduce a la siguiente partición:

$$\{0,\ 3,\ 6\}, \qquad \{1,\ 4\}, \qquad \{2,\ 5\}.$$

Luego, el teorema anterior que describe las propiedades de las cleses de equivalencia puede ser reformulado en términos de las particiones de la siguiente manera.

TEOREMA 4.2. *Sea $\equiv$ una relación de equivalencia sobre un conjunto A no vacío. La familia de clases de equivalencia es una partición de A.*

5. Relaciones de orden

La idea de "orden" (en un sentido quizá más laxo que el usual) queda capturada a través de 3 de las propiedades antes estudiadas:

DEFINICIÓN 5.1. Un *orden parcial R* sobre un conjunto es una relación que es reflexiva, antisimétrica y transitiva.

Cuando R sea un orden parcial usaremos la notación $a \preceq b$ si $(a,b) \in R$.

Del trabajo previo ya disponemos de varios ejemplos de relaciones de orden:

(1) La relación $\leq$ sobre $\mathbb{R}$ (o $\mathbb{Z}$)
(2) La relación "divide" (usamos el símbolo $|$), sobre $\mathbb{N}$
(3) La relación $\subseteq$ sobre $\mathcal{P}(A)$

5.1. Diagramas de Hasse. Las relaciones de orden sobre conjuntos finitos pueden ser visualizadas a través de dibujos llamados *digramas de Hasse*. Se adoptó ese nombre en honor al matemático Helmut Hasse (1898-1979).

La idea del diagrama de Hasse (y de todos los diagramas en general) es eliminar información superflua y concentrarse en la información más relevante relativa al orden. Esta (para los conjuntos finitos) es convenientemente capturada por la noción de cubrimiento.

DEFINICIÓN 5.2. Sea A un conjunto finito parcialmente ordenado. Sean $a, b \in A$ elementos distintos. Decimos que b *cubre* a a si $a \preceq b$ y no existe c distinto de a y b tal que $a \preceq c$ y $c \preceq b$.

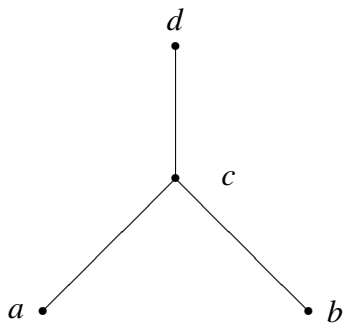
Por ejemplo, sea $X = \{1,2,3,6,12\}$, con la relación dada por la relación "divide". Entonces 2 cubre a 1, pero no cubre a 3. Por otro lado, 6 cubre a 2 y 3, pero no cubre a 1 ni 12.

Un *diagrama de Hasse* para un conjunto parcialmente ordenado finito consiste de puntos en el plano llamados *vértices* que representan los elementos del conjunto y de arcos o segmentos ascendentes que unen dos vértices a y b si y sólo si b cubre a a.

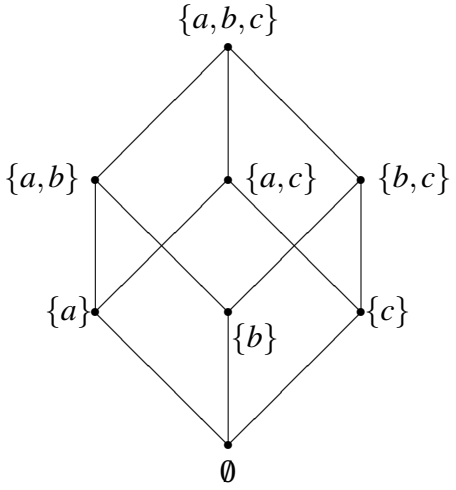
Por ejemplo, sea $P = \{a,b,c,d\}$, y sea $\preceq$ la relación dada por el conjunto

$$\{(a,a),(b,b),(c,c),(d,d),(a,c),(a,d),(c,d),(b,c),(b,d)\}$$

Entonces el diagrama de Hasse correspondiente es:



EJEMPLO 5.1. El diagrama de Hasse para $\mathcal{P}(\{a,b,c\})$ ordenado por inclusión es el siguiente.



6. Ejercicios

(1) Determine si las siguientes relaciones sobre $\mathbb{N}$ son reflexivas, simétricas, antisimétricas o transitivas:
   (a) $(x,y) \in R$ sii $x^2 = y^2$
   (b) $(x,y) \in R$ sii $x > y$
   (c) $(x,y) \in R$ sii $x \geq y$
   (d) $(x,y) \in R$ sii si el m.c.d. de x e y es 1
   (e) $(x,y) \in R$ sii $x \neq y$

(2) Sea $\simeq$ la relación "módulo 5", dada en $\mathbb{Z}$ por $x \simeq y$ si y sólo si *5 divide a* $x - y$. Verifique que es una relación de equivalencia.

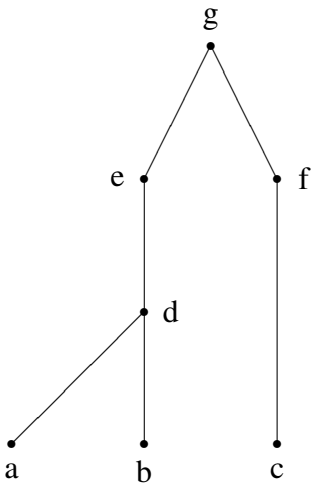
(3) Dé ejemplos de relaciones sobre $\{1,\ 2,\ 3,\ 4\}$ que cumplan las propiedades:
   (a) Reflexiva, simétrica, no transitiva.
   (b) Reflexiva, no simétrica, no transitiva.
   (c) Reflexiva, antisimétrica, no transitiva.
   (d) No reflexiva, simétrica, no antisimétrica, transitiva.
   (e) No reflexiva, no simétrica, transitiva.

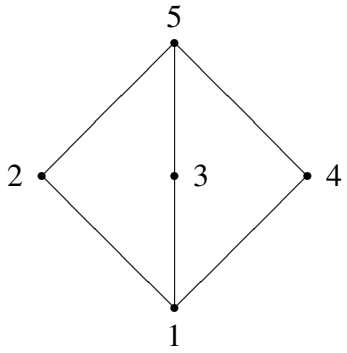
(4) Determine si la relación dada es una relación de equivalencia sobre $\{1,\ 2,\ 3,\ 4,\ 5\}$. Si la relación es de equivalencia, indique las clases de equivalencia.
   (a) $\{(1,1),\ (2,2),\ (3,3),\ (4,4),\ (5,5),\ (1,3),(3,1)\}$

(b) $\{(1,1),\ (2,2),\ (3,3),\ (4,4),\ (5,5),\ (1,3),(3,1)\ (3,4),\ (4,3)\}$
(c) $\{(1,1),\ (2,2),\ (3,3),\ (4,4)\}$
(d) $\{(1,1),\ (2,2),\ (3,3),\ (4,4),\ (5,5),\ (1,5),\ (5,1),\ (3,5),\ (5,3),\ (1,3),\ (3,1)\}$
(e) $\{(x,y) \mid 1 \leq x \leq 5,\ 1 \leq y \leq 5\}$
(f) $\{(x,y) \mid 4$ divide a $x-y\}$
(g) $\{(x,y) \mid 4$ divide a $x+y\}$
(h) $\{(x,y) \mid x$ divide a $2-y\}$

(5) Liste los pares de la relación de equivalencia sobre $\{1,\ 2,\ 3,\ 4\}$ definida por la partición dada. También señale las clases de equivalencia [1], [2], [3] y [4].
(a) $\{1,2\},\{3,4\}$
(b) $\{1\},\{2\},\{3\},\{4\}$
(c) $\{1,2,3,4\}$
(d) $\{1\},\{2,4\},\{3\}$

(6) Dados los siguientes diagramas de Hasse, liste todos los pares ordenados de la relación:





(7) Dibuje los diagramas de Hasse para cada uno de los siguientes conjuntos con la relación de divisibilidad: $n \sim m$ si y sólo si n divide a m:
(a) $\{1,\ 2,\ 3,\ 4,\ 6,\ 12\}$
(b) $\{1,\ 2,\ 4,\ 5,\ 10,\ 20\}$
(c) $\{1,\ 2,\ 4,\ 8,\ 16,\ 32\}$
(d) $\{1,\ 2,\ 3,\ 5,\ 6,\ 10,\ 15,\ 30\}$

(8) Sea A un conjunto de personas. ¿Bajo qué circunstancias la relación

$$x \preceq y \text{ si y sólo si } x \text{ es más joven o tiene la misma edad que } y$$

define un orden parcial sobre A?

(9) Dibuje el diagrama de Hasse para el conjunto de subconjuntos propios de $\{a,\ b,\ c,\ d\}$ ordenado por inclusión.

7. Problemas

(1) Sea R la relación en $D_{60} = \{d \mid d \text{ divide a } 60\}$ dada por:

$$(a,b) \in R \Leftrightarrow 5 \text{ divide a } (a-b)$$

(a) ¿Sirve la prueba dada en el ejercicio 2 concluir que R es de equivalencia? ¿Cómo lo justificaría?
(b) Hallar todas las clases de equivalencia.

(2) Sea A el conjunto formado por todas las palabras del alfabeto $\{a,b,c\}$. Considere las palabras como secuencias finitas de símbolos del alfabeto. Por ejemplo *acaaab*, a y ε (la palabra vacía) son elementos de A.
(a) Defina la relación $\trianglelefteq$, que representa el orden lexicográfico (o sea el del diccionario) sobre A
(b) Pruebe $\trianglelefteq$ es una relación de orden.

(3) Sea A un conjunto de tres elementos, y sea R una relación de orden parcial sobre A. ¿Cuántos tipos diferentes de diagramas de Hasse de A son posibles? De esta manera sabemos cuántos ordenes parciales diferentes pueden ser definidos sobre un conjunto con tres elementos. Piense detenidamente cuando dos diagramas pueden ser considerados "iguales" y cuando diferentes. Por ejemplo, ¿importan las longitudes de los arcos?, o ¿importan las pendientes de los arcos?

(4) Repita la consigna anterior para conjuntos de cuatro elementos. (Ayuda: hay 16.)

8. *Operadores

Con las relaciones podemos operar de la misma manera que con las funciones. Un operación muy común en el manejo de funciones es la *composición* (denotada mediante $\circ$), y su definición puede ser extendida a las relaciones. En este contexto, esta operación hace referencia al resultado de "conectar" relaciones. Por ejemplo, si x es padre de y, e y es padre de z, entonces x es abuelo de z. Esto lo podemos decir escribiendo: $padre \circ padre = abuelo$. Vamos ahora a introducir formalmente esta y otras operaciones que también serán de utilidad.

DEFINICIÓN 8.1. Sea R_1 una relación de A a B y sea R_2 una relación de B a C. La *composición* de R_2 con R_1 es la relación entre A y C dada por:

$$R_2 \circ R_1 = \{(a,c) \mid \text{ existe algún } b \in B \text{ tal que } (a,b) \in R_1 \text{ y } (b,c) \in R_2\}$$

DEFINICIÓN 8.2. Δ_A denotará la relación sobre A definida mediante:

$$\Delta_A = \{(x,x) : x \in A\}.$$

A tal relación la llamaremos *diagonal*.

DEFINICIÓN 8.3. Sea R una relación entre A y B. Entonces la relación *inversa*, es una relación entre B y A que está dada por:

$$R^{-1} = \{(b,a) \mid b \in B, a \in A \text{ y } (a,b) \in R\}.$$

EJEMPLO 8.1. Sean $A = \{1,2,3,4,5\}$, y las siguientes relaciones sobre A:

$R = \{(a,b) \mid a > b\}$
$S = \{(a,b) \mid b - a = 3\}$.

Describamos por extensión los conjuntos R^{-1}, S^{-1} y $S \circ R$.

Por definición $R^{-1} = \{(b,a) \mid (a,b) \in R\} = \{(b,a) \mid b < a\} = R$, luego

$$R^{-1} = \{(1,2),(1,3),(1,4),(1,5),(2,3),(2,4),(2,5),(3,4),(3,5),(4,4),(4,5)\}.$$

Por otro lado $S^{-1} = \{(b,a) \mid (a,b) \in S\} = \{(b,a) \mid b - a = 3\}$, es decir

$$S^{-1} = \{(5,2),(4,1)\}.$$

Finalmente, $S \circ R = \{(a,c) \mid$ existe algún b tal que $(a,b) \in R$ y $(b,c) \in S\} = \{(a,c) \mid$ existe algún b tal que $a > b$ y $c - b = 3\}$. Comprobando elemento por elemento (buscando siempre un "pivot", el elemento b), obtenemos

$$S \circ R = \{(2,4),(3,4),(3,4),(5,4),(3,5),(4,5),(5,5)\}$$

Para los primeros 4 elementos se toma como pivot a 1, para los últimos 3 se toma como pivot a 2. $\square$

Con estas nuevas incorporaciones a nuestro lenguaje matemático podemos reescribir las propiedades estudiadas de una forma más sintética.

Propiedades de las relaciones

(a) R es reflexiva si y sólo si $\Delta_A \subseteq R$.

(b) R es simétrica si y sólo si $R \subseteq R^{-1}$.

(c) R es antisimétrica si y sólo si $R \cap R^{-1} \subseteq \Delta_A$.

(d) R es transitiva si y sólo si $R \circ R \subseteq R$.

8.1. Ejercicios.

(1) Sean R_1 y R_2 las relaciones dadas sobre $\{1,\ 2,\ 3,\ 4\}$ por:

$$R_1 = \{(1,1),\ (1,2),\ (3,4),\ (4,2)\}$$

$$R_2 = \{(1,1),\ (2,1),\ (3,1),\ (4,4),\ (2,2)\}$$

Liste los elementos de $R_2 \circ R_1$ y de $R_1 \circ R_2$.

(2) Pruebe que para cualquier relación R vale

$$R = R^{-1} \quad \Leftrightarrow \quad R \subseteq R^{-1} \quad \Leftrightarrow \quad R^{-1} \subseteq R.$$

(3) Analice la validez de las siguientes afirmaciones, para una relación cualquiera R no vacía:

(a) Si R no es simétrica entonces $R \cap R^{-1} \subseteq \Delta_A$.

(b) $R \circ R^{-1} \subseteq \Delta_A$.

(c) $\Delta_A \subseteq R \circ R^{-1}$.

(d) Si R es simétrica entonces $R \circ R^{-1} \subseteq \Delta_A$.

(e) Si R es simétrica y transitiva entonces $R \circ R^{-1} \subseteq R$.

(4) Sea R una relación de A a B, y sea S una relación de B a C, y sea T una relación de C a D. Muestre que $(T \circ S) \circ R = T \circ (S \circ R)$.

CAPÍTULO 2

Conjuntos parcialmente ordenados

En este capítulo nos dedicamos a estudiar las relaciones de orden, y comenzaremos a preguntarnos sobre las distintas estructuras que el orden genera sobre el conjunto en el cual está definido. Este conjunto será llamado conjunto parcialmente ordenado.

DEFINICIÓN 0.4. Un par $(P,\leq)$ donde P es un conjunto y $\leq$ es un orden parcial sobre P se llama *conjunto parcialmente ordenado*.

Note que utilizamos para denotar una relación de orden parcial genérica el símbolo $\leq$, que hasta el momento estuvo reservado para el orden de los números reales. Nos vamos a permitir esta ambiguedad de notación, que resolvermos en cada caso observando el contexto. Pero el lector debe tener presente que de ahora en más el símbolo $\leq$ no necesariamente hace referencia al orden de los números, ni siquiera hace referencia a un orden total.

Para referirnos a los conjuntos parcialmente ordenados utilizaremos en este apunte dos abreviaturas que son clásicas en la literatura: *posets* (por partially ordered sets), o *cpo's*, proveniente de la abreviatura en castellano.

Para terminar estos comentarios sobre la notación mencionamos que dado un orden parcial $\leq$ sobre P podemos definir una nueva relación $<$ sobre P de la siguiente manera: $a < b$ si y sólo si $a \leq b$ y $a \neq b$. Esta convención también es compatible con el uso que se le da habitualmente al símbolo $<$ en los reales.

1. Maximales, minimales, máximos y mínimos

En un subconjunto de $\mathbb{R}$ ordenado por la relación $\leq$ (menor o igual), podemos tener un elemento mínimo y uno máximo, o sólo uno de ellos o quizás ninguno. No es esta la única situación en la que no existen elementos extremos. Considere por ejemplo los diagramas del Problema 3 del capítulo anterior. Y algunos ejemplos más:

EJEMPLO 1.1. (1) $(\mathbb{Z},\leq)$ no tiene ningún elemento máximo ni ninguno mínimo.
(2) $(\mathbb{N},\leq)$ tiene un elemento mínimo: el 1, pero no tiene elemento máximo.
(3) $[0,1)$ tiene un elemento mínimo que es el 0 y pero no tiene máximo.
(4) Tomemos X el subconjunto de $\mathcal{P}(\{a,b,c\})$ dado por

$$X = \{\{c\},\{a,b\},\{a,b,c\}\}$$

Es el caso de los diagramas mencionados, en los cuales podemos observar un máximo, pero no hay mínimo.

Para precisar todos estos conceptos daremos las siguientes definiciones:

DEFINICIÓN 1.1. Sea $\leq$ un orden parcial sobre un conjunto P. El *elemento máximo* de P (si existe) es el elemento α en A que cumple que

$$a \leq \alpha, \qquad \text{para todo } a \in P.$$

El *elemento mínimo* de P (si existe) es el elemento β en P que cumple que

$$\beta \leq a, \qquad \text{para todo } a \in P.$$

Una notación usual consiste en utilizar el símbolo 1 para denotar el máximo del conjunto (en el caso de que exista), y 0 para el mínimo. Luego, interpretamos que cuando aparece en un contexto abstracto cualquiera de estos símbolos, entonces estamos asumiendo que el conjunto en cuestión posee máximo o mínimo, según corresponda.

En muchos ejemplos observamos que aunque no existe un elemento mínimo, encontramos elementos que no tienen ningún otro elemento menor (por ejemplo $\{a,b\}$ del ejemplo anterior). Este tipo de elementos se llamarán minimales, como lo establece la siguiente definición.

DEFINICIÓN 1.2. Sea P un conjunto parcialmente ordenado con orden parcial $\leq$. Un elemento $x \in P$ se dice *maximal* si para todo a en P, $x \leq a$ implica que $x = a$.

Un elemento $y \in P$ se dice *minimal* si para todo a en P, $a \leq y$ implica que $a = y$.

EJEMPLO 1.2. Sea $P = \{2,\ 3,\ 4,\ 5,\ 6,\ 7,\ 8\}$ ordenado por divisibilidad, esto es $a \leq b$ si y sólo si a divide a b. En este caso tenemos

4 elementos minimales: 2, 3, 5, 7;

4 elementos maximales: 5, 6, 7, 8;

y no hay elemento mínimo ni elemento máximo.

EJEMPLO 1.3. Sea $P = \{\{a\},\{b\},\{c\},\{a,b\},\{a,b,c\}\}$ con la relación de inclusión: $\subseteq$. Entonces

hay 3 elementos minimales: $\{a\}$, $\{b\}$, $\{c\}$;

hay 1 elemento maximal: $\{a,b,c\}$;

hay 1 elemento máximo: $\{a,b,c\}$;

y no hay elemento mínimo.

Observemos que un conjunto puede tener varios elementos maximales o varios elementos minimales. Sin embargo, si α es un elemento máximo de P entonces α es único, y si β es un elemento mínimo de P entonces β es también único. Esta propiedad se enuncia en el siguiente teorema. Trate de dar una prueba formal del mismo.

TEOREMA 1.1. *Sea $\leq$ un orden parcial sobre P.*

(1) *Si P tiene un elemento máximo α entonces α es maximal y no existen otros elementos maximales.*
(2) *Si P tiene un elemento mínimo β entonces β es minimal y no existen otros elementos minimales.*

Existen órdenes en los que todo par de elementos está relacionado. Por ejemplo, en el caso de $\leq$ en $\mathbb{R}$ tenemos que para todo x, y en $\mathbb{R}$ se cumple que o bien $x \leq y$ o bien que $y \leq x$. Tenemos un nombre para este tipo de conjuntos.

DEFINICIÓN 1.3. Un *orden total* sobre un conjunto P es un orden parcial $\leq$ sobre P que satisface la *ley de dicotomía*:

$$\text{para todo } a,b \in P, \quad a \leq b \ \text{ o } \ b \leq a.$$

Algunos ejemplos de órdenes totales:

(1) El orden $\leq$ en $\mathbb{R}$ y el orden $\geq$ en $\mathbb{R}$.
(2) El orden lexicográfico en un diccionario.
(3) El orden "a divide a b" en el conjunto $A = \{2^k \mid k \in \mathbb{N}\}$.

Por supuesto esta es una categoría muy particular de orden, por ejemplo si tomamos la relación $\leq$ sobre $\mathbb{N}$ dada por: $a \sim b$ si y sólo si a divide a b entonces puede ocurrir que $a \not\leq b$ y que $b \not\leq a$, por ejemplo, 5 no divide a 8 y 8 no divide a 5.

Finalmente mencionamos un hecho que no desafía nuestra intuición, relativo a los órdenes finitos. Queda como ejercicio para el lector imaginar una justificación.

TEOREMA 1.2. (1) *Sea $\leq$ un orden parcial en un conjunto finito no vacío P. Entonces P contiene al menos un elemento minimal y si existe sólo uno entonces es el mínimo.*
(2) *Sea $\leq$ un orden parcial en un conjunto finito no vacío P. Entonces P contiene al menos un elemento maximal y si existe sólo uno entonces es el máximo.*

2. Supremos e ínfimos

Seguarmente en el curso de Cálculo el lector se habrá encontrado con la noción de supremo (y su dual, ínfimo), que emerge en la recta real producto de su orden particular. Una propiedad característica de este orden es la existencia de subconjuntos acotados que no poseen último elemento, debido a que su supremo (que siempre existe) no pertenece al conjunto (por ejemplo el intervalo $[0,1)$).

Los conceptos de supremo e ínfimo adquieren una relevancia especial en el estudios de las estructuras ordenadas debido a que es justamente a través de ellos como se revela su verdadera

estructura. Vamos a continuación a definir formalmente estos conceptos, y luego veremos algunos ejemplos.

DEFINICIÓN 2.1. Sea $(P,\leq)$ un poset y sea $S \subseteq P$.

(a) Un elemento $a \in P$ se dice *cota superior* de S si para todo $b \in S$ ocurre que $b \leq a$.
(b) Un elemento $a \in P$ se dice *cota inferior* de S si para todo $b \in S$ ocurre que $a \leq b$.
(c) Un elemento $a \in P$ se dice *supremo* de S si a es una cota superior de S y para toda cota superior b de S se cumple que $a \leq b$.
(d) Un elemento $a \in P$ se dice *ínfimo* de S si a es una cota inferior de S y para toda cota inferior b de S se cumple que $b \leq a$.

EJEMPLO 2.1. Consideremos $(\mathbb{R},\leq)$ con la relación de orden usual. Entonces 4 y 5 son cotas superiores del subconjunto $[1,2)$. Notemos que 2 es el supremo, que no pertenece al conjunto y 1 es el ínfimo, que sí pertenece al conjunto.

Antes de continuar, y para poner en juego los conceptos definidos, sugerimos responder las siguientes cuestiones:

(1) Considerando $(\mathbb{R},\leq)$ como en el ejemplo anterior, responda cuáles de las siguientes condiciones son necesarias, y cuáles son suficientes para que el subconjunto $S \subseteq \mathbb{R}$ tenga supremo dentro de S.
    (a) S es finito
    (b) S es acotado superiormente
    (c) S es un intervalo cerrado
    (d) S es unión de intervalos
    (e) S es unión de intervalos cerrados
(2) Sea $P = \{a,b,c,d,e\}$. Construya diagramas de Hasse que representen posets formados por estos 4 elementos, y que satisfagan:
    (a) El supremo de $\{a,b\}$ es c, y el ínfimo es d. Además el ínfimo de P es e.
    (b) El supremo de $\{a,b\}$, el supremo de $\{a,c\}$ y el supremo de $\{b,c\}$ coinciden, y son todos el elemento d.
    (c) P no tiene supremo ni ínfimo.
    (d) El supremo de $\{a,b\}$ no existe puesto que $\{a,b\}$ no tienen cotas superiores.
    (e) Aunque $\{a,b\}$ tiene cotas superiores, el supremo de $\{a,b\}$ no existe.

Dado $(P,\leq)$ un poset, para referirnos al supremo e ínfimo de un subconjunto S utilizaremos en general la notación $\sup(S)$ e $\inf(S)$, respectivamente. En el caso de existir 0 y 1, tendremos que $0 = \inf(P)$ y $1 = \sup(P)$.

EJEMPLO 2.2. Consideremos el poset $(\mathcal{P}(\mathbb{R}),\subseteq)$. Se debe tener presente que ahora los elementos de nuestro poset son conjuntos. Luego, cuando buscamos supremo o ínfimo de un

conjunto $\mathcal{S} \subseteq \mathcal{P}(\mathbb{R})$, estamos buscando el conjunto más chico que contenga a cada uno de los conjuntos que viven en $\mathcal{S}$. Dado que manejamos tres categorías de objetos, conviene adoptar una convención sobre la notación y el estilo de letra utilizada:

(1) los elementos de $\mathbb{R}$ son denotados mediante $a, x, ...$
(2) los elementos de $\mathcal{P}(\mathbb{R})$ son denotados con $A, B, ...$
(3) los subconjuntos de $\mathcal{P}(\mathbb{R})$ son denotados mediante $\mathcal{S}, \mathcal{A}, ...$

Por ejemplo, sean A, B subconjuntos de $\mathbb{R}$ (o sea elementos de $\mathcal{P}(\mathbb{R})$). El supremo del conjunto $\mathcal{S} = \{A, B\}$ será $A \cup B$, por ser este el conjunto más chico que contiene a ambos, A y B. En general se tiene que dado $\mathcal{S} \subseteq \mathcal{P}(\mathbb{R})$,

$$\sup\ \mathcal{S} := \bigcup \mathcal{S} = \{r \in \mathbb{R} \mid r \in A, \text{ para algún } A \in \mathcal{S}\},$$

$$\inf\ \mathcal{S} := \bigcap \mathcal{S} = \{r \in \mathbb{R} \mid r \in A, \text{ para todo } A \in \mathcal{S}\},$$

lo cual en particular nos dice que

$$\sup\{A, B\} = A \cup B, \quad \inf\{A, B\} = A \cap B$$

□

Por último vamos a estudiar en general que ocurre con $\sup(S)$ e $\inf(S)$ cuando $S = \emptyset$. Notemos que todo elemento de P es cota superior e inferior del conjunto $\emptyset$. De modo que

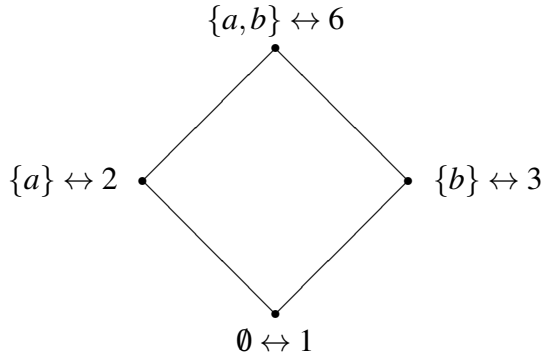
(1) $\sup(\emptyset)$ existe si y sólo si P tiene menor elemento,
(2) $\inf(\emptyset)$ existe si y sólo si P tiene mayor elemento.

3. Isomorfismos de posets

La noción de poset, como concepto abstracto, tiene por objetivo capturar los aspectos relevantes relativos al orden de una estructura, ignorando otros aspectos particulares que no se consideran relevantes. Por ejemplo, los conjuntos

$$\mathcal{P} = \{\emptyset, \{a\}, \{b\}, \{a, b\}\} \qquad P = \{1, 2, 3, 6\}$$

están formados por objetos de distinta naturaleza, pero cuando consideramos los posets $(\mathcal{P}, \subseteq)$ y $(P, |)$ (es decir, P con la relación "divide") comienza a haber ciertas similitudes. Podemos establecer una conexión entre los objetos de $\mathcal{P}$ y los objetos de P de manera de hacer corresponder los roles que cada uno ocupa en sus respectivas estructuras ordenadas. Por ejemplo, a $\emptyset \in \mathcal{P}$ le corresponde $1 \in P$, debido a que ambos son los menores elementos. Dicho de otra manera: los posets poseen el mismo diagrama de Hasse. En la siguiente figura, el símbolo $\leftrightarrow$ significa "se corresponde con".



En matemática, la noción de isomorfismo trata de capturar la idea de que dos estructuras son indistinguibles cuando nos concentramos en ciertos aspectos relevantes, ignorando otros aspectos particulares. En este lenguaje diríamos que $(\mathcal{P}, \subseteq)$ y $(P, |)$ son isomorfas.

DEFINICIÓN 3.1. **Isomorfismo de posets.** Sean $(P, \leq)$, $(Q, \leq')$ dos posets, y sea $f : P \to Q$ una función. Diremos que f es un *isomorfismo* si f es biyectiva y para todo $x, y \in P$, se cumple que $x \leq y$ si y sólo si $f(x) \leq' f(y)$. Si existe un isomorfismo entre $(P, \leq)$ y $(Q, \leq')$ diremos que estos posets son *isomorfos* y escribiremos $(P, \leq) \cong (Q, \leq')$.

EJEMPLO 3.1. Sean $A = \{1, 2, 3, 4\}$ y $B = \{2, 4, 8, 16\}$, y consideremos los posets $(A, \leq)$ con la relación de orden usual y $(B, |)$ donde $x|y$ significa que x divide a y. Luego la función $f : A \mapsto B$ dada por $f(n) = 2^n$ es un isomorfismo de posets.

El siguiente lema muestra que dos posets isomorfos tienen las mismas propiedades matemáticas, en lo que se refiere a las relaciones de orden.

LEMA 3.1. *Sean $(P, \leq)$ y $(Q, \leq')$ posets. Sea $f : P \to Q$ un isomorfismo.*

(a) *Para cada $S \subseteq P$, se tiene que existe $\sup(S)$ si y sólo si existe $\sup(f(S))$ y en el caso de que existan tales elementos se tiene que $f(\sup(S)) = \sup(f(S))$.*
(b) *Para cada $S \subseteq P$, se tiene que existe $\inf(S)$ si y sólo si existe $\inf(f(S))$ y en el caso de que existan tales elementos se tiene que $f(\inf(S)) = \inf(f(S))$.*
(c) *P tiene 1 (resp. 0) si y sólo si Q tiene 1 (resp. 0) y en tal caso se tiene que $f(1) = 1$ y $f(0) = 0$.*
(d) *Para cada $p \in P$, p es maximal (respectivamente minimal) si y sólo si $f(p)$ es maximal (respectivamente minimal).*

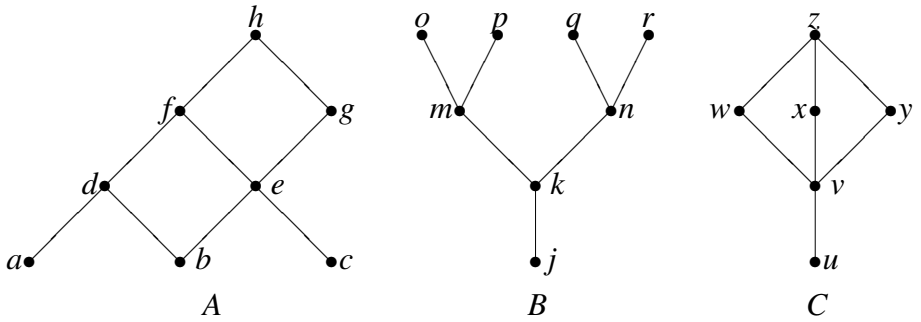
DEMOSTRACIÓN. Notemos que si f es un isomorfismo entonces su inversa f^{-1} también es un isomorfismo. Probemos el inciso (a). Si existe $a = \sup(S)$ entonces $x \leq a$ para todo $x \in S$. Luego $f(x) \leq' f(a)$ para todo $f(x) \in f(S)$. Esto dice que $f(a)$ es una cota superior de $f(S)$.

Veamos ahora que $f(a)$ es la menor cota superior. Supongamos que b es una cota superior de $f(S)$, o sea $f(x) \leq b$ para todo $x \in S$. Entonces $x = f^{-1}(f(x)) \leq f^{-1}(b)$ para todo $x \in S$. Como a es el supremo de S, y $f^{-1}(b)$ resultó ser una cota superior de S, entonces $a \leq f^{-1}(b)$. Luego $f(a) \leq b$, lo que indica que $f(a)$ es la menor cota superior de $f(S)$.

Las demás demostraciones son análogas y se dejan a cargo del lector. □

4. Ejercicios

(1) La siguiente figura muestra los diagramas de Hasse de tres conjuntos parcialmente ordenados.



(a) ¿Cuáles son los elementos maximales y minimales de estos conjuntos?
(b) ¿Cuáles de estos conjuntos tienen mínimo, cuáles máximo?
(c) ¿Qué elementos cubren e?
(d) Encuentre cada uno de los siguientes, si es que existe. En cada caso determine previamente el conjunto de cotas correspondiente.

$$\sup\{d,c\}, \quad \sup\{w,y,v\}, \quad \sup\{p,m\}, \quad \inf\{a,g\}. \quad \sup\{m,n\} \quad \inf\{g,a,f\}$$

(2) Considere el conjunto parcialmente ordenado $(D_{90}, |)$
(a) Dibuje el diagrama de Hasse.
(b) Calcule $\sup\{6,10\}$, $\inf\{6,10\}$, $\sup\{30,9\}$ y $\inf\{9,30\}$.
(c) ¿Cuál es el subconjunto más grande que encuentra dentro de D_{90} que constituya en sí mismo un orden total?

(3) Considere el poset $(\mathbb{N}, |)$; recuerde que $m|n$ si m divide a n.
(a) ¿Cuál es el elemento mínimo?
(b) ¿Tiene $\mathbb{N}$ un elemento máximo?
(c) Describa $\sup\{n,m\}$ e $\inf\{n,m\}$, para cualquier m,n.

(4) Determine cuales de los siguientes mapas de P a Q son isomorfismos. En caso de no serlo determine qué es lo que falla.
   (a) $P = Q = \mathbb{Z}$ (con el orden usual), $f(x) = x+1$
   (b) $P = Q = \mathbb{Z}$ (con el orden usual), $f(x) = 2x$
   (c) $P = Q = \mathbb{Z}$ (con el orden usual), $f(x) = -x$
   (d) $P = Q = \mathcal{P}(\{a,b,c\})$ (con la inclusión). La función f está definida de la siguiente manera. Si a,b están ambos en A, o no están ninguno de los dos en A, entonces $f(A) = A$. En otro caso f quita de A al que está y pone al que no está. Por ejemplo, $f(\{a\}) = \{b\}$ $\quad$ $f(\{a,c\}) = \{b,c\}$.
   (e) $P = Q = \mathcal{P}(\{a,b,c\})$ (con la inclusión), y $f(A) = A^c$.

5. Problemas

(1) En la noción de isomorfismo podemos observar que se recurre a un "si y sólo si" para capturar la idea de que el orden es el mismo en las dos estructuras. Considere esta definición alternativa de isomorfismo:

Sean $(P,\leq)$, $(Q,\leq')$ dos posets, y sea $f : P \to Q$ una función. Diremos que f es un *isomorfismo* si f es biyectiva y para todo $x,y \in P$, se cumple que $x \leq y$ implica $f(x) \leq' f(y)$.

¿Es equivalente a la anterior? ¿Que problemas tendría el adoptar esta definición?

(2) Determine si es posible encontrar dentro del poset $(\mathcal{P}(\{a,b,c,d\}),\subseteq)$ un subconjunto que visto como poset sea isomorfo a D_{90}

(3) La Tabla 1 fue llenada parcialmente. La misma da los valores de $\sup\{x,y\}$ para x e y en cierto poset $(S,\preceq)$. Por ejemplo $\sup\{b,c\} = d$.
   (a) Llene el resto de la tabla.
   (b) ¿Cuál es el mínimo y el máximo de S?
   (c) Muestre que $f \preceq c \preceq d \preceq e$.
   (d) Dibuje el diagrama de Hasse asociado a $(S,\preceq)$.

(4) Supongamos que un poset tiene la siguiente propiedad: para todo $a,b \in P$ se tiene que $\sup\{a,b\}$ existe. ¿Podemos concluir que $\sup(S)$ existe para cualquier S finito?

(5) Supongamos que un poset tiene la siguiente propiedad: para todo subconjunto finito S de P se tiene que $\sup(S)$ existe. ¿Podemos concluir que $\sup(S)$ existe para cualquier S?

sup	a	b	c	d	e	f
a		e	a	e	e	a
b			d	d	e	b
c				d	e	c
d					e	d
e						e
f						

TABLA 1

(6) Supongamos que un poset tiene la siguiente propiedad: para todo subconjunto S de P se tiene que $\sup(S)$ existe (en particular existe $\sup(P)$ y $\sup(\emptyset)$). ¿Podemos concluir que $\inf(S)$ existe para cualquier S?

(7) En un poset, un subconjunto D se dice *decreciente* si cada vez que un elemento está en D, también están los más chicos. En símbolos: si $d \in D$ y $c \leq D$, entonces $c \in D$. Sea f de P en Q una función. Probar que son equivalentes:
   (a) f preserva el orden, o sea, $x \leq y$ implica $f(x) \leq' f(y)$.
   (b) si D es un subconjunto decreciente de Q, entonces $f^{-1}(D)$ es decreciente en P.

(8) Determine cuantos isomorfismo hay de $(\mathcal{P}(\{a,b,c\}),\subseteq)$ en sí mismo.

CAPÍTULO 3

Álgebras de Boole y Reticulados

Los conjuntos parcialmente ordenados constituyen un marco abstracto apropiado para modelar una enorme cantidad de fenómenos, resultando así una herramienta teórica de mucha utilidad, sobre todo a la hora de establecer las bases fundacionales de las Ciencias de la Computación. Por ejemplo, permiten introducir la noción de *dominio*, pilar del desarrollo de la semántica denotacional de los lenguajes de programación. Por otro lado, los conjunto parcialmente ordenados son la puerta de entrada a las estructuras que permiten la algebrización de la lógica, constituyéndose así en un concepto central en los desarrollos de la Lógica Matemática, la Teoría de Pruebas, la Teoría de Modelos y el Álgebra Universal.

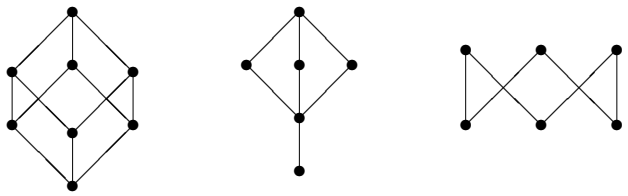
En este capítulo vamos a introducir dos estructuras fundamentales para la lógica: los Reticulados y las Álgebras de Boole. Estudiaremos sus propiedades fundamentales y sus distintas formas de presentación.

1. Reticulados como posets (o posets reticulados)

Los reticulados son una estructura matemática que posee distintas formas de presentación. Introducimos por primera vez esta noción a través del concepto de poset.

DEFINICIÓN 1.1. **Poset reticulado.** Diremos que un poset $(L, \leq)$ es un poset *reticulado* si para todo $a, b \in L$, existen $\sup(\{a,b\})$ e $\inf(\{a,b\})$.

EJEMPLO 1.1. De los tres órdenes siguientes, los dos primeros son posets reticulados y el tercero no.

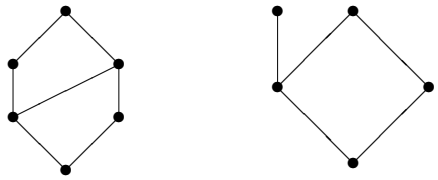


Dado que un poset reticulado garantiza la existencia de ínfimos y supremos de pares de elementos, podemos introducir dos operaciones binarias definidas en todo poset reticulado (para todo par de elementos) representando las operaciones de "tomar el supremo del par" y "tomar el ínfimo del par". Utilizaremos la notación:

$$a \bigvee b = \sup\{a,b\}, \qquad a \bigwedge b = \inf\{a,b\}.$$

Antes de continuar, y como una manera de fijar los conceptos, sugerimos responder lo siguiente:

(1) Determine como y cuando están definidas las operaciones $\bigcirc\!\!\!\wedge$, $\bigcirc\!\!\!\vee$ en los siguientes posets. Considere todos los pares de elementos posibles.
   (a) $(\mathbb{N}, |)$ (aquí $x|y$ si y solo si x divide a y)
   (b) $(\{1,2,3,4,6,12\}, |)$
   (c) $(\{1,2,3,4,6\}, |)$
(2) ¿Cuáles de los anteriores posets son posets reticulados?
(3) Relacione los siguientes digramas de Hasse con los anteriores posets.



EJEMPLO 1.2. Sea $n \in \mathbb{N}$, entonces definimos $D_n = \{k \in \mathbb{N} : k|n\}$. Es decir D_n es el conjunto de divisores de n. Probemos que $(D_n, |)$ es un reticulado (observar que el conjunto de 1b. es D_{12}).

DEMOSTRACIÓN. Sean x, y elementos de D_n, entonces $\text{mcm}(x,y)$ es también elemento de D_n, pues como n es múltiplo de x e y y $\text{mcm}(x,y)$ es el mínimo común múltiplo entre x e y, se deduce que $\text{mcm}(x,y)|n$. Como en 1a. es fácil ver entonces que $x \bigcirc\!\!\!\vee y = \text{mcm}(x,y)$. En forma análoga se ve que $x \bigcirc\!\!\!\wedge y = \text{mcd}(x,y)$.

□

Dado que tenemos definidas dos operaciones binarias sobre todos los posets reticulados, podemos comenzar a indagar que leyes (propiedades) estas satisfacen. Por ley entendemos una expresión que vincula mediante los conectivos lógicos usuales ciertas ecuaciones (o inecuaciones). Cada una de éstas relacionan dos términos que denotan elementos del poset. Cuando decimos que un poset reticulado satisface una ley o propiedad, estamos afirmando que para toda posible elección de elementos, la propiedad se satisface. Para chequear esto debemos comprobar la propiedad para toda posible forma de reemplazar las variables no cuantificadas por elementos del poset reticulado.

Para comprobar si este punto ha quedado claro, sugerimos completar la siguiente actividad: para cada propiedad de la lista siguiente, dé digramas de Hasse representando reticulados que la satisfagan, y que no la satisfagan, en el caso de existir.

(1) $(x \bigcirc\!\!\!\wedge y = y) \vee (x \bigcirc\!\!\!\wedge y = x)$
(2) $x \bigcirc\!\!\!\wedge y = y$
(3) $\exists x\, x \bigcirc\!\!\!\wedge y = x$

El siguiente lema nos muestra una serie de propiedades básicas que son válidas en todos los reticulados.

LEMA 1.1. *Dado un reticulado $(L,\leq)$, y elementos $x,y,z,w \in L$, se cumplen las siguientes propiedades:*

(a) $x \leq x \veebar y$
(b) $x \wedge y \leq x$
(c) $x \leq y \quad \Leftrightarrow \quad x \veebar y = y \quad \Leftrightarrow \quad x \wedge y = x$,
(d) *leyes de idempotencia:*

$$x \veebar x = x \wedge x = x$$

(e) *leyes conmutativas:*

$$x \veebar y = y \veebar x, \qquad x \wedge y = y \wedge x$$

(f) *leyes de absorción:*

$$x \veebar (x \wedge y) = x, \qquad x \wedge (x \veebar y) = x$$

(g) *ley de compatibilidad*

$$x \leq z \ \ e \ \ y \leq w \quad \textit{implican} \quad x \veebar y \leq z \veebar w, \quad x \wedge y \leq z \wedge w$$

(h) *desigualdades distributivas*

$$x \veebar (y \wedge z) \leq (x \veebar y) \wedge (x \veebar z) \qquad (x \wedge y) \veebar (x \wedge z) \leq x \wedge (y \veebar z)$$

(i) *leyes asociativas:*

$$(x \veebar y) \veebar z = x \veebar (y \veebar z), \qquad (x \wedge y) \wedge z = x \wedge (y \wedge z)$$

DEMOSTRACIÓN. Las pruebas de los incisos (a) hasta (f) son dejados al lector. Veamos (g). Puesto que

$$x \leq z \leq z \veebar w, \qquad y \leq w \leq z \veebar w,$$

tenemos que $z \veebar w$ es cota superior de $\{x,\ y\}$ lo cual dice que $x \veebar y \leq z \veebar w$. La demostración para la otra desigualdad es análoga.
(h) Veamos la segunda desigualdad. La primera queda para el lector. Puesto que $(x \wedge y) \leq x$, $(x \wedge z) \leq x$, $(x \wedge y) \leq y \veebar z$ y $(x \wedge z) \leq y \veebar z$, tenemos que

$$(x \wedge y) \leq x \wedge (y \veebar z) \quad \text{y} \quad (x \wedge z) \leq x \wedge (y \veebar z),$$

por lo cual $(x \wedge y) \veebar (x \wedge z) \leq x \wedge (y \veebar z)$.
(i) Notemos que por (i), $x \veebar (y \veebar z)$ es cota superior del conjunto $\{x \veebar y,\ z\}$, lo cual dice que $(x \veebar y) \veebar z \leq x \veebar (y \veebar z)$. Análogamente se puede probar que $x \veebar (y \veebar\ z) \leq (x \veebar y) \veebar z$.

□

Dado que la distribución de paréntesis en una expresión del tipo

$$(...(x_1 \bigvee x_2) \bigvee ...) \bigvee x_n,$$

es irrelevante (ya que $\bigvee$ es asociativa), en general suprimiremos los paréntesis.

2. Reticulados como estructuras algebraicas (o simplemente reticulados)

Los reticulados poseen una propiedad notable: pueden ser presentados (o definidos) de dos manera muy diferentes, que sorprendentemente resultan equivalentes. La primera es la que vimos anteriormente: un reticulado es presentado como un poset con la propiedad de poseer supremo e infimo de todo par de elementos. La siguiente definición describe a los reticulados como un tipo de estructura algebraica. Por estructura algebraica entendemos un conjunto no vacío dotado de algunas operaciones.

DEFINICIÓN 2.1. **Reticulado.** Una estructura del tipo $\langle L, \bigvee, \bigwedge \rangle$, donde L es un conjunto no vacío cualquiera y $\bigvee$ e $\bigwedge$ son dos operaciones binarias sobre L será llamada un *reticulado*, si se satisfacen las siguientes identidades:

R1 leyes de idempotencia:

$$x \bigvee x = x \bigwedge x = x,$$

R2 leyes conmutativas:

$$x \bigvee y = y \bigvee x, \qquad x \bigwedge y = y \bigwedge x,$$

R3 leyes asociativas:

$$(x \bigvee y) \bigvee z = x \bigvee (y \bigvee z), \qquad (x \bigwedge y) \bigwedge z = x \bigwedge (y \bigwedge z),$$

R4 leyes de absorción:

$$x \bigvee (x \bigwedge y) = x, \qquad x \bigwedge (x \bigvee y) = x.$$

Notemos que no toda estructura del tipo $\langle L, \bigvee, \bigwedge \rangle$ es un reticulado. Por ejemplo la estructura $\langle \mathbb{R}, +, \cdot \rangle$ donde $+$ y $\cdot$ son las operaciones de suma y producto usuales de $\mathbb{R}$, no es un reticulado ya que no cumple, por ejemplo, la primera de las identidades.

Está claro desde el lema 1.1 que un poset reticulado $(L, \leq)$ puede "mutar" para convertirse en un reticulado (como estructura algebraica): la estructura $\langle L, \bigvee, \bigwedge \rangle$ satisface las propiedades R1, R2, R3 y R4, en tanto definamos $x \bigwedge y = \inf\{x, y\}$, y $x \bigvee y = \sup\{x, y\}$.

El siguiente teorema muestra que podemos realizar la mutación a la inversa: toda estructura del tipo $\langle L, \bigvee, \bigwedge \rangle$ que cumpla las propiedades R1, R2, R3 y R4, determina un único poset reticulado $(L, \leq)$ en donde las nociones de supremo e ínfimos están dadas por las operaciones $\bigwedge$ y $\bigvee$.

TEOREMA 2.1. *Sea $\langle L, \bigvee, \bigwedge \rangle$ un reticulado (como estructura algebraica). La relación binaria definida por:*

$$x \leq y \text{ si y sólo si } x \bigvee y = y$$

es un orden parcial sobre L para el cual se cumple:

$$x \bigvee y = \sup\{x,y\}, \qquad x \bigwedge y = \inf\{x,y\}.$$

DEMOSTRACIÓN. Dejamos como ejercicio para el lector probar que $\leq$ es reflexiva y antisimétrica. Veamos que $\leq$ es transitiva. Supongamos que $x \leq y$ y $y \leq z$. Entonces $x \bigvee z = x \bigvee (y \bigvee z) = (x \bigvee y) \bigvee z = y \bigvee z = z$, por lo cual $x \leq z$. Veamos ahora que $x \bigvee y = \sup\{x,y\}$. Es claro que $x \bigvee y$ es una cota superior del conjunto $\{x,y\}$. Supongamos $x, y \leq z$. Entonces

$$(x \bigvee y) \bigvee z = (x \bigvee y) \bigvee (z \bigvee z) = (x \bigvee z) \bigvee (y \bigvee z) = z \bigvee z = z,$$

por lo que $x \bigvee y$ es la menor cota superior.

Para probar $x \bigwedge y = \inf(\{x,y\})$, probaremos que

$$x \leq y \text{ si y sólo si } x \bigwedge y = x,$$

lo cual le permitirá al lector aplicar un razonamiento similar al usado en el caso de la operación $\bigvee$. Supongamos que $x \bigvee y = y$. Entonces $x \bigwedge y = x \bigwedge (x \bigvee y) = x$. Recíprocamente si $x \bigwedge y = x$, entonces $x \bigvee y = (x \bigwedge y) \bigvee y = y$.

□

El teorema anterior asegura que las dos definiciones de reticulado dadas anteriormente (como poset o como estructura del tipo $\langle L, \bigvee, \bigwedge \rangle$) son equivalentes. Por esto, en lo que sigue muchas veces utilizaremos la hipótesis "L es un reticulado" sin especificar si se trata de un poset o una estructura algebraica. Una vez asumido que L es un reticulado, por lo anterior podremos disponer tanto del orden, como de las operaciones $\bigvee$ e $\bigwedge$.

EJEMPLO 2.1.
(a) Si X es un conjunto arbitrario, entonces $\langle \mathcal{P}(X), \cup, \cap \rangle$ es un reticulado. La relación binaria inducida por $\cup$ y $\cap$ es precisamente la inclusión, pues $A = A \cup B$ si y sólo si $B \subseteq A$.
(b) Si $n \in \mathbb{N}$ entonces $\langle D_n, \text{mcm}, \text{mcd} \rangle$ es un reticulado. La relación binaria inducida es la de divisibilidad, pues $\text{mcm}(x,y) = y$ si y sólo si x divide a y.

3. Isomorfismos de reticulados

Dados dos reticulados, por tener estos una estructura de posets, podemos analizar si son isomorfos o no. Pero las estructuras algebraicas poseen su propia definición de isomorfismo: dos estructuras del mismo tipo son isomorfas si existe entre ellas un biyección que preserve las operaciones de las mismas. En nuestro caso, esta definición se formaliza de la siguiente manera:

DEFINICIÓN 3.1. **Isomorfismo de reticulados (vistos como estrucuras algebraicas).** Sean $\langle L, \bigtriangledown, \bigtriangleup \rangle$ y $\langle L', \bigtriangledown', \bigtriangleup' \rangle$ reticulados. Una función $F : L \to L'$ se dice un *isomorfismo* de reticulados si F es biyectiva y para todo $x, y \in L$ se cumple que

$$F(x \bigtriangledown y) = F(x) \bigtriangledown' F(y) \qquad F(x \bigtriangleup y) = F(x) \bigtriangleup' F(y).$$

Escribiremos $\langle L, \bigtriangledown, \bigtriangleup \rangle \cong \langle L', \bigtriangledown', \bigtriangleup' \rangle$ cuando exista un isomorfismo de L en L'.

Dado que hemos dado dos presentaciones diferentes para el mismo concepto de reticulado, debemos además probar que si dos reticulados son isomorfos vistos como posets, son también isomorfos vistos como estructuras algebraicas.

LEMA 3.1. *Sean $\langle L, \bigtriangledown, \bigtriangleup \rangle$ y $\langle L', \bigtriangledown', \bigtriangleup' \rangle$ reticulados y sean $(L \leq)$ y $(L', \leq')$ los posets asociados. Entonces una función $F : L \mapsto L'$ es un isomorfismo entre las estructuras $\langle L, \bigtriangledown, \bigtriangleup \rangle$ y $\langle L', \bigtriangledown', \bigtriangleup' \rangle$ si y sólo si lo es entre los posets $(L, \leq)$ y $(L', \leq')$.*

DEMOSTRACIÓN. Sea $F : L \mapsto L'$ un isomorfismo de reticulados (vistos como estructuras algebraicas). Veamos que $x \leq y \Leftrightarrow F(x) \leq' F(y)$. Recordemos que $x \leq y$ si y sólo si $x \bigtriangledown y = y$. Luego

$$x \leq y \Leftrightarrow F(x \bigtriangledown y) = F(y) \Leftrightarrow F(x) \bigtriangledown' F(y) = F(y) \Leftrightarrow F(x) \leq' F(y).$$

La recíproca se deduce del Lema 3.1.

□

4. Reticulados acotados y complementados

En esta sección vamos a incorporar los conceptos necesarios para aproximarnos a un tipo especial de reticulado: el álgebra de Boole.

DEFINICIÓN 4.1. **Reticulado acotado.** Una estructura del tipo $\langle L, \bigtriangledown, \bigtriangleup, 0, 1 \rangle$ donde L es un conjunto no vacío, $\bigtriangledown$ e $\bigtriangleup$ son operaciones binarias sobre L y $0, 1 \in L$, se dice un *reticulado acotado* si $\langle L, \bigtriangledown, \bigtriangleup \rangle$ es un reticulado y además para cada $x, y \in L$,

$$0 \bigtriangledown x = x, \qquad x \bigtriangledown 1 = 1.$$

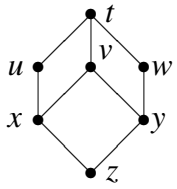
Notemos que si $(P, \leq)$ es un reticulado con máximo 1 y mínimo 0, entonces $\langle P, \sup, \inf, 0, 1 \rangle$ es un reticulado acotado. Además en virtud del Teorema 2.1 todo reticulado acotado se obtiene de esta forma.

EJEMPLO 4.1. (a) $\langle D_n, \text{mcm}, \text{mcd}, 1, n \rangle$ es un reticulado acotado.
(b) $\langle \mathbb{N}, \text{mcm}, \text{mcd} \rangle$ no tiene estructura de reticulado acotado pues no tiene máximo.
(c) Si X es un conjunto finito, entonces $\langle \mathcal{P}(X), \cup, \cap, \emptyset, X \rangle$ es un reticulado acotado.

DEFINICIÓN 4.2. **Complemento.** Sea $\langle L, \bigvee, \bigwedge, 0, 1\rangle$ un reticulado acotado. Dado $a \in L$, diremos que a es *complementado* cuando exista un elemento $b \in L$ llamado *complemento de a* tal que:

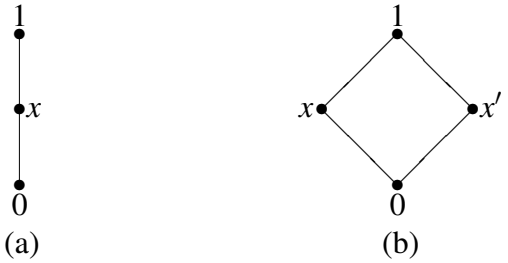
$$a \bigvee b = 1, \qquad a \bigwedge b = 0.$$

Notemos que un elemento puede no tener complementos, o tener varios complementos. Por ejemplo en el reticulado S dado por el diagrama



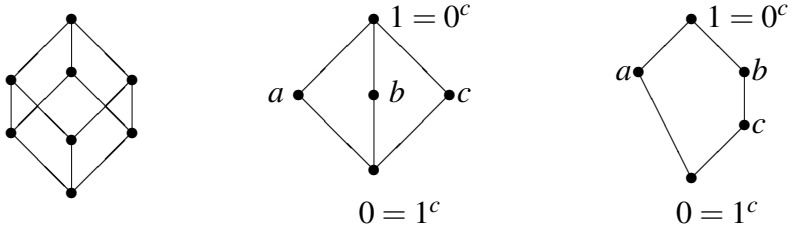
vemos que v no tiene complementos, mientras que w tiene a u y x como complementos.

Las cadenas, como por ejemplo la de la figura (a), son ejemplos de reticulados en los cuales el 0 y el 1 son los únicos elementos complementados. El reticulado de la figura (b) es un reticulado en el cual todo elemento tiene complemento



DEFINICIÓN 4.3. **Reticulado complementado.** Un reticulado complementado será un estructura del tipo Sea $\langle L, \bigvee, \bigwedge, 0, 1, ^c\rangle$ tal que $\langle L, \bigvee, \bigwedge, 0, 1\rangle$ un reticulado acotado y para todo $a \in L$ se tiene que a^c es un complemento de a.

Por ejemplo, en los siguientes reticulados podemos definir x^c para cada x, de manera de convertirlos en reticulados complementados:



Es un ejercicio útil comprobar que en el primer reticulado existe una única forma de definir x^c, para cualquier elemento x del reticulado. No ocurre lo mismo en los demás reticulados, en los cuales tenemos distintas manera de definir x^c para aquellos elementos x que poseen más de un complemento. Por ejemplo, en el reticulado del medio podemos definir $a^c = b^c = c,\ c^c = a$, pero también podríamos definir $a^c = c^c = b,\ b^c = a$, y esto no agota todas las posibilidades. Un fenómeno parecido ocurre en el último reticulado, debido a que a posee dos complementos.

5. Reticulados distributivos

El último ejemplo de la sección anterior nos muestra que la operación complemento (cuando existe) no está determinada por la estructura de poset (o sea por el orden), ya que existe un poset en el cuál podemos definir la operación complemento de distintas maneras.

Este fenómeno se revierte mediante la noción de reticulado distributivo, que nos acercará al concepto de álgebra de Boole. Vamos a introducir este concepto, y veremos que en los reticulados distributivos no pueden existir dos complementos de un mismo elemento. Luego encontraremos una caracterización sencilla de los reticulados distributivos.

Primero vamos a probar el siguiente lema.

LEMA 5.1. *Sea $\langle L, \bigvee, \bigwedge \rangle$ un reticulado; entonces son equivalentes:*

(1) $x \bigwedge (y \bigvee z) = (x \bigwedge y) \bigvee (x \bigwedge z)$, *cualesquiera sean $x, y, z \in L$,*
(2) $x \bigvee (y \bigwedge z) = (x \bigvee y) \bigwedge (x \bigvee z)$, *cualesquiera sean $x, y, z \in L$.*

DEMOSTRACIÓN. Veamos que (1) $\Rightarrow$ (2).

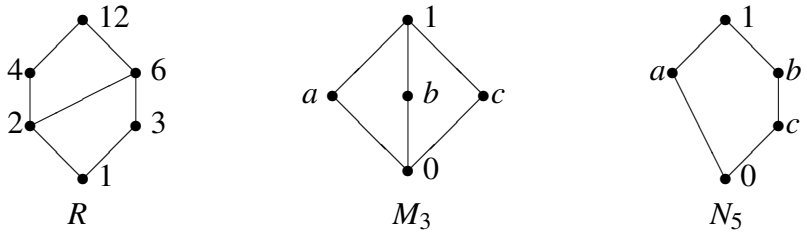
$$\begin{aligned}
\big(x \bigvee y\big) \bigwedge \big(x \bigvee z\big) &= \big(\big(x \bigvee y\big) \bigwedge x\big) \bigvee \big(\big(x \bigvee y\big) \bigwedge z\big) \\
&= \big(x \bigvee \big(z \bigwedge \big(x \bigvee y\big)\big) \\
&= \big(x \bigvee \big(\big(z \bigwedge x\big) \bigvee \big(z \bigwedge y\big)\big) \\
&= x \bigvee \big(z \bigwedge y\big) = x \bigvee \big(y \bigwedge z\big)
\end{aligned}$$

La demostración de que (2) $\Rightarrow$ (1) es similar.

□

DEFINICIÓN 5.1. **Reticulado Distributivo.** Un reticulado se llamará *distributivo* cuando cumpla alguna de las propiedades del Lema 5.1.

EJEMPLO 5.1. En los siguientes reticulados, el primero es distributivo, y los restantes no lo son. Los dos últimos tendrán una importancia relevante en el estudio de los reticulados distributivos, por eso serán llamados M_3 y N_5, respectivamente.



DEMOSTRACIÓN. Notemos que el reticulado R es el correspondiente a $(D_{12}, |)$. Haremos el caso D_n en general en el Ejemplo 5.4. El reticulado M_3 no es distributivo, pues, por ejemplo,

$$a \bigvee (b \bigwedge c) = a$$

en tanto que

$$(a \bigvee b) \bigwedge (a \bigvee c) = 1.$$

El reticulado N_5 tampoco es distributivo. Queda como ejercicio para el lector encontrar tres elementos que no satisfagan las ecuaciones requeridas. □

Como mencionamos al principio, la distributividad tiene una consecuencia importante sobre los complementos:

LEMA 5.2. *Si* $\langle L, \bigvee, \bigwedge, 0, 1\rangle$ *es un reticulado acotado y distributivo, entonces todo elemento tiene a lo sumo un complemento.*

DEMOSTRACIÓN. Supongamos $x \in L$ tiene complementos y, z. Luego

$$y = y \bigwedge 1 = y \bigwedge (x \bigvee z) = (y \bigwedge x) \bigvee (y \bigwedge z) = 0 \bigvee (y \bigwedge z) = (y \bigwedge z),$$

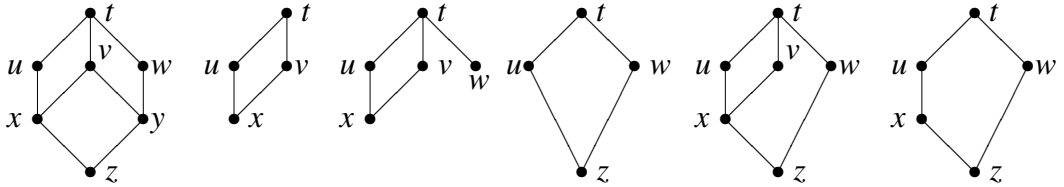
por lo cual $y \leq z$. En forma análoga se puede mostrar que $z \leq y$ y por lo tanto $z = y$. □

Existe una caracterización muy sencilla de los reticulados distributivos que consiste en observar la forma que tienen sus subreticulados de 5 elementos. Para formular esta caracterización vamos primero a precisar algunos conceptos.

DEFINICIÓN 5.2. **Subreticulado.** Sea $\langle L, \bigvee, \bigwedge\rangle$ un reticulado. Un subconjunto $M \subseteq L$ será llamado *subestructura* o *subreticulado* de $\langle L, \bigvee, \bigwedge\rangle$ si

(a) $M \neq \emptyset$,
(b) para todo x, $y \in M$, se tiene que $x \bigvee y$, $x \bigwedge y \in M$.

EJEMPLO 5.2. Consideremos el reticulado $\langle S, \bigvee, \bigwedge\rangle$ de la siguiente figura.



La figura también muestra los diagramas de Hasse de cinco subconjuntos parcialmente ordenados de $(S,\leq)$. El subconjunto M_1 es un subreticulado ya que $a \bigvee b \in M_1$, para todo a, $b \in M_1$. El subconjunto M_2 no es un subreticulado pues, en particular, no es un reticulado. El subconjunto M_3 es un subreticulado. El subconjunto M_4 es por si mismo un reticulado, pero como $v \bigwedge w = y$ e $y \notin M_4$, entonces no es un subreticulado. Por último el conjunto M_5 es también un subreticulado.

El siguiente teorema resulta muy útil cuando se desea determinar si un reticulado es distributivo. Sólo daremos el enunciado y remitimos al libro de Davey and Priestley, *Introduction to lattices and order*, Teorema 6.10 para quien desee conocer una demostración del mismo.

TEOREMA 5.3. *Un reticulado es distributivo si y sólo si no contiene subreticulados isomorfos a M_3 y N_5 del Ejemplo 5.1.*

EJEMPLO 5.3. (1) El reticulado $\mathcal{P}(A)$ de los subconjuntos de un conjunto A es distributivo como ya fue probado anteriormente.

(2) Cualquier orden total da un reticulado distributivo. Se puede ver usando el Teorema 5.3 o directamente de la siguiente forma: claramente $x \bigvee y = \max\{x,y\}$ y $x \bigwedge y = \min\{x,y\}$, entonces una de las leyes distributivas dice

$$\max\{x,\min\{y,z\}\} = \min\{\max\{x,y\},\max\{y,z\}\}.$$

Para verificar esto, primero supondremos que $y \leq z$, entonces $\min\{y,z\} = y$ y $\max\{x,y\} \leq \max\{y,z\}$, luego el lado izquierdo y derecho de la ecuación queda igual a $\max\{x,y\}$. El caso $y \geq z$ tiene una verificación similar.

EJEMPLO 5.4. Veamos que D_n es distributivo y para ello usemos el Teorema 5.3. Supongamos que tenemos en D_n un subreticulado de la forma de la figura M_3 del Ejemplo 5.1, luego $\text{mcd}(a,b) = \text{mcd}(b,c) = \text{mcd}(c,a) = 0^{M_3}$. Supongamos que $0^{M_3} = k \in D_n$. Tenemos entonces que $a = k.a'$, $b = k.b'$ y $c = k.c''$, y además a',b',c' son coprimos entre si, pues sino algún máximo común divisor sería más grande. Ahora bien, por el diagrama tenemos también que

$$\text{mcm}(a,b) = \text{mcm}(b,c) = \text{mcm}(c,a) = 1^{M_3} \qquad (I),$$

pero por lo anterior $\text{mcm}(a,b) = k.a'.b'$ y $\text{mcm}(a,c) = k.a'.c'$ que son claramente diferentes (pues al ser coprimos b' y c' no son iguales). Esto contradice (I).

Supongamos ahora que tenemos en D_n un subreticulado de la forma de la figura N_5 del Ejemplo 5.1, luego $\text{mcd}(a,b) = \text{mcd}(a,c) = 0^{N_5}$. Como antes, llamemos $k = 0^{M_3}$. Tenemos que $a = k.a'$, $b = k.b'$, $c = k.c'$, y además a' es coprimo con b' y c'. Por otro lado $\text{mcm}(a,b) = \text{mcm}(a,c) = 1^{N_5}$, y por las fórmulas anteriores tenemos que $\text{mcm}(a,b) = k.a'.b'$ y $\text{mcm}(a,c) = k.a'.c'$, de lo cual concluimos que $b' = c'$, que implica que $b = k.b' = k.c' = c$, absurdo.

Es decir, suponiendo que D_n tiene un subreticulado de la forma M o N del Ejemplo 5.1 llegamos a un absurdo. Entonces el Teorema 5.3 implica que D_n es distributivo.

6. Álgebras de Boole

Cuando dotamos a un reticulado con la operación complemento, incluimos su máximo y mínimo como operaciones (de aridad 0), y pedimos distributividad para evitar los problemas antes mencionados, estamos en presencia de un álgebra de Boole, estructura fundamental de la lógica, y modelo abstracto de la teoría de conjuntos.

DEFINICIÓN 6.1. **Álgebra de Boole.** Una estructura del tipo $\langle B, \bigvee, \bigwedge, ^c, 0, 1\rangle$, donde B es un conjunto no vacío, $\bigvee$ e $\bigwedge$ son operaciones binarias sobre B, ^{c} es una operación unaria sobre B y $0, 1 \in B$, será llamada un *álgebra de Boole* si $\langle B, \bigvee, \bigwedge, 0, 1\rangle$ es un reticulado acotado y distributivo y además para cada $x \in L$,

$$x \bigvee x^c = 1, \qquad x \bigwedge x^c = 0 \qquad \text{Ley de Complementación.}$$

EJEMPLO 6.1. Sea X un conjunto finito. Entonces $\langle \mathcal{P}(X), \cup, \cap, ^c, \emptyset, X\rangle$ es un álgebra de Boole, donde $A^c = X - C$ para cada $A \subseteq X$.

El ejemplo anterior tiene una importancia fundamental, puesto que el álgebra de Boole se introduce para modelar abstractamente el álgebra de conjuntos. A tal punto este modelado resulta acertado, que veremos más adelante que toda álgebra de Boole finita es escencialmente un álgebra de conjuntos.

EJEMPLO 6.2. Veamos que D_n tiene estructura de álgebra de Boole si y sólo si n es producto de factores primos distintos (i.e. $n = p_1 \dots p_k$, con $p_i \neq p_j$ si $i \neq j$).

DEMOSTRACIÓN. Ya hemos visto que para todo n, $\langle D_n, \text{mcm}, \text{mcd}, 1, n\rangle$ es un reticulado acotado distributivo. Veamos en qué casos todo elemento de D_n tiene complementos. Supongamos que n es producto de factores primos distintos. Sea x en D_n, es decir $x|n$, luego $n = x.k$, como n es producto de factores primos distintos, es claro que x y k son coprimos, luego $\text{mcd}(x,k) = 1$ y $\text{mcm}(x,k) = n$, es decir que $x \bigwedge k = 0$ y $x \bigvee k = 1$. Luego x tiene complemento. Notemos que $x^c = n/x$.

Supongamos ahora que n no es producto de factores primos distintos, luego $n = p^2.r$ para algún p primo. Veamos que p no tiene complemento. Si lo tuviera existiría y tal que $\text{mcd}(p,y) = 1$ y $\text{mcm}(p,y) = n$, ahora bien, la primera igualdad implica que p e y son coprimos y por lo tanto $p.y = \text{mcm}(p,y)$, es decir que $p.y = n$, luego $y = p.r$ (pues $n = p^2.r$). Pero entonces $\text{mcd}(p,y) = \text{mcd}(p,p.r) = p$ y llegamos a una contradicción.

□

Leyes fundamentales que el lector seguramente recordará del álgebra de conjuntos se repreducen en el álgebra de Boole.

TEOREMA 6.1. *Sea* $\langle B, \veebar, \wedge, ^c, 0, 1\rangle$ *un álgebra de Boole, entonces se cumplen las* ***leyes De Morgan****:*

$$(x \veebar y)^c = x^c \wedge y^c \qquad (x \wedge y)^c = x^c \veebar y^c$$

para todo x e y en B.

DEMOSTRACIÓN. Probando que $(x \veebar y) \wedge (x^c \wedge y^c) = 0$ y $(x \veebar y) \veebar (x^c \wedge y^c) = 1$, se deduce de la unicidad del complemento que $x^c \wedge y^c = (x \veebar y)^c$. Probemos primero que $(x \veebar y) \wedge (x^c \wedge y^c) = 0$:

$$\begin{aligned}
(x \veebar y) \wedge (x^c \wedge y^c) &= (x^c \wedge y^c) \wedge (x \veebar y) && \text{Ley Conmutativa}\\
&= \big((x^c \wedge y^c) \wedge x\big) \veebar \big((x^c \wedge y^c) \wedge y\big) && \text{Ley Distributiva}\\
&= \big(x \wedge (x^c \wedge y^c)\big) \veebar \big((x^c \wedge y^c) \wedge y\big) && \text{Ley Conmutativa}\\
&= \big((x \wedge x^c) \wedge y^c\big) \veebar \big(x^c \wedge (y^c \wedge y)\big) && \text{Ley Asociativa}\\
&= \big((x \wedge x^c) y^c\big) \veebar \big(x^c (y \wedge y^c)\big) && \text{Ley Conmutativa}\\
&= (0 \wedge y^c) \veebar (x^c \wedge 0) && \text{Ley de Complementación}\\
&= (y^c \wedge 0) \veebar (x^c \wedge 0) && \text{Ley Conmutativa}\\
&= 0 \veebar 0 && \text{Ley de Acotación}\\
&= 0 && \text{Ley de Identidad.}
\end{aligned}$$

Ahora probemos que $(x \bigotimes y) \bigotimes x^c y^c = 1$:

$$\begin{aligned}
(x \bigotimes y) \bigotimes (x^c \bigoplus y^c) &= \big((x \bigotimes y) \bigotimes x^c\big) \bigoplus \big((x \bigotimes y) \bigotimes y^c\big) && \text{Ley Distributiva} \\
&= \big((y \bigotimes x) \bigotimes x^c\big) \bigoplus \big((x \bigotimes y) \bigotimes y^c\big) && \text{Ley Conmutativa} \\
&= \big(y \bigotimes (x \bigotimes x^c)\big) \bigoplus \big(x \bigotimes (y \bigotimes y^c)\big) && \text{Ley Asociativa} \\
&= (y \bigotimes 1) \bigoplus (x \bigotimes 1) && \text{Ley de Complementación} \\
&= 1 \bigoplus 1 && \text{Ley de Absorción} \\
&= 1 && \text{Ley de Identidad.}
\end{aligned}$$

□

DEFINICIÓN 6.2. **Isomorfismo de Álgebras de Boole.** Si $\langle B, \bigotimes, \bigoplus, ^c, 0, 1\rangle$ y $\langle B', \bigotimes', \bigoplus', ^{c'}, 0', 1'\rangle$ son álgebras de Boole, entonces una función $F : B \to B'$ será llamada un *isomorfismo* si F es biyectiva $F(0) = 0'$, $F(1) = 1'$ y para todo $x, y \in B$ se cumple que

$$F(x \bigotimes y) = F(x) \bigotimes' F(y), \qquad F(x \bigoplus y) = F(x) \bigoplus' F(y), \qquad F(x^c) = F(x)^{c'}.$$

La propiedad de distributividad juega un rol fundamental para determinar la estructura de un álgebra de Boole, como lo muestra el siguiente resultado. El mismo no puede ser probado sin la hipótisis de la distributividad, como lo muestra el problema 1.

TEOREMA 6.2. *Sean $\langle B, \bigotimes, \bigoplus, ^c, 0, 1\rangle$ y $\langle B', \bigotimes', \bigoplus', ^{c'}, 0', 1'\rangle$ álgebras de Boole, y sea $F : B \to B'$. Entonces F es un isomorfismo de álgebras de Boole si y sólo si es un isomorfismo de posets.*

DEMOSTRACIÓN. Ya hemos probado que F es in isomorfismo de reticulados acotados si y sólo si es un isomofismo de poset. Resta ver que si F es un isomorfismo de posets entonces $F(x^c) = (F(x))^{c'}$, para todo $x \in B$. Note que, si vemos que $F(x^c)$ es un complemento de $F(x)$ en B', entonces por la unicidad del complemento tendremos que $F(x^c) = (F(x))^{c'}$. Que $F(x^c)$ es un complemento de $F(x)$ sale inmediatamente del hecho que F preserva las operaciones $\bigoplus$ y $\bigotimes$. □

7. Ejercicios

(1) Considere el reticulado L_2 de la Fig. 1.b. Encuentre $v \bigotimes x$, $s \bigotimes v$ y $u \bigotimes v$.

(2) Considere el reticulado L_1 de la Fig. 1.a. Para los elementos $1, b, c$, muestre todas las formas posibles de escribirlo como supremo. Por ejemplo, una manera sería $1 = d \bigotimes c$.

(3) De un ejemplo de un poset finito **no** reticulado P con la siguiente propiedad: para todo $S \subseteq P$, los conjuntos $\{x : x \text{ es cota superior de } S\}$ y $\{x : x \text{ es cota inferior de } S\}$ son ambos no vacíos.

(4) Considere el reticulado L_2 de la Fig. 1.b.
  (a) ¿Es L_2 un reticulado con complementos?
  (b) Encuentre un elemento con dos complementos.
  (c) ¿Es L_2 un reticulado distributivo?

(5) Considere el reticulado L_1 de la Fig. 1.a.
  (a) Dé los complementos, si es que existen, de los siguientes elementos: a, b, d, 0.
  (b) ¿Es L_1 un reticulado con complementos?
  (c) ¿Es L_1 un reticulado distributivo?

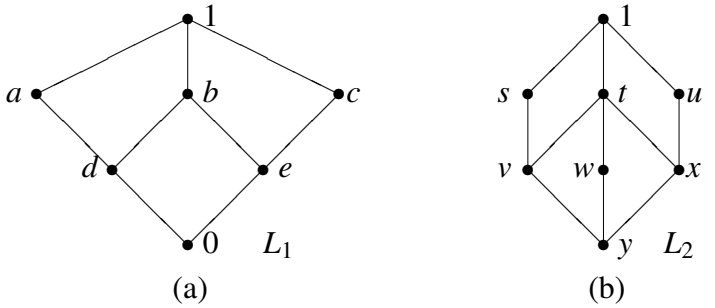


FIGURA 1

(6) (a) Dibuje el diagrama de Hasse para el reticulado $(D_{24}, |)$.
  (b) ¿Es D_{24} un reticulado con complementos?
  (c) ¿Es D_{24} un reticulado distributivo?
  (d) ¿Es D_{24} un álgebra de Boole?

(7) (a) Muestre que las figuras 2.b y 2.c son diagramas de Hasse de reticulados distributivos.
  (b) ¿Es Fig. 2.b un reticulado con complementos?
  (c) ¿Es Fig. 2.c un reticulado con complementos?

(8) (a) Muestre que la Fig. 2.a es el diagrama de Hasse para $(D_{36}, |)$.
  (b) ¿Es D_{36} un reticulado distributivo?
  (c) ¿Es D_{36} un reticulado con complementos?

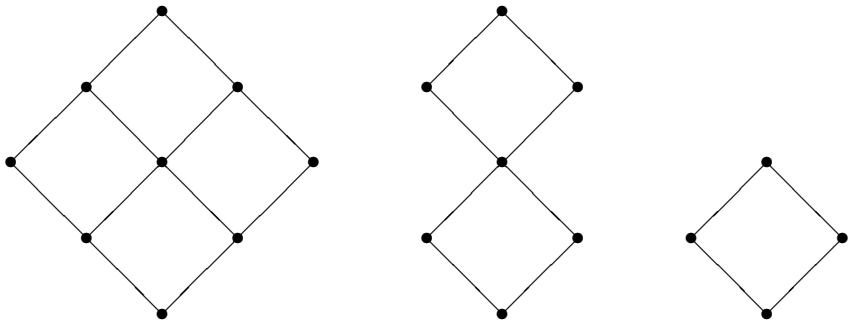


FIGURA 2

(9) Sea $(S, \preceq)$ un reticulado. Demuestre que si $x \preceq y$, entonces $x \bigtriangledown (z \bigtriangleup y) \preceq (x \bigtriangledown z) \bigtriangleup y$ para toda z en S.

(10) Considere los diagramas de la Fig. 3.

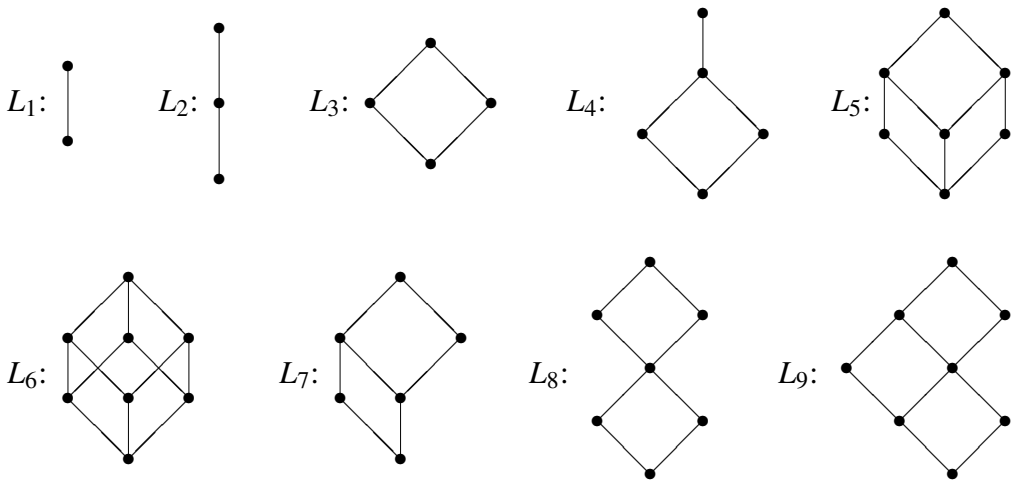


FIGURA 3

(a) Determine cuáles son reticulados distributivos.
(b) Determine cuáles son álgebras de Boole. Determine en cada caso los subreticulados que son álgebras de Boole (no considere el álgebra de Boole trivial $\{0,1\}$).
(c) Encuentre para L_i con $i = 1,2,3,7,8$ un álgebra de Boole B_i tal que L_i sea subreticulado de B_i.

(11) Sea S un reticulado distributivo. Demuestre que si x e y en S satisfacen $x \bigvee a = y \bigvee a$ y $x \bigwedge a = y \bigwedge a$ para alguna a en S, entonces $x = y$.

(12) Sea B un álgebra de Boole y $\preceq$ el orden asociado a B. Demuestre que
   (a) si $x \preceq y$ entonces $y^c \preceq x^c$;
   (b) si $y \bigwedge z = 0$ entonces $y \preceq z^c$;
   (c) si $x \preceq y$ e $y \bigwedge z = 0$ entonces $z \preceq x^c$.

(13) Sea L un reticulado. Pruebe o refute cada una de las siguientes desigualdades:

1. $x \bigvee (y \bigwedge z) \leq (x \bigvee y \bigvee z) \bigwedge (x \bigvee y)$
2. $x \bigvee (y \bigwedge z) \leq (x \bigvee y) \bigwedge (y \bigvee z)$
3. $x \bigwedge (y \bigvee z) \leq (x \bigwedge y) \bigvee (x \bigwedge y)$
4. $a \geq c \Rightarrow a \bigwedge (b \bigvee c) \geq (a \bigwedge b) \bigvee c$
5. $(a \bigwedge b) \bigvee (a \bigwedge c) \bigvee (b \bigwedge c) \leq (a \bigvee b) \bigwedge (a \bigvee c) \bigwedge (b \bigvee c)$
6. $(a \bigwedge b) \bigvee (a \bigwedge c) \bigvee (b \bigwedge c) \geq (a \bigvee b) \bigwedge (a \bigvee c) \bigwedge (b \bigvee c)$

(14) Sea C un orden total (o sea una cadena). Pruebe la identidad

$$x \bigvee (y \bigwedge z) = (x \bigvee y) \bigwedge (x \bigvee z),$$

que demuestra la distributividad de C.

8. Problemas

(1) Recordamos que pudimos concluir que $f : L \to L'$ es un isomorfismo de reticulados si y sólo si es un isomorfismo de posets (Lema 3.1). Lamentablemente, este resultado no puede ser extendido a la estructura de reticulado complementado. Supongamos se define *isomorfismo* de reticulados complementados como un isomorfismo f de reticulados que satisface las ecuaciones

$$f(0) = 0' \qquad f(1) = 1' \qquad f(x^c) = (f(x))^{c'}.$$

Encuentre dos reticulados complementados, y un iso de posets f entre ellos que no sea iso de reticulados complementados.

(2) Una mapa $f : L_0 \to L_1$ se dice un homomorfismo de reticulados si satisface las ecuaciones

$$f(x \owedge y) = f(x) \owedge f(y) \qquad f(x \ovee y) = f(x) \ovee f(y).$$

De las siguientes propiedades, que son válidas si f es un isomorfismo, diga cuáles son válidas si f es un homomorfsmo.

1. f preserva orden,
2. si L_0 es distributivo entonces L_1 también lo es,
3. si x es complementado entonces $f(x)$ también lo es,
4. $f(L_0)$ es subreticulado de L_1,
5. si L_0 tiene una copia de M_3 entonces L_1 también,

(3) Para los puntos 1,...,5 del ejercicio anterior, determinar en que casos los falsos se vueleven verdaderos:

a. f inyectiva,
b. f sobre,
c. f biyectiva.

(4) Sean L, M dos poset. Considere el conjunto $L \times M$ con la relación $\preceq$ definida:

$$(x_1, y_1) \preceq (x_2, y_2) \text{ sii } x_1 \leq_L x_2 \text{ y } y_1 \leq_M y_2.$$

i. Dé los digramas de hasse de :

a. $\mathbf{2} \times \mathbf{3}$ (aquí $\mathbf{n}$ denota la cadena de n elementos).
b. $\mathcal{P}(\{a,b\}) \times \mathbf{4}$.

ii. Pruebe que si L, M son reticulados entonces $L \times M$ es un reticulado. Dé explicitamente las operaciones

$$(x_1, y_1) \owedge (x_2, y_2)$$
$$(x_1, y_1) \ovee (x_2, y_2)$$

iii. Pruebe que si L, M (vistos como estructuras algebraicas) son distributivos, entonces $L \times M$ también lo es.

iv. Utilizando como guía lo hecho en ii defina el producto $B_0 \times B_1$ de las álgebras de Boole B_0 y B_1.

(5) Sea L un reticulado. Pruebe que L es distributivo si y sólo si para todo $a \in L$, el mapa

$$f_a(x) = (x \owedge a, x \ovee a)$$

definido en L, con imagen en $(a] \times [a)$, es un homomorfismo inyectivo.

(6) En un reticulado distributivo $(L, \barwedge, \barvee, 0)$ el pseudocomplemento de x es el máximo elemento z (si existe) que satisface $x \barwedge z = 0$. Pruebe que en las álgebras de Boole el pseudocomplemento es exactamente el complemento.

(7) Pruebe que si B es un álgebra de Boole finita entonces B es isomorfa a $\mathbf{2}^n$ para algún $n \geq 1$. (Aquí $\mathbf{2}^n$ denota al álgebra de Boole $\mathbf{2} \times ... \times \mathbf{2}$).

CAPÍTULO 4

Teoremas de representación

El primer objetivo de este capítulo será demostrar que toda álgebra de Boole finita es isomorfa al álgebra de subconjuntos de un conjunto finito (o sea $\mathcal{P}(X)$). Luego llegaremos a un resultado análogo para los reticulados distributivos finitos. En este caso ya no podremos hablar del álgebra de todos los subconjuntos de un conjunto finito (puesto que no todo reticulado distributivo finito es complementado), pero podremos establecer un resultado similar quedándonos con un subreticulado de tal álgebra de conjuntos.

1. Álgebras de Boole finitas

Procedamos ahora con la construcción de un conjunto X asociado a un álgebra de Boole finita B, tal que $\mathcal{P}(X)$ resulte isomorfo a B. El conjunto X que necesitamos estará formado por elementos de B. Una particularidad que tendrá este conjunto es que a partir de él podemos generar todos los elementos del álgebra a través de la operación supremo (o sea para todo $x \in B$ existe $S \subseteq B$ tal que $x = \sup S$). Es un buen ejercicio bucar en los diagramas de las álgebras de Boole de 4 y 8 elementos que subconjuntos tiene esta particularidad, y cuál de todos es el más chico.

DEFINICIÓN 1.1. **Átomo.** Sea B un álgebra de Boole. Un elemento $a \in B$ será llamado átomo si a cubre a 0. Mediante $At(B)$ denotamos el conjunto de todos los átomos de B.

El siguiente lema muestra que $At(B)$ es el conjunto que buscábamos.

LEMA 1.1. *Sea B un álgebra de Boole finita. Entonces todo elemento de B se escribe de manera única como supremo de átomos. O sea: para todo $x \in B$ se tiene:*

(1) $x = sup\{a \in At(B) : a \leq x\}$,
(2) *si* $A \subseteq At(B)$ *y* $x = sup\, A$, *entonces* $A = \{a \in At(B) : a \leq x\}$.

Vamos a proceder ahora con la prueba de este lema. Para esto necesitamos los siguientes resultados. El primero se prueba fácilmente.

LEMA 1.2. *Sea B un álgebra de Boole finita. Para todo $x \in B$ distinto de 0 existe $a \in At(B)$ tal que $a \leq x$.*

LEMA 1.3. *Sea B un álgebra de Boole finita, y sean $x, y \in B$ tales que $x \not\leq y$. Entonces existe $a \in At(B)$ tal que $a \leq x$ y $a \not\leq y$.*

DEMOSTRACIÓN. Veamos que $x \owedge y^c \neq 0$. Si $x \owedge y^c = 0$, entonces $y \ovee (x \owedge y^c) = y \ovee 0$, o sea $y \ovee x = y$, lo que contradice la hipótesis del lema. Luego $x \owedge y^c \neq 0$. Por el lema anterior existe $a \in At(B)$ tal que $a \leq x \owedge y^c$. Claramente $a \leq x$, veamos que $a \not\leq y$. Si $a \leq y$, como $a \leq x \owedge y^c$ tendríamos $a \leq x \owedge y^c \owedge y = 0$, lo que es absurdo puesto que a es un átomo. Luego $a \not\leq y$. □

DEMOSTRACIÓN. (del **Lema 1.1**)
Sea $A_x = \{a \in At(B) : a \leq x\}$, y sea $y = sup\{a \in At(B) : a \leq x\} = sup\, A_x$.

(1) Como x es cota superior de A_x, entonces claramente $y \leq x$. Supongamos ahora que $x \not\leq y$. Por el lema 1.3, existe $a \in At(B)$ tal que $a \leq x$ y $a \not\leq y$. Pero esto es absurdo, pues $a \leq x$ implica $a \in A_x$, lo que indica que $a \leq y$. Concluimos que $x \leq y$, y por lo tanto $x = y$.

(2) Que $A \subseteq A_x$ es inmediato: $a \in A$ implica $a \leq x$, lo que indica que $a \in A_x$. Veamos que $A_x \subseteq A$. Sea $b \in A_x$, y supungamos que $A = \{a_1, ..., a_n\}$. Como $x = a_1 \ovee ... \ovee a_n$ y B es distributiva, tenemos que

$$b = b \owedge x = (b \owedge a_1) \ovee ... \ovee (b \owedge a_n)$$

Como b es un átomo (y por lo tanto cubre al 0), tenemos que para todo $i = 1, ..., n$, el elemento $b \owedge a_i$ debe ser 0 o b, y además no puede ser que $b \owedge a_i = 0$ para todo i (sino b sería 0). Luego existe i tal que $b \owedge a_i = b$, lo que indica que $b = a_i$, puesto que ambos son átomos. Luego $b \in A$.

□

Ahora si estamos en condiciones de probar lo que era nuestro objetivo.

TEOREMA 1.4. *Sea* $\langle B, \ovee, \owedge, ^c, 0, 1\rangle$ *un álgebra de Boole finita, y sea* $X = At(B)$. *La función*

$$\begin{aligned} F : B &\longrightarrow \mathcal{P}(X) \\ x &\longrightarrow \{a \in X : a \leq x\} \end{aligned}$$

es un isomorfismo entre $\langle B, \ovee, \owedge, ^c, 0, 1\rangle$ *y* $\langle \mathcal{P}(X), \cup, \cap, ^c, \emptyset, X\rangle$.

DEMOSTRACIÓN. Sea $A_x = \{a \in At(B) : a \leq x\}$. Vemos primero que el mapa definido mediante $F(x) = A_x$ es una biyección entre B y $\mathcal{P}(At(B))$.

F es **uno-a-uno** porque $A_x = A_y$ implica $sup\, A_x = sup\, A_y$, lo que nos permite concluir desde el lema 1.1 (inciso 1) que $x = y$, dado que $x = sup\, A_x$ y $y = sup\, A_y$.

Vemos que F es **sobre**. Sea $A \subseteq At(B)$. Definamos $x = sup\, A$, y verifiquemos que $F(x) = A$. Por el lema 1.1 (inciso 2) tenemos que $A = A_x$, lo que indica que $F(x) = A$.

Veamos ahora que F es un isomorfismo. Hemos probado en el capítulo anterior que F es isomorfismo de álgebras de Boole si y sólo si es isomorfismo de posets. Luego, resta verificar:

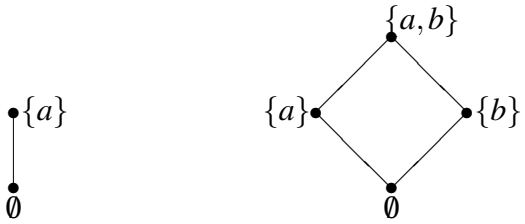
$$x \leq y \;\Leftrightarrow\; A_x \subseteq A_y,$$

o sea,

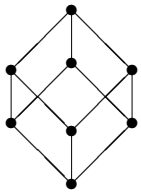
$$sup\, A_x \leq sup\, A_y \;\Leftrightarrow\; A_x \subseteq A_y.$$

La implicación ($\Leftarrow$) no presenta dificultades, y se deja como ejercicio para el lector. Supongamos $sup\, A_x \leq sup\, A_y$, y tomemos $a \in A_x$. Entonces $a \leq x = sup\, A_x \leq sup\, A_y = y$. Luego $a \in A_y$. □

Desde el Teorema anterior, concluimos que las dos álgebras de Boole no triviales más chicas son $\mathcal{P}(\{a\})$ y $\mathcal{P}(\{a,b\})$, cuyos diagramas son



Luego, le siguen en orden creciente $\mathcal{P}(X)$, con $|X| = 3, 4, \ldots$. El caso $|X| = 3$ tiene aún un diagrama fácil de dibujar:



Por último, el Teorema anterior nos permite responder la siguiente pregunta: Para qué números naturales n existe un álgebra de Boole B tal que $|B| = n$? La respuesta es: para todo número de la forma 2^k, con $k = 0, 1, \ldots$.

Antes de pasar al estudio de los reticulados distributivos, es natural preguntarse si el Teorema anterior puede extenderse a las álgebras de Boole infinitas. Lamentablemente, la respuesta es negativa, como lo demuestra el siguiente resultado.

LEMA 1.5. *Existe un álgebra de Boole infinita que no es isomorfa $\mathcal{P}(X)$, para ningún X.*

DEMOSTRACIÓN. Un subconjunto de números naturales se dice *cofinito* si su complemento es finito. Definamos

$$\mathcal{B} = \{X \subseteq \mathbf{N} : X \text{ es finito o cofinito}\}.$$

Note que las operaciones $\cup, \cap$ y ^{c} están bien definidas sobre $\mathcal{B}$ puesto que

$X \in \mathcal{B},\ Y \in \mathcal{B} \ \Rightarrow\ (X \cup Y) \in \mathcal{B}$,

$X \in \mathcal{B},\ Y \in \mathcal{B} \ \Rightarrow\ (X \cap Y) \in \mathcal{B}$,

$X \in \mathcal{B} \ \Rightarrow\ X^c \in \mathcal{B}$.

Luego, la estructura $\langle \mathcal{B}, \cup, \cap, \emptyset, \mathbf{N}, ^c \rangle$ es claramente un álgebra de Boole.

Veamos que no puede ser isomorfa a $\mathcal{P}(X)$, para ningún X. Para esto veremos que es imposible encontrar una función biyectiva entre $\mathcal{B}$ y $\mathcal{P}(X)$, cualquiera sea el X. Si X es finito, entonces $\mathcal{P}(X)$ es finito, por lo tanto es imposible encontrar tal biyección puesto que $\mathcal{B}$ es infinito.

El caso X infinito requiere un poco más de trabajo. Primero notemos que el conjunto $\mathcal{B}$ es infinito y numerable (o sea que puede ponerse en correspondencia biyectiva con los números naturales). En efecto, es sabido que $\{X \subseteq \mathbf{N} : X \text{ es finito}\}$ es numerable, y el mapa $X \to X^c$ es una biyección entre $\{X \subseteq \mathbf{N} : X \text{ es finito}\}$ y $\{X \subseteq \mathbf{N} : X \text{ es cofinito}\}$. Luego $\mathcal{B}$ resulta numerable puesto que es unión de dos conjunto numerables.

Por otro lado, es sabido que si X es un conjunto infinito, entonces $\mathcal{P}(X)$ es un conjunto infinito no numerable, luego no puede ponerse de ninguna manera en correspondencia con $\mathcal{B}$, que es numerable. □

2. Reticulados distributivos finitos

Anteriormente mencionamos que un reticulado ditributivo finito no necesariamente será de la forma $\mathcal{P}(X)$, puesto que no todos son complementados. Vamos a introducir ahora un álgebra (incompleta) de conjuntos, que jugará para los reticulados distributivos finitos el mismo rol que jugaba $\mathcal{P}(X)$ para las álgebras de Boole.

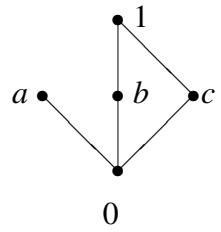
Dado un poset $(P, \leq)$, diremos que un subconjunto $D \subseteq P$ es *decreciente* si para todo $x, z \in P$ se tiene que:

$$x \in D \ \text{ y } z \leq x \implies z \in D.$$

O sea, un conjunto decreciente satisface que si un elemento se encuentra en el conjunto, entonces todos los elementos menores también están.

Por ejemplo, considere el poset P de abajo. El conjunto $\{0, a\}$ es decreciente, pero el conjunto $\{0, b, 1\}$ no lo es, porque 1 está en el conjunto y c no.

P



Denotaremos mediante $\mathcal{D}(P)$ a la familia de todos los subconjuntos decrecientes de P:

$$\mathcal{D}(P) = \{D \subseteq P : D \text{ es decreciente}\}.$$

Notemos que $\emptyset$ y P son decrecientes. Además la unión e intersección de subconjuntos decrecientes es decreciente.

Las observaciones anteriores nos dicen que dado un poset finito $(P, \leq)$, tenemos asociado naturalmente el reticulado acotado

$$\langle \mathcal{D}(P), \cup, \cap, \emptyset, P \rangle.$$

Además tal reticulado es distributivo, puesto que en el álgebra de conjuntos se satisfacen las leyes distributivas.

Tenemos entonces un reticulados formado por conjuntos que jugará para los reticulados distributivos finitos el rol que jugaba $\mathcal{P}(X)$ para las álgebras de Boole.

Qué elementos desempeñarán el rol de los átomos?

DEFINICIÓN 2.1. Sea $\langle L, \bigvee, \bigwedge, 0, 1 \rangle$ un reticulado acotado. Un elemento $x \in L$ será llamado $\bigvee$-*irreducible* si

(1) $x \neq 0$,
(2) si $x = y \bigvee z$, entonces $x = y$ o $x = z$, para todo $y, z \in L$.

Notemos que la condición (2) es equivalente a la siguiente condición, la cual es mas fácil de chequear en los diagramas de Hasse:

($2'$) si $y < x$ y $z < x$, entonces $y \bigvee z < x$, para todo y, $z \in L$.

En el caso en que L es finito, nótese que $2'$ es equivalente a

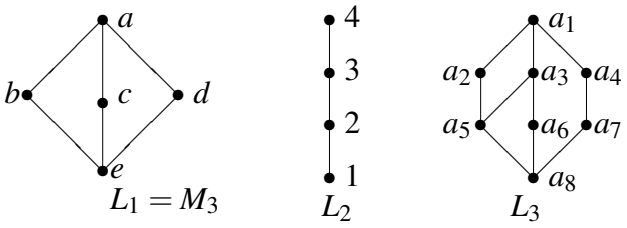
($2''$) existe un z tal que $z < x$ y $z = \sup\{y \mid y < x\}$.

De la condición ($2''$) podemos concluir fácilmente que todo átomo de $\langle L, \bigvee, \bigwedge, 0, 1 \rangle$ es un elemento $\bigvee$-irreducible.

Si $\langle L, \bigvee, \bigwedge, 0, 1 \rangle$ es un reticulado acotado, denotaremos

$$Irr(L) = \{i \in L : i \text{ es } \bigvee\text{-irreducible}\}.$$

Por ejemplo, considere los siguientes reticulados.



Los conjuntos de átomos e irreducibles son dados a continuación.

$$At(L_1) = \{b,\ c,\ d\} \qquad Irr(L_1) = \{b,\ c,\ d\},$$
$$At(L_2) = \{2\} \qquad Irr(L_2) = \{2,\ 3,\ 4\},$$
$$At(L_3) = \{a_5, a_6, a_7\}, \qquad Irr(L_3) = \{a_2,\ a_4,\ a_5,\ a_6,\ a_7\}.$$

Si X es un conjunto finito, entonces ya hemos mencionado que los subconjuntos que contienen un solo elemento son los átomos de $\mathcal{P}(X)$. Es un buen ejercicio comprobar en algunos casos ($|X| = 2, 4, 8$) que estos además son los únicos elementos irreducibles de $\mathcal{P}(X)$. La noción de $\bigvee$-irreducible se reduce a la noción de átomo, en el caso de las álgebras de Boole. Esto establece el siguiente Lema.

LEMA 2.1. *Sea $\langle B, \bigvee, \bigwedge, ^c, 0, 1\rangle$ un álgebra de Boole. Un elemento $x \in B$ es $\bigvee$-irreducible si y sólo si x es un átomo.*

DEMOSTRACIÓN. Como ya se observó anteriormente todo átomo es $\bigvee$-irreducible. Supongamos ahora que $x \in Irr(B)$ y sea $y \in B$ tal que $0 \leq y < x$. Veremos que $y = 0$. Tenemos

$$x = x \bigwedge 1 = x \bigwedge (y \bigvee y^c) = y \bigvee (x \bigwedge y^c)$$

En consecuencia $x \leq x \bigwedge y^c$, lo cual implica que $x = x \bigwedge y^c$, es decir $x \leq y^c$. Pero entonces tenemos que $y \leq x^c$, lo cual nos dice que $y = 0$, ya que $y < x$.

□

Nuestro próximo objetivo es demostrar que todo reticulado distributivo finito es isomorfo al reticulado de los decrecientes de un poset P. Seguiremos exactamente los pasos que efectuamos para el caso de las álgebras de Boole. En particular, el candidato a ser el poset $(P, \leq)$ asociado la reticulado L es $(Irr(L), \leq)$, donde $\leq$ es el orden heredado de L.

Vamos ahora a probar una serie de lemas que nos permitiran estructurar para los reticulados una demostración similar a la desarrollada para el caso de las àlgebras de Boole.

LEMA 2.2. *Sea $\langle L, \bigvee, \bigwedge, 0, 1\rangle$ un reticulado acotado distributivo y sea $x \in Irr(L)$. Si $x_1, ..., x_n \in L$ y $x \leq x_1 \bigvee x_2 \bigvee \ldots \bigvee x_n$, entonces $x \leq x_i$, para algún $i = 1, ..., n$.*

DEMOSTRACIÓN. Haremos la prueba haciendo inducción en n. Si $n = 1$ el resultado es obvio. Probemos para $n = 2$. Entonces si $x \leq x_1 \bigvee x_2$ tenemos que

$$\begin{aligned} x &= x \bigwedge (x_1 \bigvee x_2) \qquad (\text{ pues } x \leq x_1 \bigvee x_2) \\ &= (x \bigwedge x_1) \bigvee (x \bigwedge x_2) \qquad (\text{ pues } L \text{ es distributivo}). \end{aligned}$$

Dado que x es irreducible debe ser $x = x \bigwedge x_1$ o $x = x \bigwedge x_2$, luego $x \leq x_1$ o $x \leq x_2$.

Supongamos ahora que si $x \leq x_1 \bigvee x_2 \bigvee \ldots \bigvee x_k$, entonces $x \leq x_i$, para algún $i = 1, ..., k$. Veamos que lo mismo ocurre si $n = k+1$. Por lo visto en el caso $n = 2$ podemos ver que si $x \leq (x_1 \bigvee x_2 \bigvee \ldots \bigvee x_k) \bigvee x_{k+1}$ entonces $x \leq (x_1 \bigvee x_2 \bigvee \ldots \bigvee x_k)$ o $x \leq x_{k+1}$. Aplicando la hipótesis inductiva concluimos que $x \leq x_i$ para algún i, $1 \leq i \leq k$ o $x \leq x_{k+1}$; luego $x \leq x_i$ para algún i, $1 \leq i \leq x_{k+1}$.

□

LEMA 2.3. *Sea L un reticulado finito, y sean $x, y \in L$ tales que $x \not\leq y$. Entonces existe $i \in Irr(L)$ tal que $i \leq x$ e $i \not\leq y$.*

DEMOSTRACIÓN. Sea $I = \{z \in L : i \leq x \text{ e } i \not\leq y\}$. Claramente I no es el conjunto $\emptyset$, puesto que x es un elemento de I. Entonces como L es finito, I posee un elemento minimal, llamémoslo i. Para concluir la prueba, sólo tenemos que ver que $i \in Irr(L)$. Sean u, v tales que $i = u \bigvee v$, sin pérdida de generalidad, supongamos que $i \neq u$. Veamos que $i = v$.

Claramente $v \leq x$. Veamos primero que $v \not\leq y$. Supongamos por un momento que $v \leq y$. Entonces $u \not\leq y$, porque de lo contrario tendríamos $i = u \bigvee v \leq y$, lo que es una contradicción. Pero entonces $u \in I$, lo que es un absurdo, puesto que i es minimal de I.

Entonces tenemos que $v \not\leq y$, lo que indica que $v \in I$. Como i es minimal, concluimos que $v = i$. □

LEMA 2.4. *Sea L un reteticulado distributivo finito. Entonces para todo $x \in L$ se tiene:*

(1) $x = sup\{i \in Irr(L) : i \leq x\}$,
(2) *si $D \subseteq Irr(L)$ es decreciente, y $x = sup\, D$, entonces $D = \{i \in Irr(L) : i \leq x\}$.*

DEMOSTRACIÓN. Sea $D_x = \{i \in Irr(L) : i \leq x\}$, y sea $y = sup\{i \in Irr(L) : i \leq x\}$.

(1) Se repite exactamente el argumento desarrollado para las álgebras de Boole.
(2) El argumento es muy similar al desarrollado para el caso de la álgebras de Boole. De todas maneras la haremos en detalle.

Que $D \subseteq D_x$ es inmediato: $i \in D$ implica $i \leq x$, lo que indica que $i \in D_x$. Veamos que $D_x \subseteq D$. Sea $j \in D_x$, y supongamos que $D = \{i_1, ..., i_n\}$. Como $x = i_1 \bigvee ... \bigvee i_n$ y L es distributiva, tenemos que

$$j = j \bigwedge x = (j \bigwedge i_1) \bigvee ... \bigvee (j \bigwedge i_n)$$

Como $j \in Irr(L)$, tenemos que existe k tal que $j \bigwedge i_k = j$, lo que indica que $j \leq i_k$. Como D es decreciente tenemos que $j \in D$.

□

Después de esta maratón de lemas estamos en condiciones de probar nuestro (mejor dicho, de Birkhoff) teorema de representación para reticulados distributivos finitos.

TEOREMA 2.5. ***(Teorema de representación de Birkhoff)*** *Sea* $\langle L, \bigvee, \bigwedge, 0, 1\rangle$ *un reticulado acotado distributivo finito, y sea* $P = Irr(L)$. *Entonces la función*

$$\begin{aligned} F : L &\longrightarrow \mathcal{D}(P) \\ x &\longrightarrow \{y \in P : y \leq x\} \end{aligned}$$

es un isomorfismo entre $\langle L, \bigvee, \bigwedge, 0, 1\rangle$ *y* $\langle \mathcal{D}(P), \cup, \cap, \emptyset, P\rangle$.

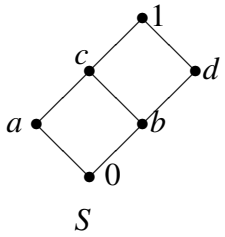
DEMOSTRACIÓN. Sea $D_x = \{i \in Irr(L) : i \leq x\}$. Para ver que el mapa definido mediante $F(x) = D_x$ es una biyección entre L y $\mathcal{D}(Irr(L))$, repetimos exactamente el argumento hecho para el caso de las álgebras de Boole, pero usando en este caso el Lema 2.4.

Veamos ahora que F es un isomorfismo. Hemos probado en el capítulo anterior que F es isomorfismo de reticulados distributivos acotados si y sólo si es isomorfismo de posets. Luego, resta verificar:

$$x \leq y \;\Leftrightarrow\; D_x \subseteq D_y,$$

Aquí nuevamente repetimos el argumento hecho para el caso de las álgebras de Boole. □

Por ejemplo, consideremos el siguiente reticulado S.



S

Aquí $Irr(S) = \{a, b, d\}$. La familia de subconjuntos decrecientes de $Irr(S)$ es

$$\mathcal{D}\big(Irr(S)\big) = \{\emptyset, \{a\}, \{b\}, \{b,d\}, \{a,b\}, \{a,b,d\}\}.$$

La correspondencia F dada por el Teorema 2.5 es:

$$\begin{aligned} 0 &\to \emptyset & a &\to \{a\} \\ b &\to \{b\} & d &\to \{b,d\} \\ c &\to \{a,b\} & 1 &\to \{a,b,d\} \end{aligned}$$

Finalmente, consideremos el reticulado $L = D_{36}$. Entonces $Irr(L) = \{2,3,4,9\}$. La familia de subconjuntos decrecientes de $Irr(L)$ es:

$$\mathcal{D}(Irr(L)) = \{\emptyset, \{2\}, \{2,4\}, \{3\}, \{3,9\}, \{2,3\}, \{2,4,3\}, \{2,3,9\}, \{2,3,4,9\}\}.$$

La correspondencia entre L y $\mathcal{D}(Irr(L))$ está dada por

$$F(n) = \{k \in Irr(L) \mid k \text{ divide a } n\},$$

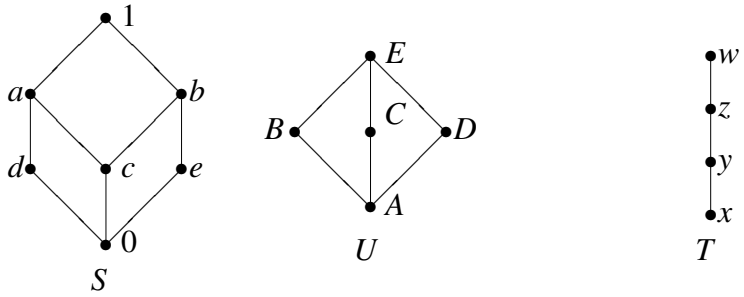
por ejemplo, $f(18) = \{2,3,9\}$.

Finalmente, los teoremas probados en las secciones anteriores nos permiten probar fácilmente la siguiente caracterización de las álgebras de Boole finitas:

COROLARIO 2.6. *Si $\langle L, \bigvee, \bigwedge, 0, 1\rangle$ es un reticulado acotado distributivo finito, entonces $\langle L, \bigvee, \bigwedge, 0, 1\rangle$ es álgebra de Boole si y sólo si $Irr(L) = At(L)$.*

3. Ejercicios

(1) Considere los reticulados S, T y U de la siguiente figura:



(a) Calcule el conjunto de átomos de cada reticulado.
(b) Calcule el conjunto de irreducibles de cada reticulado.
(c) ¿Tiene alguno de esos reticulados elementos irreducibles que no sean átomos?

(2) (a) Encuentre los átomos de $(D_{12}, |)$.
(b) Muestre que los elementos 2 y 6 en D_{12} no tiene complementos.
(c) Encuentre los elementos irreducibles de D_{12}.
(d) Escriba el elemento máximo de D_{12} como supremos de elementos irreducibles.

(3) (a) Encuentre los átomos de $(D_{36}, |)$.
(b) Encuentre los elementos irreducibles de D_{36}.
(c) Escriba al elemento máximo de D_{36} como unión de elementos irreducibles.

(4) Considere los diagramas de la Fig. 1.

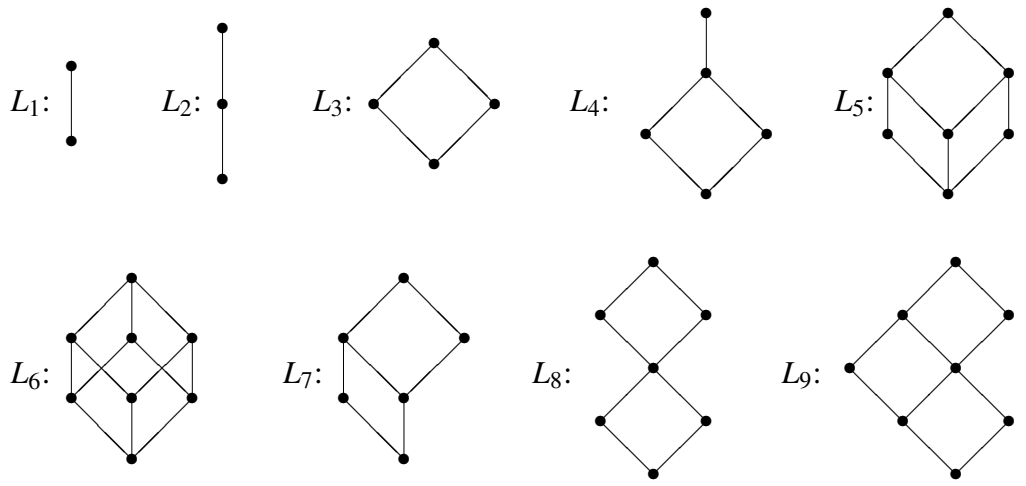


FIGURA 1

(a) Halle en cada caso $At(L)$.
(b) Halle en cada caso $Irr(L)$.
(c) Dibuje en cada caso el diagrama de Hasse de $\mathcal{P}(At(L))$.
(d) Dibuje en cada caso el diagrama de Hasse de $\mathcal{D}(Irr(L)$.
(e) Determine cuáles son álgebras de Boole.

(5) Para cada uno de los reticulados de la Fig. 1, determine cuales satisfacen las hipótiesis del Teorema 1.4. En tal caso dar explícitamente el mapa F.

(6) Para cada uno de los reticulados de la Fig. 1, determine cuales satisfacen las hipótiesis del Teorema 2.5. En tal caso dar explícitamente el mapa F. Qué propiedades tiene?

(7) Para cada uno de los reticulados de la Fig. 1, utilice el Teorema 2.5 para determinar si el reticulado es distributivo o no.

4. Problemas

(8) Encuentre todos los reticulados distributivos con exactamente 3 join irreducibles.

(9) Encuentre todos los reticulados distributivos con exactamente 4 join irreducibles y un sólo atomo (que está contado entre los 4 join irred.).

(10) Efectúe un rastreo en la prueba del Teorema de Birkhoff para comprobar frente a la ausencia de la propiedad de distributividad, aún se puede probar que el mapa F es inyectivo. Utilice este hecho para demostrar que la siguiente propiedad es válida para todos los reticulados finitos:

*L **no** es distributivo si y sólo si $|L| < |\mathcal{D}(Irr(L))|$.*

(11) Supongamos que n es producto de primos distintos p_1, p_2, ..., p_k, ¿cuáles son los elementos irreducibles de D_n?.

(12) Encuentre un homomorfismo f de D_{90} en $\mathbf{2}$ que que separe el 10 del 9, o sea que $f(10) \neq f(9)$.

(13) En ejercicios anteriores hemos definido el producto de reticulados. Pruebe lo siguiente:
   (a) Si $i \in Irr(L)$ entonces $(i, 0_G) \in Irr(L \times G)$
   (b) Si $j \in Irr(G)$ entonces $(0_L, j) \in Irr(L \times G)$
   (c) Si $(x,y) \in Irr(L \times G)$, entonces ocurre una de las dos siguientes posibilidades:
      (i) $x = 0_L$ e $y \in Irr(G)$
      (ii) $y = 0_G$ y $x \in Irr(L)$.

(14) Queremos abordar en este problema la siguiente cuestión, formulada para reticulados distributivos finitos L y G.

Supongamos que $L \times G = \mathcal{D}(P)$, para cierto poset P. ¿Qué relación tiene P con $Irr(L)$ e $Irr(G)$?

   (a) Estudie varios ejemplos (puede experimentar con $\mathbf{2} \times \mathbf{3}$, $\mathbf{2} \times \mathbf{3}$ y $\mathbf{2} \times \mathcal{P}(\{a,b\})$ por ejemplo).
   (b) Formule una conjetura.
   (c) Trate luego de obtener una prueba formal de la conjetura.

(15) (a) Describa de la forma más clara posible los elementos irreducibles de D_n.
   (b) Determine $Irr(D_{300})$. Escriba a D_{300} como producto de cadenas.
   (c) Pruebe que D_n es isomorfo a un producto de cadenas.

CAPÍTULO 5

Filtros maximales en álgebras de Boole

En esta sección vamos a estudiar la noción de *filtro maximal*, que tiene fundamental importancia en la teoría de las álgebras de Boole. Los conceptos y resultados vertidos en esta sección será utilizados en Lógica.

En esta sección B será siempre un álgebra de Boole. Un subconjunto no vacío $F \subseteq B$ se dice *filtro* si es *creciente* y *cerrado para el* $\bigwedge$, o sea:

(1) $x \in F$, $x \leq y$ implica $y \in F$,

(2) $x \in F$, $y \in F$ implican $x \bigwedge y \in F$.

Un filtro P se dice *propio* si está contenido estrictamente en B (o sea, si visto como cunjunto es distinto de B).

Un filtro propio P se dice *primo* si cada vez que $x \bigvee y \in P$ se tiene que $x \in P$ o $y \in P$.

Por último, un filtro F se dice *maximal* si no existe otro filtro propio Q (distinto de F) que lo contenga. Esto es lo mismo que decir que F es un elemento maximal del poset formado por todos los filtros propios.

El siguiente lema muestra que el concepto de filtro maximal, de vital importancia en el estudio de las álgebras de Boole infinitas, queda reducido al concepto de átomo para el caso de las finitas.

LEMA 0.1. *Sea B un álgebra de Boole finita. Entonces M es filtro maximal si y sólo si existe un átomo a tal que* $M = \{x \in B : x \geq a\}$.

Prueba: ($\Rightarrow$) Si $M = \{x_1, ..., x_n\}$ entonces M se puede expresar de la siguiente manera:

$M = \{x \in B : x \geq x_1 \bigwedge .... \bigwedge x_n\}$

Como no puede existir otro filtro propio más grande que M, entonces $x_1 \bigwedge .... \bigwedge x_n$ debe ser un átomo.

($\Leftarrow$) Sea Q un filtro que contiene a M estrictamente, y sea $x \in Q - M$. Veamos que $Q = B$. Como a no es menor o igual que x, entonces tenemos que $x \bigwedge a = 0$. Como $x, a \in Q$ se tiene que $0 = x \bigwedge a \in Q$. Luego $Q = B$.

El siguiente lema muestra tres formas distintas de caracterizar los filtros maximales.

LEMA 0.2. *Sea B un álgebra de Boole, y sea P un filtro propio de B. Entonces son equivalentes:*

i. P es primo.

ii. P es maximal.

iii. Para todo $x \in B$ se tiene $x \in P$ o $x^c \in P$.

Prueba: Supongamos *i*. Sea Q tal que $P \subset Q$ y Q es un filtro. Veamos que $Q = B$. Tomemos $a \in Q - P$. Como $a \bigvee a^c = 1 \in P$ y $a \notin P$, se tiene que $a^c \in P \subset Q$. Luego $a, a^c \in Q$ y por lo tanto $0 = a \bigwedge a^c \in Q$. Como Q es un filtro se tiene que $Q = B$. Luego hemos demostrado *ii*.

Supongamos *ii*. Sea $a \notin P$, $a \neq 0$. Definamos

$$Q = \{x \in L : x \geq a \bigwedge p \text{ para algún } p \in P\}.$$

Veamos que Q es un filtro. Dejamos al lector el verificar que Q es creciente. Sea $x, y \in Q$, veamos que $x \bigwedge y \in Q$. Sabemos que existen p, q tales que $x \geq a \bigwedge p$ y $y \geq a \bigwedge q$. Luego

$$x \bigwedge y \geq (a \bigwedge p) \bigwedge (a \bigwedge q) = a \bigwedge (p \bigwedge q).$$

Como P es un filtro se tiene que $p \bigwedge q \in P$. Luego $x \bigwedge y \in Q$. Concluimos entonces que Q es un filtro.

Q no puede ser propio, pues sino $P = Q$ (porque P es maximal) y tendríamos entonces $a \in P$, que es una contradicción. Luego $Q = B$. Como $0 \in Q$ deducimos que existe $p \in P$ tal que $0 \geq a \bigwedge p$, o sea $0 = a \bigwedge p$. Entonces

$$a^c = a^c \bigvee (a \bigwedge p) = (a^c \bigvee a) \bigwedge (a^c \bigvee p) = a^c \bigvee p.$$

Esto dice que $a^c \geq p \in P$, por lo tanto $a^c \in P$. Luego hemos probado *iii*.

Supongamos por último *iii*. Sean $x, y \in L$ tales que $x \bigvee y \in P$. Supongamos además que $x \notin P$. Entonces $x^c \in P$, y también

$$x^c \bigwedge (x \bigvee y) = (x^c \bigwedge x) \bigvee (x^c \bigwedge y) = x^c \bigwedge y \in P.$$

Como $y \geq x^c \bigwedge y \in P$ se tiene que $y \in P$. Luego hemos demostrado que para todo par $x, y \in L$ tales que $x \bigvee y \in P$, se tiene $x \in P$ o $y \in P$. Esto concluye la demostración de *i*.

CAPÍTULO 6

Reticulados completos y operadores clausura

En esta sección L será un reticulado cualquiera, no necesariamente distributivo como en las secciones anteriores. Diremos que L es *completo* si todo subconjunto S (aún los infinitos) poseen supremo e ínfimo. Generalizando la notación de reticulados, utilizaremos $\bigvee S$ y $\bigwedge S$ para denotar al supremo y al ínfimo de S, respectivamente. El objetivo principal de esta sección es encontrar una representación de los reticulados completos como "algebras de conjuntos", de manera similar a lo hecho para las álgebras de Boole y los reticulados distributivos finitos. Dado que no son necesariamente distributivos, no podremos representar ambas operaciones $\bigvee, \bigwedge$ como $\cup, \cap$ simultaneamente.

Note que todo reticulado completo L posee necesarimamente un menor elemento $0 = \bigwedge L$, y un mayor elemento, $1 = \bigvee L$. Por otro lado todo reticulado finito es completo.

LEMA 0.3. *Sea P un poset. Son equivalentes:*
i. P es un reticulado completo.
ii. Existe $\bigwedge S$ para todo subconjunto S de P.
iii. P tiene mayor elemento 1 y existe $\bigwedge S$ para todo subconjunto no vacío S de P.

Prueba: La equivalencia entre *ii* y *iii* sale observando que $\bigwedge \emptyset$ es siempre el mayor elemento del poset.

La implicación $i \Rightarrow ii$ es trivial.

Supongamos *ii*. Sea $S \subseteq P$. Sea

$$S_1 = \{x \in P : x \text{ es cota superior de } S\}.$$

Definamos $a = \bigwedge S_1$, vamos a demostrar que a es el supremo de S. Si $S = \emptyset$ entonces $S_1 = P$, y por lo tanto $\bigwedge S_1 = 0 = \bigvee \emptyset$. Supongamos ahora que $S \neq \emptyset$, y sea $z \in S$. Entonces para todo $x \in S_1$ se tiene $z \leq x$, lo que indica que $z \leq \bigwedge S_1 = a$. Luego hemos demostrado que a es cota superior de S. Por definición de S_1 se tiene que a es la menor de las cotas superiores.

LEMA 0.4. *Sea L un reticulado completo. Entonces para todo $S, T \subseteq L$ se tiene:*

$$\bigvee(S \cup T) = (\bigvee S) \bigvee \left(\bigvee T\right)$$
$$\bigwedge(S \cup T) = (\bigwedge S) \bigwedge \left(\bigwedge T\right).$$

La prueba del lema se deja como ejercicio.

Para la representación de los reticulados completos como "algebras de conjuntos" necesitamos el siguiente concepto.

DEFINICIÓN 0.1. Sea X cualquier conjunto. Una función $C : \mathcal{P}(X) \to \mathcal{P}(X)$ se dice un operador clausura sobre X si para todo $A, B \subseteq X$ se cumple:

1. $A \subseteq C(A)$,
2. $A \subseteq B \Rightarrow C(A) \subseteq C(B)$,
3. $C(C(A)) = C(A)$.

Sea C un operador clausura sobre X. Diremos que un subconjunto A de X es *cerrado* si $C(A) = A$. Note que el conjunto total X es cerrado, por la propiedad 1. Una propiedad importante de los operadores clausura es que la intersección de cerrados es cerrada. No ocurre necesariamente lo mismo con la unión.

LEMA 0.5. *La intersección de una familia arbitraria de cerrados es también un cerrado. En símbolos, si $\{A\}_{i\in I}$ es una familia de subconjuntos cerrados de X, y*

$$B = \cap_{i\in I} A_i,$$

entonces $C(B) = B$.

Prueba. Por la condición 1, basta ver que $C(B) \subseteq B$. Para cada $i \in I$ se tiene que $B \subseteq A_i$, y por la condición 2 tenemos

$$C(B) \subseteq C(A_i) = A_i.$$

Luego

$$C(B) \subseteq \cap_{i\in I} A_i = B. \square$$

Notemos que un caso particular contemplado en el Lema es es caso $I = \emptyset$. En tal caso $B = X$ que por la condición 1 es claramente un cerrado.

Asociado a un operador clausura tenemos siempre un reticulado completo. En efecto, sea C un operador clausura sobre X y sea Γ_C el conjunto de los subconjuntos cerrados de X. Consideremos la estructura de poset de Γ_C dada por la relación $\subseteq$.

Vamos a dar ahora algunos ejemplos.

(1) Sea P un poset, y sea C el operador sobre P definido

$$C(X) = \{z \in P : z \geq x, \text{ para algún } x \in X\}.$$

Es fácil verificar que C es un operador clausura. Los cerrados del operador clausura son los subconjuntos crecientes de P. Este operador clausura satisface claramente que la unión de cerrados es cerrado.

(2) Sea L un reticulado y sea C el operador clausura en L definido de la siguiente manera: para cada subconjunto X del reticulado, $C(X)$ es el subreticulado más chico que contiene a X. Entonces C es un operador clausura, y los cerrados del operador C sun exactamente los subreticulados de L. Claramente este operador no satisface que la unión de cerrados es cerrado

COROLARIO 0.6. *(Del Lema 0.3)* Γ_C *es un reticulado completo.*

Prueba: Dado que la intersección de una familia arbitraria de miembros de Γ_C está en Γ_C, tenemos que el ínfimo de S existe para todo $S \subseteq \Gamma_C$. Por el Lema 0.3 tenemos que Γ_C es un reticulado completo, y la operación $\bigwedge$ coincide con el operador $\cap$. $\square$

Si nos fijamos en la prueba del Lemma 0.3 podremos determinar inmediatamente como está definida la operación $\bigvee$ en Γ_C. En efecto, sea $S = \{A_i\}_{i\in I}$ una familia de cerrados. Entonces

$$\bigvee S = \cap S_1,$$

donde

$$S_1 = \{B \in \Gamma_C : A_i \subseteq B \text{ para todo } i \in I\}.$$

Claramente $A_i \subseteq \cap S_1$, lo que implica

$$\cup_{i\in I} A_i \subseteq \cap S_1.$$

Como $\cap S_1$ es cerrado tenemos

$$C\left(\cup_{i\in I} A_i\right) \subseteq \cap S_1.$$

Por otro lado, $C\left(\cup_{i\in I} A_i\right)$ es cerrado y es cota superior de S, lo que indica que

$$\cap S_1 \subseteq C\left(\cup_{i\in I} A_i\right),$$

pues $\cap S_1$ es el supremo de S. Concluimos que para todo familia de cerrados $\{A_i\}_{i\in I}$ se tiene:

$$\bigvee \{A_i\}_{i\in I} = C\left(\cup_{i\in I} A_i\right).$$

Hemos probado entonces el siguiente corolario:

COROLARIO 0.7. *Sea X un conjunto y sea C un operador clausura sobre X. Definamos para $S \subseteq \Gamma_C$:*

$$\bigvee S = C(\cup S).$$

Entonces $(\Gamma_C, \cap, \bigvee)$ *es un reticulado completo.*

1. Ejercicios

(1) Sea P un poset, y sea C el operador sobre P definido

$$C(X) = \{z \in P : z \geq x, \text{ para algún } x \in X\}.$$

Pruebe que C es un operador clausura y que los cerrados de C son exactamente los subconjuntos crecientes.

(2) Sea L un reticulado y sea C el operador clausura en L definido de la siguiente manera: para cada subconjunto X del reticulado, $C(X)$ es el subreticulado más chico que contiene a X. Pruebe que C es un operador clausura, y los cerrados del operador C sun exactamente los subreticulados de L.

Apunte de Introducción a la Lógica y la Computación
Lógica Proposicional

Pedro Sánchez Terraf (CIEM-FAMAF)

13 de octubre de 2022

Este apunte está basado principalmente en el primer capítulo del libro *"Logic and Structure"* de Dirk van Dalen (tercera edición, Springer)[1], y se nutrió con las sugerencias y correcciones indicadas por H. Gramaglia, M. Pagano, D. Alonso, J. Venzon, P. Dal Lago y Luis M. Ferrer Fioritti, entre otros. Agradezco a Pablo Villalba por los apuntes tomados durante el segundo semestre de 2003.

Índice

1. La Lógica Proposicional: lo básico

1.1. Introducción: Semántica versus Sintaxis

Dividiremos el análisis de la sintaxis proposicional en dos partes, primero presentando su lenguaje y en segundo término dando un procedimiento para "derivar" proposiciones

a partir de otras. Entre esas dos partes formalizaremos la semántica. Para finalizar, investigaremos las relaciones que surgen entre las nociones definidas por vía sintáctica y por vía semántica.

1.2. Lenguaje de la Lógica Proposicional

La lógica proposicional se escribirá con el siguiente alfabeto:

1. ***Símbolos proposicionales*** (en cantidad numerable): $p_0, p_1, \ldots, p_n, \ldots$;
2. ***Conectivos***: $\bot$, $\wedge$, $\vee$, $\to$.
3. Símbolos ***auxiliares***: '(' y ')'.

Los símbolos proposicionales y $\bot$ son los ***átomos*** o "proposiciones atómicas", y los designaremos con el nombre At. El conectivo "$\bot$" es "nulario" (corresponde a una *constante*) y el resto son binarios. Para designar un operador binario arbitrario, utilizaremos el símbolo "$\odot$".

Definición 1. El conjunto de las *proposiciones*, $PROP$, es el menor conjunto X que cumple con las siguientes propiedades:

$\boxed{\varphi \in At}$ Para todo $\varphi \in At$, $\varphi \in X$.

$\boxed{(\varphi \odot \psi)}$ Para todas φ, ψ en X, $(\varphi \odot \psi)$ está en X.

También usaremos los nombres ***fórmulas proposicionales*** o simplemente "fórmulas" para nombrar a las proposiciones. Utilizaremos las letras griegas φ, ψ, χ, ...[2] para nombrar proposiciones. Así, la expresión $(\varphi \odot \psi)$ puede ser $(\varphi \wedge \psi)$, ó $(\varphi \to \psi)$, o bien $(\varphi \vee \psi)$, etcétera. Sería bueno notar que los símbolos que utilizamos que no están entre los enumerados más arriba, **no son** proposiciones ni pueden formar parte de ellas. Digamos, el símbolo φ se utilizó más arriba para *designar* una proposición, para ser el nombre, no la proposición misma. Claramente, la letra griega φ no es ningún p_i, ni un conectivo ni un paréntesis. Es el nombre que le damos a una proposición, y cuando decimos $(\varphi \to \psi)$ nos estamos refiriendo a cualquiera de las fórmulas que tengan dicha *estructura*, por ejemplo $(p_0 \to p_1)$, $(p_3 \to (p_1 \wedge p_4))$, etcétera.

Teorema 2 (inducción en subfórmulas)**.** *Sea A un predicado sobre $PROP$. Luego $A(\varphi)$ es verdadero para toda $\varphi \in PROP$ si y sólo si:*

$\boxed{\varphi \in At}$ *Si φ es atómica, $A(\varphi)$ vale.*

$\boxed{(\varphi \odot \psi)}$ *Si $A(\varphi)$ y $A(\psi)$ entonces $A((\varphi \odot \psi))$.*

Demostración. Sea $X = \{\varphi \in PROP : A(\varphi)\}$. X satisface las tres propiedades de la Definición 1, así que $PROP \subseteq X$ (pues $PROP$ es el **menor** conjunto con tales propiedades). Como $X \subseteq PROP$, tenemos $X = PROP$ y entonces $A(\varphi)$ vale para toda $\varphi \in PROP$. □

Veamos un ejemplo de prueba por inducción:

[2]Las letras griegas "φ", "ψ" y "χ" se llaman, respectivamente, "fi", "psi" y "ji".

Definición 3. Una sucesión de proposiciones $\varphi_1, \ldots, \varphi_n$ es una ***serie de formación*** de $\varphi \in PROP$ si $\varphi_n = \varphi$ y para todo $i \leq n$, φ_i es:

- atómica, o bien
- igual a $(\varphi_j \odot \varphi_k)$ con $j, k < i$.

Teorema 4. *Toda $\varphi \in PROP$ tiene una serie de formación.*

Demostración. Analizamos cada caso:

$\boxed{\varphi \in At}$ "φ" es una serie de formación de φ (tenemos $n = 1$, $\varphi_1 := \varphi$).

$\boxed{(\varphi \odot \psi)}$ Por hipótesis inductiva, φ y ψ tienen sendas series de formación; llamémoslas $\varphi_1, \ldots, \varphi_n (= \varphi)$ y $\psi_1, \ldots, \psi_m (= \psi)$. Luego $\varphi_1, \ldots, \varphi_n, \psi_1, \ldots, \psi_m, (\varphi \odot \psi)$ es serie de formación de $(\varphi \odot \psi)$: contrastar con las definiciones.

Con esto se concluye la prueba. □

Dado que las proposiciones tienen una construcción recursiva, uno puede definir objetos en términos de proposiciones de manera recursiva (también llamada inductiva). Veamos un ejemplo:

Definición 5. El *grado* de una proposición, $gr(\cdot)$, es la función definida de la siguiente manera.

$\boxed{\varphi \in At}$ Si $\varphi = p_n$, $gr(\varphi) := n$; $gr(\bot) := -1$.

$\boxed{(\varphi \odot \psi)}$ $gr((\varphi \odot \psi)) := \text{máx}\{gr(\varphi), gr(\psi)\}$.

Es decir, el grado de una proposición es el máximo subíndice de los símbolos proposicionales que ocurren en ella. ¿Cómo aplicamos tal definición? Calculemos $gr(((p_0 \wedge p_3) \to p_2))$:

$$gr\big(((p_0 \wedge p_3) \to p_2)\big) = \text{máx}\big\{gr\big((p_0 \wedge p_3)\big), gr(p_2)\big\} \qquad \text{Por el caso "}\odot\text{"}$$

Ahora nos haría falta hacer lo mismo para cada término, es decir, *recursivamente:*

$$\begin{aligned}
&= \text{máx}\big\{gr\big((p_0 \wedge p_3)\big), 2\big\} && \text{Por el caso "}At\text{"} \\
&= \text{máx}\big\{\text{máx}\big\{gr(p_0), gr(p_3)\big\}, 2\big\} && \text{Por el caso "}\odot\text{"} \\
&= \text{máx}\big\{\text{máx}\{0, 3\}, 2\big\} && \text{Por el caso "}At\text{"} \\
&= \text{máx}\{3, 2\} && \text{Por definición de máx} \\
&= 3 && \text{Por definición de máx}
\end{aligned}$$

El siguiente teorema nos asegura que las definiciones recursivas sobre $PROP$ funcionan bien.

Teorema 6 (definición por recursión en subfórmulas)**.** *Sea A un conjunto y supongamos dadas funciones $H_{At} : At \to A$, y $H_\odot : A^2 \to A$. Entonces hay exactamente una función $F : PROP \to A$ tal que*

$$\left\{ \begin{array}{lcl} F(\varphi) & = & H_{At}(\varphi) \quad \textit{para } \varphi \textit{ en } At \\ F((\varphi \odot \psi)) & = & H_\odot\big(F(\varphi), F(\psi)\big) \end{array} \right. \tag{1}$$

Demostración. Demostraremos solamente la unicidad del mapeo F. Esto (como se repetirá en muchísimas ocasiones más adelante) se hará por inducción en subfórmulas. Supongamos que G también satisface con las propiedades (1). Luego,

$\boxed{\varphi \in At}$ Sabemos que $F(\varphi) = H_{At}(\varphi) = G(\varphi)$, por ser φ atómica.

$\boxed{(\varphi \odot \psi)}$ Supongamos (por HI) que $F(\varphi) = G(\varphi)$ y $F(\psi) = G(\psi)$. Luego

$$F((\varphi \odot \psi)) = H_{\odot}(F(\varphi), F(\psi)) = H_{\odot}(G(\varphi), G(\psi)) = G((\varphi \odot \psi)).$$

Con esto probamos cada paso requerido por el principio de inducción en subfórmulas (Teorema 2), así que podemos concluir que F y G coinciden en todo $PROP$. □

Ejemplo 1. Para la función $gr : PROP \to \mathbb{Z}$ considerada más arriba, tenemos:

$$\begin{aligned} H_{At}(\varphi) &= \begin{cases} n & \text{si } \varphi = p_n \\ -1 & \text{si } \varphi = \bot \end{cases} \\ H_{\odot}(m, n) &= \text{máx}\{m, n\} \end{aligned}$$

Ejercicio 1. Comprobar que las funciones del Ejemplo 1 son las que corresponden.

Nota 1. La definición del conjunto $PROP$ es recursiva, pero **no** por recursión en subfórmulas, sino mediante un teorema aún más general que no enunciaremos aquí; baste observar que si en tal definición hubiésemos usado el Teorema 6, habríamos incurrido en una *petitio principi.*[3]

1.3. Semántica

Hasta ahora, nuestras proposiciones son sólo cadenas de símbolos. A continuación les daremos una noción de "significado": cuándo son ciertas y cuándo falsas. Para ello, consideraremos que un "universo posible" donde se efectúan esas preguntas viene dado por una función, que a cada símbolo proposicional le asigna 0 cuando es "falsa" y 1 cuando es "verdadera".

Definición 7. Una ***asignación*** será una función $v : \{p_0, p_1, \dots\} \to \{0, 1\}$.

Una vez determinados los valores de verdad de los símbolos proposicionales, podemos dar significado a todas las proposiciones. Nuestra noción de "significado" viene dado por la siguiente definición.

Definición 8. Una función $[\![\cdot]\!] : PROP \to \{0, 1\}$ es una ***semántica*** o ***valuación*** si:

1. $[\![\bot]\!] = 0$.
2. $[\![(\varphi \wedge \psi)]\!] = \text{mín}\{[\![\varphi]\!], [\![\psi]\!]\}$.
3. $[\![(\varphi \vee \psi)]\!] = \text{máx}\{[\![\varphi]\!], [\![\psi]\!]\}$.
4. $[\![(\varphi \to \psi)]\!] = 0$ si y sólo si $[\![\varphi]\!] = 1$ y $[\![\psi]\!] = 0$.

[3] El problema del huevo y la gallina, para decirlo en criollo.

Ejercicio 2. Aunque la Definición 8 **no es** una definición por recursión como se establece el Teorema 6 (¿por qué?), se pueden identificar las funciones $H_\odot$ (para cada "$\odot$"). Encontrarlas. ¿Qué pasa con H_{At}?

Si nuestros símbolos proposicionales codifican (para dar un ejemplo) ciertas afirmaciones sobre el mundo físico en un momento dado, digamos:

p_0	Llueve.
p_1	Hace frío.
p_2	Son las dos de la tarde
$\dots$	$\dots$
p_{100}	Cayó un meteorito
$\dots$	$\dots$,

cada asignación corresponderá a un posible momento en la historia.

Ejercicio 3. Determinar para **este** preciso momento cuáles serían los valores de $v(p_i)$ para $i = 0, 1, 2$ e $i = 100$.

Teorema 9 (de Extensión)**.** *Para toda asignación f, existe una única función semántica $[\![\cdot]\!]_f$ tal que $[\![\varphi]\!]_f = f(\varphi)$ para toda $\varphi \in At$.*

Demostración. Construiremos la valuación $[\![\cdot]\!]_f$ por recursión en subfórmulas.

$\boxed{\varphi \in At}$ Definimos $[\![\varphi]\!]_f := f(\varphi)$ si φ es un símbolo proposicional y $[\![\bot]\!] := 0$.

$\boxed{(\varphi \wedge \psi)}$ Dados $[\![\varphi]\!]_f$ y $[\![\psi]\!]_f$, definimos $[\![(\varphi \wedge \psi)]\!]_f := \min\{[\![\varphi]\!]_f, [\![\psi]\!]_f\}$.

$\boxed{(\varphi \to \psi)}$ $[\![(\varphi \to \psi)]\!]_f := 0$ si $[\![\varphi]\!]_f = 1$ y $[\![\psi]\!]_f = 0$, y $[\![(\varphi \to \psi)]\!]_f := 1$ en caso contrario.

$\boxed{(\varphi \vee \psi)}$ $[\![(\varphi \vee \psi)]\!]_f := \max\{[\![\varphi]\!]_f, [\![\psi]\!]_f\}$.

Se sigue inmediatamente de esta definición que $[\![\cdot]\!]_f$ cumple con todas las propiedades para ser una valuación; contrastar con el Ejercicio 2. Más aún, para que una función $[\![\cdot]\!]_f$ cumpla con la hipótesis del teorema debe satisfacer el caso base ($\varphi \in At$) de la definición recursiva anterior, y para que sea una valuación está obligada a satisfacer las propiedades impuestas por los otros casos de la recursión. Es decir, una valuación tal siempre va a poder obtenerse por una construcción recursiva. Y como la construcción recursiva da una única respuesta (por el Teorema 6), se concluye que la extensión de f a $PROP$ es única. □

Como consecuencia inmediata de este teorema, obtenemos:

Corolario 10. *Si $[\![\cdot]\!]_1$ y $[\![\cdot]\!]_2$ son funciones semánticas que coinciden en At (es decir, $[\![\varphi]\!]_1 = [\![\varphi]\!]_2$ para toda $\varphi \in At$), entonces $[\![\cdot]\!]_1 = [\![\cdot]\!]_2$.*

Demostración. Se puede probar por inducción en subfórmulas; esta opción queda como ejercicio. Sino, podemos considerar el siguiente argumento: las restricciones de v y v' a At (es decir, v y v' pensadas como funciones de At en $\{0, 1\}$) son iguales; llamémosle w a esa restricción común. Las valuaciones v y v' son extensiones de w a todo $PROP$ y el Teorema de Extensión obliga (por la unicidad) a que sean iguales. □

Antes de proseguir nos será sumamente útil introducir ***abreviaturas*** para ciertas fórmulas que aparecerán una y otra vez, y para las cuales también tenemos concepciones previas de lo que deben significar.

Definición 11. Para proposiciones φ y ψ, la expresión "$(\neg\varphi)$" denotará la proposición $(\varphi \to \bot)$ y "$(\varphi \leftrightarrow \psi)$" denotará $((\varphi \to \psi) \wedge (\psi \to \varphi))$.

Ejercicio 4. Probar que para toda valuación $[\![\cdot]\!]$, $[\![(\neg\varphi)]\!] = 1 - [\![\varphi]\!]$, y que $[\![(\varphi \leftrightarrow \psi)]\!] = 1$ si y sólo si $[\![\varphi]\!] = [\![\psi]\!]$.

Siguiendo la analogía de una valuación con un momento posible de la historia, nuestro próximo concepto caracteriza a las proposiciones que son válidas "siempre" (o en todo universo posible).

Definición 12. φ es una ***tautología*** (escribimos "$\models \varphi$") si y sólo si $[\![\varphi]\!]_v = 1$ para toda asignación v. Sea $\Gamma \subseteq PROP$; diremos que φ es ***consecuencia*** de Γ (escribimos "$\Gamma \models \varphi$") si y sólo si para toda asignación v tal que

> Para toda $\psi \in \Gamma$, $[\![\psi]\!]_v = 1$

se da

> $[\![\varphi]\!]_v = 1$.

Se ve fácilmente que "$\models \varphi$" es lo mismo que "$\emptyset \models \varphi$". Si $\Gamma \subseteq PROP$ y v es tal que para toda $\psi \in \Gamma$, $[\![\psi]\!]_v = 1$, diremos que v ***valida*** Γ, o más informalmente, que es una "***asignación de*** Γ", y escribiremos $[\![\Gamma]\!]_v = 1$.

Ejemplo 2.

1. $\models (\varphi \to \varphi)$.

   Sea v una asignación arbitraria. Como los únicos valores posibles de $[\![\cdot]\!]_v$ son 0 y 1, ver que da 1 es lo mismo que ver que no da 0.

   $$\begin{aligned} [\![\varphi \to \varphi]\!]_v = 0 &\iff [\![\varphi]\!]_v = 0 \text{ y } [\![\varphi]\!]_v = 1 \\ &\iff \text{contradicción} \end{aligned}$$

   Como suponer $[\![\varphi \to \varphi]\!]_v = 0$ es contradictorio, concluimos $[\![\varphi \to \varphi]\!]_v = 1$.

2. $\models ((\neg(\neg\varphi)) \to \varphi)$ (ejercicio, recordar el Ejercicio 4 aquí).

3. $\{\varphi, (\varphi \to \psi)\} \models \psi$. Sea v una asignación tal que $[\![\varphi]\!]_v = [\![(\varphi \to \psi)]\!]_v = 1$. Luego, sabemos que **no** es el caso que $[\![\varphi]\!]_v = 1$ y $[\![\psi]\!]_v = 0$ (por la segunda igualdad); es decir: o bien $[\![\varphi]\!]_v = 0$, ó $[\![\psi]\!]_v = 1$, ó ambos. Como $[\![\varphi]\!]_v = 1$, debe ser $[\![\psi]\!]_v = 1$, que era lo que debíamos probar.

4. $\not\models p_1$

   Sale negando la definición: p_1 **no** es una tautología si y sólo si existe alguna v tal que $[\![p_1]\!]_v = 0$. Basta entonces tomar $v(p_n) := 0$ para todo $n \in \mathbb{N}_0$ y tenemos el resultado, dado que $[\![p_1]\!] = v(p_1) = 0$ por definición de $[\![\cdot]\!]_v$.

Para probar que una proposición φ en particular es una tautología, no hace falta ver que para cada una de las infinitas[4] valuaciones posibles v se da $[\![\varphi]\!]_v = 1$, sino que esto se puede saber revisando una cantidad finita de casos. Para ello, necesitaremos un resultado de índole práctica, que nos asegura una forma *finitista* de aplicar valuaciones varias a una proposición.

[4]De hecho, hay tantas como números reales (!). Probar esto es un ejercicio interesante.

Lema 13 (de Coincidencia)**.** *Si $v(p_i) = v'(p_i)$ para todos los p_i que ocurran en φ, entonces $[\![\varphi]\!]_v = [\![\varphi]\!]_{v'}$.*

Demostración.

$\boxed{\varphi \in At}$ Como la única fórmula atómica que ocurre en φ es φ, tenemos que $[\![\varphi]\!]_v = v(\varphi) = v'(\varphi) = [\![\varphi]\!]_{v'}$, en el caso de que φ sea algún p_n. Para concluir, notar que $[\![\bot]\!]_v = [\![\varphi]\!]_{v'} = 0$ siempre.

$\boxed{(\varphi \wedge \psi)}$ Supongamos que v y v' coinciden en los átomos de $(\varphi \wedge \psi)$; pero estos átomos incluyen los de φ y ψ, y luego por hipótesis inductiva $[\![\varphi]\!]_v = [\![\varphi]\!]_{v'}$ y $[\![\psi]\!]_v = [\![\psi]\!]_{v'}$. Ahora bien,

$$[\![(\varphi \wedge \psi)]\!]_v = \text{mín}\{[\![\varphi]\!]_v, [\![\psi]\!]_v\} = \text{mín}\{[\![\varphi]\!]_{v'}, [\![\psi]\!]_{v'}\} = [\![(\varphi \wedge \psi)]\!]_{v'}.$$

$\boxed{(\varphi \odot \psi)}$ El resto de los casos queda como ejercicio. □

Otra forma de enunciar el Lema de Coincidencia es la siguiente: si dos valuaciones coinciden en los símbolos proposicionales que forman φ, entonces coinciden en φ. Es precisamente este lema el que da utilidad a las ***tablas de verdad***. Veámoslo con un ejemplo.

Ejemplo 3. Queremos ver que $\models ((p_0 \wedge p_2) \to p_2)$. Deberíamos (por lo menos en principio) revisar todas las posibles asignaciones. Cada una de ellas es una asignación de un 0 ó un 1 a cada símbolo proposicional:

	p_0	p_1	p_2	p_3	p_4	$\dots$
v	1	0	1	1	0	$\dots$

Luego, tomemos **todas** las asignaciones y nos fijemos a qué evalúa nuestra proposición, considerando sus extensiones mediante el Teorema de Extensión:

	p_0	p_1	p_2	p_3	$\dots$	$(p_0 \wedge p_2)$	$((p_0 \wedge p_2) \to p_2)$
v_1	1	0	1	1	$\dots$	1	1
v_2	1	1	0	1	$\dots$	0	1
v_3	1	0	1	0	$\dots$	1	1
v_4	0	0	1	1	$\dots$	0	1
v_5	0	1	0	0	$\dots$	0	1
$\vdots$			$\vdots$		$\ddots$		

Hasta ahora, sólo nos dio 1, pero una prueba a ciegas no nos asegura nada. De todos modos, si entendimos la definición de valuación, nos daremos cuenta que en cada caso sólo necesitamos conocer el valor de p_0 y p_2 en la respectiva asignación; por ejemplo, como v_1 y v_3 coinciden en p_0 y p_2, también coinciden en $(p_0 \wedge p_2)$. En fin, nos podemos quedar con la parte de la tabla que contiene a p_0 y a p_2:

	p_0	p_2	$(p_0 \wedge p_2)$	$((p_0 \wedge p_2) \to p_2)$
v_1	1	1	1	1
v_2	1	0	0	1
v_3	1	1	1	1
v_4	0	1	0	1
v_5	0	0	0	1
$\vdots$		$\vdots$		

y por último, eliminamos la parte repetida:

	p_0	p_2	$(p_0 \wedge p_2)$	$((p_0 \wedge p_2) \to p_2)$
v_1	1	1	1	1
v_2	1	0	0	1
v_4	0	1	0	1
v_5	0	0	0	1

con lo cual nos queda una tabla de verdad como Dios[5] manda.

Definición 14. La ***sustitución*** del símbolo proposicional p **por** la proposición ψ **en** φ, denotada por $\varphi[\psi/p]$ (léase: "φ, con ψ en lugar de p") se define de la siguiente manera:

$\boxed{\varphi \in At}$ Si $\varphi = p$ entonces $\varphi[\psi/p] := \psi$. Caso contrario, $\varphi[\psi/p] := \varphi$

$\boxed{(\varphi \odot \chi)}$ $(\varphi \odot \chi)[\psi/p] := (\varphi[\psi/p] \odot \chi[\psi/p])$.

Ejercicio 5. Probar que $(\neg\varphi)[\psi/p] = (\neg\varphi[\psi/p])$ y $(\varphi \leftrightarrow \chi)[\psi/p] = (\varphi[\psi/p] \leftrightarrow \chi[\psi/p])$.

Los próximos resultados mostrarán que la sustitución funciona bien con la relación de consecuencia.

Lema 15. *Para cada asignación v, su valuación asociada $[\![\cdot]\!] := [\![\cdot]\!]_v$ cumple con que $[\![\varphi_1]\!] = [\![\varphi_2]\!]$ implica $[\![\psi[\varphi_1/p]]\!] = [\![\psi[\varphi_2/p]]\!]$,*

Demostración. Fijemos una $[\![\cdot]\!]$; probaremos el resultado por inducción en ψ suponiendo, en cada paso de la prueba inductiva, válido el antecedente para probar el consecuente.

$\boxed{\psi \in At}$ Aplicamos directamente la definición: Si $\psi = p$, entonces $[\![\psi[\varphi_1/p]]\!] = [\![\varphi_1]\!]$ y $[\![\psi[\varphi_2/p]]\!] = [\![\varphi_2]\!]$. Como los segundos miembros son iguales por hipótesis, son iguales también los primeros miembros. Si $\psi \neq p$, entonces $[\![\psi[\varphi_1/p]]\!] = [\![\psi]\!] = [\![\psi[\varphi_2/p]]\!]$, con lo que queda probado este caso.

$\boxed{(\varphi \odot \chi)}$ Supongamos $[\![\varphi_1]\!] = [\![\varphi_2]\!]$. Por hipótesis inductiva obtenemos $[\![\varphi[\varphi_1/p]]\!] = [\![\varphi[\varphi_2/p]]\!]$ y $[\![\psi[\varphi_1/p]]\!] = [\![\psi[\varphi_2/p]]\!]$. Ahora bien,

$$\begin{aligned} [\![(\varphi \odot \chi)[\varphi_1/p]]\!] &= [\![(\varphi[\varphi_1/p] \odot \chi[\varphi_1/p])]\!] && \text{por caso "}\odot\text{" de sustitución} \\ &= H_\odot([\![\varphi[\varphi_1/p]]\!], [\![\chi[\varphi_1/p]]\!]) && \text{por definición de valuación.} \end{aligned}$$

Aquí "$H_\odot$" son las del Ejercicio 2.

$$\begin{aligned} &= H_\odot([\![\varphi[\varphi_2/p]]\!], [\![\chi[\varphi_2/p]]\!]) && \text{por hipótesis inductiva} \\ &= [\![(\varphi[\varphi_2/p] \odot \chi[\varphi_2/p])]\!] && \text{por definición de valuación} \\ &= [\![(\varphi \odot \chi)[\varphi_2/p]]\!] && \text{por caso "}\odot\text{" de sustitución.} \end{aligned}$$

Queda entonces probado $[\![(\varphi \odot \chi)[\varphi_1/p]]\!] = [\![(\varphi \odot \chi)[\varphi_2/p]]\!]$. □

Teorema 16 (Regla de Leibnitz)**.** *Si $\models \varphi_1 \leftrightarrow \varphi_2$ entonces $\models \psi[\varphi_1/p] \leftrightarrow \psi[\varphi_2/p]$.*

Demostración. Supongamos que $\models \varphi_1 \leftrightarrow \varphi_2$. Luego, dada una asignación arbitraria y su valuación asociada $[\![\cdot]\!]$, tenemos $[\![\varphi_1 \leftrightarrow \varphi_2]\!] = 1$ y entonces $[\![\varphi_1]\!] = [\![\varphi_2]\!]$ por el Ejercicio 4. Por el Lema 15, obtenemos $[\![\psi[\varphi_1/p]]\!] = [\![\psi[\varphi_2/p]]\!]$ y luego, nuevamente por el Ejercicio 4, tenemos $[\![\psi[\varphi_1/p] \leftrightarrow \psi[\varphi_2/p]]\!] = 1$. Como $[\![\cdot]\!]$ era arbitraria, sabemos que *para toda v* se da $[\![\psi[\varphi_1/p] \leftrightarrow \psi[\varphi_2/p]]\!] = 1$, pero esto no es otra cosa que $\models \psi[\varphi_1/p] \leftrightarrow \psi[\varphi_2/p]$. □

[5] Reemplace por la deidad que más le agrade.

1.4. Completitud Funcional

En vista de los resultados anteriores, se deduce que la semántica de un conectivo (i.e., cómo se comporta con respecto a una asignación) viene dada por su tabla de verdad. Ahora, una tabla de verdad no es otra cosa que una función de n variables (para el caso de los conectivos binarios, $n = 2$; unarios, $n = 1$ y etcétera) que toma valores 0 ó 1 y devuelve un valor 0 ó 1. Retomando el Ejemplo 3, la tabla de verdad de $(p_0 \wedge p_2)$ es

	p_0	p_2	$p_0 \wedge p_2$
v_1	1	1	1
v_2	1	0	0
v_4	0	1	0
v_5	0	0	0

Es decir, es exactamente la función $H_\wedge$ que va de $\{0,1\}^2$ a $\{0,1\}$. En general, para cada función de $\{0,1\}^2$ a $\{0,1\}$ podemos definir un nuevo conectivo correspondiente. Si por ejemplo tomamos la función $H_|$ definida por esta tabla:

x	y	$H_\mid(x,y)$
1	1	0
1	0	1
0	1	1
0	0	1

obtenemos un nuevo conectivo binario, la "rayita" (en inglés, *stroke*) de Scheffer, "|". Si v es una asignación, obtenemos $[\![(\varphi \mid \psi)]\!]_v = H_\mid([\![\varphi]\!]_v, [\![\psi]\!]_v)$, o en otros términos,

$$[\![(\varphi \mid \psi)]\!]_v := 1 - \text{mín}\{[\![\varphi]\!]_v, [\![\psi]\!]_v\}. \tag{2}$$

Ejercicio 6. Comprobar esto último.

Se pueden definir conectivos ternarios, cuaternarios, ... En general, (la semántica de) un conectivo n-ario "#" vendrá dado por una función $H_\# : \{0,1\}^n \to \{0,1\}$ y tendremos

$$[\![\#(\varphi_1, \ldots, \varphi_n)]\!] := H_\#([\![\varphi_1]\!], \ldots, [\![\varphi_n]\!]).$$

Un ejemplo de conectivo ternario sería el siguiente:

p_0	p_1	p_2	$\#(p_0,p_1,p_2)$
1	1	0	0
1	0	0	1
0	1	0	0
0	0	0	0
1	1	1	0
1	0	1	0
0	1	1	0
0	0	1	0

Volvamos por un momento a la ecuación (2). Mediante un simple cálculo, se tiene que

$$\begin{aligned}
[\![(\varphi \mid \psi)]\!]_v &= 1 - \text{mín}\{[\![\varphi]\!]_v, [\![\psi]\!]_v\} && \text{definición de } | \\
&= 1 - [\![(\varphi \wedge \psi)]\!]_v && \text{definición de valuación} \\
&= [\![(\neg(\varphi \wedge \psi))]\!]_v && \text{Ejercicio 4}
\end{aligned}$$

Luego, a nivel semántico, decir "$\varphi \mid \psi$" es exactamente lo mismo que decir "$(\neg(\varphi \wedge \psi))$" (una es "cierta" si y sólo si la otra lo es). Diremos entonces que $\mid$ *es expresable en términos de* $\neg$ *y* $\vee$.

Ejercicio 7. Comprobar que $[\![\#(p_0, p_1, p_2)]\!]_v = [\![(p_0 \wedge (\neg(p_1 \vee p_2)))]\!]_v$ para toda v. Es decir, $\#$ es expresable en términos de $\wedge$, $\neg$ y $\vee$.

Estos ejemplos son testigos de un resultado muy general, que es consecuencia de la teoría de álgebras de Boole.

Teorema 17. *Todo conectivo se puede expresar en términos de* $\wedge$, $\vee$ *y* $\neg$.

Definición 18. Un conjunto $\mathcal{C}$ de conectivos es ***funcionalmente completo*** si y sólo si todo conectivo es expresable en términos de elementos de $\mathcal{C}$.

Luego, el Teorema 17 dice que $\{\wedge, \vee, \neg\}$ es funcionalmente completo.

1.5. Ejercicios

1. Dé series de formación de las siguientes proposiciones:

   a) $\big((((p_1 \to p_2) \to p_1) \to p_2) \to p_1\big)$.

   b) $(p_3 \vee (p_1 \leftrightarrow p_2))$ (ojo con la abreviatura).

   c) $\big((p_4 \wedge (\neg p_2)) \to p_1\big)$.

2. Demuestre que si φ, ψ y ξ son proposiciones, también lo es $((\varphi \wedge \psi) \to \xi)$.

3. Demuestre que toda $\varphi \in PROP$ tiene tantos "(" como ")". Además, vea que la cantidad de paréntesis ("abre" y "cierra", todos juntos) es igual a doble de la cantidad de conectivos distintos de $\bot$ que ocurren.

4. Demuestre que $(p_0) \notin PROP$.

5. Defina recursivamente una función $\Sigma(\varphi)$ que devuelva una serie de formación de φ para cada $\varphi \in PROP$.

6. Ídem al anterior para "complejidad de φ", donde la complejidad de una proposición viene dada por la cantidad de ocurrencias de conectivos en la proposición.

7. Ídem al anterior para "longitud de φ", considerando a φ como una sucesión de símbolos (incluyendo paréntesis).

8. Se define la noción de ***subfórmula*** de la siguiente manera (recursiva):

   $\boxed{\varphi \in At}$ ψ es subfórmula de φ si $\psi = \varphi$.

   $\boxed{(\varphi \odot \chi)}$ ψ es subfórmula de $(\varphi \odot \chi)$ si ψ es igual a $(\varphi \odot \chi)$ ó si es subfórmula de φ ó de χ.

   a) Demostrar que si ψ es subfórmula de φ, entonces ψ es un término de la sucesión $\Sigma(\varphi)$ del Ejercicio 5. En general, toda subfórmula aparecerá en cada serie de formación de φ.

b) (*) Encontrar el conjunto A y las funciones $H_\bullet$ del Teorema 6 para esta definición[6] (Ayuda mezquina: $A \neq PROP$).

9. Pruebe el Corolario 10 por inducción en subfórmulas.

10. Determine $\varphi[((\neg p_0) \to p_3)/p_0]$ para $\varphi = ((p_1 \wedge p_0) \to (p_0 \to p_3))$ y $\varphi = ((p_3 \leftrightarrow p_0) \vee (p_2 \to (\neg p_0)))$.

11. Suponga $\varphi_1, \ldots, \varphi_n = \varphi$ es serie de formación de φ.

    a) Probar que $\varphi_1[\bot/p_0], \ldots, \varphi_n[\bot/p_0]$ es serie de formación de $\varphi[\bot/p_0]$.

    b) ¿Vale en general que $\varphi_1[\psi/p_0], \ldots, \varphi_n[\psi/p_0]$ es serie de formación de $\varphi[\psi/p_0]$ para todas φ, ψ?

12. Decida si las siguientes funciones de $PROP$ a $\{0, 1\}$ son valuaciones:

    a) $[\![\varphi]\!] := 1$ para toda $\varphi \in PROP$.

    b) $[\![\varphi]\!] := 0$ para toda $\varphi \in PROP$.

    c) Dada una asignación v, defino $V : PROP \to \{0, 1\}$ como $V(\varphi) := [\![\varphi[\bot/p_0]]\!]_v$. Además de decidir, describa a V con sus palabras. Pregunte a su compañero/a si entiende su definición ◡̈.

13. Escribamos $\varphi \models \psi$ cuando $\{\varphi\} \models \psi$. Pruebe que la relación así definida ("ser consecuencia de") es transitiva y reflexiva en $PROP$.

14. Pruebe que $\models \varphi \to \psi$ si y sólo si $\varphi \models \psi$.

15. Pruebe que $p_0 \not\models (p_0 \wedge p_1)$ y que $\{p_0, (p_0 \to (p_1 \vee p_2)\} \not\models p_2$.

16. Sea $\varphi \in PROP$. Demostrar que si $\models p \to \varphi$ para todo $p \in At$, entonces $\models \varphi$.

17. Diremos que φ es *(semánticamente) equivalente a* ψ si $\models \varphi \leftrightarrow \psi$. Encontrar proposiciones equivalentes a las siguientes que sólo contengan los conectivos $\to$ y $\neg$:

    a) $(p_0 \wedge p_1)$

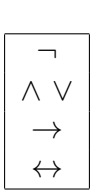
    b) $((p_0 \vee p_2) \leftrightarrow p_3)$

18. Demostrar que $\{\to, \bot\}$ y $\{|\}$ son funcionalmente completos.

2. Deducción Natural

Si uno piensa a la lógica como una codificación del razonamiento, entonces debería analizarse de cerca el proceso de realizar inferencias, es decir, obtener conclusiones a partir de premisas de manera *correcta*. Estudiaremos entonces un sistema de deducción (formalizado por G. Gentzen en 1934) que modela el modo de razonamiento (en su faceta proposicional) que se utiliza, por ejemplo, en matemática. Para hacer más simple la notación, utilizaremos una tabla de precedencia para evitar poner paréntesis:

[6] Los ejercicios con una "(*)" son un poco más duros. Por ahí conviene dejarlos para el final.



En particular, eliminaremos los paréntesis exteriores de cada fórmula. Así, en vez de

$$\big((p_7 \to \bot) \leftrightarrow (p_4 \wedge (\neg p_2)) \to p_1\big)$$

podremos escribir

$$p_7 \to \bot \ \leftrightarrow \ p_4 \wedge \neg p_2 \to p_1.$$

2.1. Reglas de Inferencia

En pocas palabras, un sistema deductivo es mecanismo formal para pasar de algunas proposiciones iniciales ("premisas") a otra ("conclusión") mediante un conjunto de reglas sintácticas. Nuestro objetivo será describir reglas que se adecuen a nuestra noción intuitiva de "razonamiento correcto". En particular, tendremos reglas *de introducción* que a partir de las premisas (por ejemplo, φ y ψ) concluyen una fórmula con un conectivo más (por ejemplo, $\varphi \wedge \psi$); y también reglas *de eliminación* en las que la conclusión se borró alguna conectiva que aparecía en las premisas (por ejemplo pasar de $\varphi \wedge \psi$ a φ). Las dos reglas que ejemplificamos se llaman, respectivamente, *introducción de* $\wedge$ y *eliminación de* $\wedge$ y se pueden representar mediante los dos primeros diagramas de los que siguen:

$$\frac{\varphi \quad \psi}{\varphi \wedge \psi}\wedge I \qquad \frac{\varphi \wedge \psi}{\varphi}\wedge E \qquad \frac{\varphi \wedge \psi}{\psi}\wedge E$$

$$\frac{\varphi}{\varphi \vee \psi}\vee I \qquad \frac{\psi}{\varphi \vee \psi}\vee I \qquad \frac{\varphi \vee \psi \qquad \begin{array}{c}[\varphi]\\ \vdots\\ \chi\end{array} \qquad \begin{array}{c}[\psi]\\ \vdots\\ \chi\end{array}}{\chi}\vee E.$$

$$\frac{\varphi \quad \varphi \to \psi}{\psi}\to E \qquad \frac{\begin{array}{c}[\varphi]\\ \vdots\\ \psi\end{array}}{\varphi \to \psi}\to I \qquad \frac{\bot}{\varphi}\bot \qquad \frac{\begin{array}{c}[\neg\varphi]\\ \vdots\\ \bot\end{array}}{\varphi}\,RAA$$

Antes de dar la definición formal de derivación, conviene hacer algún ejemplo para conocer el funcionamiento de la deducción natural de manera intuitiva:

Ejemplo 4. Tratemos de hallar una "codificación" simbólica de una demostración de $\varphi \wedge \psi \to \psi \wedge \varphi$. Para ello, nos fijamos que hay una sola regla que permite deducir una implicación de manera explícita, y es la regla $(\to I)$ ("introducción de $\to$"). Reemplazando,

$$\frac{\begin{array}{c}[\varphi \wedge \psi]_1\\ \vdots\\ \psi \wedge \varphi\end{array}}{\varphi \wedge \psi \to \psi \wedge \varphi}\to I_1$$

Esto significa que "Si tomando como *hipótesis* $\varphi \wedge \psi$ puedo deducir $\psi \wedge \varphi$, entonces tengo una prueba de $\varphi \wedge \psi \to \psi \wedge \varphi$". Los corchetes subindizados con "1" dicen que en el paso "$(\to I)$" de la derivación *cancelamos* la hipótesis $\varphi \wedge \psi$.

Ahora tenemos que reemplazar los puntos suspensivos por una deducción que lleve de $\varphi \wedge \psi$ a $\psi \wedge \varphi$. Haciendo la misma observación de hace un rato, la única regla que tiene como conclusión una conjunción es la $(\wedge I)$ ("introducción de $\wedge$"). Seguimos completando:

$$\dfrac{\dfrac{\begin{array}{c}[\varphi \wedge \psi]_1 \\ \vdots \\ \psi \quad \varphi\end{array}}{\psi \wedge \varphi} \wedge I}{\varphi \wedge \psi \to \psi \wedge \varphi} \to I_1$$

Necesitamos llegar desde nuestra hipótesis $\varphi \wedge \psi$ a cada uno de los términos de la conjunción. Pero para ello, las reglas $(\wedge E)$ ("eliminación de $\wedge$") nos vienen al pelo. Nos hacen falta dos copias de $\varphi \wedge \psi$ a tal fin (una para deducir φ y otra para ψ). No hay problema, porque como sucede en una demostración matemática, cada vez que se hace una suposición, se la puede utilizar tantas veces como uno quiera:

$$\dfrac{\dfrac{\dfrac{[\varphi \wedge \psi]_1}{\psi} \wedge E \quad \dfrac{[\varphi \wedge \psi]_1}{\varphi} \wedge E}{\psi \wedge \varphi} \wedge I}{\varphi \wedge \psi \to \psi \wedge \varphi} \to I_1 \qquad \begin{array}{ccccc} \varphi \wedge \psi & & & & \varphi \wedge \psi \\ \bullet & & & & \bullet \\ \psi & & & & \varphi \\ \bullet & & & & \bullet \\ & & \psi \wedge \varphi & & \\ & & \bullet & & \\ & & \varphi \wedge \psi \to \psi \wedge \varphi & & \\ & & \bullet & & \end{array} \tag{3}$$

Este árbol (notar su estructura a la derecha) es una "derivación" de $\varphi \wedge \psi \to \psi \wedge \varphi$.

Definiremos a continuación la noción de *derivación*. Como arriba, una derivación será un árbol (i.e., un grafo acíclico) con una raíz distinguida, tal que cada nodo (vértice) está decorado con una proposición[7]. Tomamos la convención de dibujar dichos árboles con la raíz abajo, como a continuación:

$$\begin{array}{c} \vdots \\ D \\ \vdots \\ \varphi \end{array}$$

Las proposiciones que decoran las hojas del árbol son las *hipótesis*, y la proposición que decora la raíz de D será la ***conclusión*** de la derivación y será denotada por $Concl(D)$; por ejemplo, con

$$\begin{array}{c} \psi \quad \psi' \\ \vdots \\ D \\ \vdots \\ \varphi \end{array}$$

indicaremos una derivación entre cuyas hipótesis se encuentran *eventualmente* ψ y ψ'. Si no hace falta poner nombres (por ejemplo, "D", más arriba), simplemente escribiremos

$$\begin{array}{c} \vdots \\ \varphi \end{array} \qquad\qquad \begin{array}{c} \psi \quad \psi' \\ \vdots \\ \varphi \end{array}$$

Como se habrá notado más arriba, las hipótesis a secas no son de interés, sino el subconjunto las hipótesis *no canceladas*, que definiremos formalmente en un momento (Definición 22).

Definición 19. El conjunto $\mathcal{D}$ de las ***derivaciones*** será el menor conjunto de árboles nodos decorados con proposiciones y con una raíz distinguida tal que:

[7]Notar, de todos modos, el Ejercicio 16 de la Sección 2.4 y sus comentarios.

$\boxed{PROP}$ Todo árbol de un único nodo decorado con $\varphi \in PROP$ pertenece a $\mathcal{D}$, que denotaremos simplemente por "φ".

$\boxed{\wedge I}$ Si $\begin{array}{c}\vdots\\ \varphi\end{array} D$ y $\begin{array}{c}\vdots\\ \varphi'\end{array} D'$ están en $\mathcal{D}$, entonces $D'' := \dfrac{\begin{array}{cc}\vdots D & \vdots D\\ \varphi & \varphi\end{array}}{\varphi \wedge \varphi'} \wedge I$ pertenece a $\mathcal{D}$.

$\boxed{\wedge E}$ Si $\begin{array}{c}\vdots\\ \varphi \wedge \varphi'\end{array} D \in \mathcal{D}$ entonces $D_1 := \dfrac{\begin{array}{c}\vdots D\\ \varphi \wedge \varphi'\end{array}}{\varphi} \wedge E$ y $D_2 := \dfrac{\begin{array}{c}\vdots D\\ \varphi \wedge \varphi'\end{array}}{\varphi'} \wedge E$ están en $\mathcal{D}$.

$\boxed{\to I}$ Dada $\begin{array}{c}\varphi\\ \vdots\\ \psi\end{array} D$ en $\mathcal{D}$, tenemos que $D' := \dfrac{\begin{array}{c}[\varphi]\\ \vdots D\\ \psi\end{array}}{\varphi \to \psi} \to I$ está en $\mathcal{D}$.

$\boxed{\to E}$ Si $\begin{array}{c}\vdots\\ \varphi\end{array} D$ y $\begin{array}{c}\vdots\\ \varphi \to \psi\end{array} D'$ están en $\mathcal{D}$, entonces $D'' := \dfrac{\begin{array}{cc}\vdots D & \vdots D'\\ \varphi & \varphi \to \psi\end{array}}{\psi} \to E$ está en $\mathcal{D}$.

$\boxed{\bot}$ Si tenemos $\begin{array}{c}\vdots\\ \bot\end{array} D$, entonces para toda $\varphi \in PROP$, $D' := \dfrac{\begin{array}{c}\vdots D\\ \bot\end{array}}{\varphi} \bot$ está en $\mathcal{D}$.

$\boxed{RAA}$ Dada una derivación $\begin{array}{c}\neg\varphi\\ \vdots\\ \bot\end{array} D$, $D' := \dfrac{\begin{array}{c}[\neg\varphi]\\ \vdots D\\ \bot\end{array}}{\varphi} RAA$ está en $\mathcal{D}$.

$\boxed{\vee I}$ Si $\begin{array}{c}\vdots\\ \varphi\end{array} D \in \mathcal{D}$, entonces $\dfrac{\begin{array}{c}\vdots D\\ \varphi\end{array}}{\varphi \vee \psi} \vee I \in \mathcal{D}$. Lo mismo con $\begin{array}{c}\vdots\\ \psi\end{array} D'$ y $\dfrac{\begin{array}{c}\vdots D'\\ \psi\end{array}}{\varphi \vee \psi} \vee I$.

$\boxed{\vee E}$ Dadas $\begin{array}{c}\vdots\\ \varphi \vee \psi\end{array} D$, $\begin{array}{c}\varphi\\ \vdots\\ \chi\end{array} D'$ y $\begin{array}{c}\psi\\ \vdots\\ \chi\end{array} D''$ en $\mathcal{D}$, entonces $\dfrac{\begin{array}{ccc}\vdots D & [\varphi]\ \vdots D' & [\psi]\ \vdots D''\\ \varphi \vee \psi & \chi & \chi\end{array}}{\chi} \vee E \in \mathcal{D}$.

En las reglas $(\to I)$ y (RAA) no es necesario que las hipótesis φ y $\neg\varphi$ (respectivamente) aparezcan en la derivación D. Similarmente, no es necesario que φ ni ψ ocurran explícitamente en la subderivaciones D' y D'' que figuran en $(\vee E)$.

Como $\mathcal{D}$ tiene una definición recursiva, podemos establecer análogos a los Teoremas 2 y 6.

Teorema 20 (inducción en derivaciones)**.** *Sea A un predicado definido en $\mathcal{D}$. Luego $A(D)$ es verdadero para toda $D \in \mathcal{D}$ si y sólo si:*

$\boxed{PROP}$ *Si* $\varphi \in PROP$, $A(\varphi)$ *vale.*

$\boxed{\wedge I}$ *Si se dan* $A\begin{pmatrix}\vdots\\ \varphi\end{pmatrix}$ *y* $A\begin{pmatrix}\vdots\\ \varphi'\end{pmatrix}$, *entonces se da* $A\left(\dfrac{\begin{matrix}\vdots & \vdots\\ \varphi & \varphi'\end{matrix}}{\varphi \wedge \varphi'}\,\wedge I\right)$.

$\boxed{\wedge E}$ *Si se da* $A\begin{pmatrix}\vdots\\ \varphi \wedge \varphi'\end{pmatrix}$, *entonces se dan* $A\left(\dfrac{\begin{matrix}\vdots\\ \varphi \wedge \varphi'\end{matrix}}{\varphi}\,\wedge E\right)$, $A\left(\dfrac{\begin{matrix}\vdots\\ \varphi \wedge \varphi'\end{matrix}}{\varphi'}\,\wedge E\right)$.

$\boxed{\to I}$ $A\begin{pmatrix}\varphi\\ \vdots\\ \psi\end{pmatrix}$ *implica* $A\left(\dfrac{\begin{matrix}[\varphi]\\ \vdots\\ \psi\end{matrix}}{\varphi \to \psi}\,\to I\right)$.

$\boxed{\to E}$ *Si se dan* $A\begin{pmatrix}\vdots\\ \varphi\end{pmatrix}$ *y* $A\begin{pmatrix}\vdots\\ \varphi \to \psi\end{pmatrix}$, *entonces se da* $A\left(\dfrac{\begin{matrix}\vdots & \vdots\\ \varphi & \varphi \to \psi\end{matrix}}{\psi}\,\to E\right)$.

$\boxed{\bot}$ *Si tenemos* $A\begin{pmatrix}\vdots\\ \bot\end{pmatrix}$, *entonces para toda* $\varphi \in PROP$, *se da* $A\left(\dfrac{\begin{matrix}\vdots\\ \bot\end{matrix}}{\varphi}\,\bot\right)$.

$\boxed{RAA}$ *Si se da* $A\begin{pmatrix}\neg\varphi\\ \vdots\\ \bot\end{pmatrix}$ *entonces se da* $A\left(\dfrac{\begin{matrix}[\neg\varphi]\\ \vdots\\ \bot\end{matrix}}{\varphi}\,RAA\right)$.

$\boxed{\vee I}$ *Si se da* $A\begin{pmatrix}\vdots\\ \varphi\end{pmatrix}$, *entonces se da* $A\left(\dfrac{\begin{matrix}\vdots\\ \varphi\end{matrix}}{\varphi \vee \psi}\,\vee I\right)$. *Si se da* $A\begin{pmatrix}\vdots\\ \psi\end{pmatrix}$, *entonces se da* $A\left(\dfrac{\begin{matrix}\vdots\\ \psi\end{matrix}}{\varphi \vee \psi}\,\vee I\right)$.

$\boxed{\vee E}$ *Si se dan* $A\begin{pmatrix}\vdots\\ \varphi \vee \psi\end{pmatrix}$, $A\begin{pmatrix}\varphi\\ \vdots\\ \chi\end{pmatrix}$ *y* $A\begin{pmatrix}\psi\\ \vdots\\ \chi\end{pmatrix}$, *entonces se da*

$$A\left(\frac{\begin{matrix}\vdots & [\varphi] & [\psi]\\ \varphi \vee \psi & \chi & \chi\end{matrix}}{\chi}\,\vee E\right).$$

Teorema 21 (recursión en derivaciones)**.** *Sea A un conjunto y sean dadas funciones H_P de $PROP$ en A, $H_{\wedge E1}$, $H_{\wedge E2}$, $H_\bot$, H_{RAA}, $H_{\to I}$, $H_{\vee I1}$ y $H_{\vee I2}$ de A en A, $H_{\wedge I}, H_{\to E}$ de*

$A \times A$ en A y $H_{\vee E} : A^3 \to A$. Luego existe una única función $F : \mathcal{D} \to A$ tal que se cumplen recursiones análogas a las del Teorema 6 según la Definición 19. Es decir, F satisface,

$$
\begin{array}{ll}
F(\varphi) = H_P(\varphi) & \textit{si } \varphi \in PROP \\
F(D) = H_{\wedge I}(F(D_1), F(D_2)) & \textit{si } D \textit{ se obtiene de } D_1 \textit{ y } D_2 \textit{ mediante una aplicación de } (\wedge I) \\
F(D) = H_{\wedge E1}(F(D')) & \textit{si } D \textit{ se obtiene de } D' \textit{ mediante una aplicación del primer caso de } (\wedge E) \\
F(D) = H_{\wedge E2}(F(D')) & \textit{si } D \textit{ se obtiene de } D' \textit{ mediante una aplicación del segundo caso de } (\wedge E) \\
\quad \ldots & \quad \ldots
\end{array}
$$

y así sucesivamente.

Aplicaremos inmediatamente el Teorema de Recursión en Derivaciones en la próxima definición.

Definición 22. El conjunto $Hip(D)$ de las ***hipótesis no canceladas*** de D está dado por la siguiente recursión:

$\boxed{PROP}$ Si $\varphi \in PROP$, $Hip(\varphi) := \{\varphi\}$.

$\boxed{\wedge I}$

$$Hip\left(\begin{array}{c} \vdots D \quad \vdots D' \\ \varphi \qquad \varphi' \\ \hline \varphi \wedge \varphi' \end{array} \wedge I\right) := Hip\begin{pmatrix} \vdots D \\ \varphi \end{pmatrix} \cup Hip\begin{pmatrix} \vdots D' \\ \varphi' \end{pmatrix}.$$

$\boxed{\wedge E}$

$$Hip\left(\begin{array}{c} \vdots \\ \varphi \wedge \varphi' \\ \hline \varphi \end{array} \wedge E\right) = Hip\left(\begin{array}{c} \vdots \\ \varphi \wedge \varphi' \\ \hline \varphi' \end{array} \wedge E\right) := Hip\begin{pmatrix} \vdots \\ \varphi \wedge \varphi' \end{pmatrix}.$$

$\boxed{\to I}$

$$Hip\left(\begin{array}{c} [\varphi] \\ \vdots \\ \psi \\ \hline \varphi \to \psi \end{array} \to I\right) := Hip\begin{pmatrix} \varphi \\ \vdots \\ \psi \end{pmatrix} \smallsetminus \{\varphi\}.$$

$\boxed{\to E}$

$$Hip\left(\begin{array}{c} \vdots \quad \vdots \\ \varphi \quad \varphi \to \psi \\ \hline \psi \end{array} \to E\right) := Hip\begin{pmatrix} \vdots \\ \varphi \end{pmatrix} \cup Hip\begin{pmatrix} \vdots \\ \varphi \to \psi \end{pmatrix}.$$

$\boxed{\bot}$ $\qquad\qquad$ $\boxed{RAA}$

$$Hip\left(\begin{array}{c}\vdots\\ \bot\\ \hline \varphi\end{array}\bot\right) := Hip\begin{pmatrix}\vdots\\ \bot\end{pmatrix} \qquad Hip\left(\begin{array}{c}[\neg\varphi]\\ \vdots\\ \bot\\ \hline \varphi\end{array} RAA\right) := Hip\begin{pmatrix}\neg\varphi\\ \vdots\\ \bot\end{pmatrix}\smallsetminus\{\neg\varphi\}.$$

$\boxed{\vee I}$

$$Hip\left(\begin{array}{c}\vdots\\ \varphi\\ \hline \varphi\vee\varphi'\end{array}\vee I\right) = Hip\left(\begin{array}{c}\vdots\\ \varphi\\ \hline \varphi'\vee\varphi\end{array}\vee I\right) := Hip\begin{pmatrix}\vdots\\ \varphi\end{pmatrix}.$$

$\boxed{\vee E}$

$$Hip\left(\begin{array}{ccc} & [\varphi] & [\psi]\\ \vdots D_1 & \vdots D_2 & \vdots D_3\\ \varphi\vee\psi & \chi & \chi\\ \hline & \chi & \end{array}\vee E\right) := Hip(D_1)\cup\big(Hip(D_2)\smallsetminus\{\varphi\}\big)\cup\big(Hip(D_3)\smallsetminus\{\psi\}\big).$$

Notemos que en la aplicación de $(\vee E)$, la hipótesis φ se cancela en la segunda subderivación (es decir, D_2), pero **no** en las otras (ni en D_1 ni en D_3). Lo mismo ocurre con ψ y D_3.

Ejemplo 5. Veamos que la derivación que habíamos esbozado de $\varphi\wedge\psi\to\psi\wedge\varphi$ está en $\mathcal{D}$.

1. $\varphi\wedge\psi$ es una derivación por la regla $(PROP)$. Tenemos $Concl(\varphi\wedge\psi)=\varphi\wedge\psi$ y $Hip(\varphi\wedge\psi)=\{\varphi\wedge\psi\}$.

2. Como $\varphi\wedge\psi$ es una derivación, puedo aplicar la regla $(\wedge E)$ y obtener las derivaciones $D_1 := \dfrac{\varphi\wedge\psi}{\psi}\wedge E$ y $D_2 := \dfrac{\varphi\wedge\psi}{\varphi}\wedge E$, donde hemos tomado $D=\varphi\wedge\psi$, que cumple con la condición necesaria para aplicar esta regla. La única hipótesis no cancelada de D_1 y D_2 es $\varphi\wedge\psi$.

3. Usando las derivaciones D_1 y D_2 podemos aplicar el caso $(\wedge I)$ (nuestras D, D' son D_1, D_2, respectivamente). Obtenemos una nueva derivación

$$D_3 := \begin{array}{c}\varphi\wedge\psi \qquad \varphi\wedge\psi\\ \vdots D_1 \qquad \vdots D_2\\ \psi \qquad\quad \varphi\\ \hline \psi\wedge\varphi\end{array}\wedge I \quad = \quad \begin{array}{c}\dfrac{\varphi\wedge\psi}{\psi}\wedge E \quad \dfrac{\varphi\wedge\psi}{\varphi}\wedge E\\ \hline \psi\wedge\varphi\end{array}\wedge I.$$

La conclusión de D_3 es $\psi\wedge\varphi$ como se indica, y las hipótesis son las hipótesis de D_1 y D_2 en conjunto. Como sólo está $\varphi\wedge\psi$, queda ella como única hipótesis no cancelada.

4. Este paso es menos trivial, vamos a aplicar $(\to I)$. El caso nos indica que podemos pasar de $\begin{array}{c}\varphi\wedge\psi\\ \vdots\, D_3\\ \psi\wedge\varphi\end{array}$ a

$$D_4 := \dfrac{\begin{array}{c}[\varphi\wedge\psi]_1\\ \vdots\, D_3\\ \psi\wedge\varphi\end{array}}{\varphi\wedge\psi\to\psi\wedge\varphi}\to I_1 \quad = \quad \dfrac{\dfrac{\dfrac{[\varphi\wedge\psi]_1}{\psi}\wedge E \quad \dfrac{[\varphi\wedge\psi]_1}{\varphi}\wedge E}{\psi\wedge\varphi}\wedge I}{\varphi\wedge\psi\to\psi\wedge\psi}\to I_1 \quad ,$$

donde hemos cancelado la hipótesis $\varphi\wedge\psi$. Esta nueva derivación D_4 tiene por hipótesis a las de D_3 menos $\varphi\wedge\psi$, pero como ésta era la única hipótesis de D_3, D_4 tiene todas sus hipótesis canceladas; en símbolos, $Hip(D_4)=\emptyset$, i.e., todas las hipótesis fueron canceladas.

Ejemplo 6. Hallemos una derivación cuyas hipótesis estén todas canceladas y su conclusión sea $(\varphi\to\psi)\to(\neg\psi\to\neg\varphi)$.

Para demostrar este tipo de proposición, basta ver que si suponemos cierto el antecedente, podemos demostrar el consecuente. Entonces haremos una lista numerada de las hipótesis que utilizamos, que luego serán canceladas mediante la introducción de $\to$. De esta manera, este ejercicio y muchos otros similares se resuelven desde abajo para arriba, como vemos a continuación. Partimos de lo que queremos probar:

$$\begin{array}{c}\vdots\\ (\varphi\to\psi)\to(\neg\psi\to\neg\varphi)\end{array}$$

tomamos el antecedente como hipótesis y lo anotamos para uso futuro:

1. $\varphi\to\psi$

$$\dfrac{\begin{array}{c}\vdots\\ (\neg\psi\to\neg\varphi)\end{array}}{(\varphi\to\psi)\to(\neg\psi\to\neg\varphi)}\to I_1$$

Podemos hacer lo mismo ahora con $(\neg\psi\to\neg\varphi)$: sacar el antecedente y usarlo como hipótesis:

1. $\varphi\to\psi$
2. $\neg\psi$

$$\dfrac{\dfrac{\begin{array}{c}\vdots\\ \neg\varphi\end{array}}{(\neg\psi\to\neg\varphi)}\to I_2}{(\varphi\to\psi)\to(\neg\psi\to\neg\varphi)}\to I_1$$

Recordemos que $\neg\varphi$ es una abreviación de $\varphi\to\bot$, luego podemos hacer una vez más el mismo procedimiento:

1. $\varphi\to\psi$
2. $\neg\psi$
3. φ

$$\dfrac{\dfrac{\dfrac{\begin{array}{c}\vdots\\ \bot\end{array}}{\neg\varphi}\to I_3}{(\neg\psi\to\neg\varphi)}\to I_2}{(\varphi\to\psi)\to(\neg\psi\to\neg\varphi)}\to I_1$$

Ahora la tarea se reduce a llegar a $\bot$ usando nuestras tres hipótesis. Ahora comenzamos a trabajar de arriba para abajo: podemos deducir ψ de la primera y la tercera hipótesis:

$$\dfrac{\dfrac{\dfrac{\dfrac{[\varphi]_3 \quad [\varphi \to \psi]_1}{\psi}\to E}{\vdots}}{\bot}}{\neg\varphi}\to I_3}{(\neg\psi \to \neg\varphi)}\to I_2}{(\varphi \to \psi) \to (\neg\psi \to \neg\varphi)}\to I_1$$

1. $\varphi \to \psi$
2. $\neg\psi$
3. φ

Notemos que en las inferencias subindizadas con 1 y 3 se cancelan las respectivas hipótesis, así que por eso las ponemos entre corchetes (respectivamente subindizados).

De esta ψ y la segunda hipótesis obtenemos $\bot$ y queda todo conectado:

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{[\varphi]_3 \quad [\varphi \to \psi]_1}{\psi}\to E \qquad [\neg\psi]_2}{\bot}\to E}{\neg\varphi}\to I_3}{(\neg\psi \to \neg\varphi)}\to I_2}{(\varphi \to \psi) \to (\neg\psi \to \neg\varphi)}\to I_1$$

1. $\varphi \to \psi$
2. $\neg\psi$
3. φ

Ejemplo 7. Hallemos una derivación de $\varphi \to (\psi \to \varphi)$. Como hicimos anteriormente, podemos tratar de derivar $\psi \to \varphi$ tomando como hipótesis a φ. Y a su vez, para derivar $\psi \to \varphi$ suponer ψ y derivar φ. Escribimos la lista de nuestras hipótesis y a lo que queremos llegar:

$$\dfrac{\dfrac{\dfrac{\vdots}{\varphi}}{\psi \to \varphi}\to I_2}{\varphi \to (\psi \to \varphi)}\to I_1 \quad .$$

1. φ
2. ψ

Esperemos un momento: ¡lo que queremos derivar ya es parte de lo que estamos suponiendo! Es decir, podemos dejar la derivación tal como está:

$$\dfrac{\dfrac{\dfrac{[\varphi]_1}{\psi \to \varphi}\to I_2}{\varphi \to (\psi \to \varphi)}\to I_1}{}\ .$$

1. φ
2. ψ

Surge la pregunta: ¿qué hipótesis cancelamos en el primer paso? La respuesta es "ninguna", y la moraleja es: en una introducción de "$\to$", no hace falta que la hipótesis a cancelar aparezca en la derivación.

Ejercicio 8. Pensar por qué funciona así esto, dando una demostración de:

Si $x^2 + y^2 \geq 0$ para todo x e y, entonces 8 es múltiplo de 2.

Ejemplo 8. Haremos una derivación que requiere el uso esencial de (RAA), que es la regla menos *intuitiva*[8]. Veamos que hay una derivación de $(\neg\psi \to \neg\varphi) \to (\varphi \to \psi)$ con todas sus hipótesis canceladas. Como antes, podemos tomar como hipótesis el antecedente y tratar de derivar el consecuente:

$$\dfrac{\begin{array}{c}[\neg\psi \to \neg\varphi]_1 \\ \vdots \\ \varphi \to \psi\end{array}}{(\neg\psi \to \neg\varphi) \to (\varphi \to \psi)} \to I_1$$

y otra vez lo mismo:

$$\dfrac{\begin{array}{cc}[\varphi]_2 & \dfrac{\begin{array}{c}[\neg\psi \to \neg\varphi]_1 \\ \vdots \\ \psi\end{array}}{\varphi \to \psi} \to I_2\end{array}}{(\neg\psi \to \neg\varphi) \to (\varphi \to \psi)} \to I_1$$

Ahora estamos trabados, porque no podemos extraer más hipótesis evidentes. Pero podemos "pedir prestada" una $\neg\psi$ (que pensamos utilizar con $\neg\psi \to \neg\varphi$) y cancelarla más tarde:

$$\dfrac{\begin{array}{cc}[\varphi]_2 & \dfrac{\begin{array}{cc}\neg\psi & [\neg\psi \to \neg\varphi]_1 \\ & \neg\varphi\end{array}}{\begin{array}{c}\vdots \\ \psi\end{array}} \to E \\ & \varphi \to \psi\end{array} \to I_2}{(\neg\psi \to \neg\varphi) \to (\varphi \to \psi)} \to I_1$$

Ahora ya podemos usar la φ que teníamos descolgada.

$$\dfrac{\begin{array}{cc}\dfrac{\begin{array}{cc}[\varphi]_2 & \dfrac{\begin{array}{cc}\neg\psi & [\neg\psi \to \neg\varphi]_1 \\ & \neg\varphi\end{array}}{\begin{array}{c}\vdots \\ \psi\end{array}} \to E\end{array}}{\varphi \to \psi} \to E \\ & \varphi \to \psi\end{array} \to I_2}{(\neg\psi \to \neg\varphi) \to (\varphi \to \psi)} \to I_1$$

Los puntos suspensivos los podemos sacar: notemos que tenemos una $\neg\psi$ sin cancelar y (RAA) nos permite cancelarla si tenemos una derivación con conclusión $\bot$, obteniendo

[8] En la lógica proposicional *intuicionista*, se permite el uso de todas las reglas menos la reducción al absurdo. El Intuicionismo está relacionado con la corriente *constructivista* en matemática (L.E.J. Brouwer, A. Heyting) y tiene importantes consecuencias en ciencias de la computación. Un ejemplo paradigmático de esto es el *Isomorfismo de Curry-Howard*, que muestra que hay una equivalencia entre pruebas intuicionistas y algoritmos.

así ψ (Oh, ¡caramba! ¡Qué coincidencia!).

$$\dfrac{\dfrac{\dfrac{[\neg\psi]_3 \quad [\neg\psi \to \neg\varphi]_1}{\to E} \quad [\varphi]_2}{\to E} \quad \neg\varphi}{\dfrac{\dfrac{\bot}{\psi} RAA_3}{\varphi \to \psi} \to I_2}{(\neg\psi \to \neg\varphi) \to (\varphi \to \psi)} \to I_1$$

Resulta que en general, (RAA) funciona pidiendo las hipótesis que hagan falta, y luego todo cierra *casi* por arte de magia.

Como se vio más arriba, también se puede derivar $(\varphi \to \psi) \to (\neg\psi \to \neg\varphi)$, pero esta última es "constructiva" (no utiliza (RAA)). En conjunto (y aplicando la regla $(\wedge I)$), hemos derivado $(\varphi \to \psi) \leftrightarrow (\neg\psi \to \neg\varphi)$.

Ejemplo 9. La regla $(\vee E)$ es un modelo de *prueba por casos*: si puedo probar χ cuando φ es cierta, y también puedo hacerlo cuando ψ es cierta, entonces puedo probar χ bajo la única suposición de que alguna de las dos es cierta (i.e., que $\varphi \vee \psi$ es cierta).

$$\dfrac{\dfrac{\neg\varphi \vee \psi \qquad \dfrac{\dfrac{[\varphi]_3 \quad [\neg\varphi]_1}{\to E}}{\bot} \bot}{\psi} \qquad [\psi]_2}{\psi} \vee E_{1,2}}{\varphi \to \psi} \to I_3$$

Arriba, en la aplicación de $(\vee E)$, primero se prueba ψ usando como hipótesis al primer término de la disyunción $\neg\varphi$ (cancelándose ésta en esa subderivación) y luego usando hipótesis ψ (que sólo se cancela en la segunda subderivación).

Definición 23. Sean $\Gamma \subseteq PROP$, $\varphi \in PROP$. Decimos que φ ***se deduce de*** Γ (y escribimos "$\Gamma \vdash \varphi$") si y sólo si existe una derivación con conclusión φ tal que todas sus hipótesis no canceladas estén en Γ. Diremos que φ es un ***teorema*** cuando $\emptyset \vdash \varphi$, y abreviaremos por "$\vdash \varphi$".

Dicho más brevemente, $\Gamma \vdash \varphi$ si y sólo si existe $D \in \mathcal{D}$ tal que $Concl(D) = \varphi$ y $Hip(D) \subseteq \Gamma$, y $\vdash \varphi$ si existe $D \in \mathcal{D}$ tal que $Concl(D) = \varphi$ y tiene todas sus hipótesis canceladas.

Ejemplo 10.

1. $\Gamma \vdash \varphi$ siempre que $\varphi \in \Gamma$.

   En este caso debemos considerar a φ como un elemento de $\mathcal{D}$: tiene como conclusión φ (ella misma) y todas sus hipótesis (viz., $\{\varphi\}$) están en Γ trivialmente.

2. $\{\varphi, \varphi \to \psi\} \vdash \psi$.

   Basta ver la regla $(\to E)$ para darse cuenta de esto.

3. $\vdash \varphi \wedge \psi \to \psi \wedge \varphi$.

   La derivación del Ejemplo 5 tiene todas sus hipótesis canceladas, así que nos sirve.

4. El Ejemplo 9 muestra que $\{\neg\varphi \vee \psi\} \vdash \varphi \to \psi$.

5. Por el Ejemplo 6 obtenemos $\vdash (\varphi \to \psi) \to (\neg\psi \to \neg\varphi)$, y con el Ejemplo 8 conseguimos $\vdash (\neg\psi \to \neg\varphi) \to (\varphi \to \psi)$.

6. $\Gamma \vdash \bot$ implica $\Gamma \vdash \varphi$ para toda φ en $PROP$.

   Supongamos que hay una derivación $\begin{array}{c}\vdots\\ \bot\end{array}\ D$ con $Hip(D) \subseteq \Gamma$ y sea $\varphi \in PROP$ arbitraria. Luego, usando $(\bot)$ sabemos que $\dfrac{\begin{array}{c}\vdots\\ \bot\end{array}\ D}{\varphi}\ \bot$ es una derivación, y por construcción tiene las mismas hipótesis (no canceladas) que D (revisar la Definición 19). Luego, sus hipótesis no canceladas están en Γ y entonces $\Gamma \vdash \varphi$. Como φ era arbitraria, esto vale en general.

Ejemplo 11. Ver que $\neg\varphi \vdash \varphi \to \psi$. Procedemos como en el Ejemplo 4: si queremos derivar una implicación, basta conseguir una derivación del consecuente cuyas hipótesis pueden (o no) incluir el antecedente. En nuestro caso, la única hipótesis distinta de tal antecedente (viz., φ) que puede aparecer es $\neg\varphi$.

$$\dfrac{\begin{array}{cc}[\varphi]_1 & \neg\varphi\\ & \vdots\\ & \psi\end{array}}{\varphi \to \psi} \to I_1$$

Pero de φ y $\neg\varphi = \varphi \to \bot$ podemos derivar $\bot$ mediante una eliminación de "$\to$":

$$\dfrac{\begin{array}{cc}[\varphi]_1 & \neg\varphi\\ & \vdots\\ & \psi\end{array}}{\varphi \to \psi} \to I_1 \qquad \dfrac{[\varphi]_1 \quad \neg\varphi}{\bot} \to E$$

Por último, usando la regla $(\bot)$ puedo obtener ψ a partir de $\bot$:

$$\dfrac{\begin{array}{cc}[\varphi]_1 & \neg\varphi\\ & \vdots\\ & \psi\end{array}}{\varphi \to \psi} \to I_1 \qquad \dfrac{\dfrac{[\varphi]_1 \quad \neg\varphi}{\bot} \to E}{\psi}\ \bot$$

Con esto completamos una derivación de $\varphi \to \psi$ con (única) hipótesis no cancelada $\neg\varphi$.

2.2. Teorema de Completitud

Hasta ahora, tenemos dos familias de conceptos aparentemente diferentes entre sí:

Semántica		**Cálculo**
Tautologías (valuar 1)	$\longleftrightarrow$	Teoremas (derivable)
$\models$	$\longleftrightarrow$	$\vdash$
Asignaciones (modelo)	$\longleftrightarrow$	Derivaciones (pruebas formales)

Todos estos conceptos son *equivalentes* en el sentido del siguiente

Teorema 24 (Completitud y Corrección de la Lógica Proposicional)**.** *Para todos* $\Gamma \subseteq PROP$ *y* $\varphi \in PROP$*, se tiene*

$$\Gamma \models \varphi \text{ si y sólo si } \Gamma \vdash \varphi$$

Estrictamente hablando, se llama ***completitud*** a la implicación directa (es decir, el "*sólo si*"), mientras que a la vuelta (el "*si*") se la llama ***corrección***, porque asegura que no se pueden deducir cosas falsas.

Teorema 25 (Corrección)**.** *Si* $\Gamma \vdash \varphi$*, entonces* $\Gamma \models \varphi$*.*

Demostración. Probaremos por inducción en derivaciones que el siguiente enunciado

"Para todo Γ tal que $Hip(D) \subseteq \Gamma$, se da $\Gamma \models Concl(D)$",

vale para toda derivación D.

$\boxed{PROP}$ Supongamos $D = \varphi$. Si $Hip(D) = \{\varphi\} \subseteq \Gamma$, tenemos $\varphi \in \Gamma$, e inmediatamente $\Gamma \models \varphi$.

$\boxed{\wedge I}$ Supongamos que $\begin{matrix}\vdots\\ \varphi\end{matrix} D$, $\begin{matrix}\vdots\\ \varphi'\end{matrix} D'$ satisfacen la hipótesis inductiva, y supongamos que las hipótesis no canceladas de $D'' := \dfrac{\begin{matrix}\vdots\\ \varphi\end{matrix} D \quad \begin{matrix}\vdots\\ \varphi'\end{matrix} D'}{\varphi \wedge \varphi'} \wedge I$ están incluidas en Γ. Como $Hip(D'') = Hip(D) \cup Hip(D')$, Γ contiene tanto las hipótesis de D como las de D', así que $\Gamma \models \varphi$ y $\Gamma \models \varphi'$ por hipótesis inductiva. Sea v una asignación que valide[9] Γ. Luego obtenemos $[\![\varphi]\!]_v = [\![\varphi']\!]_v = 1$, y por definición de valuación tenemos $[\![\varphi \wedge \varphi']\!]_v = 1$. Como v era arbitraria, se sigue que $\Gamma \models \varphi \wedge \varphi'$.

$\boxed{\wedge E}$ Supongamos que $\begin{matrix}\vdots\\ \varphi \wedge \varphi'\end{matrix} D$ satisface la hipótesis inductiva y tomemos

$$D_1 := \frac{\begin{matrix}\vdots\\ \varphi \wedge \varphi'\end{matrix} D}{\varphi} \wedge E \qquad D_2 := \frac{\begin{matrix}\vdots\\ \varphi \wedge \varphi'\end{matrix} D}{\varphi'} \wedge E.$$

Veamos el caso de D_1; sea Γ que contenga $Hip(D_1)$. Como $Hip(D_1) = Hip(D)$, sabemos (por hipótesis inductiva) que $\Gamma \models \varphi \wedge \varphi'$. Sea v una asignación tal que $[\![\Gamma]\!]_v = 1$. Tenemos entonces que $[\![\varphi \wedge \varphi']\!]_v = 1$, y por definición de valuación, $[\![\varphi]\!]_v = 1$. Como v era una asignación arbitraria que validaba Γ, esto muestra que $\Gamma \models \varphi$. El caso de D_2 es igual.

[9] Ver el párrafo siguiente a la Definición 12.

$\boxed{\to I}$ Supongamos que $\begin{array}{c}\varphi\\ \vdots\\ \psi\end{array} D \in \mathcal{D}$ satisface la hipótesis inductiva. Veamos que la derivación D' de la derecha también la satisface. Sea Γ conteniendo las hipótesis de D', es decir, Γ contiene las hipótesis de D menos φ. Tomemos ahora $\Gamma' = \Gamma \cup \{\varphi\}$; Γ' contiene todas las hipótesis de D. Por hipótesis inductiva, $\Gamma' \models \psi$. Sea v tal que $[\![\Gamma]\!]_v = 1$. Si suponemos

$$\dfrac{\begin{array}{c}[\varphi]_1\\ \vdots D\\ \psi\end{array}}{\varphi \to \psi} \to I_1$$

que $[\![\varphi]\!]_v = 1$, v es también una asignación tal que $[\![\Gamma']\!]_v = 1$, y por ende $[\![\psi]\!]_v = 1$, así que se da $[\![\varphi \to \psi]\!]_v = 1$. Si $[\![\varphi]\!]_v = 0$ también se da $[\![\varphi \to \psi]\!]_v = 1$, y en consecuencia $\Gamma \models \varphi \to \psi$.

$\boxed{\to E}$ Sean $\begin{array}{c}\vdots D\\ \varphi\end{array}$, $\begin{array}{c}\vdots D'\\ \varphi \to \psi\end{array} \in \mathcal{D}$ satisfaciendo la HI. Sea $D'' := \dfrac{\begin{array}{c}\vdots D\\ \varphi\end{array} \qquad \begin{array}{c}\vdots D'\\ \varphi \to \psi\end{array}}{\psi} \to E$ y supongamos que Γ contiene $Hip(D'')$. Como $Hip(D'')$ es el conjunto formado por las hipótesis de D y las de D', Γ contiene a estas últimas; por hipótesis inductiva, $\Gamma \models \varphi$ y $\Gamma \models \varphi \to \psi$. Sea v tal que $[\![\Gamma]\!]_v = 1$. Luego $[\![\varphi]\!]_v = 1$ y $[\![\varphi \to \psi]\!]_v = 1$. Si $[\![\psi]\!]_v$ fuera 0, tendríamos que $[\![\varphi \to \psi]\!]_v = 0$, una contradicción. En conclusión, debe ser $[\![\psi]\!]_v = 1$ y luego (pues v era arbitraria) $\Gamma \models \psi$.

$\boxed{RAA}$ Sea $\begin{array}{c}\neg\varphi\\ \vdots\\ \bot\end{array} D$ satisfaciendo la hipótesis inductiva. Sea Γ conteniendo las hipótesis de D' de la derecha; análogamente al caso $(\to I)$, Γ contiene las hipótesis de D menos $\neg\varphi$. Supongamos (por el absurdo) que $\Gamma \not\models \varphi$; luego hay una asignación v tal que $[\![\Gamma]\!]_v = 1$ y $[\![\varphi]\!]_v = 0$, y en consecuencia $[\![\neg\varphi]\!]_v = 1$. En resumen v es una asignación tal que $[\![\Gamma']\!]_v = 1$,

$$\dfrac{\begin{array}{c}[\neg\varphi]_1\\ \vdots D\\ \bot\end{array}}{\varphi} RAA_1$$

con $\Gamma' = \Gamma \cup \{\neg\varphi\}$. Γ' contiene todas las hipótesis de D. Por hipótesis inductiva, $\Gamma' \models \bot$. Pero entonces se debería dar $[\![\bot]\!]_v = 1$, una contradicción. En consecuencia, $\Gamma \models \varphi$.

$\boxed{\bot}$ Sea $\begin{array}{c}\vdots D\\ \bot\end{array}$ que satisfaga la hipótesis inductiva, y sea $\varphi \in PROP$. La derivación $\dfrac{\begin{array}{c}\vdots D\\ \bot\end{array}}{\varphi} \bot \in \mathcal{D}$ tiene las mismas hipótesis que D. El razonamiento es similar al caso (RAA), pero sin hacerse problemas con hipótesis canceladas. Queda como ejercicio muy fácil para el lector.

$\boxed{\vee I}$ Es análoga a $(\wedge E)$.

$\boxed{\vee E}$ Hay que realizar un análisis por casos similar al de $(\to I)$, pero teniendo en cuenta los valores de verdad de las hipótesis que se cancelan en la segunda y tercera subderivaciones. Queda como ejercicio. □

Demostraremos ahora una serie de lemas que nos conducirán a la Completitud. En toda esta sección, Γ será un subconjunto de $PROP$. Comenzamos con una definición:

Definición 26. Un conjunto $\Gamma \subseteq PROP$ es ***inconsistente*** si y sólo si $\Gamma \vdash \bot$. Γ es ***consistente*** si no es inconsistente.

Esto parece un caso particular de lo que se decía más arriba; Γ consistente si no puedo deducir a partir de él *una* ($\bot$) proposición falsa, pero es totalmente general, como lo muestra el siguiente

Lema 27 (de Inconsistencia)**.** *Son equivalentes*

1. Γ *es inconsistente.*
2. *Existe* $\varphi \in PROP$ *tal que* $\Gamma \vdash \neg\varphi$ *y* $\Gamma \vdash \varphi$.
3. *Para toda* $\varphi \in PROP$ *se da* $\Gamma \vdash \varphi$.

Demostración. Es obvio que 3 $\Rightarrow$ 2. También 2 $\Rightarrow$ 1 sale fácil: supongamos que $\begin{array}{c}\vdots\\ \varphi\end{array} D$ y $\begin{array}{c}\vdots\\ \neg\varphi\end{array} D'$ tienen sus hipótesis no canceladas en Γ (i.e., $Hip(D) \subseteq \Gamma$ y $Hip(D') \subseteq \Gamma$). Luego (teniendo en cuenta que $\neg\varphi = \varphi \to \bot$) la derivación de la izquierda tiene conclusión $\bot$ e hipótesis en Γ. La prueba de 1 $\Rightarrow$ 3 está en el Ejemplo 10 ítem 6.

$$\dfrac{\begin{array}{cc}\vdots & \vdots\\ \varphi & \neg\varphi\end{array}\begin{array}{c} D \\ \\ \end{array}\quad\begin{array}{c} D' \\ \\ \end{array}}{\bot} \to E$$

□

Lema 28 (Criterio de Consistencia)**.** *Si hay una asignación* v *tal que* $[\![\psi]\!]_v = 1$ *para toda* $\psi \in \Gamma$*, entonces* Γ *es consistente.*

Demostración. Supongamos por el absurdo que $\Gamma \vdash \bot$. Por la Corrección de la lógica proposicional, $\Gamma \models \bot$, así que para toda asignación v tal que $[\![\Gamma]\!]_v = 1$, se debe dar $[\![\bot]\!]_v = 1$. Pero como para toda v, $[\![\bot]\!]_v = 0$, llegamos a una contradicción. □

Ejemplo 12. Probemos que $\Gamma := \{p_0 \to p_1,\ \neg p_2 \to p_0,\ p_5 \wedge \neg p_0\}$ es consistente. Para ello, basta encontrar una asignación que valide dicho conjunto. Utilizaremos el Teorema de Extensión a tal fin.

Sea $f : At \to \{0, 1\}$ definida de la siguiente manera: $f(\varphi) = 1$ si y sólo si $\varphi = p_2, p_5$. Como $f(\bot) = 0$, existe una valuación $[\![\cdot]\!]_f$ que extiende a f sobre $PROP$. Vemos que esta valuación es de Γ:

$$\begin{array}{rll} [\![p_0 \to p_1]\!]_f = 0 \text{ si y sólo si } [\![p_0]\!]_f = 1 \text{ y } [\![p_1]\!]_f = 0 & & \text{por definición de valuación} \\ \text{si y sólo si } 0 = 1 \text{ y } [\![p_1]\!]_f = 0 & & \text{por construcción de } f \\ \text{si y sólo si nunca.} & & \end{array}$$

$$\begin{array}{rll} [\![p_5 \wedge \neg p_0]\!]_f = \min\{[\![p_5]\!]_f, [\![\neg p_0]\!]_f\} & & \text{por definición de valuación} \\ = \min\{1, [\![\neg p_0]\!]_f\} & & \text{por construcción de } f \\ = \min\{1, 1 - [\![p_0]\!]_f\} & & \text{por Ejercicio 4} \\ = \min\{1, 1 - 0\} & & \text{por construcción de } f \\ = 1. & & \end{array}$$

$$
\begin{aligned}
[\![\neg p_2 \to p_0]\!]_f = 0 \text{ si y sólo si } & [\![\neg p_2]\!]_f = 1 \text{ y } [\![p_0]\!]_f = 0 && \text{por definición de valuación}\\
\text{si y sólo si } & 1 - [\![p_2]\!]_f = 1 \text{ y } [\![p_0]\!]_f = 0 && \text{por Ejercicio 4}\\
\text{si y sólo si } & 1 - 1 = 1 \text{ y } [\![p_0]\!]_f = 0 && \text{por construcción de } f\\
\text{si y sólo si } & 0 = 1 \text{ y } [\![p_0]\!]_f = 0 &&\\
\text{si y sólo si nunca.} &&&
\end{aligned}
$$

Definición 29. Un conjunto Γ es ***consistente maximal*** si y sólo si es consistente y para todo $\Gamma' \supseteq \Gamma$, si Γ' también es consistente, entonces $\Gamma' = \Gamma$.

Ejemplo 13. Dada una valuación v, el conjunto $\Gamma := \{\varphi \in PROP \ : \ [\![\varphi]\!]_v = 1\}$ es un conjunto consistente maximal. Por el Lema 28, Γ es consistente. Consideremos un Γ' consistente y que contenga a Γ, es decir, $\Gamma \subseteq \Gamma'$. Ahora, supongamos por el absurdo $\psi \in \Gamma' \setminus \Gamma$. Luego $[\![\psi]\!]_v = 0$ y por ende $[\![\neg\psi]\!]_v = 1$; en conclusión $\neg\psi \in \Gamma$. Pero como $\Gamma \subseteq \Gamma'$, tenemos que Γ' es inconsistente, una contradicción. Luego no hay elementos de Γ' fuera de Γ, $\Gamma = \Gamma'$.

Este ejemplo de conjunto consistente maximal aparenta ser muy específico; sin embargo, como se verá más adelante, tiene toda la generalidad posible.

Lema 30. *Todo conjunto consistente* Γ *está contenido en uno maximal* Γ^*.

Demostración. Las proposiciones forman un conjunto *numerable*, es decir, se puede hacer una lista $\varphi_0, \varphi_1, \ldots, \varphi_n, \ldots$ (con subíndices todos los números naturales) en la cual aparecen todas las proposiciones.

Ejercicio 9. (*) Pensar en un modo de llevar esto a cabo (Ayuda: considerar las proposiciones de complejidad menor que n que tengan grado[10] menor que n. Son finitas, y toda proposición tiene grado y complejidad finitos).

Definiremos una sucesión no decreciente de conjuntos Γ_i tal que la unión es consistente maximal.

$$
\begin{aligned}
\Gamma_0 &:= \Gamma,\\
\Gamma_{n+1} &:= \begin{cases} \Gamma_n \cup \{\varphi_n\} & \text{si resulta consistente} \\ \Gamma_n & \text{en caso contrario} \end{cases}\\
\Gamma^* &:= \bigcup_{n \geq 0} \Gamma_n
\end{aligned}
$$

Se puede probar por inducción en n que cada Γ_n es consistente (por construcción, Γ_0 es consistente; y si Γ_n es consistente, Γ_{n+1} es consistente, pues alguna de las dos opciones se da). Veamos que Γ^* también lo es.

Por el absurdo, supongamos que $\Gamma^* \vdash \bot$. Luego hay una derivación D con $Concl(D) = \bot$ y $Hip(D) \subseteq \Gamma^*$. Como $Hip(D)$ es un conjunto finito, hay un N suficientemente grande tal que $Hip(D) \subseteq \Gamma_{N+1}$; de hecho, tomando $N := \text{máx}\{n : \varphi_n \in Hip(D)\}$ tenemos

$$Hip(D) \subseteq \Gamma^* \cap \{\varphi_0, \ldots, \varphi_N\} \subseteq \Gamma_{N+1}.$$

Entonces, D es una derivación de $\bot$ con hipótesis en Γ_{N+1}. Esto es un absurdo, ya que Γ_{N+1} es consistente.

[10] Ver Definición 1.2 y el Ejercicio 6 de la Sección 1.5.

Γ^* es consistente maximal: para verlo, supongamos $\Gamma^* \subseteq \Delta$ con Δ consistente. Si $\psi \in \Delta$, entonces $\psi = \varphi_m$ para algún $m \geq 0$ (pues en nuestra enumeración aparecían **todas** las proposiciones). Como $\Gamma_m \subseteq \Gamma^* \subseteq \Delta$ y Δ es consistente, $\Gamma_m \cup \{\varphi_m\}$ es consistente. Luego $\Gamma_{m+1} = \Gamma_m \cup \{\varphi_m\}$, i.e. $\varphi_m \in \Gamma_{m+1} \subseteq \Gamma^*$. Esto muestra que $\Gamma^* = \Delta$. □

Lema 31. *1. Si $\Gamma \cup \{\neg\varphi\}$ es inconsistente entonces $\Gamma \vdash \varphi$.*

2. Si $\Gamma \cup \{\varphi\}$ es inconsistente entonces $\Gamma \vdash \neg\varphi$.

Demostración. En cada caso hay derivaciones $\begin{array}{c}\neg\varphi\\ \vdots\\ D\\ \vdots\\ \bot\end{array}$ y $\begin{array}{c}\varphi\\ \vdots\\ D\\ \vdots\\ \bot\end{array}$ con hipótesis no canceladas en $\Gamma \cup \{\neg\varphi\}$ y $\Gamma \cup \{\varphi\}$, respectivamente.

Luego las siguientes son derivaciones con hipótesis no canceladas en Γ:

$$\dfrac{\begin{array}{c}[\neg\varphi]_1\\ \vdots\\ D\\ \vdots\\ \bot\end{array}}{\varphi}\, RAA_1 \qquad \dfrac{\begin{array}{c}[\varphi]_1\\ \vdots\\ D\\ \vdots\\ \bot\end{array}}{\neg\varphi} \to I_1,$$

y queda probado el resultado. □

Lema 32. *Si Γ es consistente maximal entonces es **cerrado por derivaciones** (i.e., $\Gamma \vdash \varphi$ implica $\varphi \in \Gamma$).*

Demostración. Supongamos $\Gamma \vdash \varphi$, y en busca de un absurdo supongamos que $\varphi \notin \Gamma$. Luego $\Gamma \cup \{\varphi\}$ debe ser inconsistente. Entonces $\Gamma \vdash \neg\varphi$ por el Lema 31, así que Γ es inconsistente por el Lema 27. Absurdo. □

El siguiente lema se puede explicar diciendo que un conjunto consistente maximal ***realiza*** los conectivos $\neg$, $\to$ y $\vee$.

Lema 33. *Sea Γ consistente maximal. Luego, para todas para todas $\varphi, \psi \in PROP$,*

1. *$\neg\varphi \in \Gamma$ si y sólo si* [no $\varphi \in \Gamma$].
2. *$(\varphi \to \psi) \in \Gamma$ si y sólo si* [$\varphi \in \Gamma$ implica $\psi \in \Gamma$].
3. *$\varphi \vee \psi \in \Gamma$ si y sólo si* [$\varphi \in \Gamma$ ó $\psi \in \Gamma$].

Demostración. 1. ($\Rightarrow$) Si $\neg\varphi$ está en Γ, entonces φ no puede puesto que sería inconsistente.

($\Leftarrow$) Si φ no está, entonces $\Gamma \cup \{\varphi\}$ es inconsistente (por ser Γ maximal). Por los Lemas 31 y 32, $\neg\varphi \in \Gamma$.

2. ($\Rightarrow$) Supongamos $(\varphi \to \psi) \in \Gamma$, veamos que se da la implicación entre corchetes. Para ello, supongamos $\varphi \in \Gamma$. Ahora, con hipótesis $(\varphi \to \psi)$ y φ puedo derivar ψ por el Ejemplo 10(2). Como Γ es cerrado por derivaciones (Lema 32), tenemos que $\psi \in \Gamma$. Obtuvimos entonces [$\varphi \in \Gamma$ implica $\psi \in \Gamma$].

   ($\Leftarrow$) Supongamos cierta la implicación. Hacemos dos casos.

a) Si $\varphi \in \Gamma$, tenemos que $\psi \in \Gamma$ por la implicación. En particular, $\Gamma \vdash \psi$, así que podemos asegurar $\Gamma \vdash (\varphi \to \psi)$. El Lema 32 nos asegura entonces $(\varphi \to \psi) \in \Gamma$.

b) Si $\varphi \notin \Gamma$, entonces $\neg\varphi \in \Gamma$ (por lo probado anteriormente). Por el Ejemplo 11 obtenemos $\Gamma \vdash (\varphi \to \psi)$, y como es cerrado por derivaciones, $(\varphi \to \psi) \in \Gamma$.

3. Queda como ejercicio (10 de la Sección 2.4).

□

Ejercicio 10. Demostrar que los conjuntos consistentes maximales realizan la conjunción.

Lema 34. *Si* Γ *es consistente, entonces existe una asignación* v *tal que* $[\![\psi]\!]_v = 1$ *para toda* $\psi \in \Gamma$.

Demostración. Por el Lema 30, Γ está contenido en algún Γ^* maximal. Definamos: $f(p_i) := 1$ si $p_i \in \Gamma^*$ y $f(\varphi) := 0$ para toda otra $\varphi \in At$. Por el Teorema 9, f se puede extender a una valuación $[\![\cdot]\!]_f$. Veremos por inducción que $[\![\varphi]\!]_f = 1$ si y sólo si $\varphi \in \Gamma^*$.

$\boxed{\varphi \in At}$ Vale por construcción de f (notemos que $\bot \notin \Gamma^*$).

$\boxed{(\varphi \wedge \psi)}$

$[\![(\varphi \wedge \psi)]\!]_f = 1$	si y sólo si	$[\![\varphi]\!]_f = 1$ y $[\![\psi]\!]_f = 1$	por definición de valuación
	si y sólo si	$\varphi, \psi \in \Gamma^*$	por hipótesis inductiva
	si y sólo si	$(\varphi \wedge \psi) \in \Gamma^*$	por Ejercicio 10

$\boxed{(\varphi \to \psi)}$

$[\![(\varphi \to \psi)]\!]_f = 0$	si y sólo si	$[\![\varphi]\!]_f = 1$ y $[\![\psi]\!]_f = 0$	por definición de valuación
	si y sólo si	$\varphi \in \Gamma^*$ y $\psi \notin \Gamma^*$	por hipótesis inductiva
	si y sólo si	no se da: [$\varphi \in \Gamma^*$ implica $\psi \in \Gamma^*$]	
	si y sólo si	$(\varphi \to \psi) \notin \Gamma^*$	por el Lema 33(2)

$\boxed{(\varphi \vee \psi)}$

$[\![(\varphi \vee \psi)]\!]_f = 1$	si y sólo si	$[\![\varphi]\!]_f = 1$ ó $[\![\psi]\!]_f = 1$	por definición de valuación
	si y sólo si	$\varphi \in \Gamma^*$ ó $\psi \in \Gamma^*$	por hipótesis inductiva
	si y sólo si	$(\varphi \vee \psi) \in \Gamma^*$	por el Lema 33(3)

Como $\Gamma \subseteq \Gamma^*$, tenemos $[\![\psi]\!]_f = 1$ para toda $\psi \in \Gamma$. □

Corolario 35. $\Gamma \nvdash \varphi$ *implica que hay una valuación* v *tal que* $[\![\psi]\!]_v = 1$ *para todo* $\psi \in \Gamma$ *y* $[\![\varphi]\!]_v = 0$.

Demostración.

$\Gamma \nvdash \varphi$ implica $\Gamma \cup \{\neg\varphi\}$ es consistente (por el Lema 31)
si y sólo si hay valuación v tal que $[\![\psi]\!]_v = 1$ para toda $\psi \in \Gamma \cup \{\neg\varphi\}$
si y sólo si hay valuación v tal que $[\![\psi]\!]_v = 1$ para toda $\psi \in \Gamma$ y $[\![\varphi]\!]_v = 0$.

Queda demostrada la implicación. □

Prueba de Completitud. Supongamos $\Gamma \models \varphi$. Luego, para toda valuación v tal que $[\![\psi]\!]_v = 1$ para toda $\psi \in \Gamma$, se da $[\![\varphi]\!]_v = 1$. Esto equivale a decir que no hay valuación tal que $[\![\psi]\!]_v = 1$ para toda $\psi \in \Gamma$ y $[\![\varphi]\!]_v = 0$. Por la contrarrecíproca al corolario anterior, obtenemos que no se puede dar $\Gamma \nvdash \varphi$, es decir, obtenemos $\Gamma \vdash \varphi$. □

2.3. Reglas para la negación y la doble implicación

Tal como definimos la negación y la doble implicación diciendo que son abreviaturas, podemos introducir abreviaturas para las reglas de deducción que los involucran. De este modo, la regla intuitiva de introducción de $\leftrightarrow$

$$\dfrac{\begin{array}{cc}[\varphi] & [\psi] \\ \vdots & \vdots \\ \psi & \varphi\end{array}}{\varphi \leftrightarrow \psi} \leftrightarrow I.$$

que se corresponde con los razonamientos que parten de φ para llegar a ψ y viceversa, será una abreviatura de

$$\dfrac{\dfrac{\begin{array}{c}[\varphi]_1 \\ \vdots \\ \psi\end{array}}{\varphi \to \psi} \to I_1 \qquad \dfrac{\begin{array}{c}[\psi]_2 \\ \vdots \\ \varphi\end{array}}{\psi \to \varphi} \to I_2}{\varphi \leftrightarrow \psi} \wedge I.$$

También tendremos reglas de eliminación,

$$\dfrac{\varphi \quad \varphi \leftrightarrow \psi}{\psi} \leftrightarrow E \qquad\qquad \dfrac{\psi \quad \varphi \leftrightarrow \psi}{\varphi} \leftrightarrow E,$$

que abrevian, respectivamente,

$$\dfrac{\varphi \quad \dfrac{\varphi \leftrightarrow \psi}{\varphi \to \psi} \wedge E}{\psi} \to E \qquad\qquad \dfrac{\psi \quad \dfrac{\varphi \leftrightarrow \psi}{\psi \to \varphi} \wedge E}{\varphi} \to E$$

Por último, definimos abreviaturas para las reglas que involucran a la negación

$$\dfrac{\varphi \quad \neg\varphi}{\bot} \neg E \qquad \dfrac{\begin{array}{c}[\varphi] \\ \vdots \\ \bot\end{array}}{\neg\varphi} \neg I$$

Se puede ver que usando la definición de $\mathcal{D}$ y las funciones *Concl* e *Hip* se obtiene lo siguiente:

$\boxed{\leftrightarrow I}$ Dadas $\begin{array}{c}\varphi\\ \vdots\\ \psi\end{array} D$ y $\begin{array}{c}\psi\\ \vdots\\ \varphi\end{array} D'$ en $\mathcal{D}$, la siguiente

$$\dfrac{\begin{array}{cc}[\varphi] & [\psi]\\ \vdots & \vdots\\ \psi & \varphi\end{array}\begin{array}{c}D\\ \\ \end{array}\quad\begin{array}{c}D'\\ \\ \end{array}}{\varphi \leftrightarrow \psi} \leftrightarrow I,$$

es una derivación con hipótesis no canceladas las de D **sin** φ junto con las de D' **sin** ψ.

$\boxed{\leftrightarrow E}$ Si $\begin{array}{c}\\ \vdots\\ \varphi\end{array} D_1$, $\begin{array}{c}\\ \vdots\\ \psi\end{array} D_2$ y $\begin{array}{c}\\ \vdots\\ \varphi \leftrightarrow \psi\end{array} D$ son derivaciones, entonces

$$D' := \dfrac{\begin{array}{cc}\begin{array}{c}\\ \vdots\\ \varphi\end{array} D_1 & \begin{array}{c}\\ \vdots\\ \varphi \leftrightarrow \psi\end{array} D\end{array}}{\psi} \leftrightarrow E \qquad D'' := \dfrac{\begin{array}{cc}\begin{array}{c}\\ \vdots\\ \psi\end{array} D_2 & \begin{array}{c}\\ \vdots\\ \varphi \leftrightarrow \psi\end{array} D\end{array}}{\varphi} \leftrightarrow E$$

pertenecen a $\mathcal{D}$ y sus hipótesis no canceladas son las de D y D_1 en conjunto para la primera, y las de D y D_2 en conjunto para la segunda. En símbolos, $Hip(D') = Hip(D) \cup Hip(D_1)$ y $Hip(D'') = Hip(D) \cup Hip(D_2)$.

$\boxed{\neg I}$ Dada $\begin{array}{c}\varphi\\ \vdots\\ \bot\end{array} D$ en $\mathcal{D}$, entonces $\dfrac{\begin{array}{c}[\varphi]\\ \vdots\\ \bot\end{array} D}{\neg\varphi} \neg I$, es una derivación con hipótesis no canceladas $Hip(D) \setminus \{\varphi\}$ (es decir, las mismas hipótesis de D salvo eventualmente φ).

$\boxed{\neg E}$ Si tenemos derivaciones $\begin{array}{c}\\ \vdots\\ \varphi\end{array} D$ y $\begin{array}{c}\\ \vdots\\ \neg\varphi\end{array} D'$ entonces $\dfrac{\begin{array}{cc}\begin{array}{c}\\ \vdots\\ \varphi\end{array} D & \begin{array}{c}\\ \vdots\\ \neg\varphi\end{array} D'\end{array}}{\bot} \neg E$ pertenece a $\mathcal{D}$, y sus hipótesis no canceladas son las de D y D' en conjunto, $Hip(D) \cup Hip(D')$.

Nota 2. Igual como sucede con $\vee E$, hay que tener mucho cuidado cuando se aplica la regla $\leftrightarrow I$: la hipótesis φ se cancela en la primera subderivación (es decir, D), *pero* **no** *en la segunda (D').*

Como un ejemplo de derivación usando todas las reglas introducidas, demostraremos la equivalencia clásica entre la implicación $\varphi \to \psi$ y la "implicación material", $\neg\varphi \vee \psi$.

Ejemplo 14. Ver que $\vdash (\varphi \to \psi) \leftrightarrow (\neg\varphi \vee \psi)$. Es natural suponer que la última regla aplicada va a ser la introducción de $\leftrightarrow$, así que de ese modo comenzamos nuestra derivación:

$$\dfrac{\begin{array}{cc}[\varphi \to \psi] & [\neg\varphi \vee \psi]\\ \vdots & \vdots\\ \neg\varphi \vee \psi & \varphi \to \psi\end{array}}{\varphi \to \psi \leftrightarrow \neg\varphi \vee \psi} \leftrightarrow I$$

El lado izquierdo necesita una aplicación de la reducción al absurdo:

$$D_1 \;:=\; \dfrac{\dfrac{\dfrac{\dfrac{[\varphi]_1 \quad [\varphi \to \psi]_5}{\psi}\to E}{\neg\varphi \vee \psi}\vee I \qquad [\neg(\neg\varphi \vee \psi)]_2}{\dfrac{\dfrac{\bot}{\neg\varphi}\neg I_1}{\neg\varphi \vee \psi}\vee I \qquad [\neg(\neg\varphi \vee \psi)]_2}\neg E}{\bot}\neg E}{\neg\varphi \vee \psi}\, RAA_2$$

D_1 tiene como única hipótesis no cancelada a $\varphi \to \psi$ y $Concl(D_1) = \neg\varphi \vee \psi$, así que nos sirve.

El lado derecho es una ligera abreviación de la derivación del Ejemplo 9 (con $(\neg E)$ en lugar de $(\to E)$):

$$D_2 \;:=\; \dfrac{\dfrac{[\neg\varphi \vee \psi]_5 \qquad \dfrac{\dfrac{[\neg\varphi]_3 \quad [\varphi]_4}{\bot}\neg E}{\psi}\bot \qquad [\psi]_3}{\psi}\vee E_3}{\varphi \to \psi}\to I_4$$

También D_2 tiene las propiedades requeridas, así que

$$\dfrac{\begin{array}{cc}[\varphi \to \psi]_5 & [\neg\varphi \vee \psi]_5 \\ \vdots\, D_1 & \vdots\, D_2 \\ \neg\varphi \vee \psi & \varphi \to \psi\end{array}}{\varphi \to \psi \leftrightarrow \neg\varphi \vee \psi}\leftrightarrow I_5$$

es la derivación que andábamos buscando.

2.4. Ejercicios

Ayuda general: Releer las definiciones 1547 veces.

1. Hallar derivaciones con todas sus hipótesis canceladas que demuestren:
   - *a*) $\vdash \top$.
   - *b*) $\vdash (\varphi \to \neg\varphi) \to \neg\varphi$.
   - *c*) $\vdash \big(\varphi \to (\psi \to \sigma)\big) \leftrightarrow \big(\psi \to (\varphi \to \sigma)\big)$.
   - *d*) $\vdash (\varphi \to \psi) \wedge (\varphi \to \neg\psi) \to \neg\varphi$.
   - *e*) $\vdash \varphi \vee \varphi \leftrightarrow \varphi$.
   - *f*) $\vdash \varphi \vee \psi \to \psi \vee \varphi$.
   - *g*) $\vdash \varphi \vee \psi \;\to\; \neg(\neg\varphi \wedge \neg\psi)$.
   - *h*) $\vdash \top \vee \bot$.
   - *i*) $\vdash (\varphi \to \psi) \to \big((\varphi \to (\psi \to \sigma)) \to (\varphi \to \sigma)\big)$.

2. Para las derivaciones de este ejercicio es necesario utilizar la regla (RAA).
   - *a*) $\vdash ((\varphi \to \psi) \to \varphi) \to \varphi$.
   - *b*) $\vdash \varphi \leftrightarrow \neg\neg\varphi$.
   - *c*) $\vdash \neg(\neg\varphi \wedge \neg\psi) \;\to\; \varphi \vee \psi$.[11]
   - *d*) $\vdash (\varphi \leftrightarrow \bot) \vee (\varphi \leftrightarrow \top)$.
   - *e*) $\vdash \varphi \vee \psi \leftrightarrow ((\varphi \to \psi) \to \psi)$.
   - *f*) $\vdash (\varphi \to \psi) \vee (\psi \to \varphi)$.

[11] Resuelto en el Apéndice.

3. Demostrar:
   a) $\varphi \vdash \neg(\neg\varphi \wedge \psi)$.
   b) $\{\neg(\varphi \wedge \neg\psi), \varphi\} \vdash \psi$.
   c) (*) $\{(\varphi \vee \psi) \wedge (\varphi \vee \theta)\} \vdash \varphi \vee (\psi \wedge \theta)$.[12]
   d) $\neg\varphi \vdash \varphi \to \psi$.
   e) $\{(\neg\varphi \to \psi), (\varphi \to \psi)\} \vdash \psi$.

4. Probar que $\Gamma \vdash \varphi$ implica $\Gamma \cup \Delta \vdash \varphi$, y que si tenemos $\Gamma \vdash \varphi$ y $\Delta \cup \{\varphi\} \vdash \psi$ entonces $\Gamma \cup \Delta \vdash \psi$.

5. Demostrar, transformando derivaciones cuando sea necesario:

   a) $\vdash \varphi$ implica $\vdash \psi \to \varphi$
   b) Si $\varphi \vdash \psi$ y $\neg\varphi \vdash \psi$ entonces $\vdash \psi$.
   c) $\Gamma \cup \{\varphi\} \vdash \psi$ implica $\Gamma \setminus \{\varphi\} \vdash (\varphi \to \varphi) \wedge (\varphi \to \psi)$.
   d) $\Gamma \cup \{\varphi\} \vdash \psi$ implica $\Gamma \vdash \varphi \to (\psi \vee \neg\varphi)$.

6. (*) Definir por recursión el *conjunto* de proposiciones que ocurren en una derivación D.

7. Decida cuáles de los siguientes conjuntos son consistentes:

   a) $\{\neg p_1 \wedge p_2 \to p_0, p_1 \to (\neg p_1 \to p_2), p_0 \leftrightarrow \neg p_2\}$.
   b) $\{p_0 \to p_1, p_1 \to p_2, p_2 \to p_3, p_3 \to \neg p_0\}$.
   c) $\{p_0 \to p_1, p_0 \wedge p_2 \to p_1 \wedge p_3, p_0 \wedge p_2 \wedge p_4 \to p_1 \wedge p_3 \wedge p_5, \dots\}$ (pares implican impares...).
   d) $\{p_{2n} : n \geq 0\} \cup \{\neg p_{3n+1} : n \geq 0\}$.
   e) $\{p_{2n} : n \geq 0\} \cup \{\neg p_{4n+1} : n \geq 0\}$.

8. Probar que $\Gamma \cup \{\varphi \wedge \psi\}$ es consistente si y sólo si $\Gamma \cup \{\varphi, \psi\}$ es consistente (ayuda: contrarrecíproca).

9. Demostrar que $\Gamma^+ := \{\varphi \in PROP : \varphi$ no contiene los conectivos "$\neg$" ni "$\bot$"$\}$ es consistente (Ayuda: construir una v y probar por inducción en subfórmulas que $[\![\varphi]\!]_v = 1$ para toda $\varphi \in \Gamma^+$).

10. Pruebe que todo Γ consistente maximal realiza la disyunción:

    para toda φ, ψ, $\varphi \vee \psi \in \Gamma$ si y sólo si ($\varphi \in \Gamma$ *ó* $\psi \in \Gamma$).

11. Sea Γ consistente maximal y suponga $\{p_0, \neg(p_1 \to p_2), p_3 \vee p_2\} \subseteq \Gamma$. Decida si las siguientes proposiciones están en Γ. (Ayuda: usar Completitud, o la caracterización de consistente maximal).
    a) $\neg p_0$.
    b) p_3.
    c) $p_2 \to p_5$.
    d) $p_1 \vee p_6$.

12. Sea Γ consistente y cerrado por derivaciones. ¿Es maximal?

13. Considere la relación $\Gamma \vdash^+ \varphi$, definida de igual manera que $\Gamma \vdash \varphi$ salvo que se reemplaza la regla $(\to I)$ por la siguiente:

[12] Resuelto en el Apéndice.

$\boxed{\to I^+}$ Dada $\begin{array}{c}\varphi\\ \vdots\\ \psi\end{array} D$ en $\mathcal{D}$ **tal que** $\boldsymbol{\varphi \in Hip(D)}$, tenemos que $D' := \dfrac{\begin{array}{c}[\varphi]\\ \vdots\\ \psi\end{array} D}{\varphi \to \psi} \to I$ pertenece a $\mathcal{D}$.

Pruebe que $\Gamma \vdash^+ \varphi$ si y sólo si $\Gamma \vdash \varphi$.

14. *a*) (**) Demostrar que si $\vdash \varphi$, entonces existe una derivación de $\neg\neg\varphi$ con todas sus hipótesis canceladas que no utiliza la Regla del Absurdo

    b) (¡sin estrella!) Intentar nuevamente el ítem anterior aplicando el siguiente Teorema de *Forma Normal RAA*:

    > Para toda derivación D existe una derivación D' con las mismas hipótesis no canceladas y la misma conclusión, tal que D' tiene **a lo sumo una** aplicación de la regla (RAA), exactamente **al final**.

    c) (*) Probar el Teorema de Forma Normal (RAA).[13]

    d) Probar que todo teorema de la forma $\neg\varphi$ es *intuicionista*, es decir, que se puede demostrar sin usar (RAA).

15. (*) Muestre que son equivalentes:

    a) $\{\varphi_1, \ldots, \varphi_n\}$ es inconsistente.

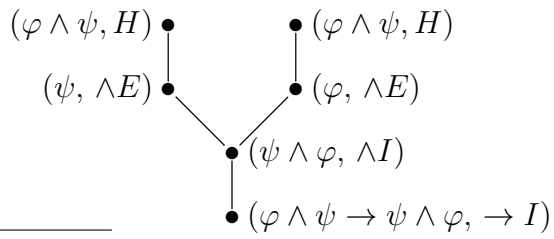
    b) $\vdash \neg(\varphi_1 \wedge \ldots \wedge \varphi_n)$.

    c) $\vdash \varphi_2 \wedge \ldots \wedge \varphi_n \to \neg\varphi_1$.

    Aquí, $\varphi_1 \wedge \ldots \wedge \varphi_n := \varphi_1 \wedge (\varphi_2 \wedge (\ldots(\varphi_{n-1} \wedge \varphi_n))\ldots)$ (Ayuda: probar un resultado más general, "$\Gamma \cup \{\varphi_1, \ldots, \varphi_n\}$ es inconsistente equivale a $\Gamma \vdash \ldots$", por inducción en n).

16. (*) Dar dos derivaciones D_1 y D_2 tales $Hip(D_1) \neq Hip(D_2)$, pero que tienen el mismo árbol con las mismas decoraciones en $PROP$.

El Ejercicio 16 muestra que nuestra presentación no es 100 % rigurosa, en el sentido que la "implementación" que propusimos para las derivaciones no es completamente fiel. Una que sí lo es se obtiene decorando cada nodo de los árboles con la proposición y la regla que nos hace obtenerla. En el caso de las derivaciones de una sola nodo, la regla puede llamarse "(H)" (por "hipótesis" u "hoja"). Para dar un ejemplo, la derivación (3), tendría la siguiente estructura ahora:



[13] Resuelto en el Apéndice.

3. Reticulados y Lógica

Un libro para consultar acerca de esta sección es *Introduction to Lattices and Order*, de B. A. Davey y H. A. Priestley (Cambridge Mathematical Texts), en el capítulo 7.

Uno se preguntará ahora, ¿por qué el ínfimo de un álgebra de Boole se denota con el mismo símbolo que la conjunción ("$\wedge$")? Si la lógica corresponde a las álgebras de Boole, ¿que significan los filtros, los filtros primos?

Antes de abordar estas cuestiones, repasemos un par de ejercicios de Deducción Natural, algunos de ellos muy triviales:

3.1. Más Ejercicios

Probar las siguientes afirmaciones.

1. *a*) $\vdash \varphi \to \varphi$.
   b) Si $\vdash \varphi \to \psi$ y $\vdash \psi \to \varphi$ entonces $\vdash \varphi \leftrightarrow \psi$.
   c) Si $\vdash \varphi \to \psi$ y $\vdash \psi \to \chi$ entonces $\vdash \varphi \to \chi$.
2. *a*) $\vdash \varphi \wedge \psi \to \varphi$.
   b) $\vdash \varphi \wedge \psi \to \psi$.
   c) Si $\vdash \chi \to \varphi$ y $\vdash \chi \to \psi$ entonces $\vdash \chi \to \varphi \wedge \psi$.
3. *a*) $\vdash \varphi \to \varphi \vee \psi$.
   b) $\vdash \psi \to \varphi \vee \psi$.
   c) Si $\vdash \varphi \to \chi$ y $\vdash \psi \to \chi$ entonces $\vdash \varphi \vee \psi \to \chi$.
4. *a*) $\vdash \varphi \to \top$, $\vdash \bot \to \varphi$.
   b) $\vdash \varphi \wedge \neg\varphi \leftrightarrow \bot$.
   c) $\vdash \varphi \vee \neg\varphi \leftrightarrow \top$.

3.2. *PROP* como poset

Con todos los elementos de la sección anterior, basta hacer un pequeño acto de abstracción para probar que "está todo conectado". Definamos una relación $\preccurlyeq$ en *PROP* de la siguiente manera:

$\varphi \preccurlyeq \psi$ si y sólo si $\vdash \varphi \to \psi$.

Ahora bien, esta relación resulta reflexiva por el Ejercicio 1*a* y transitiva por el Ejercicio 1*c*. No es antisimétrica, pues tenemos $(p_0 \wedge \neg p_0) \preccurlyeq \bot$ y $\bot \preccurlyeq (p_0 \wedge \neg p_0)$ y sin embargo $\bot \neq (p_0 \wedge \neg p_0)$. Lo que sí sabemos es que $\vdash \bot \leftrightarrow p_0 \wedge \neg p_0$, así que si consideráramos dos proposiciones equivalentes (es decir, que se pueda derivar la proposición que afirma "una si y solo si la otra") como idénticas, tendríamos la antisimetría.

Definición 36. Sea $\approx$ la relación de equivalencia dada por $\varphi \approx \psi$ si y sólo si $\vdash \varphi \leftrightarrow \psi$. Definamos $\overline{\varphi}$ como la clase de equivalencia correspondiente a φ según la relación $\approx$.

Ejercicio 11. Demostrar que la relación $\approx$ es efectivamente una relación de equivalencia.

Llamaremos $\overline{PROP}$ al conjunto de clases de equivalencia de la relación $\approx$, y denotaremos $\overline{\varphi}$ a la clase de equivalencia de φ. Por ejemplo, tenemos $\overline{\bot} = \overline{p_0 \wedge \neg p_0}$ (pues $\vdash \bot \leftrightarrow p_0 \wedge \neg p_0$). Para verlo de una forma más simple, usamos el símbolo $\overline{\varphi}$ para poder trabajar normalmente con φ, pero se la puede reemplazar indistintamente por cualquier ψ tal que $\vdash \varphi \leftrightarrow \psi$.

Se puede ahora extender la definición de $\preccurlyeq$ a $\overline{PROP}$, y se hace de la manera obvia.

Definición 37. Diremos que $\overline{\varphi} \preccurlyeq \overline{\psi}$ si $\varphi \preccurlyeq \psi$.

Para ver que esta definición es *buena*, necesitamos un resultado más:

Ejercicio 12. Supongamos $\varphi \approx \psi$ y $\chi \approx \theta$. Entonces $\varphi \preccurlyeq \chi$ si y sólo si $\psi \preccurlyeq \theta$. (Ayuda: reemplazando $\approx$ y $\preccurlyeq$ por sus definiciones respectivas, este ejercicio pide demostrar: "Dadas dos derivaciones D y D' con todas sus hipótesis canceladas y conclusión $\varphi \leftrightarrow \psi$ y $\chi \leftrightarrow \theta$, probar: existe $D_1 \in \mathcal{D}$ con $Hip(D_1) = \emptyset$ y conclusión $\varphi \to \chi$ si y sólo si existe $D_2 \in \mathcal{D}$ con $Hip(D_2) = \emptyset$ con conclusión $\psi \to \theta$".)

Las propiedades que vimos de la relación $\preccurlyeq$ siguen valiendo si ponemos "$\overline{\cdots}$" en todos lados; decimos que $\preccurlyeq$ es *preservada* por $\approx$. Por ejemplo, por el Ejercicio 1*a*, tenemos:

Para toda φ, $\overline{\varphi} \preccurlyeq \overline{\varphi}$.

El Ejercicio 1*c*, por su parte, nos dice que $\preccurlyeq$ es transitiva:

Para todas φ, ψ y χ, $\overline{\varphi} \preccurlyeq \overline{\psi}$ y , $\overline{\psi} \preccurlyeq \overline{\chi}$ implican , $\overline{\varphi} \preccurlyeq \overline{\chi}$.

Volviendo a la antisimetría, el Ejercicio 1*b* nos dice que si $\varphi \preccurlyeq \psi$ y $\psi \preccurlyeq \varphi$ obtenemos $\vdash \varphi \leftrightarrow \psi$. Esto quiere decir que $\varphi \approx \psi$ y luego están en la misma clase de equivalencia, $\overline{\varphi} = \overline{\psi}$:

Si $\overline{\varphi} \preccurlyeq \overline{\psi}$ y $\overline{\psi} \preccurlyeq \overline{\varphi}$, entonces $\overline{\varphi} = \overline{\psi}$.

En resumen: $\preccurlyeq$ define una relación de orden en $PROP$, una vez que identificamos cosas equivalentes. ¿Cómo es este poset? (o mejor, ¿para qué nos mandaron a hacer el resto de los ejercicios?).

3.3. El Álgebra de Lindenbaum

Traduciendo los ejercicios restantes, obtenemos lo siguiente. Los Ejercicios 2*a* y 2*b* nos dicen que la conjunción de dos proposiciones es una cota inferior de las mismas:

Para todas φ, ψ, $\overline{\varphi \wedge \psi} \preccurlyeq \overline{\varphi}$ y $\overline{\varphi \wedge \psi} \preccurlyeq \overline{\psi}$,

y el Ejercicio 2*c* dice que es mayor o igual que cualquier cota:

Si $\overline{\chi} \preccurlyeq \overline{\varphi}$ y $\overline{\chi} \preccurlyeq \overline{\psi}$, entonces $\overline{\chi} \preccurlyeq \overline{\varphi \wedge \psi}$.

Juntando todo, tenemos que efectivamente $\overline{\varphi \wedge \psi}$ es el ínfimo entre $\overline{\varphi}$ y $\overline{\psi}$ en $\overline{PROP}$. Con esto hemos demostrado que es lícito escribir $\overline{\varphi \wedge \psi} = \overline{\varphi} \wedge \overline{\psi}$.

Por otro lado, traduciendo correspondientemente los Ejercicios 3 deducimos que $\vee$ nos fabrica el supremo, y se puede ver que distribuye con el ínfimo:

Ejercicio 13. Probar que efectivamente $\vee$ distribuye con $\wedge$ en $\overline{PROP}$

Por último, viendo los Ejercicios 4*a*, 4*b* y 4*c* obtenemos las siguientes propiedades:

Para toda φ, $\overline{\varphi} \preccurlyeq \overline{\top}$ y $\overline{\bot} \preccurlyeq \overline{\varphi}$.
$\overline{\varphi} \wedge \overline{\neg\varphi} = \overline{\bot}$, $\overline{\varphi} \vee \overline{\neg\varphi} = \overline{\top}$.

Es decir, $\overline{\top}$ y $\overline{\bot}$ son respectivamente los elementos máximo y mínimo de $\overline{PROP}$, y $\overline{\neg\varphi}$ cumple el rol de complemento. En suma, no sólo $\overline{PROP}$ es un poset, sino que también es un álgebra de Boole $\langle \overline{PROP}, \wedge, \vee, \neg, \overline{\bot}, \overline{\top} \rangle$, que se llama *álgebra de Lindenbaum*.

Pero las "coincidencias" no terminan aquí. Definamos $\overline{\Gamma} := \{\overline{\varphi} : \varphi \in \Gamma\}$.

Lema 38. *Γ es inconsistente si y sólo si hay elementos de $\overline{\Gamma}$ cuyo ínfimo (en $\overline{PROP}$) es $\overline{\bot}$.*

Demostración. ($\Rightarrow$) Como Γ es inconsistente, tenemos una derivación D con $Hip(D) = \{\varphi_1, \ldots, \varphi_n\} \subseteq \Gamma$ y $Concl(D) = \bot$. Pero entonces $\{\varphi_1, \ldots, \varphi_n\} \vdash \bot$, y esto es lo mismo que $\vdash \varphi_1 \wedge \ldots \wedge \varphi_n \to \bot$. Además sabemos (por la regla $(\bot)$) que $\vdash \bot \to \varphi_1 \wedge \ldots \wedge \varphi_n$, así que tenemos en resumen $\overline{\bot} = \overline{\varphi_1 \wedge \ldots \wedge \varphi_n}$.

($\Leftarrow$) Supongamos que hay $\varphi_1, \ldots, \varphi_n \in \Gamma$ tales que $\overline{\bot} = \overline{\varphi}_1 \wedge \ldots \wedge \overline{\varphi}_n$. Entonces sabemos que $\vdash (\varphi_1 \wedge \ldots \wedge \varphi_n) \leftrightarrow \bot$, y en particular $\vdash (\varphi_1 \wedge \ldots \wedge \varphi_n) \to \bot$ (usando la regla $(\wedge E)$ o $(\leftrightarrow E)$ según consideremos a $\leftrightarrow$ como una definición o un nuevo conectivo, respectivamente). Entonces $\{\varphi_1, \ldots, \varphi_n\} \vdash \bot$, y en consecuencia $\Gamma \vdash \bot$. □

Otro resultado es el siguiente:

Lema 39. *Si Γ es cerrado por derivaciones entonces $\overline{\Gamma}$ es un filtro en $\overline{PROP}$.*

Demostración. Supongamos que Γ es cerrado por derivaciones. Para ver que $\overline{\Gamma}$ es un filtro, basta ver que

- Si $\overline{\varphi}, \overline{\psi} \in \overline{\Gamma}$ entonces $\overline{\varphi} \wedge \overline{\psi} \in \overline{\Gamma}$.
- $\overline{\Gamma}$ es creciente. Es decir, si $\overline{\varphi} \in \overline{\Gamma}$ y $\overline{\varphi} \preccurlyeq \overline{\chi}$ entonces $\overline{\chi} \in \overline{\Gamma}$.

Necesitaremos una cuentita auxiliar.

Afirmación. *Supongamos $\overline{\varphi} \in \overline{\Gamma}$. Si Γ es cerrado por derivaciones, entonces $\varphi \in \Gamma$.*

Prueba de la Afirmación. Si $\overline{\varphi} \in \overline{\Gamma}$, existe $\chi \in \Gamma$ tal que $\overline{\chi} = \overline{\varphi}$. Es decir, hay una derivación D con conclusión $\chi \leftrightarrow \varphi$ y todas sus hipótesis canceladas. Luego,

$$\dfrac{\dfrac{\begin{array}{c}\vdots\\ D\\ \vdots\end{array}}{\chi \leftrightarrow \varphi}\wedge E \quad \chi}{\varphi} \to E \quad \chi \to \varphi$$

es una derivación con única hipótesis no cancelada χ (que está en Γ) y conclusión φ, así que $\Gamma \vdash \varphi$ y como es cerrado por derivaciones, $\varphi \in \Gamma$. □

Supongamos ahora que $\overline{\varphi}, \overline{\psi} \in \overline{\Gamma}$. Por la Afirmación sabemos que $\varphi, \psi \in \Gamma$; entonces hay una derivación con hipótesis en Γ y conclusión $\varphi \wedge \psi$ (por la regla $(\wedge I)$). Entonces $\Gamma \vdash \varphi \wedge \psi$ y luego $\varphi \wedge \psi \in \Gamma$, con lo que probamos la primera parte

Para la segunda condición de filtro, supongamos $\overline{\varphi} \in \overline{\Gamma}$ y $\overline{\varphi} \preccurlyeq \overline{\chi}$. Por la Afirmación y puesto que $\approx$ preserva $\preccurlyeq$, puedo eliminar las barras y obtenemos $\varphi \in \Gamma$ y $\varphi \preccurlyeq \chi$, donde esta última equivale a $\vdash \varphi \to \chi$. Pero entonces $\{\varphi\} \vdash \chi$ y luego $\Gamma \vdash \chi$ pues φ pertenecía a Γ. Nuevamente, como Γ es cerrado por derivaciones, $\chi \in \Gamma$ y luego $\overline{\chi} \in \overline{\Gamma}$. □

La vuelta del lema no es cierta así como está, pero vale si se reemplaza la segunda condición por una más fuerte. Diremos que Γ es ***cerrado por*** $\approx$ si $\varphi \in \Gamma$ y $\varphi \approx \psi$ entonces $\psi \in \Gamma$.

Ejercicio 14. Probar que si $\overline{\Gamma}$ es un filtro y Γ es cerrado por $\approx$ entonces Γ es cerrada por derivaciones.

Corolario 40. *Si Γ es cerrado por derivaciones y consistente, entonces es un filtro propio.*

Demostración. Ejercicio (ayuda: un filtro es propio si no contiene al elemento mínimo). □

Como golpe de gracia, obtenemos

Lema 41. *Γ es consistente maximal implica $\overline{\Gamma}$ es un filtro primo.*

Demostración. Supongamos Γ es consistente maximal. Como Γ es cerrado por derivaciones y consistente, ya sabemos que es un filtro propio. Para ver que es primo, basta probar que para todo $\overline{\varphi}, \overline{\psi}$ en $\overline{PROP}$, $\overline{\varphi} \vee \overline{\psi} \in \overline{\Gamma}$ si y sólo si $\overline{\varphi} \in \overline{\Gamma}$ ó $\overline{\varphi} \in \overline{\Gamma}$. Pero esto último es inmediato por el Ejercicio 10 de la Sección 2.4. □

Teorema 42. *Suponga que Γ es cerrado por $\approx$. Luego Γ es consistente maximal si y sólo si $\overline{\Gamma}$ es un filtro primo.*

Demostración. Queda como ejercicio. □

3.4. Algunos Comentarios

El álgebra de Lindenbaum que definimos está basada exclusivamente en nociones sintácticas. Podemos definir otra álgebra usando la relación $\sqsubseteq$ dada por: $\varphi \sqsubseteq \psi$ si y sólo si $\models \varphi \to \psi$, que correspondería a las nociones semánticas. Se puede dar una nueva prueba de la completitud de la lógica proposicional usando estas dos álgebras, que resultan ser isomorfas, y en las que los filtros primos son la realidad subyacente a conjuntos consistentes maximales (por el lado sintáctico) y valuaciones (por el lado semántico). Estas ideas se generalizan a la lógica de primer orden, que incorpora los cuantificadores $\exists$ y $\forall$.

3.5. Ejercicios

1. Supongamos $\varphi \approx \psi$ y $\chi \approx \theta$. Entonces $\varphi \wedge \chi \approx \psi \wedge \theta$.

2. Encontrar Γ y φ tales que $\overline{\varphi} \in \overline{\Gamma}$ pero $\varphi \notin \Gamma$.

3. ¿Son los elementos de $\overline{At}$ átomos del álgebra de Boole $\overline{PROP}$?

4. *a*) Sea $h : \overline{PROP} \to \mathbf{2}$ un homomorfismo. Probar que la función $v : PROP \to \{0,1\}$ definida como $[\![\varphi]\!]_v := h(\overline{\varphi})$ es una valuación.

   b) Probar que toda valuación se obtiene de esa manera.

5. *a*) Sean $\varphi, \psi \in PROP$ tales que $\overline{\varphi} \preccurlyeq \overline{\psi}$ pero $\overline{\varphi} \neq \overline{\psi}$. Demostrar que si p es un átomo que no ocurre en φ ni en ψ, entonces $\nvdash \varphi \vee (p \wedge \psi) \to \varphi$ y $\nvdash \psi \to \varphi \vee (p \wedge \psi)$. (Ayuda: usar Completitud).

b) Probar que $\overline{PROP}$ es densa, es decir, si $\overline{\varphi} \preccurlyeq \overline{\psi}$ y $\overline{\varphi} \neq \overline{\psi}$ entonces existe $\overline{\chi} \in \overline{PROP}$ distinta de las anteriores tal que $\overline{\varphi} \preccurlyeq \overline{\chi} \preccurlyeq \overline{\psi}$.

c) Concluir que el álgebra de Boole $\overline{PROP}$ no tiene átomos, y por ende no es isomorfa a $\mathcal{P}(X)$ para ningún X.

4. Axiomatización[14]

Estudiaremos un par (de dos o más) de conceptos relacionados con la axiomatización de teorías (proposicionales, en nuestro caso).

Definición 43. Una ***teoría*** será un subconjunto de $PROP$.

La tarea de hombres y mujeres de ciencia en general consiste en analizar y organizar el conjunto "total" de afirmaciones sobre el Mundo (una pequeña fracción de ellas podría ser la que se halla antes del Ejemplo 3). Considerando a una asignación v como un "mundo posible", la teoría relacionada con ese mundo es el conjunto consistente maximal $\Gamma_v := \{\varphi : [\![\varphi]\!]_v = 1\}$. Para poder estudiar dicha teoría, conviene "simplificarla" para hacerla más manejable. Por ejemplo, sabemos que si φ y ψ están en Γ_v, entonces $\varphi \wedge \psi$ está; así que para "entender" a Γ_v tener a φ, ψ y a $\varphi \wedge \psi$ es redundante, ya que sabemos que una vez que tenemos a las dos primeras podemos deducir la tercera. Sería de interés encontrar un conjunto de proposiciones de las que se pueda deducir lo mismo que se puede deducir de Γ_v, pero que sea más resumido. Las siguientes definiciones capturan algunos de dichos conceptos.

Definición 44.

1. Γ es ***independiente*** si y sólo si para toda $\varphi \in \Gamma$, $\Gamma \setminus \{\varphi\} \nvdash \varphi$.

2. φ es ***indecidible*** para Γ si $\Gamma \nvdash \varphi$ ni $\Gamma \nvdash \neg\varphi$.

3. Sea Γ una teoría. Una teoría Δ es un conjunto de *axiomas para* (o que *es equivalente a*) Γ, si para toda $\varphi \in PROP$ se da [$\Gamma \vdash \varphi$ si y sólo si $\Delta \vdash \varphi$].

Proposición 45. *Si para toda $\varphi \in \Gamma$, φ es indecidible para $\Gamma \setminus \{\varphi\}$, entonces Γ es independiente.*

Demostración. Estamos diciendo en las hipótesis que para toda φ se dan $\Gamma \setminus \{\varphi\} \nvdash \varphi$ y $\Gamma \setminus \{\varphi\} \nvdash \neg\varphi$. En particular, podemos afirmar "para toda φ, $\Gamma \setminus \{\varphi\} \nvdash \varphi$", que es lo mismo que dice la Definición 44. □

Damos seguidamente un criterio para decidir si una proposición es indecidible para una teoría.

Lema 46. *Si hay asignaciones v_0, v_1 de Γ tales que $v_0(\varphi) = 0$, $v_1(\varphi) = 1$, entonces φ es indecidible para Γ.*

Demostración. Probamos la contrarrecíproca, es decir, supongamos que φ es decidible para Γ. Como primer caso, supongamos que $\Gamma \vdash \varphi$. Por la Corrección de la lógica proposicional, tenemos que $\Gamma \models \varphi$ y por ende toda asignación que valida Γ valúa φ en 1, así que no se puede dar el antecedente. En segundo caso, si $\Gamma \vdash \neg\varphi$, tenemos $\Gamma \models \neg\varphi$ y luego toda asignación v que valide Γ hará $[\![\neg\varphi]\!]_v = 1$, y por definición de valuación, $[\![\varphi]\!]_v = 0$, cosa que también contradice el antecedente. □

[14] Bonus Track.

Otra propiedad interesante del conjunto Γ_v es la siguiente consecuencia del Ejemplo 13 y el Lema 33: para toda φ, $\Gamma_v \vdash \varphi$ ó $\Gamma_v \vdash \neg\varphi$. Es decir, Γ_v "decide" cualquier proposición: una vez que supusimos Γ_v, la verdad o falsedad de cada φ (en términos de derivabilidad) queda determinada.

Definición 47. Una teoría Γ es ***completa*** si y sólo si para toda $\varphi \in PROP$, $\Gamma \vdash \varphi$ ó $\Gamma \vdash \neg\varphi$.

En la terminología de la Definición 44, una teoría es completa si no hay proposición indecidible para ella. Nuestros primeros ejemplos de conjuntos completos son los obvios.

Ejemplo 15.

1. $\{\bot\}$ es completo.

   Queda como ejercicio fácil (ver el Lema 27);

2. Si Γ es consistente maximal, entonces es completo.

   Una aplicación trivial del Lema 33.

Ejemplo 16. (Uno no tan obvio). El conjunto $\Pi := \{p_0, p_1, p_2, \dots\}$ es completo (y consistente). Sea $\varphi \in PROP$ y supongamos que $\Pi \nvdash \varphi$. Entonces por Completitud de la lógica proposicional, $\Pi \not\models \varphi$ y en consecuencia hay una asignación v que valida Π tal que $[\![\varphi]\!]_v = 0$, es decir, $[\![\neg\varphi]\!]_v = 1$. Por otro lado, si v_1 y v_2 son asignaciones que vaildan Π, entonces $v_1(p_i) = v_2(p_i) = 1$ para todo p_i, así que coinciden en At y en consecuencia son iguales, $v_1 = v_2$. Como hay una única asignación que valida Π, entonces decir "hay una asignación v que valida Π tal que $[\![\neg\varphi]\!]_v = 1$" es lo mismo que decir "para toda asignación v que valida Π, $[\![\neg\varphi]\!]_v = 1$" y esto último equivale a $\Pi \models \neg\varphi$. Usando completitud nuevamente, obtenemos $\Pi \vdash \neg\varphi$.

Las teorías completas y consistentes son muy especiales, ya que para ellas el Lema 30 vale en una forma mucho más fuerte.

Teorema 48. *Γ es consistente y completa si y sólo si existe un único Γ^* consistente maximal que lo contiene.*

Demostración. Probaremos que las respectivas negaciones son equivalentes.

($\Rightarrow$) Sabemos que hay por lo menos un consistente maximal que lo contiene. Supongamos que hubiera dos distintos Γ_1 y Γ_2. Sabemos que hay una $\varphi \in \Gamma_1 \setminus \Gamma_2$,

Ejercicio 15. ¿Por qué?

así que en particular $\varphi \notin \Gamma_2$, y por el Lema 33, $\neg\varphi \in \Gamma_2$. Si $\Gamma \vdash \varphi$, entonces $\Gamma \cup \{\neg\varphi\}$ sería inconsistente, y por ende Γ_2 lo sería (pues $\Gamma \cup \{\neg\varphi\} \subseteq \Gamma_2$), absurdo. Si $\Gamma \vdash \neg\varphi$, $\Gamma \cup \{\varphi\}$ sería inconsistente, y por ende Γ_1 también, otro absurdo. Luego Γ no puede ser completo.

($\Leftarrow$) Supongamos que Γ es incompleto y entonces hay una φ tal que $\Gamma \nvdash \varphi$ y $\Gamma \nvdash \neg\varphi$. Por las contrarrecíprocas al Lema 31, tenemos que $\Gamma \cup \{\neg\varphi\}$ y $\Gamma \cup \{\varphi\}$ deben ser consistentes, así que por el Lema 30 deben haber conjuntos consistentes maximales que contengan a cada uno. Pero no pueden ser iguales ya que en uno está $\neg\varphi$ y en el otro está φ (y ambas no pueden pertenecer simultáneamente a un conjunto consistente). □

Este teorema refleja en alguna medida una de las características que deseábamos cumpla nuestro "resumen" de las verdades de un mundo posible: si nuestro conjunto reducido de afirmaciones es completo, determina totalmente el conjunto total de afirmaciones.

Por último enunciamos sin prueba un teorema sobre conjuntos independientes de axiomas.

Teorema 49. *Toda teoría admite un conjunto independiente de axiomas.*

Ejemplo 17. Para el conjunto $\{(p_0 \wedge p_1), (p_3 \to p_1), (p_1 \vee p_2)\}$, un conjunto de axiomas independientes es $\{p_0, p_1\}$. Otro posible es $\{p_0 \wedge p_1\}$.

Resumiendo esta sección: para cada "mundo posible" (léase, asignación) su teoría es completa y se puede elegir un conjunto de axiomas sin redundancia (léase, independiente) para ella.

Ejemplo 18. El conjunto Π del Ejemplo 16 es (además de completo y consistente) independiente. Pues para cada n, la función $v : At \to \{0, 1\}$ definida de la siguiente manera

$$v(\varphi) := \begin{cases} 0 & \text{si } \varphi = p_n, \bot \\ 1 & \text{caso contrario,} \end{cases}$$

asignación v que valida $\Pi \setminus \{p_n\}$ y tal que $[\![p_n]\!]_v = 0$, así que $\Pi \setminus \{p_n\} \not\models p_n$, y por Corrección, $\Pi \setminus \{p_n\} \not\vdash p_n$.

4.1. Ejercicios

1. Mostrar que $p_1 \to p_2$ es indecidible para $\{p_1 \leftrightarrow p_0 \wedge \neg p_2, p_2 \to p_1\}$.
2. Hallar conjuntos independientes que sean equivalentes a los siguientes
   a) $\{p_0, p_1 \vee p_3, p_4\}$.
   b) $\{p_1,\ (p_1 \wedge p_2 \to p_3)\ ,\ (p_1 \to p_2)\}$.
   c) $\{p_1,\ (p_1 \wedge p_2 \to p_3)\ , p_3\}$.
   d) $\{(p_1 \to p_3), (p_1 \wedge p_2 \to \neg p_3), p_2\}$.
   e) $\{p_1, (p_1 \wedge p_2), (p_1 \wedge p_2 \wedge p_3), \dots\}$.
3. Hallar dos ejemplos de conjuntos independientes, consistentes y completos (Ayuda: usar el Ejemplo 16). Justificar.

A. Apéndice: Algunos ejercicios (difíciles) resueltos

Teorema 50 (Forma Normal (RAA))**.** *Para toda derivación D existe una derivación D' con las mismas hipótesis no canceladas y la misma conclusión, tal que D' tiene* a lo sumo una *aplicación de la regla (RAA), exactamente* al final.

Demostración. En primer lugar, para cualquier D que no incluya la regla (RAA) podemos tomar simplemente $D' := D$. Eliminado este caso trivial, vamos a probar por inducción en derivaciones que siempre podemos obtener una D' con *exactamente* una aplicación de dicha regla, al final de D'. Para aplicar el razonamiento inductivo, dividiremos en casos de acuerdo a cuál es la última regla de inferencia que se utiliza en D.

$\boxed{PROP}$ Si $D = \varphi$ tomo como D' la siguiente derivación: $\dfrac{\dfrac{\varphi \quad [\neg\varphi]_1}{\bot}\,\neg E}{\varphi}\,RAA_1$.

$\boxed{\wedge I}$ Supongamos que la derivación es de la forma $\dfrac{\begin{array}{cc}\vdots D_1 & \vdots D_2\\ \varphi & \psi\end{array}}{\varphi\wedge\psi}\wedge I$; por hipótesis inductiva sabemos que hay derivaciones en forma normal (RAA) de φ y ψ, es decir, derivaciones

$$\dfrac{\begin{array}{c}[\neg\varphi]\\ \vdots D_3\\ \bot\end{array}}{\varphi}RAA \qquad \dfrac{\begin{array}{c}[\neg\psi]\\ \vdots D_4\\ \bot\end{array}}{\psi}RAA$$

que satisfacen la conclusión del teorema. Es decir, D_3 y D_4 no utilizan la regla (RAA). Usaremos estas derivaciones para conseguir la que buscamos.

Comenzaremos reemplazando cada ocurrencia de $\neg\varphi$ como una hipótesis no cancelada de D_3 por la siguiente derivación:

$$\dfrac{\dfrac{[\varphi]_1\psi}{\varphi\wedge\psi}\wedge I \qquad \neg(\varphi\wedge\psi)}{\dfrac{\bot}{\neg\varphi}\neg I_1}\neg E$$

De esta manera obtenemos una derivación:

$$\dfrac{\dfrac{[\varphi]_1\psi}{\varphi\wedge\psi}\wedge I \qquad \neg(\varphi\wedge\psi)}{\dfrac{\bot}{\neg\varphi}\neg I_1}\neg E \qquad \begin{array}{c}\vdots D_3\\ \bot\end{array}$$

y luego

$$\dfrac{\dfrac{[\varphi]_1[\psi]_2}{\varphi\wedge\psi}\wedge I \qquad \neg(\varphi\wedge\psi)}{\dfrac{\bot}{\neg\varphi}\neg I_1}\neg E \qquad \begin{array}{c}\vdots D_3\\ \bot\end{array}$$

Reemplazamos ahora cada ocurrencia de $\neg\psi$ como una hipótesis no cancelada de D_4 por la segunda derivación y obtenemos una derivación de $\varphi\wedge\psi$ que tiene una única aplicación de (RAA), exactamente al final:

$$\dfrac{\dfrac{\dfrac{\dfrac{[\varphi]_1[\psi]_2}{\varphi\wedge\psi}\wedge I \qquad [\neg(\varphi\wedge\psi)]_3}{\dfrac{\bot}{\neg\varphi}\neg I_1}\neg E \qquad \begin{array}{c}\vdots D_3\\ \bot\end{array}}{\dfrac{\bot}{\neg\psi}\neg I_2} \qquad \begin{array}{c}\vdots D_4\\ \bot\end{array}}{\varphi\wedge\psi}RAA_3$$

$\boxed{\wedge E}$ Hacemos el mismo procedimiento; el esquema general que se obtiene es el siguiente:

$$\dfrac{\dfrac{\dfrac{[\varphi \wedge \psi]_1}{\varphi}\wedge E \qquad [\neg\varphi]_2}{\bot}\neg E}{\neg(\varphi\wedge\psi)}\neg I_1 \\ \vdots \\ \dfrac{\bot}{\varphi} RAA_2$$

$\boxed{\to I}$ Es análogo a los anteriores, con una salvedad: partimos de una derivación de la forma $\dfrac{\begin{array}{c}[\varphi]\\ \vdots D_1\\ \psi\end{array}}{\varphi\to\psi}\to I$ y por hipótesis inductiva sabemos que hay una derivación en forma normal (RAA) de ψ *con la hipótesis adicional* φ:

$$\dfrac{\begin{array}{cc}[\neg\psi] & \varphi\\ & \vdots D\\ & \bot\end{array}}{\psi} RAA$$

Usamos esta D.

$$\dfrac{\dfrac{\dfrac{\dfrac{[\psi]_1}{\varphi\to\psi}\to I \qquad [\neg(\varphi\to\psi)]_3}{\bot}\neg E}{\neg\psi}\neg I_1 \qquad [\varphi]_2 \\ \vdots D \\ \dfrac{\bot}{\psi}\bot}{\varphi\to\psi}\to I_2 \qquad [\neg(\varphi\to\psi)]_3}{\bot}\neg E \\ \dfrac{\bot}{\varphi\to\psi} RAA_3$$

Los otros casos quedan como ejercicio (ahora fácil). □

Ejercicio. Probar $\vdash \neg(\neg\varphi\wedge\neg\psi) \to \varphi\vee\psi$.

Demostración.

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{[\neg\varphi]_1 \quad [\neg\psi]_2}{\neg\varphi \wedge \neg\psi}\wedge I \qquad [\neg(\neg\varphi\wedge\neg\psi)]_4}{\bot}\neg E}{\varphi}RAA_1}{\varphi\vee\psi}\vee I \qquad [\neg(\varphi\vee\psi)]_3}{\bot}\neg E}{\psi}RAA_2}{\varphi\vee\psi}\vee I \qquad [\neg(\varphi\vee\psi)]_3}{\bot}\neg E}{\varphi\vee\psi}RAA_3}{\neg(\neg\varphi\wedge\neg\psi) \;\to\; \varphi\vee\psi}\to I_4$$

□

Ejercicio. Probar $\vdash \varphi\vee\psi \;\leftrightarrow\; ((\varphi\to\psi)\to\psi)$.

Demostración. La derivación buscada es

$$\dfrac{\begin{array}{cc}[\varphi\vee\psi] & [(\varphi\to\psi)\to\psi] \\ \vdots\, D_1 & \vdots\, D_2 \\ (\varphi\to\psi)\to\psi & \varphi\vee\psi\end{array}}{\varphi\vee\psi \;\leftrightarrow\; ((\varphi\to\psi)\to\psi)}\leftrightarrow I$$

donde D_1 y D_2 vienen dadas por:

$$D_1 \;:=\; \dfrac{\varphi\vee\psi \qquad \dfrac{\dfrac{[\varphi]_2[\varphi\to\psi]_1}{\psi}\to E \qquad [\psi]_2}{\psi}\vee E_2}{(\varphi\to\psi)\to\psi}\to I_1$$

$$D_2 \;:=\; \dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{[\varphi]_3}{\varphi\vee\psi}\vee I \qquad [\neg(\varphi\vee\psi)]_4}{\bot}\neg E}{\psi}\bot}{\varphi\to\psi}\to I_3 \qquad (\varphi\to\psi)\to\psi}{\psi}\to E}{\varphi\vee\psi}\vee I \qquad [\neg(\varphi\vee\psi)]_4}{\bot}\neg E}{\varphi\vee\psi}RAA_4$$

□

Ejercicio. Probar $\{(\varphi \vee \psi) \wedge (\varphi \vee \theta)\} \vdash \varphi \vee (\psi \wedge \theta)$.

Demostración.

$$\dfrac{\dfrac{(\varphi \vee \psi) \wedge (\varphi \vee \theta)}{\varphi \vee \psi} \wedge E \quad \dfrac{[\varphi]_1}{\varphi \vee (\psi \wedge \theta)} \vee I \quad \dfrac{\dfrac{(\varphi \vee \psi) \wedge (\varphi \vee \theta)}{\varphi \vee \theta} \wedge E \quad \dfrac{[\varphi]_2}{\varphi \vee (\psi \wedge \theta)} \vee I \quad \dfrac{\dfrac{[\psi]_1 [\theta]_2}{\psi \wedge \theta} \wedge I}{\varphi \vee (\psi \wedge \theta)} \vee I}{\varphi \vee (\psi \wedge \theta)} \vee E_2}{\varphi \vee (\psi \wedge \theta)} \vee E_1$$

□

Índice alfabético

Introducción a la Lógica y la Computación

Parte III: Lenguajes y Autómatas

Autores: Alejandro Tiraboschi y colaboradores

Contenidos

1 Introducción

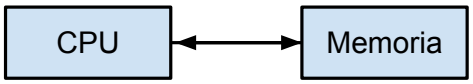
En esta parte de la materia vamos a empezar a estudiar *modelos de computación*. Es decir, modelos teóricos que aproximan el comportamiento de una computadora. Este tema se verá en más detalle en otras materias más avanzadas, cuando se estudien las *Máquinas de Turing* que nos servirán para definir exactamente que significa *computar* y cuales son sus límites.

En esta materia estudiaremos un tipo de máquinas más simples que las máquinas de Turing que se llaman *autómatas*, y veremos que existe una relación muy fuerte entre modelos de computación y lenguajes formales.

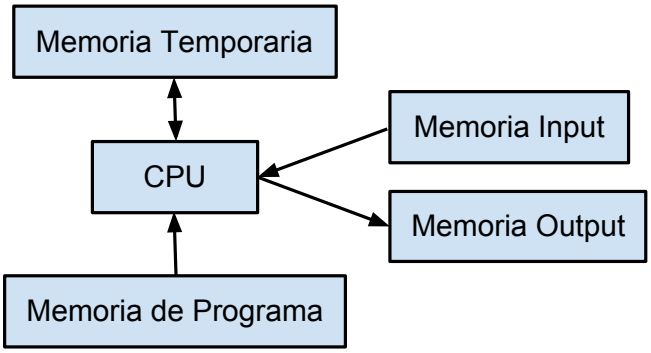
Una computadora que resuelve un problema mediante algún algoritmo puede verse en forma abstracta como "hablando el lenguaje" $\{(\text{input}_1,\text{output}_1), \ldots, (\text{input}_n,\text{output}_n), \ldots\}$. Notar que podemos pensar que este lenguaje es un conjunto de cadenas $\{\text{input}_1\#\text{output}_1, \ldots, \text{input}_n\#\text{output}_n, \ldots\}$, donde

es algún símbolo que no apareza en los inputs o los outputs. Por ejemplo la funcion $f(x) = x^3$ puede asociarse con el conjunto de cadenas {0#0, 1#1, 2#8, 3#9, ... }. Es por esto que una forma de estudiar las cosas que una computadora puede realmente computar, es investigando que lenguajes que distintos tipos de máquinas pueden producir.

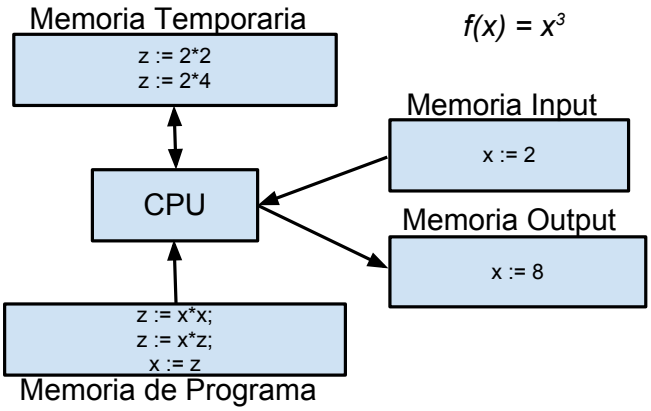
Miremos un poco mas en detalle a que nos referimos cuando decimos 'máquinas'. De forma muy abstracta, podemos representar una computadora simplemente con el siguiente diagrama:



donde sólo diferenciamos una unidad con capacidad de procesamiento (el CPU) de una unidad con capacidad de almacenamiento (la memoria). Nos va a interesar particularmente distinguir distintos tipos de memoria:



No nos interesa, en este momento, definir más precisamente cuales son las diferencias entre los distintos tipos de memoria, y nos alcanza con el siguiente ejemplo:



Ahora sí, vamos a considerar que las 'máquinas' que vamos a estudiar siempre tienen memoria input, memoria output y memoria de programa (esto simplemente quiere decir que toman un input, producen un output y siguen algún tipo de programa), pero difieren en el tipo de *memoria temporaria* que pueden tener:

- **Autómatas de Estados Finitos** (Finite State Automata): no tienen ningún tipo de memoria temporaria.

- **Autómatas con Pila** (Pushdown Automata): tienen una pila como memoria temporaria.
- **Máquinas de Turing** (Turing Machines): tienen memoria de lectura y escritura de acceso aleatoria.

Cada tipo de máquina tiene distinto poder de cómputo, que se va incrementando a medida que el tipo de memoria es menos restrictivo. Como dijimos al principio, en esta materia no llegaremos a estudiar las Máquinas de Turing, pero sí veremos distinto stipos de Autómatas de Estados Finitos y con Pila y estudiaremos sus propiedades.

2 Autómatas finitos y expresiones regulares

2.1 Cadenas, alfabetos y lenguajes

Un “símbolo" es una entidad abstracta que no definiremos formalmente, así como en geometría no definimos los “puntos" y las “líneas". Letras y dígitos son ejemplos de símbolos frecuentemente usados. Una *cadena* o *palabra* es una secuencia finita de símbolos yuxtapuestos. Por ejemplo a, b y c son símbolos y *abbba*, *abcc* y *bcccc* son cadenas. La *longitud* de una cadena w, que denotaremos $\mid w \mid$, es el número de símbolos que componen la cadena. Por ejemplo la cadena *aabc* tiene longitud 4, o en símbolos, $\mid w \mid = 4$. La *cadena vacía*, que denotaremos ε, es la cadena que tiene cero símbolos, luego $\mid \varepsilon \mid = 0$. Es importante notar que, por ejemplo, la cadena *aba* es igual a la cadena *ab*ε*a* o $\varepsilon\varepsilon$*aba* pues ε es la cadena vacía y no es símbolo.

Una *subcadena* de una cadena es una secuencia de símbolos incluida en la cadena, por ejemplo *ab* y ε son subcadenas de *cccabccc*, mientras que *cb* no lo es. Un *prefijo* de una cadena w, es una subcadena de w que comienza con el primer símbolo de w. Un *sufijo* de una cadena w, es una subcadena de w que termina con el último símbolo de w. Por ejemplo la cadena *abc* tiene como prefijos a las cadenas ε, a, *ab* y *abc*, y como sufijos a ε, c, *bc* y *abc*.

La *concatenación* de dos cadenas w y x, que se denota wx, es la cadena formada escribiendo la primera cadena seguida de la segunda, sin espacios en el medio. Por ejemplo, la concatenación de *aaa* y *bca* es *aaabca*. Observemos que la cadena vacía concatenada con otra cadena no la modifica, es decir, $\varepsilon w = w\varepsilon = w$.

Un *alfabeto* es un conjunto finito de símbolos. Un *lenguaje (formal)* es un conjunto de cadenas de símbolos de un alfabeto dado. El conjunto vacío, $\emptyset$, y el conjunto $\{\varepsilon\}$, es decir el conjunto que sólo tiene la cadena vacía, son lenguajes. Observar que estos dos lenguajes son distintos, el primero no tiene elementos, mientras que el segundo tiene un elemento.

Ejemplo 2.1. El conjunto de *palindromes* o *capicúas* sobre el alfabeto $\{0,1\}$ es un lenguaje infinito. Algunos elementos de este lenguaje son ε, 0, 1, 00, 11, 000, 010, 101, 111.

Dado un alfabeto Σ, denotaremos con Σ^* al lenguaje formado por todas las cadenas sobre el alfabeto.

Ejemplo 2.2. Sea el alfabeto $\Sigma = \{a\}$, entonces $\Sigma^* = \{\varepsilon, a, aa, aaa, aaaa, \ldots\}$. Si $\Sigma = \{0,1\}$, entonces $\Sigma^* = \{\varepsilon, 0, 1, 00, 01, 10, 11, 000, \ldots\}$

2.2 Sistemas de estados finitos

Los autómatas finitos son modelos matemáticos de sistemas, con entradas y salidas discretas. El sistema puede estar en una, entre un número finito, de configuraciones internas o “estados". El estado

del sistema es la única información que se debe tomar en cuenta para pasar a otro estado con una entrada o input dado. Un buen ejemplo, que desarrollaremos más adelante, es el mecanismo de control de un ascensor. El mecanismo no recuerda los pedidos previos, si no solamente el piso en que está el ascensor, la dirección de movimiento (arriba, abajo o quieto) y la colección de pedidos no satisfechos.

En ciencias de la computación es posible encontrar numerosos ejemplos de sistemas de estados finitos, y la teoría de autómatas es una herramienta útil para dichos sistemas. Ciertos programas tales como editores de texto y analizadores léxicos, encontrados en la mayoría de los compiladores, son a menudo designados como sistemas de estados finitos. Por ejemplo, un analizador léxico recorre los símbolos de un programa de computación para localizar las cadenas de caracteres correspondientes a identificadores, constantes numéricas, palabras reservadas, etc.

Antes de definir formalmente un sistema de estados finito consideremos el siguiente ejemplo.

Ejemplo 2.3. Un hombre con un lobo, una cabra y un repollo está en la ribera izquierda de un río. Hay un bote que puede transportar solamente al hombre y a alguno de los otros tres. El hombre quiere transportar al lobo, la cabra y el repollo al otro lado del río, sin embargo si deja solos al lobo y la cabra, el lobo devorará la cabra, y si deja solos a la cabra y el repollo, la cabra devorará al repollo. >Es posible cruzar el río sin que el lobo coma a la cabra y sin que la cabra coma el repollo?

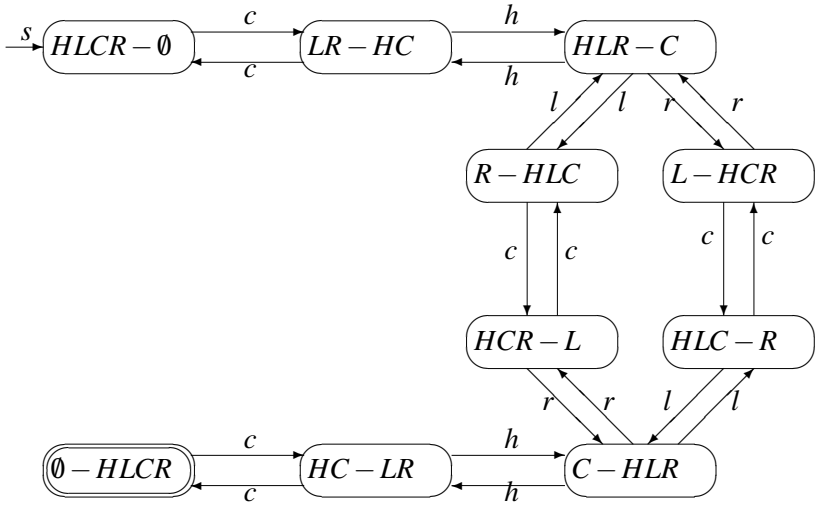


Figura 1: Diagrama de transición para el problema del hombre, el lobo, la cabra y el repollo.

El problema se modela observando que la información pertinente es la de quien ocupa cada ribera del río después de un cruce, esos serán los posibles estados, 16 en total. Un estado se denotará, por ejemplo, $LR-HC$ que significará que en la ribera izquierda están el lobo y el repollo y en la ribera derecha están el hombre y la cabra. Algunos de los 16 estados posibles, tal como $HL-CR$, son fatales y por lo tanto nunca debemos “entrar" a ellos.

Los “inputs" del sistema son las acciones que el hombre toma. Él puede cruzar sólo (input h), con el lobo (input l), con la cabra (input c) o con el repollo (input r). El estado inicial es $HLCR-\emptyset$ y el final es $\emptyset-HLCR$. El diagrama de transición (de todos los estados posibles no fatales y sus inputs) se muestra en la Fig. 1.

Hay dos soluciones cortas a este problema, tal como se deduce del gráfico: son caminos que salen del estado inicial (el círculo marcado con una flecha) y llegan al estado final (en doble círculo). Hay también un número infinito de soluciones, que, excepto las dos cortas tienen ciclos inútiles. Dado un sistema de estados finitos podemos definir el siguiente lenguaje: el conjunto de todas las cadenas de inputs que llevan del estado inicial al estado final. Por ejemplo *chlcrhc*, *chrclhc*, *chhhlcrhc* son cadenas de este lenguaje (las dos primeras corresponden a las soluciones cortas).

Finalmente debemos observar que este ejemplo tiene dos características particulares que en general los sistemas de estados finitos no tienen. La primera es que hay un solo estado final, cosa que en general no ocurre y la segunda es que por cada transición hay una transición inversa.

2.3 Autómatas finitos

Definición 2.1. Un *autómata finito determinístico (DFA)* es una 5-upla $(Q, \Sigma, \delta, q_0, F)$ que consiste de

1. un conjunto finito $Q = \{q_1, q_2, \ldots, q_m\}$ de *estados*,
2. un conjunto finito $\Sigma = \{a_1, a_2, \ldots, a_n\}$ de *símbolos de entrada* o *input*,
3. un conjunto de *reglas de transición* que transforma un estado con un input dado en otro estado, independientemente de lo que haya sucedido en las lecturas anteriores. Formalmente las reglas de transición se definen con una función $\delta : Q \times \Sigma \to Q$.
4. Además habrá un *estado inicial* q_0 y un conjunto de estados F, que llamaremos *estados finales*.

Muchas veces representaremos las reglas de transición por una tabla a doble entrada: buscando el cruce entre la entrada de un estado y la entrada de un input sabemos como cambia el estado.

Ejemplo 2.4. Sea M un autómata finito determinístico con estados $\{q_0, q_1, q_2\}$, símbolos de entrada $\Sigma = \{a, b\}$ y las siguientes reglas de transición: el estado q_0 se transforma en q_1 si el input es a y en q_0 si el input es b; el estado q_1 se transforma en q_1 si el input es a y en q_2 si el input es b; finalmente, el estado q_2 se transforma en q_2 si el input es a y en q_0 si el input es b. Es decir $\delta : Q \times \Sigma \to Q$ está definida por:

$$\begin{array}{llll} \delta(q_0, a) & = q_1, & \delta(q_0, b) & = q_0 \\ \delta(q_1, a) & = q_1, & \delta(q_1, b) & = q_2 \\ \delta(q_2, a) & = q_2, & \delta(q_2, b) & = q_0 \end{array}$$

que puede ser representada (en forma más compacta) con la siguiente tabla.

	a	b
q_0	q_1	q_0
q_1	q_1	q_2
q_2	q_2	q_0

Hablando en forma estricta el Ejemplo 2.3 no es un autómata en el sentido de la Definición 2.1 pues hay estados que no aceptan ciertos inputs. Sin embargo, en la sección siguiente extenderemos el concepto de autómata, con lo cual este tipo de situaciones será admisible.

Ejemplo 2.5. En el Ejemplo 2.3 los estados son:

$$HCLR-,\ LR-HC,\ etc.$$

el estado inicial es $HCLR-$ y el estado final es $-HCLR$. los símbolos de input son h, c, l, r y las reglas de transición son: $HCLR-$ con input c va a $LR-HC$, etc.

Como se observa en los Ejemplos 2.3 y 2.5 hay una relación estrecha entre un autómata finito determinístico y “grafos". A cada autómata finito determinístico se le puede asignar un dibujo, que llamaremos grafo, de la siguiente manera: a cada estado le corresponde un círculo con el nombre del estado dentro, el estado inicial está distinguido con una flecha apuntando al círculo y los estados finales se dibujan con doble círculo. En lo sucesivo llamaremos a estos círculos y doble círculos *nodos*. Si hay una transición del estado q al estado p con input a, entonces se dibuja una *flecha* que va del nodo correspondiente a q al nodo correspondiente a p, además a esta flecha le ponemos la etiqueta a. A este tipo de grafo lo llamaremos un *diagrama de transición*.

Es claro además, que si tenemos un diagrama de transición hay un autómata finito asociado a él; y por lo tanto en el futuro no distinguiremos autómatas finitos y diagramas de transición.

Ejemplo 2.6. Sea M un autómata con estados $\{q_0, q_1, q_2, q_3\}$, símbolos de input $\{0, 1\}$, estado inicial q_0 y estado final q_0 también. Las reglas de transición están dadas por la Tabla 1.

	0	1
q_0	q_2	q_1
q_1	q_3	q_0
q_2	q_0	q_3
q_3	q_1	q_2

Tabla 1:

El diagrama de transición de M será el dado por la Fig. 2

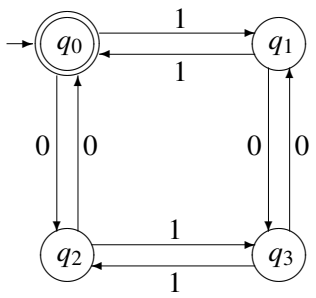


Figura 2:

Ejemplo 2.7. El siguiente autómata sirve para modelar el mecanismo de control de un ascensor automático, en este caso de un edificio con pisos 0, 1 y 2.

Un estado debe describir el piso en que se encuentra el ascensor (piso 0, 1 o 2) , si está detenido (p) o va hacia arriba (u) o abajo (d), y las demandas no satisfechas. Por ejemplo podríamos representar el estado: el ascensor está en el piso 1, va hacia arriba y debe ir al piso 0, por la upla $(1, u, V, F, F)$, las últimas tres componentes de la upla denotan los tres pisos posibles, una V denota que debe ir a ese piso, mientras que F denota que no debe ir. Los posibles inputs son los llamados que hace la gente a ciertos pisos (no es necesario distinguir si el llamado se hace desde dentro del ascensor o desde un piso) por lo tanto los inputs posibles son 0, 1 y 2. Además hay otro input que no viene dado por los usuarios si no por el movimiento del ascensor: cuando, por ejemplo, el ascensor tiene el piso 1 como

demanda insatisfecha y llega a este piso, entonces la demanda deja de ser insatisfecha. Denotemos este input con m. Notemos que hay ciertos estados que no son posibles, como por ejemplo $(1,u,F,V,F)$, pues si estamos en cierto piso no tiene sentido pedir que vaya al mismo piso. Tampoco tienen sentido, por ejemplo, $(1,u,F,F,F)$ o $(1,u,V,F,F)$, pues si no hay demandas insatisfechas arriba del piso en que está el ascensor, este no puede ir hacia arriba. Otro caso no admisible es un estado en que el ascensor está detenido y hay demandas insatisfechas, pues si el ascensor está detenido y entra una demanda, el ascensor comienza a moverse en el sentido de la demanda. Los estados finales serán aquellos en que el ascensor está detenido. Consideremos que el estado inicial es $(0,p,F,F,F)$. La Tabla 2 da los posibles cambios de estados.

	0	1	2	m
$(0,p,F,F,F)$	$(0,p,F,F,F)$	$(0,u,F,V,F)$	$(0,u,F,F,V)$	$(0,p,F,F,F)$
$(0,u,F,V,F)$	$(0,u,F,V,F)$	$(0,u,F,V,F)$	$(0,u,F,V,V)$	$(1,p,F,F,F)$
$(0,u,F,F,V)$	$(0,u,F,F,V)$	$(0,u,F,V,V)$	$(0,u,F,F,V)$	$(1,u,F,F,V)$
$(0,u,F,V,V)$	$(0,u,F,V,V)$	$(0,u,F,V,V)$	$(0,u,F,V,V)$	$(1,u,F,F,V)$
$(1,p,F,F,F)$	$(1,d,V,F,F)$	$(1,p,F,F,F)$	$(1,u,F,F,V)$	$(1,p,F,F,F)$
$(1,u,F,F,V)$	$(1,u,V,F,V)$	$(1,u,F,F,V)$	$(1,u,F,F,V)$	$(2,p,F,F,F)$
$(1,d,V,F,F)$	$(1,d,V,F,F)$	$(1,d,V,F,F)$	$(1,d,V,F,V)$	$(0,p,F,F,F)$
$(1,u,V,F,V)$	$(1,u,V,F,V)$	$(1,u,V,F,V)$	$(1,u,V,F,V)$	$(2,d,V,F,F)$
$(1,d,V,F,V)$	$(1,d,V,F,V)$	$(1,d,V,F,V)$	$(1,d,V,F,V)$	$(0,u,F,F,V)$
$(2,p,F,F,F)$	$(2,d,V,F,F)$	$(2,d,F,V,F)$	$(2,p,F,F,F)$	$(2,p,F,F,F)$
$(2,d,V,F,F)$	$(2,d,V,F,F)$	$(2,d,V,V,F)$	$(2,d,V,F,F)$	$(1,d,V,F,F)$
$(2,d,F,V,F)$	$(2,d,V,V,F)$	$(2,d,F,V,F)$	$(2,d,F,V,F)$	$(1,p,F,F,F)$
$(2,d,V,V,F)$	$(2,d,V,V,F)$	$(2,d,V,V,F)$	$(2,d,V,V,F)$	$(1,d,V,F,F)$

Tabla 2:

Nos interesa ahora estudiar las sucesiones de símbolos de input de un autómata dado que nos lleven del estado inicial a uno final. Más precisamente

Definición 2.2. Sea $M=(Q,\Sigma,\delta,q_0,F)$ un autómata finito determinístico. Una *cadena en M* es un elemento de Σ^*. Sean p y q estados de M y $\alpha=a_0a_1\dots a_n$ una cadena en M, diremos que α *transforma* q en p (un estado de M) si partiendo del estado q y aplicando sucesivamente los inputs a_0, a_1, hasta a_n, obtenemos el estado p. También definiremos que ε transforma un estado en si mismo. Finalmente, diremos que una cadena α *es aceptada por M*, si α transforma el estado inicial en uno final.

A nivel de diagramas de transición, podemos mirar el diagrama como un conjunto de sitios o ciudades (los nodos) con rutas que conectan a algunos de ellos (las flechas), entonces una cadena $\alpha=a_0a_1\dots a_n$, que también llamaremos *recorrido*, es aceptada si partiendo del sitio q_0 y yendo por las rutas $a_0,a_1,\dots,a_n$ llegamos a un sitio final.

Observación 2.1. Denotaremos el hecho de que un símbolo de input a transforma un estado q en p, por

$$q \xrightarrow{a} p.$$

Sea $\alpha=a_0a_1\dots a_n$ una cadena que transforma un estado q en p, entonces se denota

$$q \xrightarrow{a_0} q_1 \xrightarrow{a_1} \cdots \xrightarrow{a_{n-1}} q_{n-1} \xrightarrow{a_n} p \quad \text{o bien} \quad q \xrightarrow{\alpha}$$

Observemos que $q \xrightarrow{\varepsilon} q$, para todo estado q.

Notemos que esta notación tiene la siguiente propiedad : si $\alpha = \beta\gamma$, donde α, β y γ son cadenas, entonces

$$q \xrightarrow{\alpha} p \text{ si y sólo si existe } r \in Q \text{ tal que } q \xrightarrow{\beta} r \xrightarrow{\gamma} p. \tag{1}$$

Definición 2.3. Sea M un autómata finito determinístico, entonces el *lenguaje aceptado por M*, denotado $L(M)$, es el conjunto de cadenas aceptadas por el autómata. Si M y M' son dos autómatas finitos diremos que son *equivalentes* si $L(M) = L(M')$.

En la notación explicada más arriba si $M = (Q, \Sigma, \delta, q_0, F)$ es un autómata finito determinístico, entonces

$$L(M) = \{\alpha \in \Sigma^* \mid \text{ existe } p \in F \text{ tal que } q_0 \xrightarrow{\alpha} p\}.$$

Ejemplo 2.8. El lenguaje aceptado por el autómata del Ejemplo 2.6 es $L(M) =$ el conjunto de cadenas que tienen un número par de 0's y un número par de 1's.

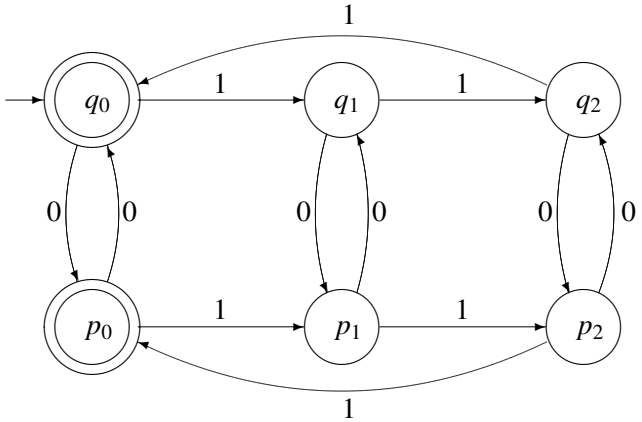
Demostración. Si observamos el diagrama de transición correspondiente notamos que ir de izquierda a derecha o de derecha a izquierda significa aplicar el input 1. Ir de arriba a abajo o de abajo a arriba significa aplicar el input 0. Ahora bien, como q_0 es el estado inicial y final, una palabra aceptada va "recorriendo" el diagrama de tal forma que cuando termina subió tantas veces como bajó y fue a la izquierda tantas veces como fue a la derecha. Es decir que esta palabra tiene un número par de 0's y un número par de 1's.

Por otro lado, si una palabra tiene un número par de 0's y un número par de 1's es claro, por las mismas razones expuestas arriba, que termina su recorrido en q_0.

Observemos que ε es una cadena aceptada, debido a que q_0 es estado final y que $q_0 \xrightarrow{\varepsilon} q_0$. Claramente, la cadena ε tiene un número par de ceros y unos, pues tiene 0 ceros y 0 unos.

□

Ejemplo 2.9. El lenguaje reconocido por el siguiente autómata es el de las cadenas de 0's y 1's con tres o un múltiplo de tres 1's. Por ejemplo las cadenas 00, 01011, 110011011, pertenecen a este lenguaje.



Demostración. Como en el anterior ejemplo la idea es geométrica: observemos que para que una cadena termine en q_0 o p_0, debe haber ido a un estado con subíndice 1, luego a otro con subíndice 2 y luego volver a un estado con subíndice 0, y esto se puede hacer un número arbitrario de veces. Ahora

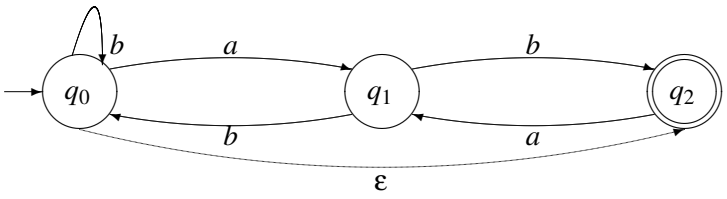
bien, cada vez que se hace esto estamos usando tres 1's de la cadena. Además, el número de ceros es arbitrario. Luego una cadena aceptada por el autómata tiene la propiedad requerida. La recíproca es similar. □

2.4 Autómatas no determinísticos

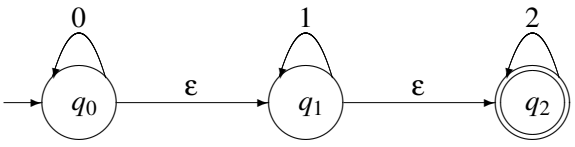
Ahora introduciremos la noción de autómata finito no determinístico. No es difícil ver que un conjunto aceptado por un autómata no determinístico también es aceptado por uno determinístico y viceversa. Sin embargo, el autómata finito no determinístico resultará útil para probar ciertos resultados y para ciertas aplicaciones. Daremos una definición no formal de este tipo de autómatas. El lector interesado en la definición formal puede consultar la subsección 2.4.1.

Modifiquemos la definición de autómata finito permitiendo cero, una, o más transiciones de un estado con el mismo símbolo de input. Es decir de un estado pueden partir un número arbitrario de flechas cada una etiquetada con cualquier símbolo de input. También permitamos transiciones sin inputs. Esta nueva definición nos da los *autómatas finitos no determinísticos (NFA) con* ε*-movimientos*. A nivel de diagramas de transición los grafos que representan estos autómatas serán de la siguiente forma: de cada nodo puede partir ninguna, una o varias flechas con la misma etiqueta de input. Además puede haber flechas que no se correspondan a ningún input, los ε-movimientos. A estas flechas les pondremos la etiqueta ε. Como en el caso de los autómatas determinísticos los NFA tienen un estado inicial y estados finales, que se denotarán en el diagrama de transición de la misma forma que en el caso determinístico

Ejemplo 2.10. El siguiente es el diagrama de un autómata finito no determinístico con ε-movimientos: los estados son q_0, q_1 y q_2. Los símbolos de input son a y b. El estado inicial es q_0 y el estado final es q_2.



Ejemplo 2.11. En este NFA los estados son q_0, q_1 y q_2. Los símbolos de input son 0, 1 y 2. El estado inicial es q_0 y el estado final es q_2.



El nombre "no determinístico" viene del hecho que si un estado recibe un input, entonces no está determinado cual es el próximo estado, si no que hay ciertos estados posibles. Los ε-movimientos reflejan la posibilidad de cambio de estado sin que necesariamente haya un input. Observar también, que existe la posibilidad que dado un estado no salga ninguna flecha de él. Esto denotará que en este

estado el autómata “aborta". Es claro que no es fácil imaginarse una “máquina" que se comporte de esta manera, sin embargo el concepto de NFA con ε-movimientos resulta muy útil en la teoría de lenguajes.

Las tablas que describirán las reglas de transición de los NFA con ε-mov serán similares a las tablas que hemos hecho para los autómatas finitos determinísticos. Las diferencias son que cuando aplicamos un input a un estado obtenemos un conjunto de estados (hacia donde van las flechas) o $\emptyset$ en el caso que no salga ninguna flecha. Además debemos agregar una entrada ε en la tabla para reflejar los posibles cambios de estado que producen los ε-movimientos.

Ejemplo 2.12. La tabla de transición correspondiente al NFA con ε-mov del ejemplo 2.10 es dada por la siguiente tabla:

	a	b	ε
q_0	q_1	q_0	q_2
q_1	$\emptyset$	q_0, q_2	$\emptyset$
q_2	q_1	$\emptyset$	$\emptyset$

Ejemplo 2.13. La tabla de transición correspondiente al NFA con ε-mov del ejemplo 2.11 es dada por la siguiente tabla:

	0	1	2	ε
q_0	q_0	$\emptyset$	$\emptyset$	q_1
q_1	$\emptyset$	q_1	$\emptyset$	q_2
q_2	$\emptyset$	$\emptyset$	q_2	$\emptyset$

Veamos ahora el lenguaje asociado a un NFA con ε-movimientos.

Definición 2.4. Sea M un NFA con ε-movimientos con símbolos de input Σ. Si $\alpha \in \Sigma^*$, diremos que α *transforma* p en q y lo denotaremos $p \xrightarrow{\alpha} q$, si existen $a_0, a_1, \ldots, a_n \in \Sigma$ tal que $\alpha = a_1 a_2 \ldots a_n$ y tal que existe un recorrido por flechas con etiquetas $a_0, a_1, \ldots, a_n$ que va del estado p al estado q. En el recorrido se pueden intercalar arbitrariamente flechas con etiqueta ε.

Es conveniente también definir que ε transforma todo estado en sí mismo, es decir $q \xrightarrow{\varepsilon} q$.

Ejemplo 2.14. En el autómata del ejemplo 2.11 tenemos, por ejemplo, que $q_0 \xrightarrow{01} q_1$, pues hay un camino que lleva q_0 a q_0 por 0, q_0 a q_1 por ε y q_1 a q_1 por 1.

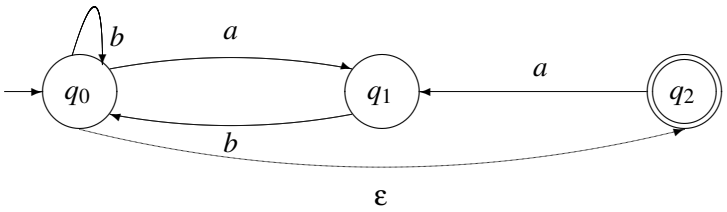
Observación 2.2. Si p y q estados, a símbolo de input y $p \xrightarrow{a} q$, no podemos asegurar que de p podemos pasar directamente a q por una flecha con etiqueta a. Por definición $p \xrightarrow{a} q$ significa que podemos llegar de p a q usando ε-movimientos además de una transición por a. Observando el autómata del ejemplo 2.11 vemos que $q_0 \xrightarrow{0} q_1$, pero no es cierto que hay una flecha con etiqueta 0 que manda q_0 a q_1.

En gran parte de las situaciones que necesitemos usar transiciones podremos tomarnos la licencia de usar la notación

$$p \xrightarrow{a_0} q_1 \xrightarrow{a_1} \cdots \xrightarrow{a_{n-1}} q_n \xrightarrow{a_n} q,$$

para indicar que hay un camino de flechas con etiquetas $a_0, \ldots, a_n$ que pasa por los estados $q_1, \ldots, q_n$ y que va de p a q (aquí permitiremos que los a_i puedan ser ε's).

Ejemplo 2.15. Sea M un NFA con ε-movimientos,con diagrama de transición :



Entonces, la tabla de transición es:

	a	b	ε
q_0	q_1	q_0	q_2
q_1	$\emptyset$	q_0	$\emptyset$
q_2	q_1	$\emptyset$	$\emptyset$

Veamos cuales son las transformaciones correspondientes a un input o un ε-movimiento:

$$\begin{array}{lll} q_0 \xrightarrow{\varepsilon} q_0, q_2 & q_0 \xrightarrow{a} q_1 & q_0 \xrightarrow{b} q_0, q_2 \\ q_1 \xrightarrow{\varepsilon} q_1 & q_1 \xrightarrow{b} q_0, q_2 & q_2 \xrightarrow{a} q_1 \\ q_2 \xrightarrow{\varepsilon} q_2 & q_2 \xrightarrow{a} q_1 & \end{array}$$

donde denotamos $p \xrightarrow{a} q, q'$ cuando $p \xrightarrow{a} q$ y $p \xrightarrow{a} q'$. La mayoría de las transformaciones son obvias y las otras se deducen de la definición. Sin embargo, creemos conveniente explicar algunas de ellas. Por ejemplo, recordemos que, por definición, $q \xrightarrow{\varepsilon} q$ para todo q estado, de allí que se justifica la primera columna de transformaciones. La transformación $q_1 \xrightarrow{b} q_2$ se deduce de que podemos llegar de q_1 a q_2 primero aplicando b y luego ε. En forma análoga se justifica $q_0 \xrightarrow{b} q_2$.

Como en el caso de autómatas determinísticos los estados finales nos determinan el lenguaje asociado al autómata.

Definición 2.5. Una cadena α es *aceptada* por el autómata si α transforma el estado inicial en uno final. Finalmente el *lenguaje aceptado por M* es el conjunto $L(M)$ de todas las cadenas aceptadas. Simbólicamente:

$$L(M) = \{\alpha \in \Sigma^* \mid q_0 \xrightarrow{\alpha} p, \text{ para algún } p \in F\}.$$

A nivel de diagramas de transición podemos entender que el lenguaje aceptado por M, un NFA con ε-movimientos, está formado por las cadenas $\alpha = b_0 b_1 \dots b_n$ tal que existe un recorrido por flechas con etiquetas $b_0, b_1, \dots, b_n$ que va del estado inicial a uno final, en el recorrido se pueden intercalar arbitrariamente flechas con etiqueta ε.

Observemos que como el ε es la cadena vacía el intercalar flechas con etiqueta ε no agrega nada a la palabra, y la palabra o cadena definitiva es aquella que no tiene ningún ε. También observemos que dada una cadena puede haber muchos recorridos partiendo del estado inicial, algunos de ellos pueden terminar en un estado final y otros no, sin embargo lo que interesa, para saber si una cadena es aceptada o no, es si por lo menos un recorrido alcanza un estado final.

Ejemplo 2.16. Averigüemos el lenguaje del autómata definido en el Ejemplo 2.11: observemos primero que si abandonamos el estado q_0 entonces vamos al estado q_1 y no podemos volver atrás. Análogamente si abandonamos el estado q_1, vamos al estado q_2 y ya no podemos salir de este estado. Luego si una cadena w alcanza el estado final, entonces comienza por q_0, pasa por q_1 y llega a q_2.

Es decir que w es de la forma $0^i1^j2^k$ (i,j,k enteros mayores o iguales a 0). Recíprocamente una cadena de la forma $0^i1^j2^k$ puede alcanzar el estado final con el siguiente recorrido: $0,0,\ldots,0$ (i-veces), ε, $1,1,\ldots,1$ (j-veces), ε, $2,2,\ldots,2$ (k-veces). Es decir que el lenguaje asociado a este autómata es $\{0^i1^j2^k : 0 \leq i,j,k\}$.

2.4.1 Formalización de los NFA

La definición formal de NFA es:

Definición 2.6. Un *autómata finito no determinístico con* ε*-movimientos (NFA con* ε*-mov)* es una 5-upla (Q,Σ,δ,q_0,F) que consiste de

1. un conjunto finito $Q = \{q_1,q_2,\ldots,q_n\}$ de *estados*,
2. un conjunto finito $\Sigma = \{a_1,a_2,\ldots,a_n\}$ de *símbolos de entrada* o *input*,
3. un conjunto de *reglas de transición* que transforma un estado con un input dado, en un conjunto posible de estados. También las reglas de transición describen la transformación de un estado en un conjunto de estados sin haber recibido input (ε-movimiento). Formalmente, las reglas de transición son dadas por una función $\delta : Q \times (\Sigma \cup \{\varepsilon\}) \to \mathcal{P}(Q)$, donde $\mathcal{P}(Q)$ es el conjunto formado por los subconjuntos de Q.
4. Además, como en el caso del DFA, hay un *estado inicial* q_0 y un conjunto de estados finales F.

Veamos ahora el lenguaje asociado a un NFA con ε-movimientos:

Definición 2.7. Sea $M = (Q,\Sigma,\delta,q_0,F)$ un NFA con ε-movimientos. Si $\alpha \in \Sigma^*$, diremos que α *transforma* p en q y lo denotaremos $p \xrightarrow{\alpha} q$, si existen $a_0,a_1,\ldots,a_n \in \Sigma \cup \{\varepsilon\}$ y $q_1,\ldots,q_n$ en Q , tal que $\alpha = a_1a_2\ldots a_n$ y

$$q_1 \in \delta(p,a_0),\ q_2 \in \delta(q_1,a_1),\ \ldots q_{i+1} \in \delta(q_i,a_i),\ \ldots q_n \in \delta(q_{n-1},a_{n-1}),\ q \in \delta(q_n,a_n).$$

Es conveniente también definir que ε transforma todo estado en sí mismo, es decir $q \xrightarrow{\varepsilon} q$.

Observación 2.3. Sea $M = (Q,\Sigma,\delta,q_0,F)$ un NFA con ε-movimientos.

1. Como dijimos antes, si $a \in \Sigma$, entonces si $q \in \delta(p,a)$ tenemos que $p \xrightarrow{a} q$. Pero no necesariamente si $p \xrightarrow{a} q$, entonces se cumple que $q \in \delta(p,a)$. La definición de $p \xrightarrow{a} q$ denota el hecho de que podemos llegar de p a q mediante ε-movimientos, luego a, luego ε-movimientos. En el autómata del ejemplo 2.11 es cierto que $q_0 \xrightarrow{0} q_1$, pues $q_0 \in \delta(q_0,0)$ y $q_1 \in \delta(q_0,\varepsilon)$, pero no es cierto que $q_1 \in \delta(q_0,0)$.
2. Cuando un autómata no tiene ε-mov, entonces coinciden la función de transición y las transformaciones por símbolos de input, es decir si p,q son estados y a es un símbolo de input,

   $$p \in \delta(q,a) \quad \text{ si y sólo si } \quad q \xrightarrow{a} p.$$

   Por lo tanto, la notación $p \xrightarrow{\alpha} q$ para NFA con ε-mov, es consistente con la misma notación para DFA.
3. Es claro, en forma intuitiva, que si $\alpha,\beta \in \Sigma^*$, entonces para $p,q \in Q$,

   $$p \xrightarrow{\alpha\beta} q \quad \text{ si y sólo si existe } r \in Q \text{ tal que } \quad p \xrightarrow{\alpha} r \xrightarrow{\beta} q. \tag{2}$$

4. Sea $M' = (Q, \Sigma, \delta', q_0, F)$ con δ' definida de la siguiente manera:

$$q \in \delta'(p,a) \quad \text{sii} \quad p \xrightarrow{a} q.$$

Entonces M' resulta ser un NFA con ε-mov que acepta el mismo lenguaje que M. La demostración se deja como ejercicio.

5. Debido a la observación anterior, en gran parte de las situaciones que necesitemos usar transiciones podremos tomarnos la licencia de usar la notación

$$p \xrightarrow{a_0} q_1 \xrightarrow{a_1} \cdots \xrightarrow{a_{n-1}} q_n \xrightarrow{a_n} q,$$

para indicar que hay un camino de flechas con etiquetas $a_0, \ldots, a_n$ que pasa por los estados $q_1, \ldots, q_n$ y que va de p a q. Dicho más formalmente:

$$q_1 \in \delta(p, a_0),\ q_2 \in \delta(q_1, a_1),\ \ldots q_n \in \delta(q_{n-1}, a_{n-1}),\ q \in \delta(q_n, a_n).$$

Repetimos la

Definición 2.8. Una cadena α es *aceptada* por el autómata si α transforma el estado inicial en uno final. Finalmente el *lenguaje aceptado por M* es el conjunto $L(M)$ de todas las cadenas aceptadas. Simbólicamente:

$$L(M) = \{\alpha \in \Sigma^* \mid q_0 \xrightarrow{\alpha} p, \text{ para algún } p \in F\}.$$

El siguiente Teorema fue anunciado al comienzo de la sección:

Teorema 2.1. *Los lenguajes aceptados por los autómatas finitos determinísticos son los mismos que los aceptados por los autómatas finitos no determinísticos con ε-movimientos. Más precisamente, dado un DFA (NFA con ε-movimientos) M, existe un NFA con ε-movimientos (resp. DFA) M', tal que $L(M) = L(M')$.*

Demostración. Si $M = (Q, \Sigma, \delta, q_0, F)$ un DFA, entonces sea $M' = (Q, \Sigma, \delta', q_0, F)$ el NFA definido por $\delta'(q,a) = \{\delta(q,a)\}$ para $q \in Q$ y $a \in \Sigma$. Es trivial comprobar, entonces, que $L(M) = L(M')$.

Veremos ahora que, dado $M = (Q, \Sigma, \delta, q_0, F)$ un NFA con ε-mov, podemos construir $M' = (Q'$, Σ, δ', $q'_0, F')$ un DFA, tal que $L(M) = L(M')$.

Sea $q \in M$, definamos $[q] = \{p \in Q \mid q \xrightarrow{\varepsilon} p\} \subseteq Q$. Sean $q_1, \ldots, q_i$ elementos de Q, entonces denotemos

$$[q_1, \ldots, q_i] = \cup_{j=1}^{i} [q_j].$$

Definamos M' de la siguiente manera: $Q' = \{[q_1, \ldots, q_i] \mid q_1, \ldots, q_i \in Q\}$. El nuevo conjunto de estados finales, F', es el conjunto de estados que contienen un estado que se transforma por ε en un estado final de M. El estado inicial será $q'_0 = [q_0]$. Finalmente, definimos $\delta'([q_1, \ldots, q_i], a)$ igual al conjunto $\{p \in Q \mid \text{existe } q_s \text{ tal que } q_s \xrightarrow{a} p\}$. Es decir, un $a \in \Sigma$ transforma $q' \in Q'$ en $p' \in Q'$, si el conjunto p' está formado por todos los elementos de Q que son los transformados por a de los elementos de q'. Resumiendo, si $q' \in Q'$ y $a \in \Sigma$:

$$\begin{aligned}
Q' &= \{[q_1, \ldots, q_i] \mid q_1, \ldots, q_i \in Q\} \subseteq \mathcal{P}(Q) \\
q'_0 &= [q_0] \\
F' &= \{[q_1, \ldots, q_i] \mid \text{ existe } k \text{ y } p \in F \text{ tal que } q_k \xrightarrow{\varepsilon} p\} \\
\delta(q', a) &= \{p \in Q \mid \exists q \in q' \text{ tal que } q \xrightarrow{a} p\} \quad \text{o equivalentemente} \\
q' &\xrightarrow{a} \{p \in Q \mid \exists q \in q' \text{ tal que } q \xrightarrow{a} p\}.
\end{aligned}$$

No es difícil comprobar que $M' = (Q', \Sigma, \delta', q_0', F')$ es un autómata finito determinístico.

No es difícil comprobar que si $q' \in Q'$ entonces

$$q' = \{p \in Q \mid \exists q \in q' \text{ tal que } q \xrightarrow{\varepsilon} p\}. \tag{3}$$

Esto se debe a que como $q' = [q_1, \ldots, q_i]$, por definición

$$\begin{aligned} q' &= \cup_{j=1}^{i} [q_j] = \cup_{j=1}^{i} \{p \in Q \mid q_j \xrightarrow{\varepsilon} p\} \\ &= \{p \in Q \mid \exists q_j \text{ tal que } q_j \xrightarrow{\varepsilon} p\} \subset \{p \in Q \mid \exists q \in q' \text{ tal que } q \xrightarrow{\varepsilon} p\}. \end{aligned}$$

Para demostrar la otra inclusión, consideremos p tal que $q \xrightarrow{\varepsilon} p$ con $q \in q'$. Como $q \in q'$ existe q_j tal que $q_j \xrightarrow{\varepsilon} q$, luego $q_j \xrightarrow{\varepsilon} q \xrightarrow{\varepsilon} p$, de lo cual se deduce que $q_j \xrightarrow{\varepsilon} p$ y por lo tanto $p \in \{p \in Q \mid \exists q_j \text{ tal que } q_j \xrightarrow{\varepsilon} p\}$.

Veamos ahora que si $\alpha \in \Sigma^*$, q' en Q', entonces:

$$q' \xrightarrow{\alpha} \{p \in Q \mid \exists q \in q' \text{ tal que } q \xrightarrow{\alpha} p\}.$$

Supongamos que $q' \xrightarrow{\alpha} p'$ y veamos que $p' = \{p \in Q \mid \exists q \in q' \text{ tal que } q \xrightarrow{\alpha} p\}$: hagamos inducción sobre $|\alpha|$. Cuando $|\alpha| = 0$, es decir cuando $\alpha = \varepsilon$, tenemos que $q' \xrightarrow{\varepsilon} q'$ y el párrafo anterior demuestra el resultado. Cuando $|\alpha| = 1$ el resultado se deduce trivialmente por definición de transición. Si $|\alpha| > 1$, entonces $\alpha = \beta a$, con $|\beta| \geq 1$ y $a \in \Sigma$. Por (1) de la observación 2.1 sabemos que existe $r' \in Q'$ tal que $q' \xrightarrow{\beta} r' \xrightarrow{a} p'$. Por hipótesis inductiva y el caso de longitud 1 tenemos que

$$\begin{aligned} q' \xrightarrow{\beta} r' &= \{r \in Q \mid \exists q \in q' \text{ tal que } q \xrightarrow{\beta} r\}, \\ r' \xrightarrow{a} p' &= \{p \in Q \mid \exists r \in r' \text{ tal que } r \xrightarrow{a} p\}. \end{aligned}$$

Si $p \in p'$, existe $r \in r'$ tal que $r \xrightarrow{a} p$, como $r \in r'$, existe $q \in q'$, tal que $q \xrightarrow{\beta} r$. Es decir que $q \xrightarrow{\beta a} p$ o equivalentemente $q \xrightarrow{\alpha} p$. Por lo tanto está probado $p' \subseteq \{p \in Q \mid \exists q \in q' \text{ tal que } q \xrightarrow{\alpha} p\}$. Por otro lado, si $p \in \{p \in Q \mid \exists q \in q' \text{ tal que } q \xrightarrow{\alpha} p\}$, existe $q \in q'$ tal que $q \xrightarrow{\alpha} p$, luego existe $r \in Q$ tal que $q \xrightarrow{\beta} r \xrightarrow{a} p$, por lo tanto $r \in r'$ y $p \in p'$. Esto prueba la otra inclusión y por lo tanto la igualdad deseada.

La aplicación directa del resultado anterior a q_0', nos dice que dada $\alpha \in \Sigma^*$, entonces

$$q_0' \xrightarrow{\alpha} \{p \in Q \mid q_0 \xrightarrow{\alpha} p\}. \tag{4}$$

Ahora bien, sea $\alpha \in L(M')$, es decir $q_0' \xrightarrow{\alpha} p'$, con $p' \in F'$. Como p' es final de M', entonces existe p en p' que también pertenece a F, luego por (4) obtenemos que $q_0 \xrightarrow{\alpha} p$, es decir $\alpha \in L(M)$. Recíprocamente, sea $\alpha \in L(M)$, es decir existe $p \in F$ tal que $q_0 \xrightarrow{\alpha} p$, por lo tanto (usando nuevamente (4)) $p \in p'$ con $q_0' \xrightarrow{\alpha} p'$. De esto se deduce que $p' \in F'$ y $\alpha \in L(M')$.

Finalmente, el párrafo anterior prueba que $L(M) = L(M')$.

□

Ejemplo 2.17. Sea M el NFA con ε-mov con estados q_0, q_1, un solo símbolo de input a, estado inicial q_0, estado final q_1 y las siguientes leyes de transición:

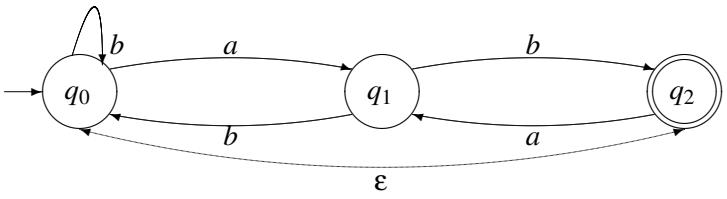
$$\delta(q_0, a) = \{q_0\}, \qquad \delta(q_0, \varepsilon) = \{q_1\}.$$

Encontremos un autómata determinístico con el mismo lenguaje que M. Por la demostración del teorema el nuevo autómata, M', tendrá estados $\emptyset, [q_0]$. Observar que $[q_0] = [q_1] = [q_0, q_1] = \{q_0, q_1\}$,

pues $q_0 \xrightarrow{\varepsilon} q_1$. El estado final e inicial es entonces $[q_0]$. Por definición, del estado $\emptyset$ no sale ninguna flecha y $\delta([q_0],\varepsilon) = [q_0]$, $\delta([q_0],a) = [q_0]$. Claramente, el lenguaje aceptado por ambos autómatas es el de todas las cadenas de a's.

Veamos un ejemplo menos trivial.

Ejemplo 2.18. Sea $M = (Q,\Sigma,\delta,q_0,F)$ un NFA con ε-movimientos definido por el siguiente diagrama de transición.



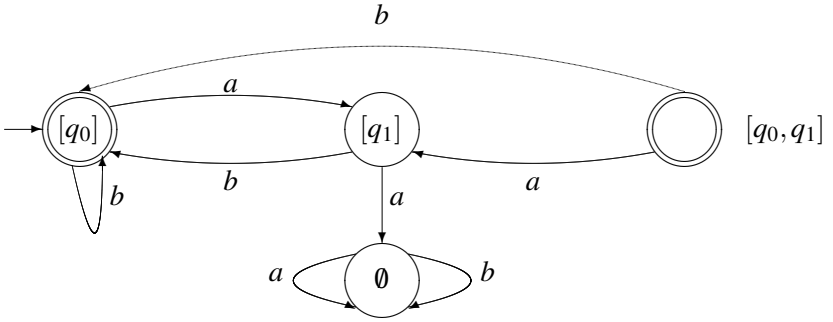
Encontremos un DFA que acepte el mismo lenguaje. Debemos primero establecer los estados del nuevo autómata:

$$\begin{aligned} [q_0] &= \{p \in Q \mid q_0 \xrightarrow{\varepsilon} p\} = \{q_0, q_2\} \\ [q_1] &= \{p \in Q \mid q_1 \xrightarrow{\varepsilon} p\} = \{q_1\} \\ [q_0,q_1] &= [q_0] \cup [q_1] = \{q_0,q_1,q_2\} \end{aligned}$$

Otro estado, siempre presente es $\emptyset$. Observemos que cualquier otro posible estado es uno de los ya listados más arriba, por ejemplo, $[q_2] = [q_0]$ y $[q_1,q_2] = [q_1] \cup [q_2] = [q_0,q_1]$. Establezcamos ahora las transiciones.

$$\begin{aligned} [q_0] &\xrightarrow{a} \{p \in Q \mid \exists q \in [q_0] \text{ tal que } q \xrightarrow{a} p\} = \\ &\qquad = \{p \in Q \mid q_0 \xrightarrow{a} p\} \cup \{p \in Q \mid q_2 \xrightarrow{a} p\} = \{q_1\} \cup \{q_1\} = [q_1] \\ [q_0] &\xrightarrow{b} \{p \in Q \mid q_0 \xrightarrow{b} p\} \cup \{p \in Q \mid q_2 \xrightarrow{b} p\} = \{q_0,q_2\} \cup \emptyset = [q_0] \\ [q_1] &\xrightarrow{a} \{p \in Q \mid q_1 \xrightarrow{a} p\} = \emptyset \\ [q_1] &\xrightarrow{b} \{p \in Q \mid q_1 \xrightarrow{b} p\} = \{q_0,q_2\} = [q_0] \\ [q_0,q_1] &\xrightarrow{a} \{p \in Q \mid \exists q \in [q_0,q_1] \text{ tal que } q \xrightarrow{a} p\} = [q_1] \cup \emptyset = [q_1] \\ [q_0,q_1] &\xrightarrow{b} [q_0] \cup [q_0] = [q_0] \end{aligned}$$

Por definición, es claro que toda transición que sale de $\emptyset$ vuelve a $\emptyset$. Finalmente, el estado inicial y final es $[q_0]$. El diagrama de transición del autómata recién construido es:



2.5 Expresiones regulares

Los lenguajes aceptados por los autómatas finitos se pueden describir fácilmente por expresiones simples llamadas expresiones regulares. En esta sección introduciremos las operaciones de concatenación y clausura sobre conjuntos de cadenas, definiremos expresiones regulares y daremos la demostración de que una expresión regular define un lenguaje que puede ser descrito por un autómata finito. La recíproca de este resultado también es cierta y veremos su demostración en la subsección final.

Definición 2.9. Sea Σ un conjunto finito de símbolos y sean L, L_1 y L_2 conjuntos de cadenas de Σ^*. La *concatenación* de L_1 y L_2, denotada L_1L_2 es un conjunto formado por cadenas de L_1 seguidas por cadenas de L_2, es decir: $L_1L_2 = \{xy \mid x \text{ está en } L_1 \text{ e } y \text{ está en } L_2\}$. Definamos $L^0 = \{\varepsilon\}$ y $L^i = LL^{i-1}$ para $i \geq 1$. La *clausura de Kleene* (o sólo *clausura*) de L, denotada L^*, es el conjunto

$$L^* = \cup_{i=0}^{\infty} L^i$$

y la clausura positiva de L, denotada L^+, es el conjunto

$$L^+ = \cup_{i=1}^{\infty} L^i.$$

L^* denota las cadenas construidas por concatenación de un número arbitrario de cadenas de L. L^+ es igual pero sin poner L^0. Nótese que L^+ contiene ε si y sólo si L lo contiene. Recordemos que en la Sección 2.1 definimos Σ^*, y observemos que ese conjunto coincide con el Σ^* que surge de la definición anterior, considerando a Σ como un lenguaje donde las cadenas son los símbolos del alfabeto.

Ejemplo 2.19. Sea $L_1 = \{11, 0\}$ y $L_2 = \{001, 10\}$, entonces

$$L_1L_2 = \{11001, 1110, 0001, 010\}.$$

Por otro lado,

$$L_1^* = \{11, 0\}^* = \{\varepsilon, 11, 0, 1111, 110, 011, 00, 111111, 11110, 11011, 1100, \ldots\}.$$

L_1^+ es L_1^* menos el ε.

Definición 2.10. Sea Σ un alfabeto. Las *expresiones regulares* sobre Σ y los *conjuntos* que ellas denotan se definen recursivamente de la siguiente manera:

1. $\emptyset$ es una expresión regular y denota el conjunto vacío.
2. ε es un expresión regular y denota el conjunto $\{\varepsilon\}$.
3. Para cada a en Σ, a es una expresión regular que denota el conjunto $\{a\}$.
4. Si r y s son expresiones regulares que denotan los conjuntos R y S, respectivamente, entonces $(r+s)$, (rs) y (r^*) son expresiones regulares que denotan los conjuntos $R \cup S$, RS y R^*, respectivamente.

Si r es una expresión regular escribiremos $L(r)$ al conjunto (o lenguaje) denotado por r.

Cuando escribimos expresiones regulares podemos omitir muchos paréntesis si asumimos que $*$ tiene más alta precedencia que la concatenación o $+$, y la concatenación tiene más alta precedencia que $+$. Por ejemplo $((0(1^*))+0)$ puede ser escrita 01^*+0. Finalmente, si r es una expresión regular denotemos r^i a la expresión $rr\cdots r$ (i veces).

Debemos tener mucho cuidado en no confundir cadenas con expresiones regulares y para ello debemos aclarar debidamente a qué nos estamos refiriendo. Por ejemplo la expresión regular ab denota un conjunto cuyo único elemento es la cadena ab.

Ejemplo 2.20. Los siguientes son ejemplo de expresiones regulares:

1. 00 es una expresión regular que representa $\{00\}$.
2. $(0+1)^*$ denota todas las cadenas de 0's y 1's.
3. $(0+1)^*00(0+1)^*$ denota todas las cadenas de 0's y 1's con al menos dos 0's consecutivos.
4. $(1+10)^*$ denota todas las cadenas de 0's y 1's que comienzan con 1 y no tienen dos 0's consecutivos. También pertenece a este conjunto la cadena vacía.

   Demostración. Probemos por inducción que $(1+10)^i$ no tiene dos 0's consecutivos. El caso $i=0$ es trivial, y si suponemos que $(1+10)^{i-1}$ no tiene dos 0's consecutivos, entonces, es claro que $(1+10)^i = (1+10)^{i-1}(1+10)$ no tiene dos 0's consecutivos, pues en el final las palabras son de la forma $\cdots\mathbf{1}1$ o $\cdots\mathbf{1}10$ o $\cdots\mathbf{10}1$ o $\cdots\mathbf{10}10$, donde las letras en negrita denotan palabras de $(1+10)^{i-1}$. De esta forma probamos que $(1+10)^i$ no tiene dos 0's consecutivos para todo i y por lo tanto $(1+10)^*$ no tiene dos 0's consecutivos. Por otro lado, dada cualquier cadena que comience con 1 y no tenga dos 0's consecutivos, se puede hacer una partición de la cadena de tal forma que si un 1 no es seguido de un 0, se toma el 1 (que pertenece a $(1+10)$) y si un 1 es seguido de un 0, se toma el 10 (que pertenece a $(1+10)$). Por ejemplo, 11010111 es particionado $1-10-10-1-1-1$ que pertenece a $(1+10)^6$. □
5. Basado en lo anterior, es claro que $(0+\varepsilon)(1+10)^*$ denota las cadenas de 0's y 1's que no tienen dos 0's seguidos.
6. $(0+1)^*011$ denota las cadenas de 0's y 1's que terminan en 001.
7. $0^*1^*2^*$ denota las cadenas que están formadas por cierta cantidad de 0's, luego cierta cantidad de 1's y luego cierta cantidad de 2's. Observar que este es lenguaje del autómata definido en el ejemplo 2.11.

Veremos ahora como a partir de una expresión regular r podemos construir un autómata no determinístico M que defina el mismo lenguaje, es decir tal que $L(r) = L(M)$. En realidad lo que construiremos son autómatas no determinísticos con un sólo estado final. Para las expresiones regulares ε, $\emptyset$ o a con $a \in \Sigma$ los autómatas no determinísticos de la Fig. 3 definen el lenguaje correspondiente:

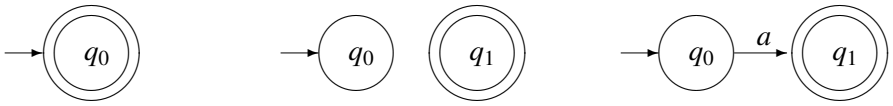


Figura 3:

A un autómata M con estado inicial q y estado final f lo dibujaremos de la siguiente manera (Fig. 4):

Ahora haremos lo siguiente: dadas expresiones regulares r_1 y r_2 a las cuales ya les hayamos asociado los autómatas M_1, M_2 correspondientes, como en la Fig. 5 encontraremos los autómatas correspondientes a $r_1 + r_2$, $r_1 r_2$ y r_1^*. Dicho de otra manera: si L_1 es el lenguaje de M_1 y L_2 es el lenguaje de M_2, entonces encontraremos autómatas correspondientes a $L_1 \cup L_2$, $L_1 L_2$ y L_1^*. El autómata correspondiente a $r_1 + r_2$ será el de la Fig. 6.

El autómata correspondiente a $r_1 r_2$ será el de la Fig. 7.

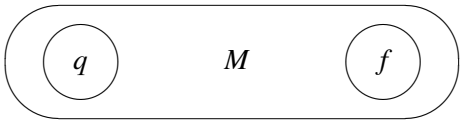


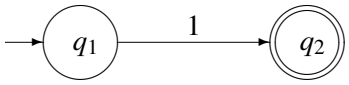
Figura 4:

Figura 5: Los autómatas de r_1 y r_2.

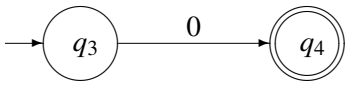
El autómata correspondiente a r_1^* será el de la Fig. 8
De lo anterior se deduce

Teorema 2.2. *Sea L una lenguaje denotado por la expresión regular r. Entonces existe M un DFA con* ε*-mov tal que* $L = L(M)$.

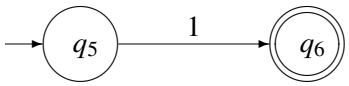
Ejemplo 2.21. Construyamos un autómata para la expresión regular $01^* + 1$. Por lo dicho anteriormente (respecto a los paréntesis), esta expresión es en realidad $((0(1^*)) + 1)$, es decir es de la forma $r_1 + r_2$, donde $r_1 = 01^*$ y $r_2 = 1$. El autómata para r_2 es fácil



Podemos expresar r_1 como $r_3 r_4$, donde $r_3 = 0$ y $r_4 = 1^*$. El autómata de r_3, también es fácil:



Ahora bien, r_4 es r_5^*, con $r_5 = 1$. Un autómata para r_5 es



Observemos que para poder construir el autómata correspondiente a la expresión regular necesitamos distinguir los estados de r_2 y r_5, aunque representen la misma expresión. Basándonos en las explicaciones anteriores un autómata para r_4 será

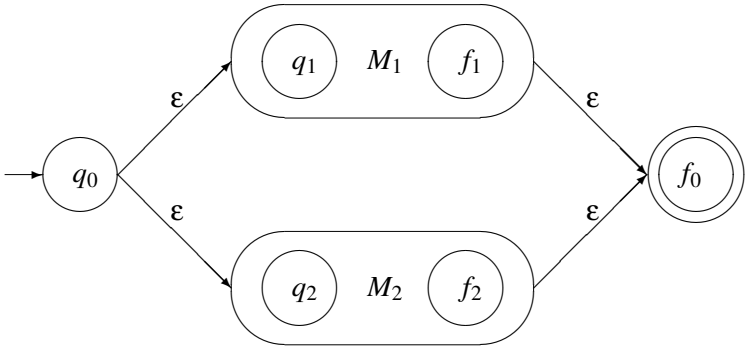


Figura 6: El autómata de $r_1 + r_2$.

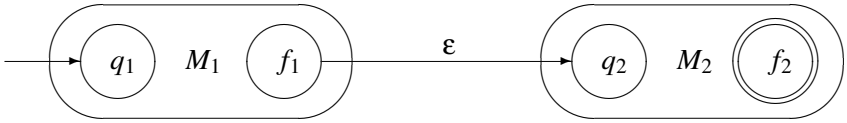
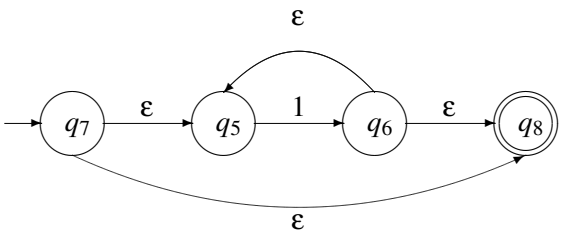
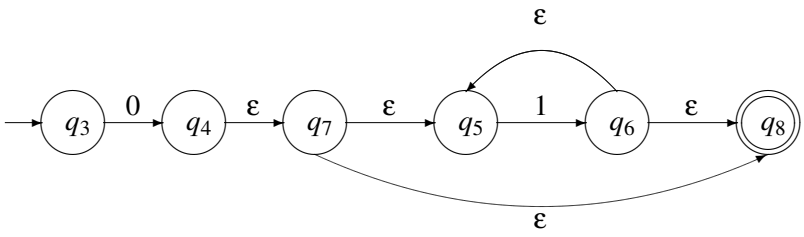


Figura 7: El autómata de $r_1 r_2$.



Luego el autómata correspondiente a la expresión regular $r_1 = 01^*$ será



Finalmente el autómata correspondiente a $01^* + 1$ será el de la Fig. 9.

2.5.1 Teorema de Kleene

Estudiemos el problema, un poco más complicado, de construir una expresión regular que denote el lenguaje de un autómata dado. Esta construcción se basa en la idea de encontrar, en forma algorítmica, una expresión regular que involucra resolver el problema para un autómata con menos estados. Repitiendo el proceso un número conveniente de veces lograremos la expresión regular buscada. Para

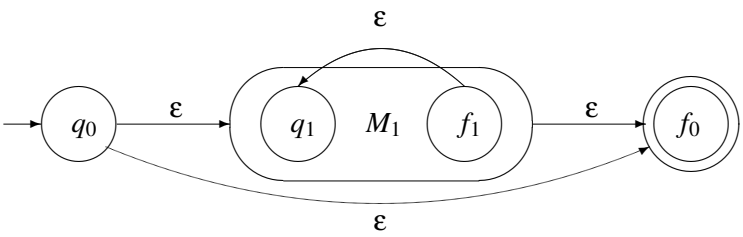


Figura 8: El autómata de r_1^*.

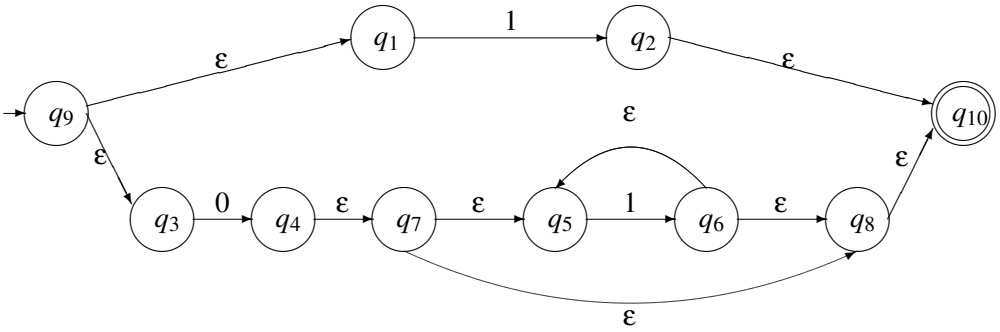


Figura 9: Un autómata correspondiente a la expresión regular $01^* + 1$.

describir el procedimiento, comenzaremos con el caso más simple en el cual nuestro autómata M tiene un solo estado final, q_n y su estado inicial es q_0. Es claro que $L(M) = L_{0n}$, el lenguaje de las cadenas que comienzan en q_0 y llegan a q_n. Llamemos I_0 al lenguaje formado por las cadenas que salen de q_0 y vuelven a q_0 *sin pasar nuevamente por él.* Si el estado inicial es el estado final, (es decir, $n = 0$) entonces claramente $L_{0n} = I_0^*$, puesto que con repetir caminos que salgan de q_0 y vuelvan a él obtengo todas las palabras aceptadas. Si $n \neq 0$, definamos F_{0n} como el lenguaje de las cadenas que salen de q_0 y llegan a q_n *sin pasar por* q_0. Entonces, por un razonamiento similar, $L_{0n} = I_0^* F_{0n}$.

Debemos ahora explicitar quiénes son I_0 y F_{0n}. Los elementos de I_0 son cadenas de dos formas:

1. $a\beta_0 b$, donde $a, b \in \Sigma \cup \{\varepsilon\}$ y β_0 es una cadena que parte de un estado p y tal que $q_0 \xrightarrow{a} q_i \xrightarrow{\beta_0} q_j \xrightarrow{b} q_0$ con p y q distintos de q_0 y además el camino que recorre β_0 no pasa por q_0; o bien
2. $c \in \Sigma \cup \{\varepsilon\}$, que haga un bucle de q_0 en sí mismo.

Si consideramos en el primer caso el autómata M' que se obtiene de M sacando q_0 y haciendo que q_i sea estado inicial y q_j estado final, entonces β_0 es una cadena aceptada por M'. En este caso, $L(M') = L_{ij}$, el lenguaje de las cadenas que salen de q_i y llegan a q_j sin pasar nunca por q_0, y obtenemos:

$$I_0 = aL_{ij}b + \cdots + c + \ldots$$

donde se suman todas las posibilidades para a, b, i, j tales que $q_0 \xrightarrow{a} q_i$, $q_j \xrightarrow{b} 0$, con q_0 distinto de q_i y q_j, los c tales que $q_0 \xrightarrow{c} q_0$, y en L_{ij} no consideramos q_0. Es decir, las cadenas que parten y llegan a q_0 se pueden ver como la concatenación de cadenas formadas por ciertos símbolos de entrada, con lenguajes aceptados por autómatas más chicos.

De igual modo, si q_0 no es el estado final, los elementos de F_{0n} son las cadenas que parten de q_0 y llegan al estado final q_n sin pasar por q_0. En este caso las cadenas son de la forma $a\beta$ con

$q_0 \xrightarrow{a} q_j \xrightarrow{\beta} q_n$, donde $a \in \Sigma \cup \{\varepsilon\}$. Luego esta cadena es la concatenación de un símbolo de entrada y una cadena aceptada por un autómata más chico:

$$F_{0n} = aL_{jn} + \ldots$$

donde a, j se mueven entre todas las opciones tales que $q_0 \xrightarrow{a} q_j \xrightarrow{\beta} q_n$, $j \neq 0$ y en L_{ij} (igual que arriba) no consideramos a q_0. Dado que el problema se va reduciendo a autómatas mas chicos, este algoritmo termina.

El caso de autómatas M con múltiples estados finales se trata similarmente, escribiendo

$$L(M) = aL_{jk} + \ldots,$$

donde a, j, k se mueven entre las opciones tales que $q_0 \xrightarrow{a} q_j \xrightarrow{\beta} q_k$ con q_k un estado final.

Ejemplo 2.22. Encontremos una expresión regular para el lenguaje aceptado por el autómata M descrito mediante el diagrama de transición de la Figura 10.

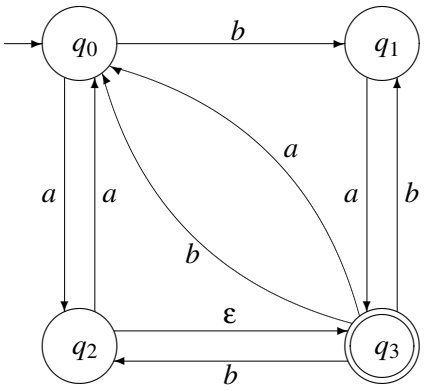


Figura 10: El autómata M

Es claro que $L(M) = L_{03}$, el lenguaje de las cadenas que comienzan en q_0 y llegan a q_3. Llamemos I_0 al lenguaje formado por las cadenas que salen de q_0 y vuelven a q_0 *sin pasar nuevamente por él* y F_{03} al lenguaje de las cadenas que salen de q_0 y llegan a q_3 *sin pasar por* q_0. Entonces, $L_{03} = I_0^* F_{03}$.

Analicemos ahora quiénes son I_0 y $F_{0,3}$.

Las formas de salir de q_0 y volver a sí mismo sin pasar nuevamente por él se pueden escribir así:

$$I_0 = bL_{12}a + bL_{13}a + bL_{13}b + aL_{23}a + aL_{23}b + aL_{22}a,$$

$$I_0 = b(L_{12}a + L_{13}(a+b)) + a(L_{23}(a+b) + L_{22}a),$$

donde L_{12} es el lenguaje aceptado por el autómata M_{12} que consta de los estados $\{q_1, q_2, q_3\}$ (hemos eliminado q_0), y que tiene por estado inicial a q_1 y final a q_2. De manera similar se definen $L_{13} = L(M_{13})$, $L_{23} = L(M_{23})$ (ver Figura 12) y L_{22} (ver Figura 11).

Las maneras de salir de q_0 y llegar al estado final q_3 sin pasar por q_0 son:

$$F_{03} = bL_{13} + aL_{23}.$$

Tenemos, como antes $L_{12} = I_1^* F_{12}$, salvo que ahora hemos eliminado q_0 de nuestras consideraciones. Es decir,

$$I_1 = aL_{33}b, \qquad F_{12} = aL_{33},$$

son las formas de ir de q_1 a q_1 y de q_1 a q_3, respectivamente, sin pasar nuevamente por q_1 (y sin utilizar q_0). Ahora, en nuestro L_{33}, no utilizaremos ni q_0 ni q_1.

> Como nuestro estado inicial es igual al estado final (q_3), tenemos
>
> $$L_{33} = I_3^*.$$
>
> En I_3 no usaremos ni q_0 ni q_1. Entonces obtenemos:
>
> $$I_3 = bL_{22}\varepsilon = b,$$
>
> con lo que
>
> $$L_{33} = b^*.$$

Ya podemos calcular los anteriores $I_1 = aL_{33}b = ab^*b$ y $F_{12} = aL_{33} = ab^*$. Luego obtenemos

$$L_{12} = (ab^*b)^*ab^*.$$

Ahora nos toca ver $L_{13} = I_1^* F_{13}$, eliminando q_0.

$$I_1 = ab^*b, \qquad F_{13} = aL_{33} = ab^*,$$

dado que, como antes, en L_{33} eliminamos q_0 y q_1. Entonces $L_{13} = (ab^*b)^*ab^*$.

Observemos que $L_{23} = I_2^* F_{23}$, eliminando q_0.

$$I_2 = \varepsilon L_{33}b = L_{33}b, \qquad F_{23} = \varepsilon L_{33} = L_{33},$$

donde en L_{33} se eliminan q_0 y q_2.

> Veamos L_{33} con q_0 y q_2 eliminados. Como antes (estado inicial es el final) tenemos
>
> $$L_{33} = I_3^*.$$
>
> En I_3 no usaremos ni q_0 ni q_2. Entonces obtenemos:
>
> $$I_3 = bL_{11}a = ba$$
>
> con lo que
>
> $$L_{33} = (ba)^*.$$

Obtenemos entonces

$$I_2 = (ba)^*b, \qquad F_{23} = (ba)^*,$$

y luego $L_{23} = ((ba)^*b)^*(ba)^*$.

Sólo nos falta $L_{22} = I_2^*$ (Figura 11). Tenemos

$$I_2 = \varepsilon L_{33} b = L_{33} b = (ba)^* b,$$

pues es el mismo L_{33} considerado anteriomente. Entonces

$$L_{22} = ((ba)^* b)^*.$$

Por último,

$$I_0 = b(L_{12}a + L_{13}(a+b)) + a(L_{23}(a+b) + L_{22}a),$$

$$I_0 = b((ab^*b)^* ab^* a + (ab^*b)^* ab^*(a+b)) + a(((ba)^*b)^*(ba)^*(a+b) + ((ba)^*b)^* a).$$

$$F_{03} = bL_{13} + aL_{23},$$

$$F_{03} = b(ab^*b)^* ab^* + a((ba)^*b)^*(ba)^*,$$

y nuestro resultado final es

$$L = L_{03} = I_0^*\, F_{03}.$$

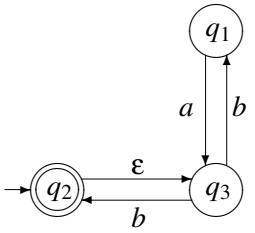


Figura 11: El autómata M_{22}

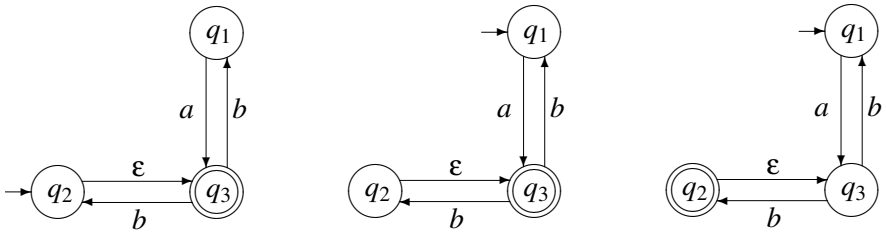


Figura 12: Los autómatas M_{23}, M_{13} y M_{12}, respectivamente

Teorema 2.3 (Teorema de Kleene)**.** *Sea M un NFA con ε-mov, entonces, existe una expresión regular r tal que $L(M)$ es el lenguaje denotado por r.*

Probaremos este teorema viendo que, dado un NFA M, el algoritmo descripto anteriormente devuelve una expresión regular que caracteriza el lenguaje aceptado por M. Para ello será necesario formalizar las definiciones de L_{nm}, I_n, F_{nm} y el proceso de "eliminar estados".

Supongamos que $M = (Q, \Sigma, \delta, q_0, F)$. Supondremos que existe un único estado final: $F = \{q_f\}$, con $0 \leq f \leq r$. Sea $Q = \{q_0, \ldots, q_r\}$ y $\Sigma = \{c_1, \ldots, c_u\}$.

Dados $0 \leq n, m \leq r$, $R \subseteq Q$, definamos recursivamente las siguientes expresiones regulares:

$$\begin{array}{rl} L_{nm}(R) & = \emptyset \qquad \text{si } n \text{ ó } m \text{ no están en } R. \\ L_{nn}(R) & = I_n(R)^* \\ L_{nm}(R) & = I_n(R)^* F_{nm}(R) \qquad \text{si } n \neq m. \\ I_n(R) & = \sum_{q_n \xrightarrow{a} q_t,\, q_s \xrightarrow{b} q_n} a L_{ts}(R \setminus \{q_n\}) b \;\; + \sum_{q_n \xrightarrow{c} q_n} c \\ F_{nm}(R) & = \sum_{q_n \xrightarrow{a} q_t} a L_{tm}(R \setminus \{q_n\}), \end{array}$$

donde $a, b, c \in \Sigma \cup \{\varepsilon\}$. En particular,

$$L_{nn}(\{q_n\}) = I_n(\{q_n\})^* = \left(\sum_{q_n \xrightarrow{c} q_n} c \right)^*.$$

En esencia, en las expresiones $L_{nm}(R)$ sólo consideramos "habilitados" los estados del conjunto R.

Por otro lado, sea $M_{nm}(R) = (R, \Sigma, \delta \mid R, q_n, \{q_m\})$ el autómata que resulta de M luego eliminar todos los estados que no están en R y sus transiciones, y estipulando que el estado inicial es q_n y el único estado final es q_m. En caso de que q_n ó q_m no estén en R, $M_{nm}(R)$ será un autómata vacío con lenguaje $\emptyset$.

Lema 2.4. *Toda cadena α representada por las expresiones regulares de arriba son aceptadas por los respectivos autómatas. Es decir: para todo $R \subseteq Q$, $L_{nm}(R) \subseteq L(M_{nm}(R))$.*

Demostración. Tenemos dos casos a probar, correspondientes a la definición de $L_{nm}(R)$:

(1) Si $q_n \in R$ y $\alpha \in I_n(R)^k$ entonces $\alpha \in L(M_{nn}(R))$.
(2) Si $n \neq m$, $q_n, q_m \in R$ y $\alpha \in I_n(R)^k F_{nm}(R)$ entonces $\alpha \in L(M_{nm}(R))$.

Procederemos simultáneamente por inducción en k y en $|\, R \,|$.

El caso base para R es el conjunto vacío, para el cual los antecedentes de (1) y (2) son falsos. Más aún, por las definiciones tenemos que si alguno de n ó m falta en R, $L_{nm}(R) = L(M_{nm}(R)) = \emptyset$. Luego, podemos considerar de ahora en adelante que $n, m \in R$.

$\boxed{k=0}$ En el caso (1) debe ser $\alpha = \varepsilon$ y claramente $\varepsilon \in L(M_{nn}(R))$ al ser q_n estado inicial y final.
En el caso (2), $\alpha \in F_{nm}(R) = \sum_{q_n \xrightarrow{a} q_t} a L_{tm}(R \setminus \{q_n\})$. Luego existen a, t, α' tales que

$$q_n \xrightarrow{a} q_t, \quad \alpha' \in L_{tm}(R \setminus \{q_n\}), \quad \alpha = a\alpha'.$$

Por hipótesis inductiva (eliminamos un estado) $\alpha' \in L(M_{tm}(R \setminus \{q_n\}))$ y luego existe una sucesión

$$q_t = s_1 \xrightarrow{a_1} s_2 \xrightarrow{a_2} \cdots \xrightarrow{a_l} s_{l+1} = q_m$$

con $s_j \in R \setminus \{q_n\}$ y $\alpha' = a_1 a_2 \ldots a_l$. Como $q_n \xrightarrow{a} q_t$ entonces $\alpha \in L(M_{nm}(R))$.

$\boxed{k+1}$ Caso (1): es muy similar al (2), lo dejamos como ejercicio.
Caso (2): Supongamos que $\alpha \in I_n(R)^{k+1} F_{nm}(R)$. Luego, hay dos opciones (para cada uno de los tipos de sumandos en la definición de I_n):

- $\alpha = a_0\alpha'\beta'$ con $q_n \xrightarrow{a_0} q_n$, $\alpha' \in I_n(R)^k$ y $\beta' \in F_{nm}(R)$. Por hipótesis inductiva (disminuimos k), $\alpha'\beta' \in L(M_{nm}(R))$. Como antes, hay una sucesión

$$q_n = s_1 \xrightarrow{a_1} s_2 \xrightarrow{a_2} \cdots \xrightarrow{a_l} s_{l+1} = q_m$$

tal que $\alpha'\beta' = a_1 a_2 \dots a_l$ y luego

$$q_n \xrightarrow{a_0} q_n \xrightarrow{a_1} s_2 \xrightarrow{a_2} \cdots \xrightarrow{a_l} s_{l+1} = q_m$$

muestra que $\alpha \in L(M_{nm}(R))$.

- $\alpha = a_0\gamma b_0\alpha'\beta'$ con $q_n \xrightarrow{a_0} s_0$, $t_0 \xrightarrow{b_0} q_n$, $\gamma \in L_{s_0 t_0}(R \setminus \{q_n\})$, $\alpha' \in I_n(R)^k$ y $\beta' \in F_{nm}(R)$. Por hipótesis inductiva $\alpha'\beta' \in L(M_{nm}(R))$ (disminuimos k) y $\gamma \in L(M_{s_0 t_0}(R \setminus \{q_n\}))$ (borramos q_n). Es decir

$$s_0 \xrightarrow{\gamma} t_0, \quad q_n \xrightarrow{\alpha'\beta'} q_m,$$

y luego

$$q_n \xrightarrow{a_0} s_0 \xrightarrow{\gamma} t_0 \xrightarrow{b_0} q_n \xrightarrow{\alpha'\beta'} q_m$$

muestra que $\alpha \in L(M_{nm}(R))$.

□

Lema 2.5. *Toda cadena α aceptada por $M_{nm}(R)$ está en el lenguaje representado por la expresión regular dada por el algoritmo. Es decir: para todo $R \subseteq Q$, $L(M_{nm}(R)) \subseteq L_{nm}(R)$.*

Demostración. En esta prueba es más conveniente por motivos técnicos considerar las sucesiones de transiciones que dan lugar a las cadenas aceptadas, que considerar las cadenas solamente. Probaremos, para toda sucesión de transiciones en M

$$q_n = s_0 \xrightarrow{a_0} s_1 \xrightarrow{a_1} \cdots \xrightarrow{a_{l-1}} s_l = q_m$$

(donde $a_i \in \Sigma \cup \{\varepsilon\}$ y $s_i \in Q$), que:

si $s_i \in R$ para todo i entonces $a_1 \dots a_{l-1} \in L_{nm}(R)$,

por inducción en la longitud l de la sucesión. Llamemos $\alpha = a_1 \dots a_{l-1}$. Consideraremos un "caso (1)" cuando $n = m$ y un "caso (2)" cuando no sean iguales.

$\boxed{l = 0}$ Caso (1): si la cantidad de transiciones es cero, no nos movemos de q_n. La cadena representada es la vacía, y obviamente está en $I_n(R)^* = L_{nn}(R)$.

Caso (2): al ser $n \neq m$, l tiene que ser mayor a cero, así que este caso base es trivial.

$\boxed{l+1}$ Caso (1): si todos los s_i son iguales a q_n, entonces claramente

$$\alpha \in \left(\sum_{q_n \xrightarrow{c} q_n} c \right)^* \subseteq I_n(R)^* = L_{nn}(R)$$

Sino, tomemos $0 \leq j \leq h \leq l$ tales que s_i es distinto de q_n para todo i entre j y h, pero $s_{j-1} = s_{h+1} = q_n$. Luego existen α', β y γ (eventualmente vacías) tales que:

$$q_n = s_0 \xrightarrow{\alpha'} s_{j-1} \xrightarrow{a_{j-1}} s_j \xrightarrow{\beta} s_h \xrightarrow{a_h} s_{h+1} \xrightarrow{\gamma} s_l = q_n$$

Por hipótesis inductiva, tenemos $\alpha', \gamma \in L_{nn}(R) = I_n(R)^*$ y $\beta \in L_{jh}(R \setminus \{q_n\})$.[1] Luego

$$\alpha \in I_n(R)^* \left(a_{j-1} L_{jh}(R \setminus \{q_n\}) a_h\right) I_n(R)^* \subseteq I_n(R)^* I_n(R) I_n(R)^* = I_n(R)^*$$

Caso (2): sea s_h es el último s_j que es igual a q_n (notar que h debe ser menor que l puesto que $n \neq m$). Luego

$$q_n = s_0 \xrightarrow{\alpha'} s_h \xrightarrow{a_h} s_{h+1} \xrightarrow{\beta'} s_l = q_m$$

con $\alpha = \alpha' a_h \beta'$ con α' y β' eventualmente vacíos. Ahora, la sucesión de transiciones correspondiente a β' no contiene a q_n; por hipótesis inductiva tenemos que $\beta' \in L_{km}(R \setminus \{q_n\})$, donde suponemos sin pérdida de generalidad que $s_{h+1} = q_k$. Y también por hipótesis inductiva $\alpha' \in L_{nn}(R) = I_n(R)^*$. Entonces

$$\alpha \in I_n(R)^* a_h L_{km}(R \setminus \{q_n\}) \subseteq I_n(R)^* F_{nm}(R) = L_{nm}(R).$$

□

Prueba del Teorema de Kleene. Supongamos $M = (Q, \Sigma, \delta, q_0, F)$ donde $Q = \{q_0, \ldots, q_r\}$ y $F = \{q_k, \ldots, q_m\}$ con $0 \leq k \leq m \leq r$. Está claro que

$$L(M) = \bigcup_{k \leq i \leq m} L(M_{0i}(Q)),$$

dado que cada cadena aceptada termina obviamente en un solo estado final. Pero ahora, por los lemas anteriores tenemos

$$L(M) = \bigcup_{k \leq i \leq m} L_{0i}(Q)$$

y por último

$$L(M) = \sum_{k \leq i \leq m} L_{0i}(Q)$$

es la expresión regular buscada. □

2.6 Pumping Lemma

En esta sección veremos que existen lenguajes que no son regulares. Demostraremos esto usando un resultado conocido como el Pumping Lemma o "Lema del Inflado".

Proposición 2.6 (Pumping Lemma)**.** *Sea L un lenguaje regular. Entonces existe un número $k > 0$ tal que para toda cadena $\alpha \in L$ con $| \alpha | \geq k$, existen cadenas β_1, β_2, γ con $| \beta_1 \gamma | \leq k$, $| \gamma | > 0$ que satisfacen*

1. $\alpha = \beta_1 \gamma \beta_2$,
2. $\beta_1 \gamma^n \beta_2 \in L$ *para todo* $n \geq 1$.

Demostración. Debido a que L es un lenguaje regular, existe M un DFA tal que $L = L(M)$. Sea k el número de estados de M y α una cadena en L con $| \alpha | \geq k$. Entonces $\alpha = a_{i_1} \cdots a_{i_t}$ donde a_{i_j} símbolo de input ($1 \leq j \leq t$) y $t \geq k$. Sean $q_{i_0}, q_{i_1}, \ldots, q_{i_t}$ estados tal que, $q_{i_0} = q_0$ el estado inicial, q_{i_t} un estado final y

$$q_{i_0} \xrightarrow{a_{i_1}} q_{i_1} \xrightarrow{a_{i_2}} \cdots \xrightarrow{a_{i_{t-1}}} q_{i_{t-1}} \xrightarrow{a_{i_t}} q_{i_t}.$$

[1] Aquí suponemos, por comodidad notacional, que $s_j = q_j$ y $s_h = q_h$.

Es claro que, como $q_{i_0}, \ldots, q_{i_k}$ es una lista de $k+1$ estados y como el autómata tiene k estados, hay al menos dos estados que se repiten en la lista, digamos q_{i_r} y q_{i_s}, con $r < s \leq k$. Sean entonces $\beta_1 = a_{i_1} \cdots a_{i_{r-1}}$, $\gamma = a_{i_r} \cdots a_{i_{s-1}}$ y $\beta_2 = a_{i_s} \cdots a_{i_t}$. Luego, $\alpha = \beta_1\gamma\beta_2$, $|\,\gamma\,| > 0$ y $|\,\beta_1\gamma\,| \leq k$. Observemos que

$$q_{i_0} \xrightarrow{\beta_1} q_{i_r} \xrightarrow{\gamma} q_{i_r} \xrightarrow{\beta_2} q_{i_t},$$

en particular tenemos que $q_{i_r} \xrightarrow{\gamma} q_{i_r}$. Por lo tanto, se cumple que para $n > 0$,

$$q_{i_r} \xrightarrow{\gamma^n} q_{i_r} \quad \text{o equivalentemente} \quad q_{i_r} \xrightarrow{\gamma} q_{i_r} \xrightarrow{\gamma} \cdots \xrightarrow{\gamma} q_{i_r} \ (n\text{-veces}).$$

Por consiguiente, también vale

$$q_{i_0} \xrightarrow{\beta_1} q_{i_r} \xrightarrow{\gamma^n} q_{i_r} \xrightarrow{\beta_2} q_{i_t},$$

lo cual implica que $\beta_1\gamma^n\beta_2$ es una cadena aceptada por M y por lo tanto que pertenece a L. Esto prueba el Pumping Lemma. □

Como mencionamos arriba el Pumping Lemma es útil para demostrar que un lenguaje no es regular.

Ejemplo 2.23. Probemos que el lenguaje $L = \{a^n b^n : n > 1\}$ no es regular.

Demostración. Hagamos la prueba por el absurdo suponiendo que L es regular. Sea k como en el Pumping Lemma. Debemos elegir $\alpha \in L$ de tal forma de obtener una contradicción que nos lleve al absurdo. Elegimos $\alpha = a^k b^k$, por el Pumping Lemma existen cadenas β_1, β_2, γ con $|\,\beta_1\gamma\,| \leq k$, $|\,\gamma\,| > 0$ que satisfacen $\alpha = \beta_1\gamma\beta_2$ y $\beta_1\gamma^n\beta_2 \in L$ para todo $n \geq 1$. Como $|\,\beta_1\gamma\,| \leq k$, γ es de la forma a^s con $1 \leq s$, luego $\alpha = a^r a^s a^t b^k$ con $r+s+t = k$ y $s > 0$. Además $a^r a^{ns} a^t b^k \in L$ para $n \geq 1$. Esto es una contradicción pues como $r+ns+t > k$ para $n > 1$, entonces $a^r a^{ns} a^t b^k \notin L$ para $n > 1$. □

El Pumping Lemma es utilizado para detectar si un lenguaje no es regular, con la siguiente estrategia:

1. Seleccionar el lenguaje L que usted desea ver que no es regular.
2. Suponer que es regular y que por lo tanto existe un k como en el Pumping Lemma.
3. Seleccionar una cadena α en L de longitud mayor que k. La elección de la cadena no es arbitraria y depende del lenguaje dado.
4. Descomponer la cadena de acuerdo al Pumping Lemma y generar nuevas cadenas en L haciendo "inflación" en el centro (es decir las cadenas $\beta_1\gamma^n\beta_2$).
5. Verificar que las cadenas "infladas" no pertenecen a L. Esto genera una contradicción que vino de suponer que L era regular.

Observar que la estrategia anterior podría no servir, pero el hecho de que no podamos aplicar el Pumping Lemma a un lenguaje L no implica que este sea regular.

Ejemplo 2.24. Demostrar que el lenguaje $L = \{0^n 001^n \mid n \in \mathbb{N}\}$ no es un lenguaje regular.

Demostración. Supongamos que L es un lenguaje regular y sea k como en el Pumping Lemma. La cadena $\alpha = 0^k001^k$ es de L y por lo tanto es aceptada por el autómata. Por el Pumping Lemma $\alpha = \beta_1\gamma\beta_2$, con $|\gamma| > 0$ y $|\beta_1\gamma| \leq k$ y tal que $\beta_1\gamma^n\beta_2$ es aceptado por el autómata para todo $n > 0$. Como $|\beta_1\gamma| \leq k$, es claro que $\beta_1 = 0^r$, $\gamma = 0^s$ y $s \geq 1$. Es decir $\beta_1\gamma^n\beta_2 = 0^{k+1+(n-1)s}001^{k+1}$. Por lo tanto, $0^{k+1}0^{(n-1)s}001^{k+1}$ es una cadena de L para todo $n > 0$. Lo cual es un absurdo, pues esa cadena no pertenece a L para $n > 1$. □

Observar que en el ejemplo anterior hay cadenas de L que son de longitud mayor que k (por ejemplo $0^{k/2}001^{k/2}$, si k es par) que no sirven para demostrar, usando el Pumping Lemma, que el lenguaje no es regular. Es decir, se debe tener mucho cuidado en la elección de la cadena, pues la elección de una cadena incorrecta hará que la utilización del Pumping Lemma no nos sirva para demostrar que el lenguaje no es regular.

2.7 Ejercicios

1. Trace los diagramas de transición de los DFA dados por las siguientes reglas de transición.

   (a) Estados $\{q_0, q_1, q_2\}$; símbolos de input $\{a, b\}$, estado inicial q_0 y estado final q_0 también.

	a	b
q_0	q_1	q_0
q_1	q_2	q_0
q_2	q_0	q_2

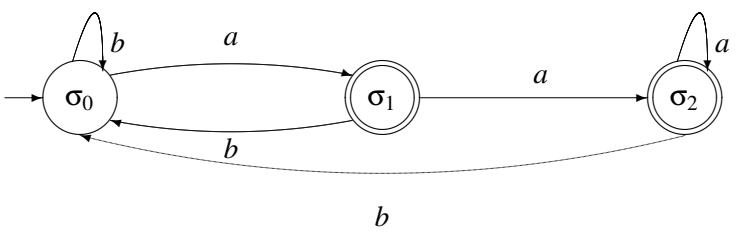
   (b) Estados $\{q_0, q_1, q_2\}$, símbolos de input $\{a, b\}$, estado inicial q_0 y estados finales q_0, q_2.

	a	b
q_0	q_1	q_1
q_1	q_0	q_2
q_2	q_0	q_1

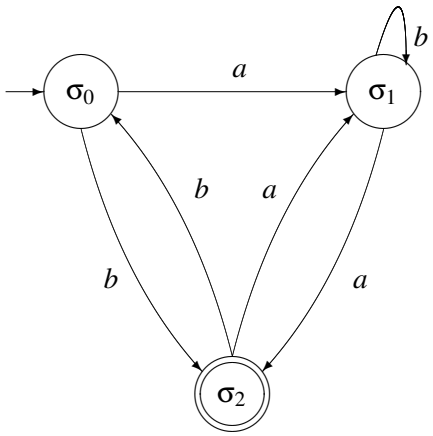
   (c) Estados $\{q_0, q_1, q_2, q_3\}$, símbolos de input $\{a, b, c\}$, estado inicial q_0 y estados finales q_1, q_2.

	a	b	c
q_0	q_1	q_0	q_2
q_1	q_0	q_3	q_0
q_2	q_3	q_2	q_0
q_3	q_1	q_0	q_1

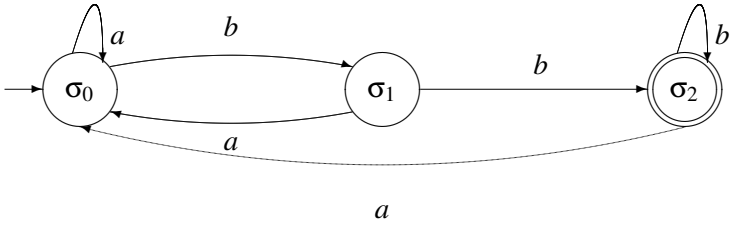
2. Determine si la cadena *abbaa* es aceptada por el DFA definido por el siguiente diagrama



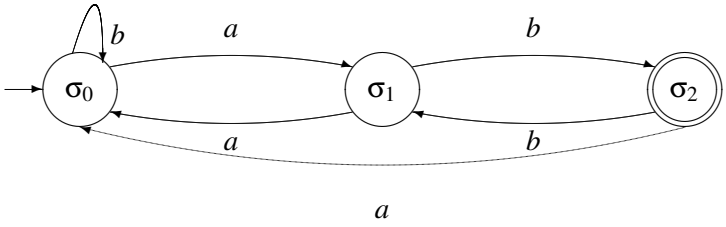
3. Determine si la cadena *abbaa* es aceptada por el DFA definido por el siguiente diagrama



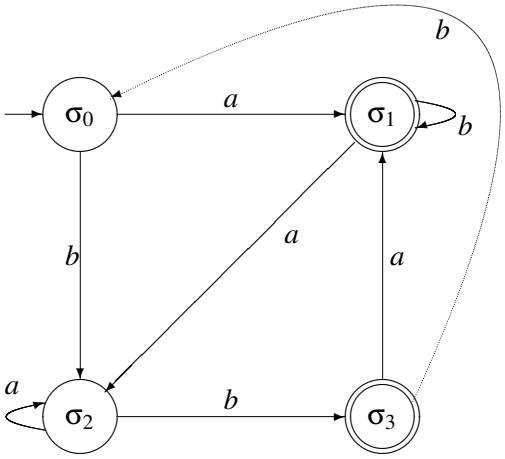
4. Determine si la cadena *aabaabb* es aceptada por el DFA definido por el siguiente diagrama



5. Determine si la cadena *aaabbbaab* es aceptada por el DFA definido por el diagrama:



6. Determine si la cadena *aaababbab* es aceptada por el DFA definido por el diagrama:



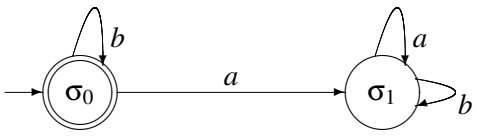
7. Pruebe que una cadena α en el alfabeto $\{a,b\}$ es aceptada por el autómata del Ejercicio 2 si y sólo si α termina en a.

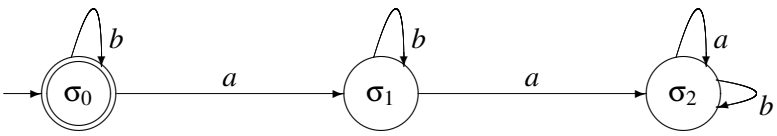
8. Pruebe que una cadena α en el alfabeto $\{a,b\}$ es aceptada por el autómata del Ejercicio 4 si y sólo si α termina en bb.

9. Trace el diagrama de transición del DFA que acepte el conjunto de cadenas en al alfabeto $\{a,b\}$ dado en cada uno de los siguientes items.

   (a) Cadenas con un número par de letras a.
   (b) Cadenas con exactamente una letra b.
   (c) Cadenas con al menos una letra b.
   (d) Cadenas con exactamente dos letras a.
   (e) Cadenas con al menos dos letras a.
   (f) Cadenas que contengan m letras a, donde m es un múltiplo de 3.
   (g) Cadenas que empiecen con baa.
   (h) Cadenas que contengan $abba$.
   (i) Cadenas donde toda letra b esté seguida de una letra a.
   (j) Cadenas que terminen con aba
   (k) Cadenas que empiecen con ab y terminen con aba

10. Demuestre que los siguientes DFA son equivalentes.





11. Sea L un conjunto finito de cadenas no vacías sobre $\{a,b\}$. Demuestre que hay un DFA que acepta L.

12. Hallar un autómata finito determinístico que acepte exactamente el lenguaje de las cadenas sobre el alfabeto $\Sigma = \{0,1\}$ que tienen una cantidad de 1's que es múltiplo de 3 y un número par de ceros.

13. Dado el DFA con alfabeto $\Sigma = \{0,\ 1\}$, estado inicial q_0 y estados finales q_1 y q_f y con la siguiente tabla de transición:

	0	1
q_0	q_2	q_1
q_1	q_0	q_f
q_2	q_f	q_2
q_f	q_0	q_1

    (a) Hacer el diagrama de transición correspondiente,

    (b) Verificar si las cadenas $w_1 = 01011$ y $w_2 = 01000$ son aceptadas o no por el autómata. Describa paso a paso.

14. Dado el DFA con alfabeto $\Sigma = \{a,\ b\}$, estado inicial q_0 y estados finales q_1 y q_f y con la siguiente tabla de transición:

	a	b
q_0	q_1	q_2
q_1	q_2	q_1
q_2	q_f	q_0
q_f	q_f	q_2

    (a) Hacer el diagrama de transición correspondiente,

    (b) Verificar si las cadenas $w_1 = baaba$ y $w_2 = aabaa$ son aceptadas o no por el autómata. Describa paso a paso.

15. Construir un autómata finito determinístico con alfabeto $\Sigma = \{a,\ b\}$ que acepte cadenas que empiecen con *ab* y terminen con *ba*.

16. Hallar un autómata finito determinístico que acepte exactamente el lenguaje de las cadenas sobre el alfabeto $\Sigma = \{0,1\}$ que tienen una cantidad par de 1's y el número de 0's es múltiplo de 3.

17. Hallar un autómata finito determinístico que acepte exactamente el lenguaje de las cadenas sobre el alfabeto $\Sigma = \{0,1\}$ que tiene un número par de ceros y terminan con dos (o más) unos.

18. Hallar un autómata finito determinístico que acepte exactamente el lenguaje de las cadenas sobre el alfabeto $\Sigma = \{0,1\}$ que no empiezan y terminan con el mismo símbolo.

19. Trace los diagramas de transición de los autómatas no determinísticos dados por las siguientes reglas de transición.

    (a) Estados $\{q_0, q_1, q_2\}$; símbolos de input $\{a,b\}$, estado inicial q_0 y estado final q_0 también y reglas de transición dadas por la siguiente tabla.

	a	b
q_0	$\emptyset$	$\{q_1, q_2\}$
q_1	$\{q_2\}$	$\{q_0, q_1\}$
q_2	$\{q_0\}$	$\emptyset$

    (b) Estados $\{q_0, q_1, q_2\}$, símbolos de input $\{a,b\}$, estado inicial q_0 y estados finales q_0, q_1 y reglas de transición dadas por la siguiente tabla.

	a	b
q_0	$\{q_1\}$	$\{q_0, q_1\}$
q_1	$\emptyset$	$\{q_2\}$
q_2	$\{q_1\}$	$\emptyset$

    (c) Estados $\{q_0, q_1, q_2, q_3\}$, símbolos de input $\{a,b\}$, estado inicial q_0 y estado final q_1 y reglas de transición dadas por la siguiente tabla.

	a	b
q_0	$\emptyset$	$\{q_3\}$
q_1	$\{q_1, q_2\}$	$\{q_3\}$
q_2	$\emptyset$	$\{q_0, q_1, q_3\}$
q_3	$\emptyset$	$\emptyset$

    (d) Estados $\{q_0, q_1, q_2\}$, símbolos de input $\{a,b,c\}$, estado inicial q_0 y estados finales q_0, q_1 y reglas de transición dadas por la siguiente tabla.

	a	b	c
q_0	$\{q_1\}$	$\emptyset$	$\emptyset$
q_1	$\{q_0\}$	$\{q_2\}$	$\{q_0, q_2\}$
q_2	$\{q_0, q_1, q_2\}$	$\{q_0\}$	$\{q_0\}$

    (e) Estados $\{q_0, q_1, q_2, q_3\}$, símbolos de input $\{a,b,c\}$, estado inicial q_0 y estados finales q_0, q_3 y reglas de transición dadas por la siguiente tabla.

	a	b	c
q_0	$\emptyset$	$\{q_3\}$	$\emptyset$
q_1	$\{q_1, q_2\}$	$\{q_3\}$	$\emptyset$
q_2	$\emptyset$	$\{q_0, q_1, q_3\}$	$\{q_0, q_2\}$
q_3	$\emptyset$	$\emptyset$	$\{q_0\}$

20. Sea M_1 es autómata correspondiente al del Ejercicio 2 y M_2 el autómata correspondiente al del Ejercicio 4. Hallar diagramas de transición de autómatas cuyos lenguajes sea $L(M_1) \cup L(M_2)$ y $L(M_1) \cap L(M_2)$ respectivamente.

21. Para cada autómata que se muestra en las Figuras 13, 14, 15, 16 y 17 establezca el conjunto de estados Q, el conjunto de símbolos de input Σ, el estado inicial q_0, el conjunto de estados finales $\mathcal{F}$.

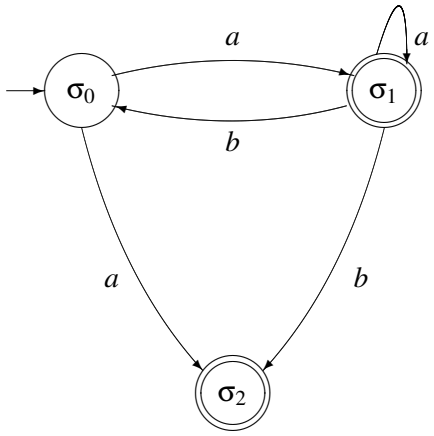


Figura 13:

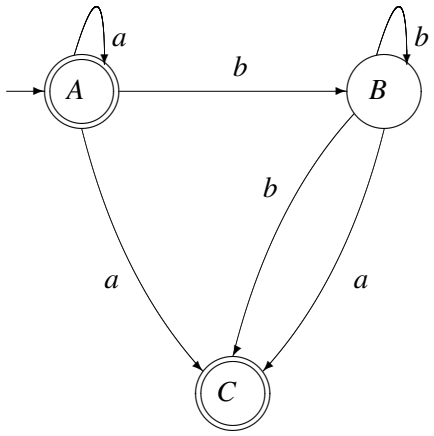


Figura 14:

22. ¿Las cadenas *bbabbb* y *bbabab* son aceptadas por el autómata de la Fig. 18? Pruebe sus respuestas.

23. Demuestre que una cadena α es aceptada por el autómata de la Fig. 18 si y sólo si α tiene una sola letra *a* y termina en *b*.

24. ¿Las cadenas *aaabba* y *aaaab* son aceptadas por el autómata de la Fig. 19? Pruebe sus respuestas.

25. Determine el lenguaje aceptado por los autómatas de las Fig. 20, 21 y 22. Encuentre un autómata determinístico con el mismo lenguaje en cada caso.

26. Caracterice los arreglos aceptados por el autómata de la Fig. 19.

27. Verifique que las cadenas aceptadas por el autómata de la Fig. 15 son las cadenas en $\{a,b\}$ que terminan en *bab*.

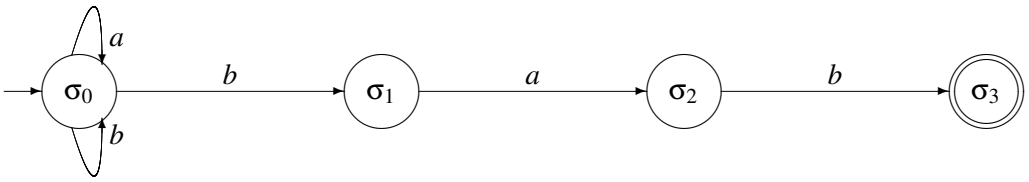


Figura 15:

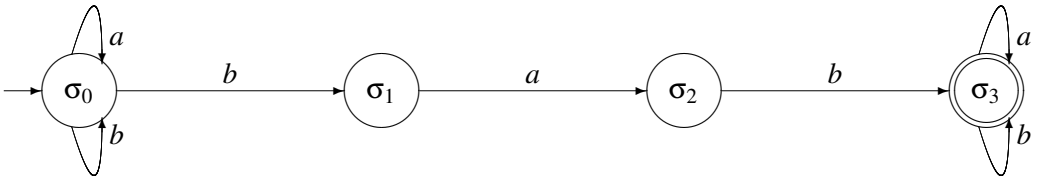


Figura 16:

28. Caracterice las cadenas aceptadas por los autómatas del Ejercicio 19.

29. Caracterice las cadenas aceptadas por los autómatas de las Figuras 13, 14, 16y 17.

30. Diseñe autómatas no determinísticos que acepten las cadenas no nulas sobre $\{a,b\}$ y que tengan las siguientes propiedades.

    (a) Comienzan con *abb* o con *ba*.

    (b) Terminan con *abb* o con *ba*.

    (c) Contienen *abb* o *ba*.

    (d) Contienen *bab* y *bb*.

    (e) Toda *b* se encuentra entre dos *a*.

    (f) Comienzan con *abb* y terminan con *ab*.

    (g) Comienzan con *ab* pero no terminan con *ab*.

    (h) No contienen *ba* o *bbb*.

    (i) No contienen *abba* o *bbb*

31. Caracterice el lenguaje aceptado por cada uno de los autómatas del Ejercicio 1.

32. Caracterice el lenguaje aceptado por los autómatas definidos en los Ejercicios 5 y 6.

33. Describir en palabras los conjuntos denotados por las siguientes expresiones regulares.

    (a) $(11+0)^*(00+1)^*$

    (b) $(1+01+001)^*(\varepsilon+0+00)$

34. Encontrar expresiones regulares en el alfabeto $\{a,b\}$ que describan los siguientes conjuntos:

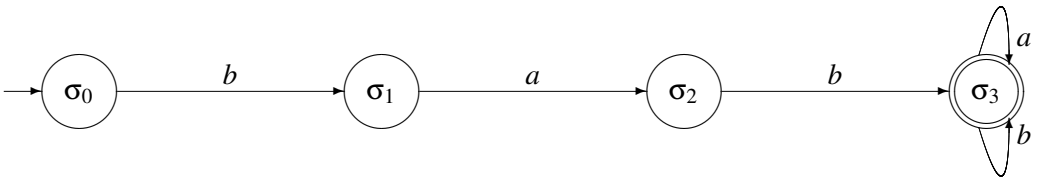


Figura 17:

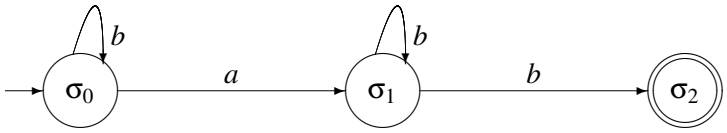


Figura 18:

(a) Cadenas con un número par de letras a.

(b) Cadenas con exactamente una letra b.

(c) Cadenas con al menos una letra b.

(d) Cadenas con exactamente dos letras a.

(e) Cadenas con al menos dos letras a.

(f) Cadenas que contengan m letras a, donde m es un múltiplo de 3.

(g) Cadenas que empiecen con baa.

(h) Cadenas que contengan $abba$.

(i) Cadenas donde toda letra b esté seguida de una letra a.

(j) Cadenas que terminen con aba

(k) Cadenas que empiecen con ab y terminen con aba

35. Construir autómatas finitos cuyo lenguaje sea dado por las siguientes expresiones regulares.

(a) $10+(0+11)0^*1$

(b) $01[((10)^*+111)^*+0]^*1$

(c) $((0+1)(0+1))^*+((0+1)(0+1)(0+1))^*$

36. Encontrar expresiones regulares en el alfabeto $\{a,b\}$ que describan los siguientes conjuntos:

(a) Comienzan con abb o con ba.

(b) Terminan con abb o con ba.

(c) Contienen abb o ba.

(d) Contienen bab y bb.

(e) Toda b se encuentra entre dos a.

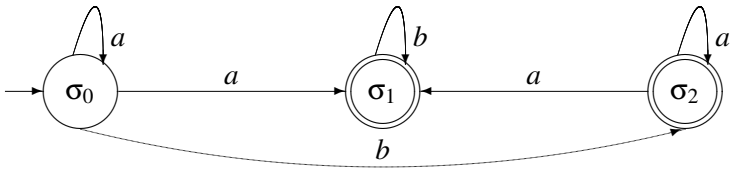


Figura 19:

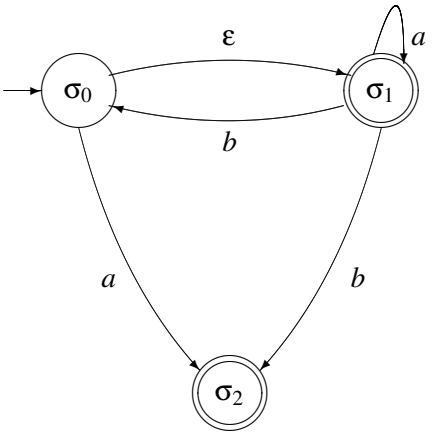


Figura 20:

(f) Comienzan con *abb* y terminan con *ab*.

(g) Comienzan con *ab* pero no terminan con *ab*.

(h) No contienen *ba* o *bbb*.

(i) No contienen *abba* o *bbb*

37. (a) Dibuje un autómata finito determinístico que acepte exactamente el lenguaje de las cadenas de 0's y 1's que no tienen más de tres ceros consecutivos.

(b) Escriba la expresión regular correspondiente.

38. Verificar si los siguientes lenguajes son o no regulares:

(a) $L_1 = \{0^n 1^n 1^m 0^m \mid n \in \mathbb{N}\}$.

(b) $L_2 = \{1^n 0^n 1^m 0^m \mid n,\ m \in \mathbb{N}\}$.

(c) $L_3 = \{0^n 110^n, n \geq 0\}$.

(d) $L_4 = \{ab^{n+2}baa^n\}$.

(e) $L_5 = \{110^n 110^n, n \geq 0\}$,

3 Gramáticas y lenguajes

En este capítulo introduciremos las gramáticas libres de contexto y regulares y los lenguajes que ellas describen (los lenguajes libres de contexto y regulares). Los lenguajes libres de contexto son de gran importancia en la definición de los lenguajes de programación, entre otra aplicaciones. Como

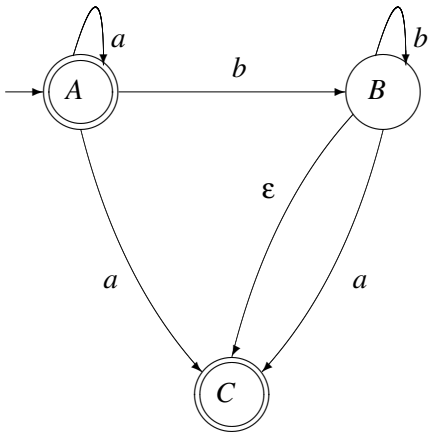


Figura 21:

un ejemplo, los lenguajes libres de contextos son útiles para describir expresiones aritméticas con paréntesis balanceados y para describir estructuras de bloque en los lenguajes de programación.

3.1 Definiciones básicas y ejemplos

Una *gramática libre de contexto* es un conjunto finito de *variables* (también llamadas *no terminales* o *categorías sintácticas*), un conjunto de símbolos llamados *terminales* y un conjunto de reglas llamadas *producciones* que transforman una variable en una cadena formada por variables y terminales.

La motivación original de las gramáticas libres de contexto fue la descripción de los lenguajes naturales. Podemos escribir reglas como:

$$\begin{aligned}
< \text{oración} > &\rightarrow < \text{sujeto} >< \text{predicado} > \\
< \text{sujeto} > &\rightarrow < \text{sujeto} >< \text{adjetivo} > \\
< \text{sujeto} > &\rightarrow \text{el perro} \\
< \text{adjetivo} > &\rightarrow \text{pequeño} \\
< \text{adjetivo} > &\rightarrow \text{grande} \\
< \text{adjetivo} > &\rightarrow \text{bueno} \\
< \text{predicado} > &\rightarrow < \text{verbo} >< \text{adjetivo} > \\
< \text{verbo} > &\rightarrow \text{es}
\end{aligned}$$

donde las categorías sintácticas son denotadas por los $<, >$ y los terminales son “el perro", “pequeño", “grande", “bueno" y “es". Ahora deseamos ver que frases se pueden armar con terminales partiendo de la variable $<$ oración $>$ y usando las reglas de producción. Por ejemplo

$$\begin{aligned}
< \text{oración} > &\Rightarrow < \text{sujeto} >< \text{predicado} > \\
&\Rightarrow < \text{sujeto} >< \text{adjetivo} >< \text{predicado} > \\
&\Rightarrow \text{el perro} < \text{adjetivo} >< \text{predicado} > \\
&\Rightarrow \text{el perro} < \text{adjetivo} >< \text{verbo} >< \text{adjetivo} > \\
&\Rightarrow \text{el perro bueno} < \text{verbo} >< \text{adjetivo} > \\
&\Rightarrow \text{el perro bueno es} < \text{adjetivo} > \\
&\Rightarrow \text{el perro bueno es grande}
\end{aligned}$$

El símbolo $\Rightarrow$ denota la acción de derivar, es decir, reemplazar una variable por el lado derecho de una producción para la variable.

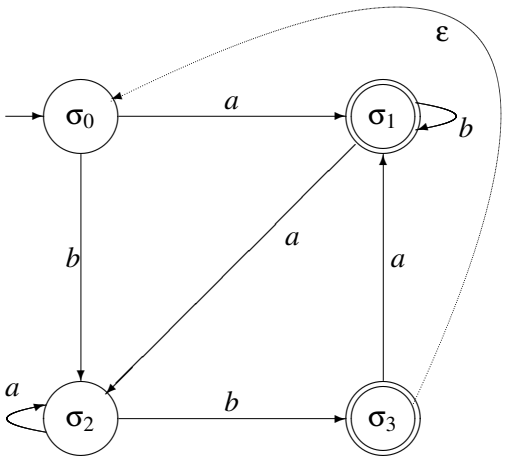


Figura 22:

Pese a su origen, las gramáticas libres de contexto no son adecuadas para la descripción de lenguajes naturales, una razón importante de este hecho es que es necesaria información semántica, además de la sintáctica, para poder construir frases correctas en castellano. Por ejemplo, aunque las reglas de producción anteriores son aparentemente “naturales" también podemos armar la siguiente frase “el perro grande es pequeño".

Ahora, formalizaremos los conceptos expresados previamente.

Definición 3.1. Una *gramática libre de contexto (CFG)* es una 5-upla $G = (V,T,P,S)$ tal que

1. V es un conjunto finito. Los elementos de V son llamados *variables* o *no terminales* o *categorías sintácticas*.

2. T es un conjunto finito disjunto con V. Los elementos de T son llamados *terminales*.

3. S es una variable especial que llamaremos el *símbolo inicial*.

4. P es un conjunto finito P, cuyos elementos son llamados *producciones*, tal que cada producción es de la forma $A \to \alpha$, donde A es una variable y α es una cadena formada por variables y terminales (que puede ser vacía), es decir $\alpha \in (V \cup T)^*$.

En el futuro usaremos la siguiente notación: si $A \to \alpha_1, A \to \alpha_2, \ldots, A \to \alpha_n$ son producciones para la variable A en alguna gramática, podemos expresar esto como

$$A \to \alpha_1 \mid \alpha_2 \mid \cdots \mid \alpha_n.$$

Definición 3.2. Sea $G = (V,T,P,S)$ una gramática. Si $A \to \alpha$ es una producción y xAy es una cadena formada por variables y terminales, se dice que $x\alpha y$ *se deriva directamente* de xAy y se escribe

$$xAy \Rightarrow a\alpha y.$$

Si $\alpha_1, \ldots, \alpha_n$ cadenas y $\alpha_1 \Rightarrow \alpha_2, \alpha_2 \Rightarrow \alpha_3, \ldots, \alpha_{n-1} \Rightarrow \alpha_n$ derivaciones directas, entonces decimos que α_n se deriva de α_1 y se escribe

$$\alpha_1 \Rightarrow \alpha_n.$$

La secuencia

$$\alpha_1 \Rightarrow \alpha_2 \Rightarrow \alpha_3 \Rightarrow \cdots \Rightarrow \alpha_{n-1} \Rightarrow \alpha_n$$

se llama *derivación* de α_1 a α_n. Por convención, cualquier cadena se deriva de si misma.

Finalmente, el *lenguaje generado* por G consiste de todas las cadenas de elementos de T que se derivan de S. Se denota $L(G)$. Los lenguajes generados por las gramáticas libres de contexto se llaman *lenguajes libres de contexto*.

Ejemplo 3.1. Consideremos la gramática con una variable E, símbolos terminales $+$, $*$, $($, $)$ y id, y producciones

$$E \to E+E \mid E*E \mid (E) \mid id \quad .$$

Entonces $(id+id)*id$ se deriva de E, pues

$$\begin{aligned} E &\Rightarrow E*E \\ &\Rightarrow (E)*E \\ &\Rightarrow (E)*id \\ &\Rightarrow (E+E)*id \\ &\Rightarrow (E+id)*id \\ &\Rightarrow (id+id)*id \end{aligned}$$

La primera línea se obtiene usando la producción $E \to E*E$, la segunda se obtiene reemplazando la primera E por el lado derecho de la producción $E \to (E)$. Las restantes líneas se obtienen de aplicar sucesivamente las producciones $E \to id$, $E \to E+E$, $E \to id$, y $E \to id$.

Ejemplo 3.2. Consideremos la gramática con una variable S , símbolos terminales a y b, y producciones $S \to aSb \mid ab$. Veamos que el lenguaje generado por esta gramática es el de todas las cadenas en el alfabeto $\{a,b\}$ que son de la forma $a^n b^n$ con $n > 0$.

Demostración. Hay esencialmente una única forma de obtener una cadena con símbolos terminales: primero aplicar repetidas veces la producción $S \to aSb$ y luego terminar con la producción $S \to ab$. Es decir que las derivaciones son del tipo

$$S \Rightarrow aSb \Rightarrow a^2Sb^2 \Rightarrow \cdots \Rightarrow a^{n-1}Sb^{n-1} \Rightarrow a^n b^n.$$

Entonces es claro que las palabras generadas por el lenguaje son de la forma $a^n b^n$ y obviamente toda cadena de la forma $a^n b^n$ se obtiene haciendo la derivación escrita más arriba. □

Ejemplo 3.3. Consideremos la gramática con una variable S, símbolos terminales a y b, y producciones $S \to aSa \mid bSb \mid a \mid b \mid \varepsilon$. Veamos que el lenguaje generado por esta gramática es el de todas las cadenas en el alfabeto $\{a,b\}$ que son capicúa.

Demostración. Por definición, una palabra capicúa es o bien de la forma $x_1x_2\ldots x_nx_n\ldots x_2x_1$ o de la forma $x_1x_2\ldots x_{n-1}x_nx_{n-1}\ldots x_2x_1$, con x_i igual a a o b. Usando sucesivamente las producciones $S \to x_1Sx_1$, $S \to x_2Sx_2$, …, $S \to x_{n-1}Sx_{n-1}$, obtenemos la derivación

$$S \Rightarrow x_1Sx_1 \Rightarrow x_1x_2Sx_2x_1 \Rightarrow \cdots \Rightarrow x_1x_2\ldots x_{n-1}Sx_{n-1}\ldots x_2x_1,$$

si ahora usamos las producción $S \to x_n S x_n$ y luego $S \to \varepsilon$ obtenemos $S \Rightarrow x_1 x_2 \ldots x_n x_n \ldots x_2 x_1$. Por otro lado si aplicamos la producción $S \to x_n$ a $x_1 x_2 \ldots x_{n-1} S x_{n-1} \ldots x_2 x_1$ obtenemos la derivación $S \Rightarrow x_1 x_2 \ldots x_{n-1} S x_{n-1} \ldots x_2 x_1 \Rightarrow x_1 x_2 \ldots x_{n-1} x_n x_{n-1} \ldots x_2 x_1$.

Veamos por inducción la recíproca, es decir que una cadena del lenguaje generado por la gramática es capicúa: la hipótesis inductiva es $S \Rightarrow \alpha S \beta$, entonces $\alpha = x_1 x_2 \ldots x_k$ y $\beta = x_k \ldots x_2 x_1$, con x_i igual a a o b. La inducción se hace sobre la longitud de α. Si la longitud de α es 1 el resultado es trivial. Supongamos que tenemos probado el resultado para cadenas de longitud $k-1$. Sea $S \Rightarrow \alpha S \beta$, con $|\alpha| = k$. Sea $\alpha = x_1 x_2 \ldots x_k$, es claro que la última producción que se usó es $S \to x_k S x_k$, luego tenemos $S \Rightarrow \alpha' S \beta' \Rightarrow \alpha' x_k S x_k \beta' = \alpha S \beta$. Claramente $\alpha' = x_1 x_2 \ldots x_{k-1}$ tiene longitud $k-1$, y entonces por hipótesis inductiva tenemos que $\beta' = x_{k-1} \ldots x_2 x_1$ y por consiguiente $\beta = x_k \ldots x_2 x_1$. Sea ahora una cadena α de símbolos terminales, tal que $S \Rightarrow \alpha$, entonces la derivación es $S \Rightarrow \alpha_1 S \alpha_2 \Rightarrow \alpha_1 x \alpha_2 = \alpha$, con x igual a a, b o ε. Por lo visto más arriba $\alpha_1 x \alpha_2 = \alpha$ es capicúa. □

Una manera alternativa de enunciar las producciones de una gramática es por el empleo de la *forma normal de Backus-Naur* o *BNF*. En una BNF los símbolos no terminales empiezan con "<" y terminan con ">". Las producción $A \to \alpha$ se expresa $A ::= \alpha$.

Ejemplo 3.4. Un entero es una cadena que consiste de un símbolo opcional ($+$ o bien $-$) seguido por un entero o cadena de dígitos (del 0 al 9). La siguiente gramática (escrita en notación BNF) genera todos los enteros.

$$\begin{aligned}
< \text{dígito} > &::= 0 \mid 1 \mid 2 \mid 3 \mid 4 \mid 5 \mid 6 \mid 7 \mid 8 \mid 9 \\
< \text{entero} > &::= < \text{entero con signo} > \mid < \text{entero sin signo} > \\
< \text{entero con signo} > &::= + < \text{entero sin signo} > \mid - < \text{entero sin signo} > \\
< \text{entero sin signo} > &::= < \text{dígito} > \mid < \text{dígito} > < \text{entero sin signo} >
\end{aligned}$$

El símbolo inicial es $<$ entero $>$, las otras variables son $<$ entero con signo $>$, $<$ entero sin signo $>$, $<$ dígito $>$, los símbolos terminales son $0,1,2,3,4,5,6,7,8,9,+,-$.

Por ejemplo la derivación del entero -452 es

$$\begin{aligned}
< \text{entero} > &\Rightarrow < \text{entero con signo} > \\
&\Rightarrow - < \text{entero sin signo} > \\
&\Rightarrow - < \text{dígito} > < \text{entero sin signo} > \\
&\Rightarrow - < \text{dígito} > < \text{dígito} > < \text{entero sin signo} > \\
&\Rightarrow - < \text{dígito} > < \text{dígito} > < \text{dígito} > \\
&\Rightarrow -4 < \text{dígito} > < \text{dígito} > \\
&\Rightarrow -45 < \text{dígito} > \\
&\Rightarrow -452
\end{aligned}$$

Claramente el lenguaje asociado a esta gramática es el de todas las cadenas que empiezan con $+$ o $-$ y sigue una sucesión de dígitos o las cadenas que son una sucesión de dígitos.

La sintaxis de algunos lenguajes de computación de alto nivel, como Pascal, C o FORTRAN, pueden expresarse en BNF.

3.2 Formas normales de Chomsky y Greibach

Dada una gramática libre de contexto G es posible hallar gramáticas con producciones de cierto tipo tal que el lenguaje generado por estas gramáticas sea del mismo que el generado por G.

Definición 3.3. Sea $G = (V,T,P,S)$ una CFG. Diremos que G está en la *forma normal de Chomsky* si todas las producciones son de la forma

$$A \to a \qquad \text{o} \qquad A \to BC,$$

donde $A,B,C \in V$ y $a \in T$. Es decir, si el lado derecho de todas las producciones es o bien un símbolo terminal o dos variables. Diremos que G está en la *forma normal de Greibach* si todas las producciones son de la forma

$$A \to a\alpha,$$

donde $A \in V$, $a \in T$ y $\alpha \in V^*$. Es decir, si el lado derecho de todas las producciones es un símbolo terminal seguido de ninguna o varias variables.

Dada una gramática G es posible hallar gramáticas G' y G'' en forma normal de Chomsky y Greibach, respetivamente, tal que el lenguaje generado por G sea el mismo que el generado por G' o G''.

En lo que resta de la sección veremos como dada una gramática G libre de contexto podremos encontrar en forma algorítmica una gramática equivalente en la forma de Chomsky. El procedimiento consta de los siguientes pasos:

1. Eliminar ε-producciones ($A \to \varepsilon$).
2. Eliminar variables que no llevan a cadenas de símbolos terminales.
3. Eliminar símbolos (variables y terminales) que no pueden ser alcanzados.
4. Eliminar producciones unitarias ($A \to B$).
5. Forma normal de Chomsky.

Comencemos a hacer el procedimiento:

(1) Eliminar ε-producciones. Con este procedimiento obtendremos una gramática con lenguaje $L(G) - \{\varepsilon\}$. Si el lenguaje original contenía a ε, entonces al final agregaremos la producción $S \to \varepsilon$. El procedimiento es como sigue. Primero sea $N = \emptyset$ y ahora:

(a) si $A \to \varepsilon$ es producción, entonces agregar A a N.

(b) Iteramos el siguiente paso hasta que N se estabilice: si $A \to \alpha$ es producción con $\alpha \in N^+$, entonces agregar A a N.

Es claro que el proceso (b) termina debido a que $N \subset V$.

El nuevo conjunto P' de producciones es el siguiente: si $A \to X_1X_2\ldots X_k$ es una producción en P donde $X_i \in V \cup T$, entonces $A \to \alpha_1\alpha_2\ldots\alpha_k$ está en P' si

1. $\alpha_i = X_i$ si $X_i \notin N$,
2. $\alpha_i = X_i$ o $\alpha_i = \varepsilon$ si $X_i \in N$,
3. $\alpha_1\alpha_2\ldots\alpha_k \neq \varepsilon$.

Es decir, si partimos de $P' = \emptyset$, para toda $A \to \alpha$ en P, se agregan a P' las producciones $A \to \alpha'$ donde $\alpha' \neq \varepsilon$ y α' es cualquier combinación que resulte de eliminar algunas variables en N de α . Observar que si S está en N, entonces ε es aceptado por el lenguaje original.

(2) Eliminar variables que no llevan a cadenas de símbolos terminales. Debemos reemplazar el conjunto de variables originales por el conjunto V' que se construye con cierto número de iteraciones: el primer V' se obtiene de agregar todas las variables que figuren del lado izquierdo de producciones del tipo $A \to \beta$ donde $\beta \in T^+$. Luego se itera el siguiente procedimiento hasta que se estabilice V'

(a) el nuevo V' se obtiene agregando todas las variables que figuren del lado izquierdo de producciones del tipo $A \to \beta$ donde $\beta \in$, es decir cuando β es una cadena donde las variables que aparecen son de V'. En pseudocódigo sería

$$V' := V' \cup \{A : A \to \beta, \beta \in (T \cup V')^+\}$$

Terminada la iteración nuestro nuevo conjunto de variables V será igual al último V'. Si S no está en V', entonces es fácil ver que el lenguaje generado por G es vacío.

(3) Eliminar símbolos que no pueden ser alcanzados. Sea $V' = \{S\}$ y $T' = \emptyset$. Iterar de la siguiente forma hasta que V' y T' se estabilicen:

(a) si $A \to \alpha$ es una producción con $A \in V'$, entonces agregar a V' todas las variables que se encuentren en α y agregar a T' todos los símbolos no terminales que se encuentren en α. En pseudocódigo sería

$$V' := V' \cup \{B \in V : A \to \alpha B \beta,\ A \in V',\ \alpha, \beta \in (T \cup V)^*\}$$
$$T' := T' \cup \{a \in T : A \to \alpha a \beta,\ A \in V',\ \alpha, \beta \in (T \cup V)^*\}$$

La iteración finaliza puesto que $V' \subset V$ y $T' \subset T$. Nuestros nuevos V y T serán V' y T', respectivamente, obtenidos por las iteraciones de más arriba.

(4) Eliminar producciones unitarias. Las producciones unitarias son aquellas de la forma $A \to B$ con $A, B \in V$. El proceso de eliminar producciones unitarias es sencillo: recorremos el conjunto de producciones y cada vez que encontramos una producción del tipo $A \to B$ la eliminamos y todas las producciones $B \to \gamma$ las cambiamos por $A \to \gamma$.

(5) Forma normal de Chomsky. En este caso suponemos que hemos completado los procedimientos (1)…(4). Si $A \to \alpha$ es una producción la *longitud de* $A \to \alpha$ se define como la longitud de la cadena α. Para hacer la forma normal de Chomsky procederemos de la siguiente manera:

(a) por cada símbolo terminal a se agrega una nueva variable A y una producción $A \to a$. Ahora, en cada producción de longitud mayor que 1 se reemplaza cada símbolo terminal a por la nueva variable A. Después de hacer esto todas las producciones son del tipo $A \to a$, con a terminal o del tipo $A \to A_1 \ldots A_n$ con $A_1, \ldots, A_n$ variables.

(b) En cada producción de longitud mayor que dos $A \to A_1 A_2 \ldots A_n$ se reemplaza $A_1 A_2$ por una nueva variable B y se agrega la producción $B \to A_1 A_2$. Este procedimiento se itera hasta que no haya producciones de longitud mayor que 2. Es claro que la iteración termina debido as que en cada paso se disminuye la longitud máxima que pueden tener las producciones.

Ejemplo 3.5. Sea $G = (V, T, P, S)$ con $V = \{S, A\}$, $T = \{a, b\}$ y producciones

$$S \to aAS \mid a \qquad A \to SbA \mid SS \mid ba$$

Ejemplo 3.6. Sea $G = (V,T,P,S)$ con $V = \{S,A\}$, $T = \{a,b\}$ y producciones

$$S \to aAS \mid a \qquad A \to SbA \mid SS \mid ba$$

Encontrar una gramática en la forma normal de Chomsky que genere el lenguaje $L(G)$.

No es difícil verificar que esta gramática no necesita que realicemos los procedimientos (1), (2), (3) y (4). Debemos entonces aplicar el procedimiento (5):

(a) Primero agregamos una variable por cada símbolo terminal, X por a e Y por b. También agregamos las producciones $X \to a$ e $Y \to b$. En las producciones de longitud mayor que 1 reemplazamos a por X y b por Y. Obtenemos entonces la gramática $G = (V,T,P,S)$ con $V = \{S,A,X,Y\}$, $T = \{a,b\}$ y producciones

$$\begin{array}{lll} S \to XAS \mid a & \qquad & A \to SYX \mid SS \mid YX \\ X \to a & & T \to b \end{array}$$

(b) Reemplazamos en $S \to XAS$ a XA por U (una nueva variable) y agregamos la producción $U \to XA$. Reemplazamos en $A \to SYX$ a SY por W (una nueva variable) y agregamos la producción $W \to SY$. Obtenemos la gramática $G = (V,T,P,S)$ con $V = \{S,A,X,Y,U,W\}$, $T = \{a,b\}$ y producciones

$$\begin{array}{lll} S \to US \mid a & \qquad & A \to WX \mid SS \mid YX \\ X \to a & & T \to b \\ U \to XA & & W \to SY \end{array}$$

La formal normal de Greibach se puede obtener a partir de la forma normal de Chomsky, pero es mucho más complicado hacerlo y queda fuera de los alcances de este escrito.

3.3 Gramáticas regulares

Los lenguajes definidos por autómatas finitos o, lo que es lo mismo, por expresiones regulares, también pueden ser vistos como los lenguajes que generan ciertas gramáticas, llamadas gramáticas regulares.

Definición 3.4. Sea G una gramática tal que toda producción es de la forma

$$A \to aB \quad \text{o bien} \quad A \to \varepsilon,$$

donde A, B son variables y a es terminal. Entonces diremos que G es una *gramática regular*.

El lenguaje generado por un una gramática regular será llamado *lenguaje regular*.

Observemos primero que podemos considerar permitidas, en las gramáticas regulares, las producciones del tipo $A \to a$ pues se pueden obtener por composición de las permitidas en la definición. Una observación importante es que cualquier cadena con símbolos terminales y no terminales que se obtiene por derivación a partir del símbolo inicial, tiene a lo sumo una variable y ésta se ubica en el extremo derecho de la cadena. Esto implica que cuando se dice que se usa ciertas producciones para realizar una derivación, no haya ambigüedad en la forma en que hay que aplicarla.

Ejemplo 3.7. Encontremos una gramática que genere el lenguaje asociado a la expresión regular a^*b^*. Es claro que el lenguaje asociado a a^*b^* es $L = \{a^n b^m : 0 \leq n, m\}$. Sea G gramática con una variable S, símbolos terminales a y b y las siguientes producciones:

$$\begin{aligned} S &\to aS \mid aT \mid bT \mid \varepsilon, \\ T &\to bT \mid \varepsilon \end{aligned}$$

entonces el lenguaje que genera G es L.

Demostración. Primero veamos que $L \subset L(G)$: sea $a^n b^m$ en L, si $n, m > 0$ usando la producción $S \to aS$, $(n-1)$-veces tenemos $S \Rightarrow a^{n-1}S$, luego usando la producción $S \to aT$ obtenemos $S \Rightarrow a^n T$, después usamos $S \to bT$ m-veces y obtenemos $S \Rightarrow a^n b^m T$, finalmente usando la producción $T \to \varepsilon$ tenemos la derivación $S \Rightarrow a^n b^m$. En el caso que $n = 0$ y $m > 0$ se hace $S \Rightarrow bT \Rightarrow b^m T \Rightarrow b^m$. El caso $n > 0$ y $m = 0$ es similar. Cuando $n = m = 0$, hacemos $S \Rightarrow \varepsilon$ usando la producción $S \to \varepsilon$.

Veamos ahora que $L(G) \subset L$: observemos que si usamos una derivación en esta gramática que agrega un b, entonces no se puede agregar más un a, como (por la definición de gramáticas regulares) los agregados de símbolos terminales sólo se pueden hacer a la derecha, es claro que las a's estarán todas a la izquierda de cualquier b.

□

Ejemplo 3.8. Encontremos una gramática regular G, cuyo lenguaje sea el lenguaje asociado a la expresión regular $r = (0+1)^*00$. Primero observemos que el lenguaje asociado a r es el de todas las cadenas de 0's y 1's que terminan en 00. Una gramática que genera el lenguaje $L(r)$ es la siguiente: los símbolos terminales serán $0, 1$, las variables serán S, T y las producciones serán

$$S \to 0S \mid 1S \mid 0T, \qquad T \to 0U, \qquad U \to \varepsilon.$$

Demostración. Veamos primero que $L(r) \subset L(G)$: sea $x = a_1 a_2 \cdots a_k 00$ una cadena en $L(r)$, es decir que los a_i son 0 o 1 (en forma arbitraria). Si a_1 es 0, hacemos la derivación $S \Rightarrow 0S$, si es 1 hacemos $S \Rightarrow 1S$, es decir que tenemos $S \Rightarrow a_1 S$, en forma análoga obtenemos la derivación

$$S \Rightarrow a_1 S \Rightarrow a_1 a_2 S \Rightarrow \cdots \Rightarrow a_1 a_2 \cdots a_k S.$$

Si a $a_1 a_2 \cdots a_k S$ le aplicamos sucesivamente las producciones $S \to 0T$ y $T \to 0$, obtenemos $S \Rightarrow a_1 a_2 \cdots a_k S \Rightarrow a_1 a_2 \cdots a_k 0T \Rightarrow a_1 a_2 \cdots a_k 00$.

Veamos ahora que $L(G) \subset L(r)$: es claro que si $S \Rightarrow x$ una derivación con x compuesta de símbolos terminales, entonces la última producción que se usó fue $T \to 0$, y entonces la penúltima fue $S \to 0T$. Como las variables siempre están en el extremo derecho, la derivación es $S \Rightarrow x'S \Rightarrow x'0T \Rightarrow x'00 = x$. Es decir $x \in L(r)$.

□

Los ejemplos anteriores se pueden generalizar con el siguiente teorema, cuyo enunciado es un tanto complicado, pero que puede entenderse bien en casos concretos.

Teorema 3.1. *Sea r una expresión regular sobre una alfabeto Σ, entonces existe H una gramática regular con símbolos terminales en Σ tal que $L(r) = L(H)$. Más explícitamente: sea Σ un alfabeto.*

(1) $L(\emptyset) = \emptyset$ es $L(G)$ donde G es una gramática sin producciones.

(2) $L(\varepsilon) = \{\varepsilon\}$es $L(G)$ donde G tiene una única producción $S \to \varepsilon$.

(3) Para cada a en Σ, $L(a) = \{a\}$ *es* $L(G)$ *donde* G *tiene producciones* $S \to Ua$ *y* $T \to \varepsilon$.

Supongamos que r *y* r' *son expresiones regulares tales que* $L(r) = L(G)$, $L(r') = L(G')$*con* $G = (V,\Sigma,P,S)$ *y* $G' = (V',\Sigma,P',S')$. *Supongamos además, sin perdida de generalidad, que* $V \cap V' = \emptyset$, *es decir que no tienen variables comunes. Debido a la definición de gramática regular, el conjunto* P *es de la forma:*

$$\{C_k \to c_k D_k\} \cup \{U_t \to \varepsilon\},$$

para $k = 1,\ldots,m$, $t = 1,\ldots,s$. *Entonces:*

(4) $L(r+r')$ *es* $L(H)$, *donde* H *tiene como variables* $V = V \cup V'$, *la variable inicial igual a* S *y con producciones* $P_H = P \cup P' \cup \{S \to \alpha : \text{ si } S' \to \alpha\}$.

(5) $L(rr') = L(H)$, *donde* H *es la gramática regular con variables* $V_H = V \cup V'$, *variable inicial igual a* S *y con producciones* $P_H = P^{(1)} \cup P'$, *donde* $P^{(1)}$ *es igual a:*

$$\{C_k \to c_k D_k\} \cup \{C_k \to c_k S' : \text{si } D_k \to \varepsilon \in P\}$$

para $k = 1,\ldots,m$.

(6) $L(r^*) = L(H)$, *donde* H *es la gramática regular con variables* $V_H = V$,*variable inicial* S *y producciones:*

$$P_H = P \cup \{S \to \varepsilon\} \cup \{C_k \to c_k S : \text{si } D_k \to \varepsilon \in P\}$$

para $k = 1,\ldots,m$.

Demostración. La demostración es bastante directa, pero a su vez es tediosa. La dejamos como ejercicio para el lector. □

Ejercicio 3.1. Repetir los Ejemplos 3.7 y 3.8 usando el método del Teorema 3.1.

Como ya mencionamos, la recíproca del Teorema 3.1 también es verdadera, es decir, dado un lenguaje generado por una gramática regular G , existe una expresión regular r que denota el mismo lenguaje. En realidad esto lo probaremos de manera indirecta: es sencillo construir a partir de G un autómata no determinístico M con el mismo lenguaje que M. Luego par obtener r usamos el teorema de Kleene.

Proposición 3.2. *Sea* $G = (V,\Sigma,P,S)$ *una gramática regular, entonces existe* $M = (Q,\Sigma,\delta,q_0,F)$ *un NFA (sin* ε*-mov) tal que* $L(M) = L(G)$. *Explícitamente,* $Q = V$, $q_0 = S$, $B \in \delta(A,a)$ *sii* $A \to aB$ *y* $F = \{A \in V : A \to \varepsilon\}$.

La demostración es muy sencilla y se deja a cargo del lector.

4 Autómatas con pila

Hemos visto que los lenguajes que generan las expresiones regulares o, equivalentemente, los lenguajes regulares se pueden obtener a partir de autómatas finitos, y viceversa. En forma análoga, a los lenguajes libres de contexto le corresponden los autómatas con pila. En esta sección veremos la definición de los autómatas con pila, los lenguajes generados por ellos y algunos ejemplos. No probaremos, por estar fuera de los alcances de este texto, la equivalencia entre los lenguajes libres de contexto y los lenguajes aceptados por los autómatas con pila.

Un autómata con pila (PDA) es esencialmente un autómata finito que posee control sobre una pila, es decir una lista de la cual solo se puede "leer", "poner" o "sacar" el primer elemento. Dado el estado actual del autómata y el primer elemento de la pila, un símbolo de input nos llevará (posiblemente en forma no determinística) el estado siguiente y a la modificación que se debe hacer en el primer elemento de la pila. Diremos que una cadena es aceptada por *pila vacía* por el PDA si cuando la aplicamos obtenemos una pila vacía. Diremos que una cadena es aceptada por *estado final* por el PDA si lleva el estado inicial a uno final. Los lenguajes aceptados por los autómatas con pila, tantos los aceptados por pila vacía o por estado final, son los mismos que los aceptados por las gramáticas libres de contexto e incluyen estrictamente a los lenguajes regulares.

Ejemplo 4.1. El lenguaje $L = \{wcw^R : w \in (0+1)^* \text{ y } c \text{ símbolo } \}$ es un lenguaje no regular. Esto es fácil de ver usando el Pumping Lemma. Sin embargo, es un lenguaje libre de contexto generado por la gramática $S \to 0S0 \mid 1S1 \mid c$. Mostraremos a continuación un autómata con pila cuyo lenguaje por pila vacía es L. Los símbolos de input serán, obviamente, 0, 1 y c. Consideremos dos estados q_1 y q_2 y que en la pila se pueden "apilar" tres tipos de objetos rojos (R), verdes (V) y azules (A). La idea para definir el autómata es la siguiente: la pila comenzará con un solo elemento R para indicar, justamente, el comienzo de la pila. Ahora, pensemos a la pila como una "memoria" donde guardaremos la forma de la primera porción de la palabra hasta c: haremos que cada vez que el input sea 0, se agregue a la pila una A y cada vez que sea 1 se agregue una V. Estos inputs no cambian el estado inicial. Cuando ingresa el input c, cambiamos de estado para indicar que tenemos que empezar a leer el final de la palabra. Ahora deberemos desapilar convenientemente, de tal forma que si después de c viene w^R, la pila quede vacía. Por ejemplo, supongamos que w termina en 1, entonces, después de aplicar c, estamos en el segundo estado y la pila tiene en la parte superior una V. Esto indica que la última letra de w es un 1 y por lo tanto, para tener esperanza de que la palabra sea aceptada, la primera letra después de c debería ser un 1. Por lo tanto decimos que si el input es 1 y la pila muestra V, se retira V. Análogamente si en la parte superior hay un A y el input es 0, se retira A. Siguiendo así, si el sufijo de c es w^R, llegaremos a una pila con un solo elemento, el R. El último movimiento, un ε-movimiento, se define de la siguiente manera: si estamos en el segundo estado y la pila muestra el R, se saca el R. Resumiendo: el autómata tendrá las siguientes reglas:

1. Comenzamos en el estado q_1 y con R en la pila.

2. En el caso en que el estado es q_1 el autómata actúa de la siguiente manera. Si se ingresa el símbolo de input 0, agregamos a la pila una A. Si el símbolo de input es 1 agregamos una V. En ambos caso el estado permanece q_1. Si la entrada es c pasamos al estado q_2 y la pila no cambia.

3. En el caso en que el estado es q_2 el autómata actúa de la siguiente manera. Si se ingresa el símbolo de input 0 y el primero de la pila es A, se retira el primer elemento de la pila (es decir la A). Si el símbolo de input es 1 y el primero de la pila es V, se retira el primer elemento de la pila. Si el primer elemento de la pila es R, se lo retira sin esperar input. En todos los casos el estado continúa siendo q_2.

4. En las situaciones no contempladas en los items anteriores, el autómata no hace nada.

Haciendo algunos ejemplos con cadenas particulares es fácil convencerse que las únicas cadenas aceptadas por pila vacía son las de la forma wcw^R.

Definición 4.1. Un *autómata con pila* es una 7-upla $M = (Q, \Sigma, \Gamma, \delta, q_0, Z_0, F)$, en donde

1. Q es un conjunto finito de *estados*;

2. Σ es un alfabeto, el *alfabeto de entrada*;

3. Γ es un alfabeto, el *alfabeto de la pila*;

4. $q_0 \in Q$, el *estado inicial*;

5. $Z_0 \in \Gamma$, el *símbolo inicial* de la pila;

6. $F \subset Q$, los *estados finales*;

7. $\delta : Q \times (\Sigma \cup \{\varepsilon\}) \times \Gamma \to \mathcal{P}_f(Q \times \Gamma^*)$, la *función de transición*, donde $\mathcal{P}_f(Q \times \Gamma^*)$ indica los subconjuntos finitos de $Q \times \Gamma^*$.

Usaremos PDA (por sus siglas en inglés) como sinónimo de "autómata con pila".

En general haremos uso de letras minúsculas del comienzo del alfabeto $(a, b, c, \ldots)$ para denotar los símbolos del alfabeto de entrada (es decir de Σ) y usaremos letras minúsculas, pero del final del alfabeto $(\ldots, x, y, z)$, para denotar cadenas en Σ. Las letras mayúsculas $(A, B, C, \ldots, X, Y, Z)$, denotarán símbolos de la pila (es decir elementos de Γ) y las letras griegas minúsculas $(\alpha, \beta, \gamma, \ldots)$ denotarán elementos de Γ^*.

Observación 4.1. Debemos hacer algunos comentarios acerca de como se debe interpretar δ. Si tenemos, por ejemplo, que $\delta(q, a, Z) = \{(p, \gamma)\}$ esto lo debemos interpretar de la siguiente manera: si q estado y Z cima de la pila, al aplicarle a del alfabeto de entrada, el estado del autómata cambia a p y en la pila se produce el reemplazo de Z por γ, quedando el primer símbolo de γ como cima de la pila. Por ejemplo si estamos en un estado q y la pila es $Z_1Z_2Z_2$, entonces aplicar a usando la regla $\delta(q, a, Z_1) = \{(p, Z_2Z_1Z_2)\}$, hace que pasemos al estado p y que la pila pase a ser $Z_2Z_1Z_2Z_2Z_2$.

Los autómatas con pila permiten acciones no determinísticas, pues

$$\delta(q, a, Z) = \{(p_1, \gamma_1), \ldots, (p_m, \gamma_m)\}$$

indica la posibilidad de hacer diferentes acciones aún con el mismo input. Por otro lado, también están permitidos los ε-movimientos, es decir

$$\delta(q, \varepsilon, Z) = \{(p_1, \gamma_1), \ldots, (p_m, \gamma_m)\}$$

está permitido y permite, sin ningún input, cambiar el estado y la pila.

Notemos que los DFA y NFA definidos en capítulos anteriores son casos especiales de autómatas con pila: basta "olvidarse" de la pila en la definición.

Ejemplo 4.2. Describamos en lenguaje formal el autómata con pila del Ejemplo 4.1: $M = (\{q_1, q_2\}, \{0, 1, c\}, \{A, V, R\}, \delta, q_1, R, \emptyset)$. Observemos que hemos determinado que el conjunto de estados finales es vacío. Esto se debe a que en este caso estamos interesados en el lenguaje aceptado por pila vacía y por lo tanto los estados finales son irrelevantes. La descripción de δ es

$$\begin{array}{lll} \delta(q_1, 0, X) = \{(q_1, AX)\}, & \delta(q_1, 1, X) = \{(q_1, VX)\}, & \delta(q_1, c, X) = \{(q_2, X)\}, \\ \delta(q_2, 0, A) = \{(q_2, \varepsilon)\}, & \delta(q_2, 1, V) = \{(q_2, \varepsilon)\}, & \delta(q_2, \varepsilon, R) = \{(q_2, \varepsilon)\}, \end{array}$$

donde X es un símbolo arbitrario de la pila. Las transiciones no descritas son $\delta(q, a, Z) = \emptyset$.

Dada una cadena del alfabeto de entrada queremos describir formalmente la situación del autómata con pila después de haberse aplicado parte de la cadena. Para ello debe registrarse, sin duda, el estado actual del autómata y el contenido de la pila. Además, registraremos la parte de la cadena que todavía no ha sido aplicada. Formalmente:

Definición 4.2. Sea $M = (Q, \Sigma, \Gamma, \delta, q_0, Z_0, F)$ un PDA. Una *descripción instantánea (ID)* es un triple (q, w, γ), donde $q \in Q$, $w \in \Sigma^*$ y $\gamma \in \Gamma^*$.

Si $(q, aw, Z\gamma)$ y $(p, w, \beta\gamma)$ son dos ID, denotaremos

$$(q, aw, Z\gamma) \vdash (p, w, \beta\gamma)$$

si $(p, \beta) \in \delta(q, a, Z)$. Dadas (q, w, γ) y (q', w', γ') descripciones instantáneas, denotaremos

$$(q, w, \gamma) \vdash^* (q', w', \gamma')$$

si existen $(q_1, w_1, \gamma_1), \ldots, (q_m, w_m, \gamma_m)$ descripciones instantáneas tales que

$$(q, w, \gamma) \vdash (q_1, w_1, \gamma_1) \vdash \cdots \vdash (q_m, w_m, \gamma_m) \vdash (q', w', \gamma').$$

Por convención, será aceptado $(q, w, \gamma) \vdash^* (q, w, \gamma)$.

Las descripciones instantáneas serán útiles para definir el lenguaje aceptado por un PDA.

Definición 4.3. Sea $M = (Q, \Sigma, \Gamma, \delta, q_0, Z_0, F)$ un PDA. Entonces, definimos $L(M)$ el *lenguaje de M por estado final*, como

$$L(M) = \{w \in \Sigma^* : (q_0, w, Z_0) \vdash^* (p, \varepsilon, \gamma) \text{ para algún } p \in F, \gamma \in \Gamma^*\}.$$

El *lenguaje de M por pila vacía* es:

$$N(M) = \{w \in \Sigma^* : (q_0, w, Z_0) \vdash^* (p, \varepsilon, \varepsilon) \text{ para algún } p \in Q\}.$$

Como ya hemos mencionado el conjunto de los lenguajes aceptados por estado final es equivalente al conjunto de los lenguajes aceptados por pila vacía. La demostración de este hecho sigue el espíritu de las demostraciones que hemos visto en teoría de lenguajes, dado un lenguaje definido de cierta manera, construimos en forma algorítmica un autómata que acepta ese lenguaje.

Recordemos que si M es un NFA, puede asociarse trivialmente M' un PDA, que esencialmente es el mismo M con una pila que no se usa. Es claro entonces, por las respectivas definiciones, que el lenguaje de M coincide con el lenguaje de M' por estado final.

Ejemplo 4.3. $L = \{aa^R : a \in \{0,1\}^*\}$

En la sección 3.2 hemos visto que toda gramática se puede reducir a una gramática equivalente en la forma normal de Greibach. La siguiente proposición probará el "implica" de la equivalencia entre lenguajes generados por gramáticas libres de contexto y lenguajes aceptados por autómatas con pila.

Proposición 4.1. *Sea G una CFG en la forma normal de Greibach. Entonces existe M un PDA tal que* $L(G) = N(M)$.

Demostración. Denotemos $L = L(G)$ y hagamos la demostración en el caso en que ε no esté en L. La demostración en el otro caso es similar y se deja a cargo del lector. G está en la forma normal de Greibach, es decir toda producción es de la forma $A \to a\gamma$ con $\gamma \in V^*$. Definamos el PDA

$$M = (\{q\}, T, V, \delta, q, S, \{\emptyset\}),$$

donde $(q, \gamma) \in \delta(q, a, A)$ si y sólo si $A \to a\gamma$ está en P. Observemos que en este caso los estados del autómata no tiene importancia (siempre estamos en el mismo estado) y lo único que importa es la

pila. Las operaciones elementales que hace δ sobre la pila claramente simulan las producciones y una composición de estas operaciones simula una derivación a izquierda. De manera formal debemos demostrar que

$$S \Rightarrow w \qquad \text{si y sólo si} \qquad (q, w, S) \vdash^* (q, \varepsilon, \varepsilon),$$

para $w \in T^*$. En realidad es más sencillo demostrar

$$S \Rightarrow w\gamma \qquad \text{si y sólo si} \qquad (q, w, S) \vdash^* (q, \varepsilon, \gamma), \tag{5}$$

para $w \in T^*$ y $\gamma \in V^*$ y se deja como ejercicio para el lector. Claramente, esto implica lo anterior haciendo $\gamma = \varepsilon$. □

Finalizaremos la sección dando la definición de autómatas con pila determinísticos.

Definición 4.4. Un *autómata con pila determinístico* es una 7-upla $M = (Q, \Sigma, \Gamma, \delta, q_0, Z_0, F)$, en donde

1. Q es un conjunto finito de *estados*;
2. Σ es un alfabeto, el *alfabeto de entrada*;
3. Γ es un alfabeto, el *alfabeto de la pila*;
4. $q_0 \in Q$, el *estado inicial*;
5. $Z_0 \in \Gamma$, el *símbolo inicial* de la pila;
6. $F \subset Q$, los *estados finales*;
7. $\delta : Q \times (\Sigma \cup \{\varepsilon\}) \times \Gamma \to \{\emptyset\} \cup (Q \times \Gamma^*)$, la *función de transición*, tal que si $\delta(q, \varepsilon, Z) \in Q \times \Gamma^*$, entonces $\delta(q, a, Z) = \emptyset$ para todo $a \in \Sigma$.

Usaremos DPDA (por sus siglas en inglés) como sinónimo de "autómata con pila determinístico".

Observemos que las transiciones de un DPDA tienen las siguientes restricciones:

1. $| \delta(q, a, Z) | = 0$ o 1.
2. Si $| \delta(q, \varepsilon, Z) | > 0$, entonces $| \delta(q, a, Z) | > 0$ para todo $a \in \Sigma$.

Haremos algunos comentarios respecto a los DPDA

1. los lenguajes aceptados por pila vacía correspondientes a DPDA son lenguajes que tienen la siguiente propiedad: si x y y son cadenas aceptadas y $x \neq y$, entonces x no puede ser prefijo de y. En particular, hay lenguajes regulares que no son aceptados por DPDA por pila vacía (por ejemplo 0^*).
2. Consideremos los lenguajes aceptados por DPDA por estado final, en este caso este conjunto de lenguajes contiene estrictamente a los lenguajes regulares y está contenido estrictamente en los lenguajes aceptados por PDA. Por ejemplo, el lenguaje $L = \{aa^R : a \in \{0,1\}^*\}$ es un lenguaje aceptado por un PDA, pero no es posible obtenerlo como lenguaje de un DPDA.

4.1 Ejercicios

1. Demostrar que los enteros se pueden generar con una gramática regular.
2. Encontrar gramáticas libres de contexto que generen los siguientes conjuntos
   (a) Todas las cadenas distintas de ε definidas sobre $\{a,b\}$.
   (b) Cadenas definidas sobre $\{a,b\}$ que empiecen con a.
   (c) Cadenas definidas sobre $\{a,b\}$ que terminen en ba.
   (d) Cadenas definidas sobre $\{a,b\}$ que contengan ba.
   (e) Cadenas definidas sobre $\{a,b\}$ que no terminen en ab.
   (f) Enteros que no empiecen con 0 (hacerlo con BNF).
   (g) Números con punto flotante (como 0.294, 89.0, 45.895).
   (h) Números exponenciales (que incluyan a los números con punto flotante y a otros como 6.9E4, 5E23, 7.5E-3, 4E-5).
3. Sea G la gramática con símbolo inicial S y derivaciones

$$S \to bS \mid aA \mid a, \qquad A \to aS \mid bB, \qquad B \to bA \mid aS \mid b$$

   (donde a y b son los símbolos terminales).
   (a) Demuestre, proporcionando la derivación correspondiente, que las siguientes cadenas pertenecen a $L(G)$

$$aaabb, \qquad bbbaaaaa, \qquad abaaabbabbaa.$$

   (b) Probar que $L(G)$ es el conjunto de todas las cadenas con un número impar de símbolos a.
4. Sea G la gramática regular definida por las producciones

$$S \to bS \mid aA \mid b, \qquad A \to aS \mid bA \mid a$$

   (donde a y b son los símbolos terminales). Demuestre que $\alpha \in L(G)$ sii $\alpha \neq \varepsilon$ y contiene un número par de símbolos a.
5. Demuestre que el lenguaje

$$\{a^n b^n c^k \mid n,k \in \mathbb{N}\}$$

   es libre de contexto, pero no es regular.
6. Obtener un autómata finito no determinístico que acepte únicamente las cadenas generadas por la siguiente gramática regular G: las variables son S y C, con S como variable inicial; las constantes son a,b y las producciones son

$$S \to bS, \quad S \to aC, \quad C \to bC, \quad C \to b.$$

7. Sea G la gramática regular definida por las producciones

$$S \to bS \mid aA \mid b, \qquad A \to aS \mid bA \mid a$$

   (donde a y b son los símbolos terminales). Obtener un autómata finito no determinístico$M$ tal que $L(M)$ es el lenguaje generado por G.

8. Sea L_1 (respectivamente L_2) el lenguaje generado por la gramática del Ejercicio 3 (respectivamente, Ejercicio 4). Encuentre una gramática regular que genere el lenguaje L_1L_2.

9. Demuestre que el conjunto L de cadenas sobre a,b, definido por

$$L = \{x_1 \cdots x_n \mid x_1 \cdots x_n = x_n \cdots x_1\}$$

(es decir las cadenas capicúas), no es un lenguaje regular.

10. Dada la expresión regular

$$b(a^* + b)^* bb^* a$$

construir

(a) Un autómata finito que acepte exactamente el lenguaje que denota la expresión regular.

(b) Una gramática que genera exactamente el lenguaje que denota la expresión regular.

11. Dada la expresión regular

$$(a + bb)^*(b^* aa + b)^*$$

construir:

(a) Un autómata finito que acepte exactamente el lenguaje que denota la expresión regular.

(b) Una gramática que genera exactamente el lenguaje que denota la expresión regular.

12. Dada la expresión regular

$$(ab)^*(ab^* aa + b)^*$$

construir:

(a) Un autómata finito que acepte exactamente el lenguaje que denota la expresión regular.

(b) Una gramática que genera exactamente el lenguaje que denota la expresión regular.

13. Dada la expresión regular

$$0(0^* + 1)^* 00^* 1$$

construir

(a) Un autómata finito que acepte exactamente el lenguaje que denota la expresión regular.

(b) Una gramática que genera exactamente el lenguaje que denota la expresión regular.

14. Probar que el lenguaje $L = \{0^n 1^n 1^m 0^m \mid n \in \mathbb{N}\}$ es libre de contexto pero no es regular.

15. Probar que el lenguaje $L = \{1^n 0^n 1^m 0^m \mid n,\ m \in \mathbb{N}\}$ es libre de contexto pero no es regular.

16. Dado el lenguaje $L = \{0^n 110^n, n \geq 0\}$,

(a) Probar que es libre de contexto.

(b) Encontrar la forma normal de Chomsky.

(c) Probar que no es regular.

17. Considere el lenguaje $L = \{ab^{n+2}baa^n\}$.

(a) Pruebe que L es libre de contexto.

(b) Encontrar la forma normal de Chomsky.

(c) Determine si L es regular o no. Justifique su respuesta.

18. Dado el lenguaje $L = \{110^n 110^n, n \geq 0\}$,

(a) Probar que es libre de contexto.

(b) Encontrar la forma normal de Chomsky.

(c) Probar que no es regular.