Tema 1: Inducción y Recurrencia

Los ejercicios importantes de este tema son:

- Inducción
- Recurrencias homogéneas y no homogéneas:

Veamos algunos ejemplos:

Ejercicios Inducción

Pasos a seguir:

1) Comprobar que se verifica para el primer natural del conjunto
2) Suponemos cierto para $n \in \mathbb{N}$ (Hipótesis de Inducción)
3) Probamos para $n+1$

① La sucesión de los números de Fibonacci se define de la siguiente forma:

$$F_0 = 0, \quad F_1 = 1, \quad F_n = F_{n-1} + F_{n-2}, \quad \forall n \geq 2$$

Demuestra cada una de las siguientes propiedades

a) $F_{n+2} > 2 \cdot F_n, \quad \forall n \geq 2$

- Para $n=2$ tenemos que $F_4 = F_3 + F_2 = 2+1 = 3 > 2 = 2 \cdot F_2$
- Suponemos cierto para $n \in \mathbb{N}$ y probamos para $n+1$:

$$F_{(n+1)+2} = F_{n+3} = F_{n+2} + F_{n+1} \underset{\text{P. H. Inducción}}{>} 2 \cdot F_n + F_{n+1} = F_n + F_n + F_{n+1} > F_{n-1} + F_n + F_{n+1} = 2F_{n+1}$$

b) $\sum_{i=0}^{n} (F_i)^2 = F_n \cdot F_{n+1}, \quad \forall n \geq 0$

- Para $n=0$ tenemos $\sum_{i=0}^{0} F_i^2 = F_0^2 = 0^2 = 0 = 0 \cdot 1 = F_0 \cdot F_1$
- Suponemos cierto para $n \in \mathbb{N}$ y probamos para $n+1 \in \mathbb{N}$:

$$\sum_{i=0}^{n+1} (F_i)^2 = F_{n+1}^2 + \sum_{i=0}^{n} (F_i)^2 \underset{\text{H. I}}{=} F_{n+1}^2 + F_n \cdot F_{n+1} = F_{n+1} (F_{n+1} + F_n) = F_{n+1} \cdot F_{n+2}$$

c) 5 divide a $F_{5n}, \quad \forall n \geq 0$

- Para $n=0$ tenemos que $F_{5 \cdot 0} = F_0 = 0$. Sabemos que $5 \mid 0 \iff \exists x \in \mathbb{N} : 5 \cdot x = 0$. Tomando $x=0$, demostramos que $5 \mid 0$
- Suponemos cierto para $n \in \mathbb{N}$ y vemos para $n+1$:

$$F_{(n+1) \cdot 5} = F_{5n+5} = F_{5n+4} + F_{5n+3} = F_{5n+3} + F_{5n+2} + F_{5n+3} = 2(F_{5n+3}) + F_{5n+2} = 2(F_{5n+2} + F_{5n+1}) + F_{5n+1} + F_{5n} =$$

$$= 2F_{5n+2} + 3F_{5n+1} + F_{5n} = 2F_{5n+1} + 2F_{5n} + F_{5n} + 3F_{5n+1} = 5F_{5n+1} + 3F_{5n}.$$

Por H. Inducción sabemos que $5 \mid 3F_n$. Por otro lado $5 \mid 5F_{5n+1}$, ya que al estar F_{5n+1} multiplicado por 5 significa que es múltiplo de 5, luego $5 \mid (5F_{5n+1} + 3F_{5n}) \Rightarrow 5 \mid F_{(n+1)5}$.

d) $\text{mcd}(F_n, F_{n+1}) = 1, \quad \forall n \geq 0$

- Para $n=0$ tenemos que $\text{mcd}(F_0, F_1) = \text{mcd}(0, 1) = 1$.
- Suponemos cierto para $n \in \mathbb{N}$ y probamos para $n+1$:

$$\text{mcd}(F_{n+1}, F_{n+2}) = \text{mcd}(F_{n+1}, F_{n+1} + F_n) = \text{mcd}(F_{n+1}, F_n) \overset{\text{H.I.}}{=} 1$$

② Demuestra que $5^{n+1} + 2 \cdot 3^n + 1$ es múltiplo de 8

- Para $n=0$ tenemos que $5^1 + 2 \cdot 3^0 + 1 = 5 + 2 + 1 = 8$, que es múltiplo de 8.
- Suponemos cierto para $n \in \mathbb{N}$ ($5^{n+1} + 2 \cdot 3^n + 1 = 8K \Rightarrow 5^{n+1} = 8K - 2 \cdot 3^n - 1$) y vemos para $n+1$:

$$5^{(n+1)+1} + 2 \cdot 3^{n+1} + 1 = 5^{n+2} + 2 \cdot 3^{n+1} + 1 = 5 \cdot 5^{n+1} + 2 \cdot 3^{n+1} + 1 \overset{\text{H.I}}{=} 5(8K - 2 \cdot 3^n - 1) + 2 \cdot 3^{n+1} + 1 =$$

$$= 40K - 10 \cdot 3^n - 5 + 6 \cdot 3^n + 1 = 40K - 4 \cdot 3^n - 4$$

. Como cada sumando es múltiplo de 8 $\Rightarrow$ la suma es múltiplo de 8.

Ejercicios Ecuaciones en Recurrencia Homogéneas

Pasos a seguir:

Opcional) Obtener ecuación lineal en recurrencias

1) Hallar ecuación característica de la recurrencia

2) Resolver la ecuación para hallar los valores propios con su multiplicidad

3) Construir solución general de la siguiente forma : $\sum_j (A_i x^i + \cdots + A_1 x^1 + A_0)(\lambda_j)^n$, donde λ_j es el valor propio e i su multiplicidad

Opcional) Hallar solución particular usando los primeros términos de la solución.

① Obtener una recurrencia lineal homogénea para cada una de las siguientes sucesiones:

a) $x_n = 4n + 1$

$$\left.\begin{array}{l} x_n = 4n+1 \\ x_{n-1} = 4(n-1)+1 = 4n-3 \end{array}\right\} \Rightarrow x_n - x_{n-1} = 4$$

es una ecuación en recurrencias. La reescribimos para que sea homogénea:

$$\left.\begin{array}{l} x_n - x_{n-1} = 4 \\ x_{n-1} - x_{n-2} = 4 \end{array}\right\} \Rightarrow x_n - 2x_{n-1} + x_{n-2} = 0$$

Otra forma:

- $4n+1 = 1^n(4n+1) \Rightarrow (x-1)^2 = x^2 - 2x + 1 \Rightarrow$

$\Rightarrow x_n - 2x_{n-1} + x_{n-2} = 0$

② Resolver las siguientes ecuaciones homogéneas en recurrencia:

a) $x_0 = 1$, $x_1 = 1$, $x_n = 2x_{n-1} - x_{n-2}$, para $n \geq 2$.

Como tenemos $x_n - 2x_{n-1} + x_{n-2} = 0 \Rightarrow$ La ecuación característica es $(x^2 - 2x + 1) = 0 \Rightarrow (x-1)^2 = 0$

La solución de la ecuación característica es:

- $x = 1$, con multiplicidad doble

La solución general es: $(An + B)(1)^n = An + B = x_n$

Para encontrar la solución particular:

$$\left.\begin{array}{l} \cdot\ x_0 = 1 = B \\ \cdot\ x_1 = 1 = A + B \end{array}\right\} \Rightarrow x_n = 1$$

b) $x_0 = 1$, $x_1 = 1$, $x_2 = 2$, $x_n = 5x_{n-1} - 8x_{n-2} + 4x_{n-3}$, para $n \geq 2$

La ecuación característica viene determinada por : $x^3 - 5x^2 + 8x - 4 = 0 \Rightarrow (x-1)(x-2)^2 = 0$

Las soluciones de la ecuación característica son:

- $x = 1$, con multiplicidad 1
- $x = 2$, con multiplicidad 2

La solución general será: $(A)(1)^n + (Bn + C)(2)^n = x_n$

Para encontrar la solución particular:

$$\left.\begin{array}{l} \cdot\ x_0 = 1 = A + C \\ \cdot\ x_1 = 1 = A + 2B + 2C \\ \cdot\ x_2 = 2 = A + 8B + 4C \end{array}\right\} \Rightarrow A = 2,\ B = \frac{1}{2},\ C = -1 \Rightarrow x_n = 2 + \left(\frac{1}{2}n - 1\right)(2)^n$$

Ejercicios Ecuaciones en Recurrencia no Homogéneas

Pasos a seguir: Igual que los homogéneas, pero en el término independiente aparecerá $(A_n x^n + \cdots + A_1 x + A_0)(\lambda)^n$, por lo que a la ecuación característica le añadimos $(x - \lambda)^{n+1}$. Si aparece el polinomio solo es porque $\lambda = 1$

① $x_0 = 0$, $x_n = 2x_{n-1} + 1$, para $n \geq 1$

Como tenemos $x_n - 2x_{n-1} = 1(1)^n \Rightarrow$ La ecuación característica es: $(x-2)(x-1) = 0$

A partir de aquí se resuelve igual.

② $x_0 = 0$, $x_n - 2x_{n-1} = (n+1)3^n$, para $n \geq 1$

Como tenemos $x_n - 2x_{n-1} = (n+1)3^n \Rightarrow$ La ecuación característica es: $(x-2)(x-3)^2 = 0$

A partir de aquí se resuelve igual

Tema 2: Álgebra de Boole

Los ejercicios importantes de este tema son:

- comprobar si un conjunto es un álgebra de Boole
- calcular átomos, coátomos y representar conjuntos D(n)
- Expresar un número como supremo de átomos o como ínfimo de coátomos
- Construir una expresión booleana
- Dada una función booleana calcular:
  - sus formas canónicas disyuntivas y conjuntivas
  - implicantes primos mediante Quine, consensos, FCC y Karnaugh
  - formas canónicas disyuntivas reducidas
  - formas no simplificables mediante Karnaugh y Petrick

Def: Un álgebra de Boole es una seis-upla $(A, \vee, \wedge, {}^*, 0, 1)$ donde A es un conjunto no vacío, $\vee$ y $\wedge$ son operaciones binarias, * es una operación monaria y 0 y 1 son elementos de A. Además, $\forall a,b,c \in A$, se cumple:

A0)	Asociatividad	$a \vee (b \vee c) = (a \vee b) \vee c$	$a \wedge (b \wedge c) = (a \wedge b) \wedge c$
A1)	conmutatividad	$a \vee b = b \vee a$	$a \wedge b = b \wedge c$
A2)	Distributividad	$a \vee (b \wedge c) = (a \vee b) \wedge (a \vee c)$	$a \wedge (b \vee c) = (a \wedge b) \vee (a \wedge c)$
A3)	complementación	$a \vee a^* = 1$	$a \wedge a^* = 0$
A4)	Identidad	$a \vee 0 = a$	$a \wedge 1 = a$

Ejercicios Comprobar si es un Álgebra de Boole:

1. Sea I el conjunto de los números reales que pertenecen al intervalo cerrado $[0,1]$. Para todo $a,b \in I$ definimos $a \vee b = \max\{a,b\}$, $a \wedge b = \min\{a,b\}$ y $\bar{a} = 1-a$. ¿Es I con estas operaciones un álgebra de Boole?

No es un álgebra de Boole porque:

$$a \vee \bar{a} = \max\{a, \bar{a}\} = \max\{a, 1-a\} = \begin{cases} a \leq 0{,}5 \Rightarrow \max\{a, 1-a\} = 1-a \neq 1, \ \forall a \\ a \geq 0{,}5 \Rightarrow \max\{a, 1-a\} = a \neq 1, \ \forall a,\ a \neq 1 \end{cases} \Rightarrow \text{No cumple A4}$$

Ejercicios D(n), átomos, coátomos y representación

1. Sea $D(70)$ el conjunto de los números naturales que son divisores de 70 con las operaciones mcm, mcd, $x^* = \frac{70}{x}$ y con "0" $= 1$ y "1" $= 70$

a) ¿Es un Álgebra de Boole? En caso afirmativo encuentra sus átomos y coátomos

¿Es Álgebra de Boole?: Para comprobar que un conjunto $D(n)$ es un álgebra de Boole con mcm, mcd, $x^* = \frac{n}{x}$, "0" $= 1$ y "1" $= n$ basta con ver si n puede descomponerse en producto de primos distintos.

$$D(70) = \{1,2,5,7,10,14,35,70\} \Rightarrow (D(70), \overset{\vee}{\text{mcm}}, \overset{\wedge}{\text{mcd}}, x^* = \tfrac{70}{x}, \overset{0}{1}, \overset{1}{70})$$

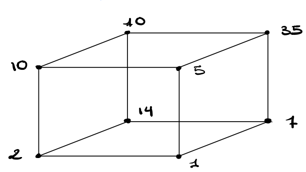
Es un álgebra de Boole, ya que 70 puede descomponerse en producto de primos distintos $(70 = 2 \cdot 5 \cdot 7)$

¿Cómo calcular átomos y coátomos?: En un álgebra de Boole $D(n)$, los átomos son los divisores primos de n y los coátomos sus complementarios

$At(D(70)) = \{1, 2, 5, 7\} \Rightarrow$ Los coátomos $(D(70)) = \{70, 35, 14, 10\}$

b) Representalo gráficamente como conjunto ordenado

¿Cómo es el orden del conjunto?: Se ordena por divisibilidad



c) calcula $35 \wedge (2 \vee 7)$ y $(2 \vee 7) \wedge (14 \wedge 10)$

- $35 \wedge (2 \vee 7) = (35 \wedge 2) \vee (35 \wedge 7) = mcm(mcd(35,2), mcd(35,7)) = mcm(1,7) = 7$
- $(2 \vee 7) \wedge (14 \wedge 10) = [(14 \wedge 10) \wedge 2] \vee [(14 \wedge 10) \wedge 7] = [14 \wedge (10 \wedge 2)] \vee [14 \wedge (10 \wedge 7)] = mcm[mcd(14, mcd(10,12)), mcd(14, mcd(10,7))] =$

$= mcm[mcd(14,2), mcd(14,1)] = mcm(2,1) = 2$

② Justifique que $D(210)$ es un álgebra de Bode y exprese los elementos 21 y 35 como supremo de átomo y como ínfimo de coátomos

$D(210) = \{1,2,3,5,6,7,10,14,15,21,30,35,42,70,105,210\}$. Como $210 = 2 \cdot 7 \cdot 3 \cdot 5$ (producto de primos distintos) $\Rightarrow$

$\Rightarrow$ Es un álgebra de Bode

$At(D(210)) = \{1,2,3,5,7\}$ Coátomos $(D(70)) = \{210, 105, 70, 42, 30\}$

¿Cómo expresar un número como supremo de átomos o ínfimo de coátomos? : Para expresar como supremo de átomos tomamos $mcm(i)$, donde $i \in At$ (el menor número de i posibles). Para expresar como ínfimo de coátomos tomamos $mcd(j)$, $j \in$ coátomos (mayor número posible).

- $21 = \vee\{3,7\} = \wedge\{210,105,42\}$
- $35 = \vee\{5,7\} = \wedge\{210,105,70\}$

$210 = 2 \cdot 3 \cdot 5 \cdot 7$ $\quad 70 = 2 \cdot 5 \cdot 7$ $\quad 30 = 2 \cdot 3 \cdot 5$

$105 = 3 \cdot 5 \cdot 7$ $\quad 42 = 2 \cdot 3 \cdot 7$

③ En el Álgebra de Bode $D(2310)$, expresa 5, 35, 154, 231, 1155 como supremo de átomos y como ínfimo de coátomos

$At(D(2310)) = \{2,3,5,7,11\}$

Coátomos $(D(2310)) = \{210, 330, 462, 770, 1155\}$

$210 = 2 \cdot 3 \cdot 7 \cdot 5$ $\quad 462 = 2 \cdot 3 \cdot 7 \cdot 11$

$330 = 2 \cdot 3 \cdot 5 \cdot 11$ $\quad 770 = 2 \cdot 5 \cdot 7 \cdot 11$

$1155 = 3 \cdot 5 \cdot 7 \cdot 11$

- $5 = \vee\{5\} = \wedge\{210, 330, 770, 1155\}$
- $35 = \vee\{5,7\} = \wedge\{210, 770, 1155\}$
- $154 = \vee\{2,7,11\} = \wedge\{462, 770\}$
- $231 = \vee\{3,7,11\} = \wedge\{462, 1155\}$
- $1155 = \vee\{3,5,7,11\} = \wedge\{1155\}$

④ Sea B un álgebra de Bode con 32 elementos. ¿Cuántos átomos tiene?

$$
\begin{array}{c|c}
1 \quad 1 & 2 \\
1 \quad 2 \quad 1 & 4 \\
1 \quad 3 \quad 3 \quad 1 & 8 \\
1 \quad 4 \quad 6 \quad 4 \quad 1 & 16 \\
1 \quad 5 \quad 10 \quad 10 \quad 5 \quad 1 & 32
\end{array}
\Rightarrow \text{Tiene 5 átomos}
$$

Ejercicios sacar una función booleana

Pasos a seguir:
1) Sacar la tabla proposicional con todos los posibles casos
2) Tomar los minterms (valores cuya $f(x) = 1$)

Opcional) Hallar forma canónica disyuntiva (No simplificada) y simplificarla por Karnaugh

① Un examen de tipo test consta de cuatro preguntas. Las respuestas correctas son: (Sí, No, Sí, Sí). Construye una función booleana que analice cada examen y distinga los aprobados de los suspensos (se considera aprobado si al menos tres respuestas son correctas).

	A	B	C	D	f
0	0	0	0	0	0
1	0	0	0	1	0
2	0	0	1	0	0
3	0	0	1	1	0

A = Pregunta 1

B = Pregunta 2

C = Pregunta 3

D = Pregunta 4

Sí = 1

No = 0

	A	B	C	D	f
4	0	1	0	0	0
5	0	1	0	1	0
6	0	1	1	0	0
7	0	1	1	1	1
8	1	0	0	0	0
9	1	0	0	1	0
10	1	0	1	0	0
11	1	0	1	1	1
12	1	1	0	0	0
13	1	1	0	1	1
14	1	1	1	0	1
15	1	1	1	1	1

Forma No simplificada:

$\boxed{f = \sum m(7, 11, 13, 14, 15)}$ → Minterm: Tomar los que tienen 1 (suma de productos)

$\boxed{f = \bar{A}BCD + A\bar{B}CD + AB\bar{C}D + ABC\bar{D} + ABCD}$ → Forma canónica disyuntiva. Consiste en una suma de productos (Niegas las posiciones que son 0 y mantienes los 1)

Nota: También puede hacerse con los maxterm:

$\boxed{f = \prod M(0,1,2,3,4,5,6,8,9,10,12)}$ → Maxterm: Tomas los que tienen 0

$\boxed{f = (A+B+C+D)(A+B+C+\bar{D})\cdots(A+B+\bar{C}+\bar{D})}$ → Productos de sumas (Mantienes las posiciones que son 0 y niegas los 1) → Forma canónica conjuntiva

2) Un comité formado por tres personas toma decisiones mediante votación por mayoría. Cada miembro del comité puede "votar SI" pulsando un botón. Diseña una red lógica mediante la cual se encienda una luz cuando y sólo cuando haya una mayoría de "votos SI".

Personas = {A, B, C}

	A	B	C	f
0	0	0	0	0
1	0	0	1	0
2	0	1	0	0
3	0	1	1	1
4	1	0	0	0
5	1	0	1	1
6	1	1	0	1
7	1	1	1	1

$\Rightarrow f = \sum m(3,5,6,7) = \bar{A}BC + A\bar{B}C + AB\bar{C} + ABC$

Ejercicios de simplificaciones Booleanas

1) Sea $f_i : B^3 \to B$ la función booleana de tres variables con $i = 63$. Halla:

a) Las formas canónicas disyuntivas y conjuntivas.

¿Cómo saco la tabla? : Pasa el número a binario y asigna los valores por orden

	x	y	z	f
0	0	0	0	0
1	0	0	1	0
2	0	1	0	1
3	0	1	1	1
4	1	0	0	1
5	1	0	1	1
6	1	1	0	1
7	1	1	1	1

$63)_{10} = 00111111)_2$

$\Rightarrow f = \sum m(2,3,4,5,6,7) = \boxed{\bar{x}y\bar{z} + \bar{x}yz + x\bar{y}\bar{z} + x\bar{y}z + xy\bar{z} + xyz}$ Canónica Disyuntiva

$\Rightarrow f = \prod M(0,1) = \boxed{(x+y+z)(x+y+\bar{z})}$ Canónica Conjuntiva

b) Implicantes primos mediante Quine, consensos, FCC y Karnaugh.

- Quine:

Monomios que se agrupan

1 1 1	1 1 –	1 – –
1 1 0	1 – 1	– 1 –
1 0 1	– 1 1	‖
0 1 1	1 – 0	x+y
1 0 0	– 1 0	
0 1 0	1 0 –	
	0 1 –	

Los implicantes primos son x e y

¿Cómo se hace Quine?:

1) Tomamos los minterm y los agrupamos en bloques según su número de 1

2) Comparamos las cadenas de un bloque con las del bloque inferior. Si encontramos dos cadenas que difieren los marcamos y, en una nueva columna representamos estas dos cadenas con "–". Si aparecen dos cadenas iguales se deja solo 1

3) Ir marcando los monomios que se agrupan y repetir hasta no poder crear más columnas. Los monomios no marcados son los integrantes primos

- Karnaugh:

¿Cómo se hace Karnaugh? (con minterm):

1) Dibuja un mapa de Karnaugh para el número de variables que tengas

2) Marca con 1 las casillas de los minterm

3) Toma agrupaciones de tamaño 2n lo más grande que puedas, recuerda que el mapa es "circular"

4) De cada bloque extrae las variables que no alteran su valor y súmalas

Recuerda que este factor está desordenado

z \ xy	00	01	11	00
0		1	1	1
1		1	1	1

y x ⇒ x+y

También, es la forma irreducible

Nos da la forma canónica disyuntiva irreducible de f

- **FCC:**

$f(x,y,z) = (x+y+z)(x+y+\bar{z}) = x + xy + x\bar{z} + yx + y + y\bar{z} + zx + zy + z\bar{z} = x + y \Rightarrow$ Implicantes primos: x, y

¿Cómo se hace Forma canónica conjuntiva (FCC)?: Desarrollamos la expresión y realizamos las absorciones necesarias. Los términos que nos quedan son los primos

- **Consensos:**

$\bar{x}y\bar{z} + \bar{x}yz + x\bar{y}\bar{z} + x\bar{y}z + xy\bar{z} + xyz =$

$= \bar{x}y\bar{z} + \bar{x}yz + \bar{x}y + x\bar{y}\bar{z} + x\bar{y}z + xy\bar{z} + xyz =$

$= \bar{x}y + x\bar{y}\bar{z} + x\bar{y}z + xy\bar{z} + xyz =$

$= \bar{x}y + x\bar{y}\bar{z} + x\bar{y}z + xy\bar{z} + xyz + x\bar{y} =$

$= \bar{x}y + xy\bar{z} + xyz + x\bar{y} =$

$= \bar{x}y + xy\bar{z} + xyz + x\bar{y} + xy =$

$= \bar{x}y + x\bar{y} + xy =$

$= \bar{x}y + x\bar{y} + xy + y =$

$= x\bar{y} + y =$

$= x\bar{y} + y + x = x + y$

¿Cómo se hace el método de los consensos?: se parte de la forma disyuntiva de la función y está basado en la siguiente igualdad del álgebra de Boole: $x\mu_1 + \bar{x}\mu_2 = x\mu_1 + \bar{x}\mu_2 + \mu_1\mu_2$, siendo x cualquier variable. Al monomio $\mu_1\mu_2$ se le llama consenso en x de los monomios $x\mu_1$ y $\bar{x}\mu_2$.

1) Partimos de la forma disyuntiva y se van añadiendo los consensos en todas las variables de todas las parejas de monomios posibles (incluidos los que van apareciendo)
2) simplifica por absorción (Elimina los monomios incluidos en otros)
3) Cuando no aparecen más consensos y se han hecho todas las absorciones, se obtiene la forma normal reducida.

c) Su forma canónica disyuntiva reducida, su forma no simplificable mediante Karnaugh y Petrick.
La forma canónica disyuntiva reducida y su forma no simplificable ya las hemos obtenido de realizar el mapa de Karnaugh en b).

- **Petrick**

¿Cómo se hace Petrick?
1) Comienza escribiendo por filas los implicantes primos y por columnas los mintérminos de la función f.
2) Marca con un aspa cada casilla en caso de que un implicante primo esté contenido en un mintérmino
3) Aquellos implicantes que tengan una columna con un solo aspa son esenciales.

	$\bar{x}y\bar{z}$	$\bar{x}yz$	$x\bar{y}\bar{z}$	$x\bar{y}z$	$xy\bar{z}$	xyz
x			✓	✓	✓	✓
y	✓	✓	✓		✓	✓

Ambos implicantes primos son esenciales $\Rightarrow f(x,y,z) = x+y$

¿Cuándo un implicante primo es esencial?: Cuando hay una columna con un solo ✓.

1) Sea $f_i: B^4 \to B$ la función booleana de 4 variables con $i = 13244$. Halla:

a) Dos formas canónicas disyuntivas y conjutivas.

$f_{13244}: B^4 \to B$, $13244)_{10}$, $0011001110111100)_2$

	x	y	z	t	f
0	0	0	0	0	0
1	0	0	0	1	0
2	0	0	1	0	1
3	0	0	1	1	1
4	0	1	0	0	0
5	0	1	0	1	0
6	0	1	1	0	1
7	0	1	1	1	1
8	1	0	0	0	1
9	1	0	0	1	0
10	1	0	1	0	1
11	1	0	1	1	1
12	1	1	0	0	1
13	1	1	0	1	1
14	1	1	1	0	0
15	1	1	1	1	0

$\Rightarrow f(x,y,z,t) = \sum m(2,3,6,7,8,10,11,12,13) =$

$= \boxed{\bar{x}\bar{y}z\bar{t} + \bar{x}\bar{y}zt + \bar{x}yz\bar{t} + \bar{x}yzt + x\bar{y}\bar{z}\bar{t} + x\bar{y}z\bar{t} + x\bar{y}zt + xy\bar{z}\bar{t} + x\bar{y}zt}$ Forma canónica disyuntiva

$f(x,y,z,t) = \prod M(0,1,4,5,9,14,15) =$

$= \boxed{(x+y+z+t)(x+y+z+\bar{t})(x+\bar{y}+z+t)(x+\bar{y}+z+\bar{t})(\bar{x}+y+z+\bar{t})(\bar{x}+\bar{y}+\bar{z}+t)(\bar{x}+\bar{y}+\bar{z}+\bar{t})}$

Forma canónica conjuntiva

b) implicantes primos

- Karnaugh

zt \ xy	00	01	11	10
00			1	1
01			1	
11	1	1		1
10	1	1		1

Implicantes primos:

$x\bar{z}\bar{t}$
$\bar{x}z$
$\bar{y}z$
$xy\bar{z}$
$x\bar{y}\bar{t}$

Forma irreducible:

$x\bar{z}\bar{t}$
$\bar{x}z$
$\bar{y}z$
$xy\bar{z}$

Forma disyuntiva irreducible:

$f(x,y,z,t) = x\bar{z}\bar{t} + \bar{x}z + \bar{y}z + xy\bar{z}$

Aclaración: En Karnaugh para los implicantes primos tomamos todas las posibles agrupaciones de tamaño 2^n más grandes que podamos, esto hace que halla términos positivos que se eliminan a la hora de calcular la forma irreducible.

- Quine

• 1 1 0 1	1 1 0 –	– 0 1 –
• 1 0 1 1	• 1 0 1 –	0 – 1 –
• 0 1 1 1	• – 0 1 1	
• 1 1 0 0	• 0 1 1 –	
• 1 0 1 0	• 0 – 1 1	
• 0 1 1 0	1 – 0 0	
• 0 0 1 1	1 0 – 0	
• 1 0 0 0	• – 0 1 0	
• 0 0 1 0	• 0 – 1 0	
	• 0 0 1 –	

$$\left.\begin{array}{l} xy\bar{z} \\ x\bar{z}\bar{t} \\ x\bar{y}\bar{t} \\ \bar{y}z \\ \bar{x}z \end{array}\right\} \text{Implicantes Primos}$$

los marcas lilas son los monomios que se agrupan

¿Qué son los implicantes? : un implicante de una función es un monomio fundamental que es menor o igual que la función.
¿Qué son los implicantes primos? : es un implicante de la función tal que al suprimir un literal deja de ser implicante.
¿Qué es un implicante primo esencial? : Es aquel que contiene mintérminos 1 no contenidos dentro de ningún otro primo

- Petrick

		$\bar{x}\bar{y}z\bar{t}$	$\bar{x}\bar{y}zt$	$\bar{x}yz\bar{t}$	$\bar{x}yzt$	$x\bar{y}\bar{z}\bar{t}$	$x\bar{y}z\bar{t}$	$x\bar{y}zt$	$xy\bar{z}\bar{t}$	$xy\bar{z}t$
A	$x\bar{z}\bar{t}$					✓			✓	
B	$\bar{x}z$	✓	✓	✓	✓					
C	$\bar{y}z$	✓	✓				✓	✓		
D	$xy\bar{z}$								✓	✓
E	$x\bar{y}\bar{t}$					✓	✓			

Los implicantes primos son: B, C, D

		$x\bar{y}\bar{z}\bar{t}$
A	$x\bar{z}\bar{t}$	✓
E	$x\bar{y}\bar{t}$	✓

$\Rightarrow$ son intercambiables, podemos elegir cualquiera, por ejemplo A:

$f(x,y,z,t) = A + B + C + D$

Tema 3: Grafos

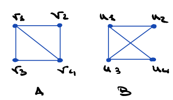
Ejercicios tipo:

- Ver si dos grafos son isomorfos
- Hallar camino mínimo usando algoritmo de Dijstra
- Pasar de grafo a matriz y viceversa
- Ver si existe un grafo de n lados que cumpla determinadas condiciones
- Determinar secuencias en grafos

Ejercicios de grafos isomorfos

¿Cuándo dos grafos son isomorfos? : Dos grafos se dicen isomorfos si existe una función entre los conjuntos de vértices que sea biyectiva y que la imagen de dos vértices adyacentes son adyacentes.
Como es una biyección conserva propiedades invariantes, como el grado.

1. ¿Son isomorfos los siguientes grafos?

a) 

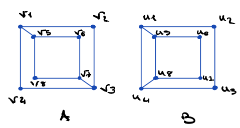
Ambos tienen 5 aristas, 4 vértices y grado 10. Tratamos de establecer el isomorfismo:

$h_V : V_{G_1} \to V_{G_2}$

$v_1 \longmapsto u_1$
$v_2 \longmapsto u_2$
$v_3 \longmapsto u_4$
$v_4 \longmapsto u_3$

$$A = \begin{array}{c} \\ v_1 \\ v_2 \\ v_3 \\ v_4 \end{array} \begin{array}{c} \begin{array}{cccc} v_1 & v_2 & v_3 & v_4 \end{array} \\ \begin{pmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{pmatrix} \end{array} \qquad H_A = \begin{array}{c} \\ u_1 \\ u_2 \\ u_4 \\ u_3 \end{array} \begin{array}{c} \begin{array}{cccc} u_1 & u_2 & u_4 & u_3 \end{array} \\ \begin{pmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{pmatrix} \end{array}$$

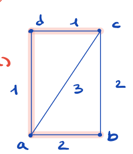
Como $A = H_A$ son isomorfos

b) 

No son isomorfos ya que en A podemos observar que v_5 (vértice de grado 3) es adyacente a dos vértices de grado 2 (v_6, v_7) y a uno de grado 3 (v_1). Sin embargo, en B todos los vértices de grado 3 son adyacentes a dos de grado 3 y uno de grado 2, por lo que no podemos establecer una biyección

Ejercicios Algoritmo de Dijstra :

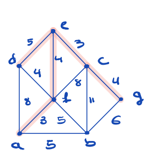
1. Halla el camino mínimo entre cada dos vértices usando el algoritmo de Dijstra:

a) 

	Paso 1	Paso 2	Paso 3	Paso 4
a	(0,a)	*	*	*
b	(2,a)	(2,a)	(2,a)	*
c	(3,a)	(2,d)	(2,d)	(2,d)
d	(1,d)	(1,d)	*	*

¿Cómo se hace Dijstra?

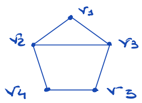
1) Escribimos los vértices por columnas y escogemos un origen.
2) En cada iteración, escribimos un par con el peso del nodo en el camino y a qué nodo está referido. Si un nodo no es adyacente a ninguno de los que conforman el camino se le asigna ∞
3) Pasamos a la siguiente columna el que pese menos, lo marcamos (lo añadimos al camino) y reajustamos los pesos de los nodos

b) 

	Paso1	Paso2	Paso3	Paso4	Paso5	Paso6	Paso7
a	∞	∞	(7,f)	(7,f)	(7,f)	(7,f)	*
b	∞	(14,c)	(9,f)	(9,f)	(9,f)	(9,f)	(9,f)
c	(3,e)	(3,e)	*	*	*	*	*
d	(5,e)	(5,e)	(5,e)	(5,e)	*	*	*
e	(0,e)	*	*	*	*	*	*
f	(4,e)	(4,e)	(4,e)	*	*	*	*
g	∞	(7,c)	(7,c)	(7,c)	(7,c)	*	*

Ejercicio Pasar de Grafo a Matriz

① Expresar en forma matricial el grafo:

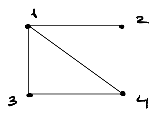


$$\Rightarrow \begin{array}{c|ccccc} & v_1 & v_2 & v_3 & v_4 & v_5 \\ \hline v_1 & 0 & 1 & 1 & 0 & 0 \\ v_2 & 1 & 0 & 1 & 1 & 0 \\ v_3 & 1 & 1 & 0 & 0 & 1 \\ v_4 & 0 & 1 & 0 & 0 & 1 \\ v_5 & 0 & 0 & 1 & 1 & 0 \end{array}$$

¿Cómo se hace?: Marca con un 1 si los nodos están conectados, 0 en caso contrario

② Representa gráficamente el grafo cuya matriz de adyacencia es:

$$\begin{array}{cccc} 1 & 2 & 3 & 4 \end{array}$$
$$\begin{pmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{pmatrix} \Rightarrow$$



Ejercicios Grafo Completo

¿Qué es un grafo completo?: Un grafo se dice completo si cada par de vértices está conectado por una arista.

Los grafos completos verifican la fórmula $\ell = \frac{n(n-1)}{2}$ (ℓ → lados; n → nº vértices)

① ¿Existe un grafo completo con 595 lados?

En un grafo completo, el número de lados es $\ell = \frac{n(n-1)}{2} \Rightarrow 595 = \frac{n^2-n}{2} \Rightarrow n^2 - n - 1190 = 0 \Leftrightarrow$

$\Leftrightarrow \begin{cases} n = 35 \checkmark \\ n = \cancel{-34} \end{cases} \Rightarrow$ Un grafo completo con 35 vértices tiene 595 lados.

② ¿Cuál es el menor número de vértices que puede tener un grafo simple con 1000 lados?

El menor número de vértices que puede tener un grafo simple se dá si todos los vértices están conectados mediante un lado, es decir, si es completo. (Sería como el 1)

③ ¿Cuál es el menor número de vértices que puede tener un grafo conexo con 1000 lados?

Un grafo es conexo si para cualesquiera dos vértices existe un camino que los une. Un grafo completo es un grafo conexo en el que dos vértices se unen mediante un solo lado, luego, nuevamente el ejercicio es como el 1.

Ejercicios de Sucesiones Gráficas

¿Qué es una sucesión gráfica?: Una sucesión finita de enteros no negativos se dice que es una sucesión gráfica si existe un grafo no dirigido simple que tenga por sucesión de grados la sucesión dada.

① Determina cuáles de las secuencias siguientes son gráficas, y para aquellas que lo sean encuentra una realización concreta:

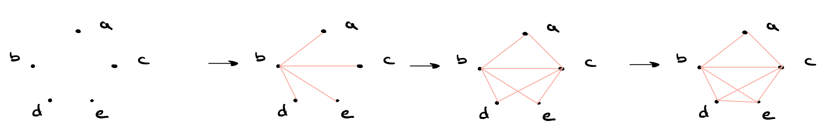
a) 2, 4, 4, 3, 3

a	b	c	d	e	
2	4	4	3	3	pivote b, elegimos a, c, d, e
1	-	3	2	2	pivote c, elegimos a, d, e
-	-	-	1	1	pivote d, elegimos e
-	-	-	-	-	

¿Cómo se hacen las conexiones?

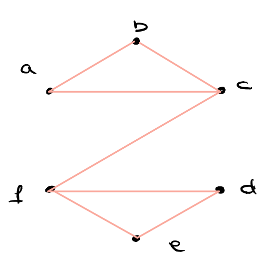
1) Tomamos como pivote el elemento de mayor grado y elegimos sus conexiones

2) Marcamos con "-" los nodos sin más conexiones disponibles y disminuimos en 1 las conexiones de los nodos usados en 1)



b) 2, 2, 3, 2, 2, 3

a	b	c	d	e	f	
2	2	3	2	2	3	pivote c, elegimos a, b, f
1	1	-	2	2	2	pivote d, elegimos e, f
1	1	-	-	1	1	pivote a, elegimos b
-	-	-	-	1	1	pivote e, elegimos f
-	-	-	-	-	-	



c) 7, 6, 5, 4, 3, 3, 2

No es una sucesión gráfica, porque al tener un nodo de grado 7 significa que necesitas conectarlo a otros 7 nodos (como el grafo es simple no puede conectarse a sí mismo creando un lazo), pero solo tienes otros 6 nodos, luego, no puedes crear un grafo simple ⇒ No es sucesión gráfica.

d) 6, 6, 5, 4, 3, 3, 1

a	b	c	d	e	f	g	
6	6	5	4	3	3	1	pivote a, elegimos b, c, d, e, f, g
-	5	4	3	2	2	-	pivote b ⇒ No puedes elegir 5 nodos porque no existen 5 nodos no nulos!!

⇒ No es sucesión gráfica

e) 3, 3, 3, 2, 2

No es una sucesión gráfica, porque necesitas un número par de elementos impares para crear un grafo simple.

a	b	c	d	e	
3	3	3	2	2	pivote a, elegimos b, c, e
-	2	2	1	2	pivote b, elegimos c, e
-	-	1	1	1	pivote c, elegimos d
-	-	-	-	1	Te queda uno suelto!!

Ejercicios de Caminos y Circuitos de Euler

¿Qué es un camino de Euler?: Es un camino que recorre todas las aristas del grafo, pasando por cada una una única vez.

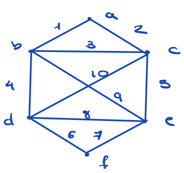
¿Qué es un circuito de Euler?: Es un camino de Euler que comienza y acaba en el mismo nodo

Nota: Diremos que un grafo es euleriano si el grado de todos sus vértices es par y diremos que es semieuleriano si tiene solo un par de vértices de grado impar

¿Cuándo puedo encontrar un circuito de Euler?: cuando el grafo es euleriano y lo hallo usando el algoritmo de Fleury sobre cualquier vértice.

¿Cuándo puedo encontrar un camino de Euler?: cuando el grafo es euleriano y hallo el camino aplicando el algoritmo de Fleury pero sobre uno de los vértices de grado impar

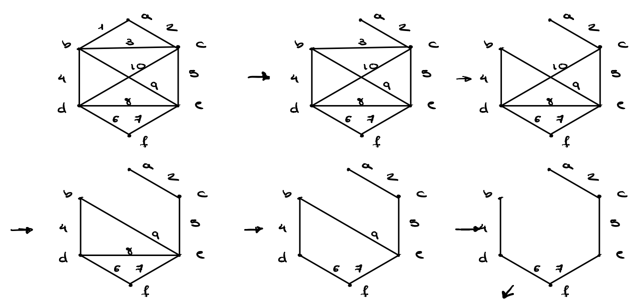
1) Encontrar un camino de Euler para estas figuras:

a) 

Como todos los vértices tienen grado par ⇒ Es un grafo euleriano y, por tanto, podemos hallar un circuito de Euler aplicando el algoritmo de Fleury sobre cualquiera de los vértices.

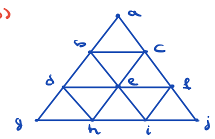
¿Cómo funciona el algoritmo de Fleury?

1) Toma el nodo inicial
2) Se selecciona una arista a partir del vértice actual para continuar que no sea puente, a no ser que no haya otra alternativa.
3) Se elimina la arista seleccionada o no se vuelve a tener en cuenta.
4) Se continúa 2), 3), 4) con un vértice inicial no aislado hasta que se hayan recorrido todas las aristas

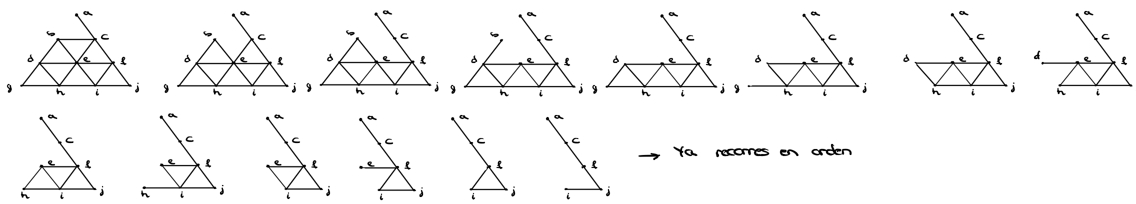


A partir de aquí ya recorres en orden

El circuito de Euler quedaría {a, b, c, d, e, b, d, f, e, c, a}

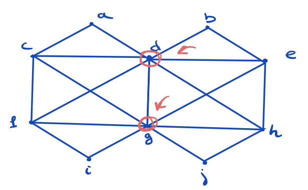
b) 

Como todos los vértices tienen grado par, entonces es un grafo euleriano y, por tanto, admite un camino de euler, que se halla aplicando el algoritmo de Fleury sobre cualquier vértice.



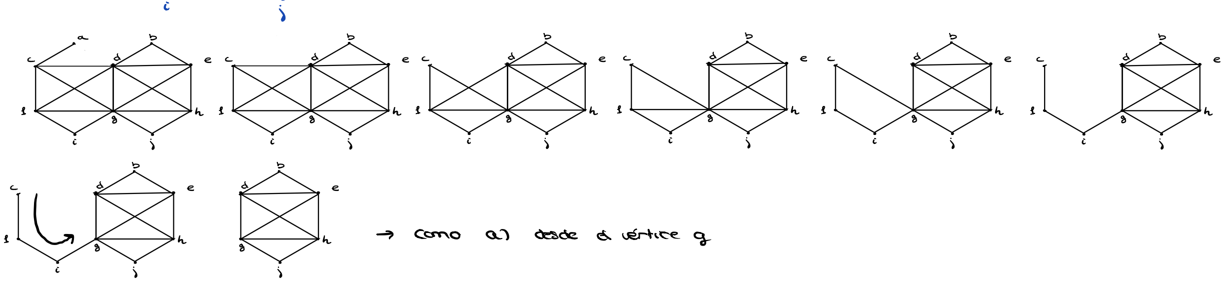
→ Ya recorres en orden

Luego, el circuito de Euler quedaría como {a, b, c, e, b, d, g, h, d, e, h, i, e, f, i, j, f, c, a}

c) 

Hay dos vértices de grado impar. Como tenemos solo dos vértices de grado impar ⇒ ⇒ es un grafo semieuleriano ⇒ admite un camino de Euler.

Aplicamos el algoritmo de Fleury desde el nodo d.



→ Como a) desde el vértice g

② ¿Para qué valores de n el grafo K_n es un circuito de Euler?

Para tener un circuito de Euler necesitamos que todos los vértices sean de grado par. Por otra parte, tenemos que el grado de todos los vértices de un grafo completo es $n-1$ (todos los vértices se conectan entre sí, menos a ellos mismos)

Por lo tanto, para que K_n sea un circuito de Euler debe tener un número de vértices (n) impar.

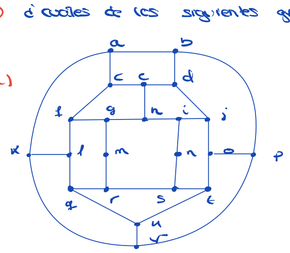
Ejercicios de Circuito Hamilton

¿Qué es un circuito Hamilton?: Es un circuito que recorre todos los vértices del grafo, pasando por cada uno una única vez.

¿Cuándo un grafo es hamiltoniano?: Sea G un grafo de n vértices, entonces tenemos:

- Si el número de lados es mayor o igual que $\frac{1}{2}(n-1)(n-2)+2$ ⇒ El grafo es hamiltoniano
- Si $n \geq 3$ y para cada par de vértices no adyacentes se verifica $gr(v)+gr(w) \geq n$ ⇒ El grafo es hamiltoniano

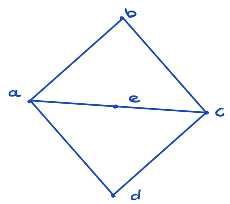
① ¿Cuáles de los siguientes grafos contiene un circuito hamiltoniano?

a) 

No contiene un circuito hamiltoniano porque ($n \geq 3$) tomando dos vértices no adyacentes, por ejemplo k y p, no se cumple que $gr(k)+gr(p) = 2+2 = 4 \geq 22$!!

$n = 22$ vértices

b)



$n = 5$

Tampoco contiene un circuito hamiltoniano porque $gr(b) + gr(d) = 4 \not> 5$

② ¿Cuándo $K_{m,n}$ es un grafo de Hamilton?

Siendo $K_{m,n}$ un grafo bipartido completo, donde m es el número de vértices de una partición y n es el número de vértices de otra partición.

Se cumple que si $n = m \Rightarrow$ grafo Hamilton

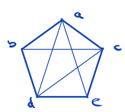
Ejercicios de grafos planos

¿Qué es un grafo plano?: Se dice que un grafo es plano cuando puede ser dibujado en el plano sin que ninguna arista se cruce.

Teorema de Kuratowski: Un grafo es plano si, y sólo si, no contiene un subgrafo que es subdivisión elemental de K_5 o $K_{3,3}$

① Determina cuáles de los siguientes grafos son planos.

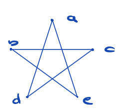
a)



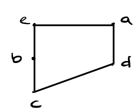
$v = 5$
$l = 9$

Aplicando el teorema de Kuratowski este grafo es plano ya que ningún subgrafo puede contraerse a K_5 o a $K_{3,3}$. Esto es porque tiene menos lados que K_5 y $K_{3,3}$, luego, no se puede realizar la contracción.

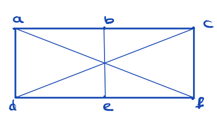
b)



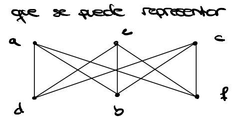
Este grafo es plano ya que ningún subgrafo puede contraerse a K_5 o $K_{3,3}$. Además, podemos encontrar una representación plana



c)



Este grafo no es plano, ya que se puede representar como $K_{3,3}$. Intercambiando posiciones de los vértices b y e:



② Demuestra que cualquier grafo con cuatro vértices o menos es siempre plano.

Esto lo podemos demostrar usando el teorema de Kuratowski, ya que al tener solo 4 vértices no lo podemos contraer a K_5 (que tiene 5 vértices) o $K_{3,3}$ (que tiene 6 vértices).

③ Demuestra que si un grafo tiene a lo sumo cinco vértices y uno de ellos es de grado entero, entonces es plano.

Un grafo de máximo 5 vértices y uno de ellos de grado 2 lo podemos contraer a uno de 4 vértices. Esto lo hacemos identificando el vértice de grado 2 con los vértices adyacentes.

④ Sea un grafo plano y conexo con nueve vértices, de los cuales, 3 son de grado 2, 3 de grado 3, 2 de grado 4 y uno de grado 5. ¿Cuántos lados hay? ¿Y caras?

Como el grafo es plano y conexo cumple la característica de Euler, es decir:

$$v - l + c = 2$$

Además sabemos que:

$$gr(v_1) + \cdots + gr(v_n) = 2l$$

Por lo tanto:

$2l = 3\cdot2 + 3\cdot3 + 2\cdot4 + 1\cdot5$

$l = 14$

$v - l + c = 2$ 9 vértices

$9 - 14 + c = 2$ $\Rightarrow$ 14 lados

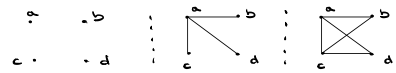
$c = 7$ 7 caras

Ejercicios polinomio cromático

(1) Encuentre, si existe, un grafo G de cuatro vértices con grados {3,2,3,2}. Utiliza el algoritmo de demolición - reconstrucción. Calcula su polinomio cromático $p_G(x)$, su número cromático y de cuántas formas se puede pintar con 6 colores.

a	b	c	d	
3	2	3	2	pivote a, elegimos b, c, d
-	1	2	1	pivote c, elegimos b, d
-	-	-	-	

Reconstrucción $\Rightarrow$

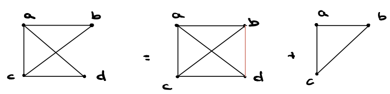


¿Qué es el polinomio cromático? : Dado un grafo no dirigido, G, y un número natural, $k \geq 1$, llamaremos polinomio cromático de G a la función de k que nos da el número de formas de colorear G con k colores ($P_G(k)$)

Algoritmo de la suma: La fórmula $P(G_e, x) = p(G, x) + p(G_e', x)$ nos indica que dado un grafo simple, G_e, se reduce a sumar los polinomios cromáticos de G y G_e', donde G tiene un lado más que G_e, y G_e' tiene un vértice menos

Usando el algoritmo de la suma tenemos que:

$P(K_n, x) = x(x-1)\cdots(x-n+1)$



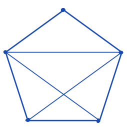
$= p(\boxed{K_4, x}) + p(K_3, x) = x(x-1)(x-2)(x-3) + x(x-1)(x-2) = P(G, x) \Rightarrow$

$\Rightarrow P(G, 6) = 6(6-1)(6-2)(6-3) + 6(6-1)(6-2) = 480$ formas distintas de colorear un grafo con 6 colores.

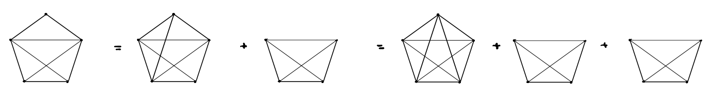
¿Qué es el número cromático? : El menor número de colores para pintar el grafo. El número cromático será el grado del sumando de menor grado (en $K_n = n$)

En nuestro caso, el número cromático es 3.

(2) Halla el polinomio cromático, el número cromático y de cuántas formas se puede colorear el siguiente grafo con 4 colores.



Usando el algoritmo de la suma tenemos:

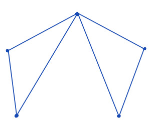


$= 2P(K_4, x) + P(K_3, x) \Rightarrow$

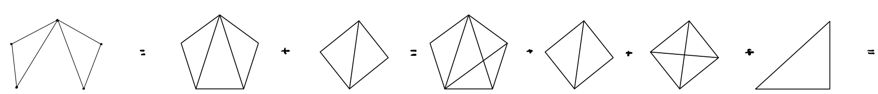
$\Rightarrow P(G, x) = 2x(x-1)(x-2)(x-3) + x(x-1)(x-2)(x-3)(x-4) \Rightarrow P(G, 4) = 48$

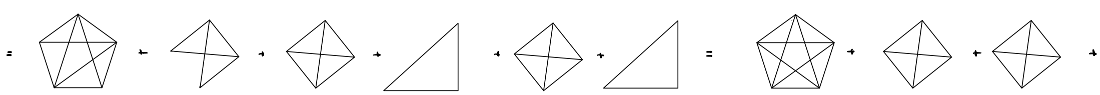
El número cromático es 4.

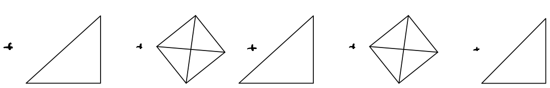
(3) Halla el polinomio cromático del siguiente grafo y calcula de cuántas formas se puede colorear con 5 colores.



Usaremos el algoritmo de la suma







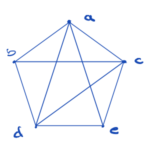
$= P(K_5, x) + 4P(K_4, x) + 3P(K_3, x)$

Ejercicios Árbol Generador

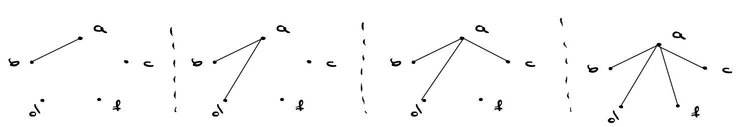
¿Qué es un árbol? : Un árbol es un grafo conexo y acíclico.

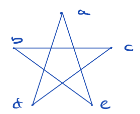
¿Qué es un árbol generador? : El árbol generador de un grafo conexo es un subgrafo que tiene todos los vértices del grafo y es un árbol.

(1) Calcular un árbol generador para los siguientes grafos:

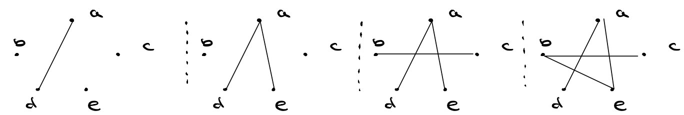
a) 

Como tiene 5 vértices, cogemos 4 lados que no formen ciclo



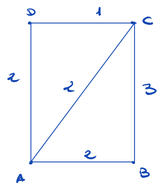
b) 

Como tenemos 5 vértices cogemos 4 lados sin ciclo



Ejercicios árbol generador mínimo (Algoritmo de Kruskal y Prim)

① Dados los grafos ponderados, hallar usando los algoritmos de Kruskal y Prim los árboles generadores de peso mínimo.

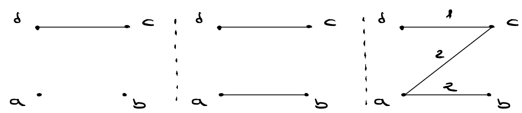
a) 

Algoritmo de Kruskal (constructivo):

ordenamos los lados por ponderación:

cd, ab, ac, ad, bc

Como tiene 4 vértices cogemos los 3 primeros lados que no formen ciclos:



¿Cómo se hace Kruskal?

1) Ordenamos los lados por peso de menor a mayor

2) Vamos tomando lados, de forma que no se originen ciclos

3) Repetir 2) hasta tener n-1 lados, donde n es el número de vértices

¿Cómo se hace el algoritmo de Prim?

Se trabaja con vértices y lados. Se parte de un vértice arbitrario v, que pasa al conjunto de los elegidos T={v} y E={ }. En cada paso se añade un vértice u a T y un lado e a E, con las siguientes condiciones:

1) u sea un vértice que no está en T.

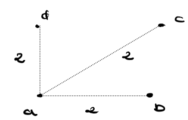
2) u sea adyacente a un vértice de T

3) El lado e no forme ciclos

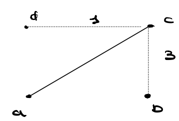
4) e sea el de peso mínimo entre los que cumplen 1), 2), 3).

Algoritmo de Prim:

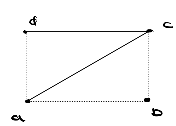
Partimos de a



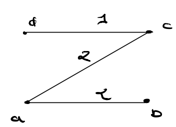
Dibujamos los lados incidentes y cogemos el que no forme ciclos y sea de peso mínimo



Repetimos con c)



Solo tenemos una opción para que no se formen ciclos

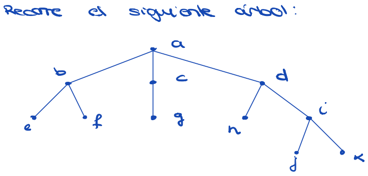


Vértices	Lados	Peso
{a}	{}	0
{a,c}	{ac}	2
{a,c,d}	{ac, cd}	3
{a,c,d,b}	{ac, cd, ab}	5

Ejercicios Recorrido Árboles

- Preorden: se recorre la raíz y después se recorren los subárboles hijos en preorder en orden creciente del rango de sus raíces.
- Postorden: Primero se recorren los subárboles hijos en postorder y después se recorre la raíz.
- Inorden: Primero se recorre el subárbol hijo de menor rango en inorder, después la raíz, y después el resto de árboles hijos.
- Top-Down: se recorren los nodos en orden creciente de profundidad o nivel.
- Bottom-up: se recorren los nodos en orden creciente de altura, dentro de la misma altura en orden creciente de profundidad y dentro de la misma altura y profundidad en orden creciente de rango.

① Recorre el siguiente árbol:



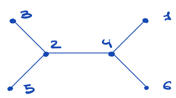
- Preorden: a b e f c g d h i j k
- Postorden: e f b g c h j k i d a
- Inorder: e b f a g c h d j i k
- Top-down: a b c d e f g h i j k
- Bottom-up: e f g h j k b c i d a

Ejercicios Códigos de Prüfer

¿ Qué es un árbol etiquetado ? : Es un árbol con n vértices en donde los vértices tienen como etiquetas números naturales.

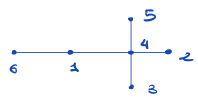
① Determinar los códigos de Prüfer de los siguientes árboles:

a)



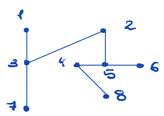
(4, 2, 2, 4)

b)



(4, 4, 4, 1)

c)



(3, 5 3, 2 5, 4)

¿ Cómo se obtienen los códigos de Prüfer ?: Suponemos los árboles etiquetados con los números naturales de 1 a n. Llamaremos código de Prüfer de un árbol etiquetado a la sucesión de longitud n-2, que se obtiene:

1) Se parte de la sucesión vacía ( ) y del árbol etiquetado T.

2) Si T dos vértices hemos acabado.

3) Se determina la hoja con menor etiqueta de T y se añade al código la etiqueta del adyacente al seleccionado.

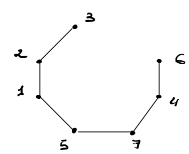
4) T pasa a ser un árbol con la hoja seleccionada suprimida.

5) Vuelve a 2) con el nuevo código y el nuevo árbol.

② Representa los árboles etiquetados con los códigos de Prüfer

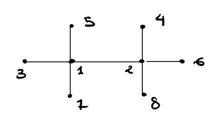
a) (2, 1, 5, 7, 4) Tiene 7 vértices y 6 lados

{1, 2, 3, 4, 5, 6, 7}



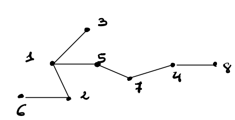
b) (1, 2, 1, 2, 1, 2) Tenemos 8 vértices y 7 lados

{1, 2, 3, 4, 5, 6, 7, 8}



c) (1, 2, 1, 5, 7, 4) Tiene 8 vértices, 7 lados

{1, 2, 3, 4, 5, 6, 7}



¿ Cómo se genera un árbol etiquetado con un código dado?

1) Se parte del código c de longitud n-2, del conjunto V = {1, 2, ..., n} y T el grafo vacío con n vértices etiquetados.

2) Si c es vacío y T tiene dos vértices se pone un lado entre los vértices de V, se devuelve T y se acaba.

3) Se determina el menor número de V que no está en c, se añade a T un lado entre ese menor número y el primer elemento del código. Se quita de V el elemento seleccionado y c su primer elemento.

4) Vuelve a 2) con el nuevo código, el nuevo conjunto V y el nuevo grafo T.

Tema 4: Lógica Proposicional

Para este tema haré un resumen de lo más importante de teoría y resolveremos algunos ejercicios.

Llamamos **conexiones lógicas** a símbolos o palabras que se utilizan para conectar dos enunciados, proposiciones o sentencias, de modo que el valor de verdad de la fórmula compuesta depende del valor de verdad de las fórmulas componentes. Diremos que una **sentencia** es **atómica** cuando los enunciados del lenguaje no se pueden dividir en otros.

Usaremos las primeras letras minúsculas del alfabeto griego para simbolizar los enunciados atómicos y las letras minúsculas a partir de la p para representar los enunciados de lo que nos interesa conocer el significado.

Para dos enunciados, α, β, las partículas conectivas que emplearemos son:

Negación ($\neg\alpha$):

α	$\neg\alpha$
0	1
1	0

Conjunción ($\alpha \wedge \beta$):

α	β	$\alpha \wedge \beta$
0	0	0
0	1	0
1	0	0
1	1	1

Disyunción ($\alpha \vee \beta$):

α	β	$\alpha \vee \beta$
0	0	0
0	1	1
1	0	1
1	1	1

Implicación ($\alpha \to \beta$):

α	β	$\alpha \to \beta$
0	0	1
0	1	1
1	0	0
1	1	1

Doble Implicación ($\alpha \leftrightarrow \beta$):

α	β	$\alpha \leftrightarrow \beta$
0	0	1
0	1	0
1	0	0
1	1	1

Llamamos **interpretación** a una aplicación $v: \Sigma \to \mathbb{Z}_2$, donde Σ es un conjunto sobre el que se construye un lenguaje proposicional. Esta aplicación le asigna un valor de verdad a cada proposición atómica. Si $v(a) = 0$ diremos que es falsa, y si $v(a) = 1$, es verdadera.

Una forma muy útil de expresar el valor de la verdad de una proposición es a través del **polinomio de Gegalkine**. Para obtenerlo, dispondremos de los siguientes cambios:

- $\alpha \vee \beta = v(\alpha) + v(\beta) + v(\alpha)v(\beta)$
- $\alpha \wedge \beta = v(\alpha)v(\beta)$
- $\alpha \to \beta = 1 + v(\alpha) + v(\alpha)v(\beta)$
- $\alpha \leftrightarrow \beta = 1 + v(\alpha) + v(\beta)$
- $\neg\alpha = 1 + v(\alpha)$

Ejemplo

Para la fórmula $\alpha = \neg(a \to b) \to (\neg a \to \neg b)$ tendríamos:

$\neg(a \to b) = 1 + 1 + a + ab = a + ab$

$\neg a \to \neg b = 1 + 1 + a + (1+a)(1+b) = a + 1 + b + a + ab = 1 + b + ab$

$\alpha = 1 + a + ab + (a+ab)(1+b+ab) = 1 + a + ab + a + ab + ab + ab + ab + ab = 1$

a	b	$a \to b$	$\neg(a \to b)$	$\neg a$	$\neg b$	$\neg a \to \neg b$	α
0	0	1	0	1	1	1	1
0	1	1	0	1	0	0	1
1	0	0	1	0	1	1	1
1	1	1	0	0	0	1	1

A las fórmulas cuyo valor de verdad sea constantemente 1 las llamaremos **tautologías**. Si la fórmula es constantemente igual a 0 la llamaremos **contradicciones**. Si dependiendo del caso, hay alguno que es cierto las llamamos **satisfacible**.

Llamamos **cláusula** a toda fórmula de la forma $\lambda_0 \vee \dots \vee \lambda_n$, donde λ_i es un **literal**.

Toda cláusula con un literal recibe el nombre de **cláusula unit**. Un **literal puro** si aparece en al menos una de las cláusulas y su complementario no aparece en ninguna.

Algoritmo de la forma clausulada

Vamos a dar un método para transformar una proposición γ en otra lógicamente equivalente y que esté clausulada.

1) Eliminación de la bicondicional: si tenemos una subfórmula $\alpha_1 \leftrightarrow \alpha_2$ la sustituimos por $(\alpha_1 \to \alpha_2) \wedge (\alpha_2 \to \alpha_1)$.
2) Eliminación de la condicional: si tenemos una subfórmula $\alpha_1 \to \alpha_2$ la sustituimos por $\neg\alpha_1 \vee \alpha_2$.
3) Interiorización de la negación. Cualquier subfórmula del tipo:
   - $\neg\neg\alpha$ la sustituimos por α
   - $\neg(\alpha_1 \vee \alpha_2)$ la sustituimos por $\neg\alpha_1 \wedge \neg\alpha_2$
   - $\neg(\alpha_1 \wedge \alpha_2)$ la sustituimos por $\neg\alpha_1 \vee \neg\alpha_2$
4) Distribución de la $\vee$ sobre la $\wedge$. Si tenemos una subfórmula de la forma $\alpha_1 \vee (\beta_1 \wedge \beta_2)$ la sustituimos por $(\alpha_1 \vee \beta_1) \wedge (\alpha_1 \vee \beta_1)$. Y lo mismo si la subfórmula es de la forma $(\alpha_1 \vee \alpha_2) \wedge \beta_1$ pasa a $(\alpha_1 \wedge \beta_1) \vee (\alpha_2 \wedge \beta_1)$.
5) Eliminación de redundancias: Absorción de cláusulas más grandes por las más chicas.

Dado un conjunto de fórmulas Γ, posiblemente vacío, y una fórmula φ decimos que Γ implica semánticamente a φ, abreviadamente $\Gamma \models \varphi$, si para toda valoración v se tiene que $v(\varphi) = 1$ siempre que para toda fórmula γ de Γ valga $v(\gamma) = 1$. Si Γ consta solamente de las fórmulas $\gamma_1, \gamma_2, \dots, \gamma_n$, en lugar de $\{\gamma_1, \dots, \gamma_n\} \models \varphi$ escribimos $\gamma_1, \dots, \gamma_n \models \varphi$ y cuando $\Gamma = \emptyset$ escribimos simplemente $\models \varphi$ en lugar de $\emptyset \models \varphi$.

Teorema de la Deducción: Sea Γ un conjunto de fórmulas (que podría ser vacío) de un lenguaje proposicional, y α, β, otras dos fórmulas. Entonces los siguientes afirmaciones son equivalentes:

1) $\Gamma \models \alpha \to \beta$
2) $\Gamma \cup \{\alpha\} \models \beta$

Teorema: Sea Γ un conjunto de fórmulas, Γ' un conjunto de fórmulas que se obtiene sustituyendo cada fórmula de Γ por una forma clausular de esa fórmula, y Γ'' el conjunto que resulta de sustituir cada fórmula de Γ' por las cláusulas que la forman. Entonces son equivalentes:

1) Γ es insatisfacible
2) Γ' es insatisfacible
3) Γ'' es insatisfacible

Algoritmo de Davis-Putman

1) Sea Σ un conjunto de cláusulas. Supongamos que en Σ hay una cláusula unit λ. Entonces Σ es insatisfacible si, y solo si, Σ_λ es insatisfacible.
2) Sea Σ un conjunto de cláusulas. Supongamos que en Σ aparece un literal puro λ, es decir, hay al menos una cláusula en la que aparece el literal λ y el literal λ^c no aparece en ninguna. Entonces Σ es insatisfacible si, y solo si, Σ_λ es insatisfacible.
3) Sea Σ un conjunto de cláusulas y sea λ un literal que aparece en alguna cláusula de Σ. Entonces: Σ es insatisfacible si, y solo si, Σ_λ y Σ_{λ^c} son insatisfacibles.

Método de resolución

Teorema: Sean α, β, γ tres fórmulas en un lenguaje proposicional. Entonces $\{\alpha \vee \beta, \neg\alpha \vee \gamma\} \models \beta \vee \gamma$.

Def: Sean C_1 y C_2 dos cláusulas. Supongamos que λ es un literal tal que aparece en la cláusula C_1 y λ^c aparece en C_2. Una cláusula que sea equivalente a $(C_1 - \lambda) \vee (C_2 - \lambda^c)$ es lo que se denomina una resolvente de C_1 y C_2.

Def: Sea Γ un conjunto de cláusulas, y C una cláusula. Una deducción (por resolución) de C a partir de Γ es una sucesión de cláusulas $C_1, \dots, C_n$, donde:

- $C_n = C$
- $C_i \in \Gamma$ o C_i es una resolvente de dos cláusulas del conjunto $\{C_1, \dots, C_{i-1}\}$

Principio de resolución: Sea Γ un conjunto de cláusulas. Entonces Γ es insatisfacible si, y sólo si, hay una deducción por resolución de la cláusula vacía.

Estrategia de saturación

1) Paramos bien cuando encontremos la cláusula vacía, bien cuando no se pueden calcular más resolventes.
2) Puesto que el número posible de cláusulas para un conjunto de n fórmulas atómicas es 3^n, este método siempre acaba, pero puede ser muy largo.
3) Para seguir un cierto orden, partimos de nuestro conjunto de cláusulas, que llamaremos Σ_0.
4) En una primera etapa calculamos todas las resolventes que podemos hacer con estas cláusulas, y las añadimos al conjunto. Tenemos así un nuevo conjunto Σ_1.
5) En la siguiente etapa calculamos todas las resolventes que podemos hacer con las cláusulas Σ_1. Al nuevo conjunto lo llamaremos Σ_2.
6) De esta forma, vamos obteniendo una sucesión de conjuntos de cláusulas $\Sigma_0 \subseteq \Sigma_1 \subseteq \dots$ que en algún momento tiene que estabilizarse. Si llegamos a ese momento sin haber obtenido la cláusula vacía el conjunto Σ satisfacible.

Resolución Lineal

Una deducción por resolución de una cláusula es **lineal** si es una sucesión de pasos de resolución que cumple que en cada paso de resolución que no sea el primero se utiliza la resolvente del paso anterior. Esto da lugar a un árbol con una estructura muy peculiar.

Estrategias Input

Dado un conjunto de cláusulas Σ, una deducción a partir de Σ es **input** si al menos una de las dos cláusulas que se usan para calcular cada una de las resolventes pertenece al conjunto Σ.

Def: Dado un lenguaje proposicional construido sobre el conjunto Σ:

1) Un literal es **positivo** si es una fórmula atómica (si pertenece a Σ)
2) Un literal es **negativo** si es el negado de una fórmula atómica.
3) Una cláusula es **negativa** si todos los literales que aparecen en ella son literales negativos.
4) Una cláusula es cláusula de **Horn** si tiene exactamente un literal positivo.
5) Un conjunto de cláusulas es un **conjunto de Horn** si tiene exactamente una cláusula negativa, y el resto de las cláusulas de Horn.

Teorema: Sea Σ un conjunto de cláusulas de un lenguaje proposicional que es un conjunto Horn. Entonces Σ es insatisfacible si, y solo si, existe una deducción lineal-input de la cláusula vacía que se inicia en la cláusula negativa.

Denominaremos **cláusula objetivo** a la cláusula negativa de un conjunto de Horn, a la cláusulas de Horn que constan de un único literal la llamaremos **hechos** y a las de más de un literal **reglas**.

Ejercicios Hallar subfórmulas

¿Cómo se halla una subfórmula?: Tomamos todos los subelementos de la fórmula que compongan una fórmula (Literalmente, es la definición de subfórmula.

1) Halla las subfórmulas de las siguientes fórmulas:

a) $a \wedge \neg b \rightarrow c \vee (e \wedge a)$

$\{a, b, c, e, a \wedge \neg b, \neg b, a \wedge \neg b \rightarrow c \vee (e \wedge a), c \vee (e \wedge a), e \wedge a\}$

b) $c \wedge (a \vee b) \rightarrow \neg a \vee b$

$\{a, b, c, a \vee b, c \wedge (a \vee b), \neg a, \neg a \vee b, c \wedge (a \vee b) \rightarrow \neg a \vee b\}$

c) $\neg(a \rightarrow b) \rightarrow a \wedge \neg(a \wedge b)$

$\{a, b, a \rightarrow b, \neg(a \rightarrow b), a \wedge b, \neg(a \wedge b), a \wedge \neg(a \wedge b), \neg(a \rightarrow b) \rightarrow a \wedge \neg(a \wedge b)\}$

d) $b \wedge a \rightarrow (\neg b \rightarrow d \wedge \neg(d \rightarrow \neg a))$

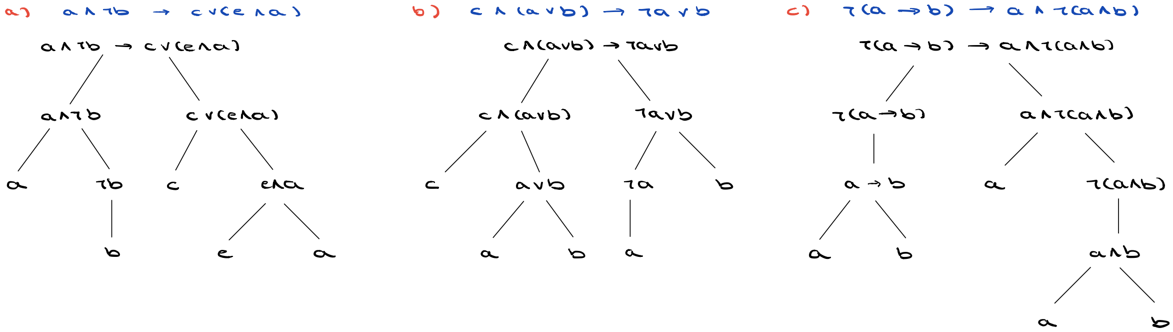
$\{a, b, d, b \wedge a, \neg b, \neg a, d \rightarrow \neg a, \neg(d \rightarrow \neg a), d \wedge \neg(d \rightarrow \neg a), \neg b \rightarrow d \wedge \neg(d \rightarrow \neg a), b \wedge a \rightarrow (\neg b \rightarrow d \wedge \neg(d \rightarrow \neg a))\}$

Ejercicios Construir Árboles de Formación

¿Cómo se hace?: Calculamos las subfórmulas como en el apartado anterior y vamos generando el árbol.

1) Construye el árbol de formación de fórmulas del ejercicio anterior:

a) $a \wedge \neg b \rightarrow c \vee (e \wedge a)$ b) $c \wedge (a \vee b) \rightarrow \neg a \vee b$ c) $\neg(a \rightarrow b) \rightarrow a \wedge \neg(a \wedge b)$



Ejercicios Calcular la veracidad de Proposiciones

¿Cómo se hace? : se hace usando que:

- $\alpha \vee \beta = v(\alpha) + v(\beta) + v(\alpha)v(\beta)$
- $\alpha \wedge \beta = v(\alpha)v(\beta)$
- $\alpha \rightarrow \beta = 1 + v(\alpha) + v(\alpha)v(\beta)$
- $\alpha \leftrightarrow \beta = 1 + v(\alpha) + v(\beta)$
- $\neg\alpha = 1 + v(\alpha)$

① Si en un determinado mundo v, α y β son verdaderas y γ es falsa ¿Cuál es el valor de verdad en dicho mundo de los siguientes proposiciones?

$\alpha = \beta = 1 \quad , \gamma = 0$

a) $\alpha \vee \gamma$

$$\alpha \vee \gamma = \alpha + \gamma + \alpha\gamma = 1 + 0 + 1 \cdot 0 = 1 \text{ (verdadero)}$$

b) $\alpha \leftrightarrow \neg\beta \vee \gamma$

$$\alpha \leftrightarrow \neg\beta \vee \gamma = 1 + \alpha + \neg\beta \vee \gamma = 1 + \alpha + \neg\beta + \gamma + \neg\beta\gamma = 1 + \alpha + 1 + \beta + \gamma + (1+\beta)\gamma = \alpha + \beta + \gamma + \gamma + \beta\gamma = 1 + 1 + 0 + 0 + 1 \cdot 0 = 0 \text{ (Falso)}$$

c) $(\beta \rightarrow \alpha) \rightarrow ((\alpha \rightarrow \neg\gamma) \rightarrow (\neg\gamma \rightarrow \beta))$

$$(\beta \rightarrow \alpha) \rightarrow ((\alpha \rightarrow \neg\gamma) \rightarrow (\neg\gamma \rightarrow \beta)) = 1 + \beta \rightarrow \alpha + (\beta \rightarrow \alpha)((\alpha \rightarrow \neg\gamma) \rightarrow (\neg\gamma \rightarrow \beta)) =$$

$$= 1 + 1 + \beta + \alpha\beta + (1 + \beta + \alpha\beta)[1 + \alpha \rightarrow \neg\gamma + (\alpha \rightarrow \neg\gamma)(\neg\gamma \rightarrow \beta)] =$$

$$= \beta + \alpha\beta + (1 + \beta + \alpha\beta)[1 + 1 + \alpha + \neg\gamma \cdot \alpha + (1 + \alpha + \neg\gamma \cdot \alpha)(1 + \neg\gamma + \neg\gamma \cdot \beta)] =$$

$$= \beta + \alpha\beta + (1 + \beta + \alpha\beta)\left[\alpha + (1+\gamma)\alpha + [1 + \alpha + (1+\gamma)\alpha][1 + 1 + \gamma + (1+\gamma)\beta]\right] =$$

$$= 1 + 1 + (1 + 1 + 1)[1 + 1 + (1 + 1 + 1)(1 + 1 + 1)] =$$

$$= 0 + 1[0 + 1 \cdot 1] = 1 \text{ (Verdadero)}$$

② Si $\alpha \rightarrow \beta$ es verdadera en un mundo v ¿Qué puedes deducir sobre el valor de verdad en dicho mundo de las siguientes proposiciones?

¿Cómo se hace? : Construyendo la tabla de verdad

a) $\alpha \vee \gamma \rightarrow \beta \vee \gamma$

α	β	γ	$\alpha \rightarrow \beta$	$\alpha \vee \gamma$	$\beta \vee \gamma$	$\alpha \vee \gamma \rightarrow \beta \vee \gamma$
0	0	0	1	0	0	1
0	0	1	1	1	1	1
0	1	0	1	0	1	1
0	1	1	1	1	1	1
1	0	0	0	1	0	0
1	0	1	0	1	1	1
1	1	0	1	1	1	1
1	1	1	1	1	1	1

Por tanto, podemos ver que si $\alpha \rightarrow \beta$ es cierto, entonces $\alpha \vee \gamma \rightarrow \beta \vee \gamma$ también lo es, luego, es una equivalencia lógica.

b) $\neg\alpha \wedge \beta \leftrightarrow \alpha \vee \beta$

α	β	$\alpha \rightarrow \beta$	$\neg\alpha$	$\neg\alpha \wedge \beta$	$\alpha \vee \beta$	$\neg\alpha \wedge \beta \leftrightarrow \alpha \vee \beta$
0	0	1	1	0	0	1
0	1	1	1	1	1	1
1	0	0	0	0	1	0
1	1	1	0	0	1	0

Solo se cumple cuando $\alpha = 0$.

③ Si $\alpha \leftrightarrow \beta$ es falsa en un mundo v, ¿Qué puedes decir sobre el valor de verdad en dicho mundo de $\alpha \wedge \beta \leftrightarrow \beta \wedge \gamma$?

α	β	γ	$\alpha \leftrightarrow \beta$	$\alpha \wedge \gamma$	$\beta \wedge \gamma$	$\alpha \wedge \beta \leftrightarrow \beta \wedge \gamma$
0	0	0	1	0	0	1
0	0	1	1	0	0	1
0	1	0	0	0	0	1
0	1	1	0	0	1	0
1	0	0	0	0	0	1
1	0	1	0	1	0	0
1	1	0	1	1	0	0
1	1	1	1	1	1	1

En el caso de $\alpha \leftrightarrow \beta$ falso, tenemos que la fórmula es verdadera si y solo si $\beta = 0$

Ejercicios clasificar Proposiciones

¿Cómo se hacen?: Desarrollando la fórmula. Si

- Si es constantemente 1, es una tautología
- Si es constantemente 0, es una contradicción
- Si depende de los valores de las variables, es contingente o satisfacible

- $\alpha \vee \beta = v(\alpha) + v(\beta) + v(\alpha)v(\beta)$
- $\alpha \wedge \beta = v(\alpha)v(\beta)$
- $\alpha \to \beta = 1 + v(\alpha) + v(\alpha)v(\beta)$
- $\alpha \leftrightarrow \beta = 1 + v(\alpha) + v(\beta)$
- $\neg \alpha = 1 + v(\alpha)$

(1) Clasifica las siguientes proposiciones:

a) $\alpha \leftrightarrow \alpha \vee \alpha$

$\alpha \leftrightarrow \alpha \vee \alpha = 1 + \alpha + \alpha \vee \alpha = 1 + \alpha + \alpha + \alpha + \alpha\alpha = 1 + \alpha + \alpha + \alpha + \alpha = 1 \Rightarrow$ Tautología

b) $\neg\alpha \to \alpha \wedge \beta$

$\neg\alpha \to \alpha \wedge \beta = 1 + \neg\alpha + \neg\alpha(\alpha \wedge \beta) = 1 + 1 + \alpha + (1+\alpha)(\alpha\beta) = \alpha + \alpha\beta + \alpha\beta = \alpha \Rightarrow$ Contingente

c) $\neg\alpha \leftrightarrow (\alpha \to \neg\alpha)$

$\neg\alpha \leftrightarrow (\alpha \to \neg\alpha) = 1 + \neg\alpha + \alpha \to \neg\alpha = 1 + 1 + \alpha + 1 + \alpha + \neg\alpha \cdot \alpha = 1 + 1 + \alpha + 1 + \alpha + \alpha + \alpha = 1 \Rightarrow$ Tautología

d) $\alpha \wedge \neg(\alpha \vee \beta)$

$\alpha \wedge \neg(\alpha \wedge \beta) = \neg(\alpha \wedge \beta) \cdot \alpha = (1 + \alpha \vee \beta)\alpha = (1 + \alpha + \beta + \alpha\beta)\alpha = \alpha + \alpha + \alpha\beta + \alpha\beta = 0 \Rightarrow$ Contradicción

Ejercicios de Equivalencia entre Proposiciones

¿Cómo se hace?: Se desarrolla la fórmula, tratando de llegar a una expresión común

- $\alpha \vee \beta = v(\alpha) + v(\beta) + v(\alpha)v(\beta)$
- $\alpha \wedge \beta = v(\alpha)v(\beta)$
- $\alpha \to \beta = 1 + v(\alpha) + v(\alpha)v(\beta)$
- $\alpha \leftrightarrow \beta = 1 + v(\alpha) + v(\beta)$
- $\neg \alpha = 1 + v(\alpha)$

(1) Estudia si las siguientes equivalencias son ciertas o no. Justifica la respuesta.

a) $a \to b \equiv \neg a \to \neg b$

$1 + a + ab \equiv (1+a) \to (1+b)$

$1 + a + ab \equiv 1 + 1 + a + (1+a)(1+b)$

$1 + a + ab \equiv a + 1 + b + a + ab$

$1 + a + ab \not\equiv 1 + b + ab$

No son equivalentes

b) $(a \vee b) \to c \equiv (a \to c) \vee (b \to c)$

$1 + a \vee b + (a \vee b)c \equiv a \to c + b \to c + (a \to c)(b \to c)$

$1 + a + b + ab + ac + bc + abc \equiv 1 + a + ac + 1 + b + bc + (1 + a + ac)(1 + b + bc)$

$1 + a + b + ab + ac + bc + abc = \cancel{a} + \cancel{b} + \cancel{ac} + \cancel{bc} + 1 + \cancel{b} + \cancel{bc} + \cancel{a} + ab + abc + \cancel{ac} + \cancel{abc} + \cancel{abc}$

$1 + a + b + ab + ac + bc + abc = 1 + ab + abc$

No son equivalentes

Ejercicio Estudiar si un conjunto de Proposiciones es satisfacible o insatisfacible

¿Cómo se hace?: Haces la tabla de verdad. Si existe un caso en el que todas las proposiciones son verdad, entonces es satisfacible, en caso contrario insatisfacible.

(1) Estudia si los siguientes conjuntos de proposiciones es satisfacible o insatisfacible:

$\Gamma = \{c \to (a \vee b),\ b \to (c \to a),\ d \wedge \neg(c \to a)\}$

a b c d	$a \vee b$	$c \to (a \vee b)$	$c \to a$	$b \to (c \to a)$	$\neg(c \to a)$	$d \wedge \neg(c \to a)$
0 0 0 0	0	1	1	1	0	0
0 0 0 1	0	1	1	1	0	0
0 0 1 0	0	0	0	1	1	0
0 0 1 1	0	0	0	1	1	1
0 1 0 0	1	1	1	1	0	0
0 1 0 1	1	1	1	1	0	0
0 1 1 0	1	1	0	0	1	0
0 1 1 1	1	1	0	0	1	1
1 0 0 0	1	1	1	1	0	0
1 0 0 1	1	1	1	1	0	0
1 0 1 0	1	1	1	1	0	0
1 0 1 1	1	1	1	1	0	0
1 1 0 0	1	1	1	1	0	0
1 1 0 1	1	1	1	1	0	0
1 1 1 0	1	1	1	1	0	0
1 1 1 1	1	1	1	1	0	0

Como no encontramos una interpretación para que todas sean verdad $\Rightarrow$

$\Rightarrow$ Insatisfacible

Ejercicios Teorema de Deducción y Polinomio de Gegalkine

Dado un conjunto de fórmulas Γ, posiblemente vacío, y una fórmula φ decimos que Γ implica semánticamente a φ, abreviadamente $\Gamma \models \varphi$, si para toda valoración v se tiene que $v(\varphi)=1$ siempre que para toda fórmula γ de Γ valga $v(\gamma)=1$. Si Γ consta solamente de las fórmulas $\gamma_1, \gamma_2, \dots, \gamma_n$, en lugar de $\{\gamma_1, \dots, \gamma_n\} \models \varphi$ escribimos $\gamma_1, \dots, \gamma_n \models \varphi$ y cuando $\Gamma = \emptyset$ escribimos simplemente $\models \varphi$ en lugar de $\emptyset \models \varphi$.

Teorema de la Deducción: Sea Γ un conjunto de fórmulas (que podría ser vacío) de un lenguaje proposicional, y α, β, otras dos fórmulas. Entonces los siguientes afirmaciones son equivalentes:

1) $\Gamma \models \alpha \to \beta$

2) $\Gamma \cup \{\alpha\} \models \beta$

① Determinar utilizando el teorema de deducción y polinomio de Gegalkine si son o no tautologías los siguientes fórmulas.

Para todos los apartados partimos de que si $\models \alpha$, entonces α es tautología.

a) $(\beta \to \alpha \vee \gamma) \to ((\alpha \to \beta) \to (\alpha \to ((\gamma \to \beta) \to \gamma)))$

Tenemos que demostrar que $\models (\beta \to \alpha\vee\gamma) \to ((\alpha\to\beta) \to (\alpha \to ((\gamma\to\beta)\to\gamma)))$. Usando el T. Deducción tenemos:

$\{\beta \to \alpha\vee\gamma\} \models (\alpha\to\beta) \to (\alpha \to ((\gamma\to\beta)\to\gamma))$

$\{\beta \to \alpha\vee\gamma,\ \alpha\to\beta\} \models \alpha \to ((\gamma\to\beta)\to\gamma)$

$\{\beta \to \alpha\vee\gamma,\ \alpha\to\beta,\ \alpha\} \models (\gamma\to\beta)\to\gamma$

$\{\beta \to \alpha\vee\gamma,\ \alpha\to\beta,\ \alpha,\ \gamma\to\beta\} \boxed{\models \gamma}$

$\Updownarrow$

$\{\beta \to \alpha\vee\gamma,\ \alpha\to\beta,\ \alpha,\ \gamma\to\beta,\ \neg\gamma\}$ es insatisfacible

γ tautología $\Rightarrow v(\gamma)=1 \Rightarrow \Gamma$ tautología $\Rightarrow v(\gamma_i)=1,\ \forall \gamma_i \in \Gamma$

Esto solo se da si $\neg\gamma$ es insatisfacible, $v(\neg\gamma)=0$. Si añado $\neg\gamma$ a Γ y compruebo que Γ es insatisfacible, equivaldría a comprobar la primera condición.

Multiplicando los polinomios de Gegalkine tenemos:

$(\beta\to\alpha\vee\gamma)(\alpha\to\beta)(\alpha)(\gamma\to\beta)(\neg\gamma) = (1+\beta+\beta(\alpha\vee\gamma))(1+\alpha+\alpha\beta)(\alpha)(1+\gamma+\gamma\beta)(1+\gamma) = (1+\beta+\beta(\alpha+\gamma+\alpha\gamma))(1+\alpha+\alpha\beta)(\alpha)(1+\gamma+\gamma\beta)(1+\gamma) =$

$= (1+\beta+\beta\alpha+\beta\gamma+\beta\alpha\gamma)(\alpha+\alpha+\alpha\beta)(1+\gamma+\gamma\beta+\gamma+\gamma+\gamma\beta) = (1+\beta+\beta\alpha+\beta\gamma+\beta\alpha\gamma)(\alpha\beta)(1+\gamma) = (1+\beta+\beta\alpha+\beta\gamma+\beta\alpha\gamma)(\alpha\beta+\gamma\alpha\beta) =$

$= \cancel{\alpha\beta}+\cancel{\alpha\beta}+\alpha\beta+\cancel{\alpha\beta\gamma}+\cancel{\gamma\beta\alpha}+\cancel{\gamma\alpha\beta}+\cancel{\gamma\alpha\beta}+\cancel{\gamma\alpha\beta}+\cancel{\gamma\alpha\beta}+\gamma\alpha\beta = \alpha\beta+\alpha\beta\gamma \Longrightarrow \Gamma$ no es insatisfacible $\Rightarrow$ la proposición no es tautología

b) $(\alpha\to\beta) \to ((\beta\to\gamma) \to (\alpha\to\gamma))$

Tenemos que demostrar que $\models (\alpha\to\beta) \to ((\beta\to\gamma)\to(\alpha\to\gamma))$, usando el teorema de deducción:

$\{\alpha\to\beta\} \models (\beta\to\gamma) \to (\alpha\to\gamma)$

$\{\alpha\to\beta,\ \beta\to\gamma\} \models \alpha\to\gamma$

$\{\alpha\to\beta,\ \beta\to\gamma,\ \alpha\} \models \gamma \iff \{\alpha\to\beta,\ \beta\to\gamma,\ \alpha,\ \neg\gamma\}$ es insatisfacible

Multiplicando los polinomios de Gegalkine:

$(\alpha\to\beta)(\beta\to\gamma)(\alpha)(\neg\gamma) = (1+\alpha+\alpha\beta)(1+\beta+\gamma\beta)\alpha(1+\gamma) =$

$= (1+\beta+\gamma\beta+\alpha+\beta\alpha+\cancel{\gamma\alpha\beta}+\cancel{\alpha\beta}+\cancel{\alpha\beta}+\cancel{\alpha\gamma\beta})(\alpha+\gamma\alpha) =$

$= \cancel{\alpha}+\cancel{\alpha\beta}+\cancel{\alpha\gamma\beta}+\cancel{\alpha}+\cancel{\beta\alpha}+\cancel{\gamma\alpha}+\cancel{\gamma\alpha\beta}+\cancel{\gamma\alpha\beta}+\cancel{\gamma\alpha}+\cancel{\beta\alpha\gamma} = 0 \Rightarrow \Gamma$ es insatisfacible $\Rightarrow$ La fórmula es tautología

② Estudia en cada caso cuáles de las siguientes consecuencias son ciertas:

a) $\underbrace{(\neg(a\wedge b),\ \neg c \vee a, b)}_{\alpha} \models \underbrace{\neg a \wedge \neg c}_{\beta}$

β es consecuencia lógica de α si para cualquier interpretación que haga todas las fórmulas de α ciertas $\Rightarrow \beta$ cierta.

a	b	c	$a\wedge b$	$\neg(a\wedge b)$	$\neg c$	$\neg c \vee a$	$\neg a$	$\neg a \wedge \neg c$
0	0	0	0	1	1	1	1	
0	0	1	0	1	0	0	1	
0	1	0	0	1	1	1	1	1
0	1	1	0	1	0	0	1	
1	0	0	0	1	1	1	0	
1	0	1	0	1	1	1	0	
1	1	0	1	0	1	1	0	
1	1	1	1	0	1	1	0	

Vemos que existe una interpretación que hace que todas las fórmulas de α son ciertas y β sea cierta $\Rightarrow$ β es consecuencia lógica de α

Ejercicios Algoritmo Davis – Putnan y Algoritmo de Resolución

¿Cómo se hacen? Seguimos los siguientes pasos:

1) Aplicamos T. Deducción, como en los ejercicios anteriores
2) Expresamos Γ en forma de cláusulas usando:
   - Eliminación de la bicondicional: si tenemos una subfórmula $\alpha_1 \leftrightarrow \alpha_2$ la sustituimos por $(\alpha_1 \to \alpha_2) \wedge (\alpha_2 \to \alpha_1)$.
   - Eliminación de la condicional: si tenemos una subfórmula $\alpha_1 \to \alpha_2$ la sustituimos por $\neg\alpha_1 \vee \alpha_2$.
   - Interiorización de la negación. Cualquier subfórmula del tipo:
     - $\neg\neg\alpha$ la sustituimos por α
     - $\neg(\alpha_1 \vee \alpha_2)$ la sustituimos por $\neg\alpha_1 \wedge \neg\alpha_2$
     - $\neg(\alpha_1 \wedge \alpha_2)$ la sustituimos por $\neg\alpha_1 \vee \neg\alpha_2$
   - Distribución de la $\vee$ sobre la $\wedge$. Si tenemos una subfórmula de la forma $\alpha_1 \vee (\beta_1 \wedge \beta_2)$ la sustituimos por $(\alpha_1 \vee \beta_1) \wedge (\alpha_1 \vee \beta_1)$. Y lo mismo si la subfórmula es de la forma $(\alpha_1 \vee \alpha_2) \wedge \beta_1$ pasa a $(\alpha_1 \wedge \beta_1) \vee (\alpha_2 \wedge \beta_1)$.
   - Eliminación de redundancias: Absorción de cláusulas más grandes por las más chicas.
3) Aplicamos Algoritmo Davis – Putnan

③ Utiliza el algoritmo de Davis – Putnan y resolución para determinar si son o no tautologías las siguientes fórmulas:

a) $(\beta \to \alpha \vee \gamma) \to ((\alpha \to \beta) \to (\alpha \to ((\gamma \to \beta) \to \gamma)))$

Tenemos que demostrar que $\models (\beta \to \alpha \vee \gamma) \to ((\alpha \to \beta) \to (\alpha \to ((\gamma \to \beta) \to \gamma)))$. Usando el T. Deducción tenemos:

$\{\beta \to \alpha \vee \gamma\} \models (\alpha \to \beta) \to (\alpha \to ((\gamma \to \beta) \to \gamma))$

$\{\beta \to \alpha \vee \gamma, \alpha \to \beta\} \models \alpha \to ((\gamma \to \beta) \to \gamma)$

$\{\beta \to \alpha \vee \gamma, \alpha \to \beta, \alpha\} \models (\gamma \to \beta) \to \gamma$

$\{\beta \to \alpha \vee \gamma, \alpha \to \beta, \alpha, \gamma \to \beta\} \models \gamma \iff \{\beta \to \alpha \vee \gamma, \alpha \to \beta, \alpha, \gamma \to \beta, \neg\gamma\}$ es insatisfacible

Convirtiéndolo a cláusulas tenemos:

$\beta \to \alpha \vee \gamma$	$\alpha \to \beta$	$\gamma \to \beta$
$\neg\beta \vee (\alpha \vee \gamma)$	$\neg\alpha \vee \beta$	$\neg\gamma \vee \beta$
$\neg\beta \vee \alpha \vee \gamma$		

$\Rightarrow$ Tenemos que ver si $\{\neg\beta \vee \alpha \vee \gamma, \neg\alpha \vee \beta, \alpha, \neg\gamma \vee \beta, \neg\gamma\}$ es insatisfacible.

Usamos ahora el algoritmo de Davis - Putnan:

$\{\neg\beta \vee \alpha \vee \gamma, \neg\alpha \vee \beta, \alpha, \neg\gamma \vee \beta, \neg\gamma\}$

$\downarrow \alpha = 1$ ← vd (α cláusula unit)

$\{\neg\gamma \vee \beta, \neg\gamma\}$ ($\neg\gamma$ cláusula unit)

$\downarrow \gamma = 0$ ← falso

$\{\ \} \Rightarrow$

$\Rightarrow$ satisfacible $\Rightarrow$ No es tautología

¿Cómo se aplica el algoritmo de Davis - Putnan?: El algoritmo se basa en tres reglas

1) Sea Σ un conjunto de cláusulas. Supongamos que en Σ hay una cláusula unit, entonces Σ insatisfacible $\iff \Sigma_\lambda$ insatisfacible
2) Supongamos que en Σ aparece un literal puro λ, entonces, Σ insatisfacible $\iff \Sigma_\lambda$ insatisfacible.
3) Sea λ un literal que aparece en alguna cláusula de Σ, entonces Σ insatisfacible $\iff \Sigma_\lambda$ y Σ_{λ^c} insatisfacible

Σ_λ es el conjunto que queda de sustraer todas las cláusulas que son ampliación de λ.

b) $(\alpha \to \beta) \to ((\beta \to \gamma) \to (\alpha \to \gamma))$

Tenemos que ver que $\models (\alpha \to \beta) \to ((\beta \to \gamma) \to (\alpha \to \gamma))$. Usando el teorema de deducción:

$\{\alpha \to \beta\} \models (\beta \to \gamma) \to (\alpha \to \gamma)$

$\{\alpha \to \beta, \beta \to \gamma\} \models \alpha \to \gamma$

$\{\alpha \to \beta, \beta \to \gamma, \alpha\} \models \gamma \iff \{\alpha \to \beta, \beta \to \gamma, \alpha, \neg\gamma\}$ insatisfacible.

Convirtiendo a cláusulas tenemos:

$\alpha \to \beta$	$\beta \to \gamma$
$\neg\alpha \vee \beta$	$\neg\beta \vee \gamma$

$\Rightarrow$ Tenemos que ver si $\{\neg\alpha \vee \beta, \neg\beta \vee \gamma, \alpha, \neg\gamma\}$ es insatisfacible:

$\{\neg\alpha \vee \beta, \neg\beta \vee \gamma, \alpha, \neg\gamma\}$

$|\ \alpha = 1$ (cláusula unit)

$\{\beta, \neg\beta \vee \gamma, \neg\gamma\}$

$|\ \beta = 1$ (cláusula unit)

$\{\gamma, \neg\gamma\}$ $\Rightarrow$ Insatisfacible $\Rightarrow$ Tautología

② Hacerlo ahora mediante el algoritmo de resolución.

a) $(\beta \to \gamma) \to (\neg(\alpha \to \gamma) \to \neg(\alpha \to \beta))$

Tenemos que demostrar que $\models (\beta \to \gamma) \to (\neg(\alpha \to \gamma) \to \neg(\alpha \to \beta))$ es tautología. Usando el teorema de deducción:

$\{\beta \to \gamma\} \models \neg(\alpha \to \gamma) \to (\alpha \to \beta)$

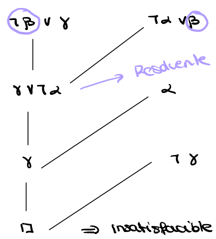
$\{\beta \to \gamma, \neg(\alpha \to \gamma)\} \models \neg(\alpha \to \beta) \iff \{\beta \to \gamma, \neg(\alpha \to \gamma), \alpha \to \beta\}$ es insatisfacible

Convirtiéndolos a cláusulas:

$\beta \to \gamma$	$\neg(\alpha \to \gamma)$	$\alpha \to \beta$
$\neg\beta \vee \gamma$	$\neg(\neg\alpha \vee \gamma)$	$\neg\alpha \vee \beta$
	$\neg\neg\alpha \vee \neg\gamma$	
	$\alpha \wedge \neg\gamma$ ↪ se separan	

$\Rightarrow \{\neg\beta \vee \gamma, \alpha, \neg\gamma, \neg\alpha \vee \beta\}$ es insatisfacible, luego:

Aplicando el algoritmo de resolución:



$\Rightarrow$ insatisfacible

$\Rightarrow$ Tautología

¿Cómo funciona el algoritmo de resolución?: sean C_1 y C_2 dos cláusulas. Supongamos que λ es un literal tal que aparece en C_1 y λ^c aparece en C_2, entonces una cláusula que sea equivalente a $(C_1 - \lambda) \vee (C_2 - \lambda^c)$ es una resolvente de C_1 y C_2.

$\downarrow$ Elimina el literal

Sea T un conjunto de cláusulas y C una cláusula. Una deducción (por resolución) de C a partir de T es una sucesión de cláusulas $C_1, \dots, C_n$, donde:

- $C_n = C$
- $C_i \in T$ o C_i es una resolvente de dos cláusulas de $\{C_1, \dots, C_{i-1}\}$

Teorema: sea T un conjunto de cláusulas, entonces T es insatisfacible si, y sólo si, hay una deducción por resolución de la cláusula vacía.

b) $((\alpha \to \beta) \to \gamma) \to (\beta \to \gamma)$

Con el T. deducción:

$\{(\alpha \to \beta) \to \gamma\} \models \beta \to \gamma$

$\{(\alpha \to \beta) \to \gamma, \beta\} \models \gamma \iff \{(\alpha \to \beta) \to \gamma, \beta, \neg\gamma\}$ insatisfacible

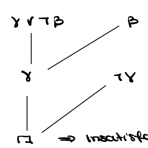
$(\alpha \to \beta) \to \gamma$

$\neg(\alpha \to \beta) \vee \gamma$

$\neg(\neg\alpha \vee \beta) \vee \gamma \qquad \Rightarrow \{\alpha \vee \gamma, \gamma \vee \neg\beta, \beta, \neg\gamma\} \qquad \Rightarrow$

$(\neg\neg\alpha \wedge \neg\beta) \vee \gamma$

$(\alpha \vee \gamma) \wedge (\gamma \vee \neg\beta)$



$\Rightarrow$ insatisfacible $\Rightarrow$ Tautología

③ Comprobar que $\{(\neg\alpha \vee \beta) \wedge (\alpha \vee \neg\beta), \neg(\alpha \leftrightarrow \beta)\}$ es insatisfacible

Lo convertimos en cláusulas

$\neg(\alpha \leftrightarrow \beta)$

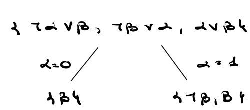
$\neg((\alpha \to \beta) \wedge (\beta \to \alpha))$

$\neg((\neg\alpha \vee \beta) \wedge (\neg\beta \vee \alpha))$

$\vdots$

$(\alpha \vee \beta) \wedge (\neg\beta \vee \alpha)$

$\Rightarrow \{\neg\alpha \vee \beta, \neg\beta \vee \alpha, \alpha \vee \beta\}$. Aplicando Davis – Putnam:



insatisfacible

Tema 5: Lógica de Primer orden

Definimos como **lenguaje de primer orden** a $\mathcal{L} = (C, F, R, p)$, donde C es el conjunto de símbolos de constante, F el de símbolos de funciones y R el de símbolos de relación, p es una aplicación de $F \cup R \to \mathbb{N}$

Ejercicios Subfórmulas, Variables Libres y Ligadas

(3) Para las siguientes fórmulas escribelas en forma de árbol. Calcula sus subfórmulas. Determina el carácter de ocurrencia de sus variables, halla sus variables libres y sus variables ligadas y di si sus fórmulas son sentencias.

a) $\forall x (R(x,y) \wedge \neg \forall y R(x,y))$

¿Cómo se colocan fórmulas en árboles?: si $\mathcal{L} = (C, F, R, p)$ es un lenguaje de primer orden. Definimos una fórmula de $\mathcal{L}$ como un árbol con etiquetas en $Cons(\mathcal{L}) \cup At(\mathcal{L})$, que cumple:

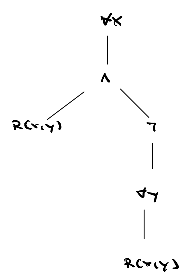
1) si un nodo tiene la etiqueta en $Cons(\mathcal{L})$, entonces la ariedad del símbolo es el número de hijos del nodo.

2) si un nodo tiene la etiqueta en $At(\mathcal{L})$, entonces el nodo es una hoja.

¿Qué es $At(\mathcal{L})$?: son las fórmulas atómicas de un lenguaje $\mathcal{L}$, es decir, un elemento de alguno de los conjuntos:

- $\{t_1 \approx t_2 : t_1, t_2 \in Ter(\mathcal{L})\}$
- $\{R(t_1, \dots, t_n) : t_1, \dots, t_n \in Ter(\mathcal{L}) \text{ y } R \in R_n\}$

¿Qué es $cons(\mathcal{L})$?: Conjunto $Cons = \{\neg, \vee, \wedge, \to, \leftrightarrow, \approx\} \cup \{\forall x : x \in V$ (conjunto de variables$\}$



¿Qué es una subfórmula?: Sea $\mathcal{L}$ un lenguaje de primer orden, y sea φ una fórmula del lenguaje. Definimos el conjunto de la subfórmula de φ, que notamos $sub(\varphi)$ como:

1) $sub(\varphi) = \{\varphi\}$ si $\varphi \in At(\mathcal{L})$
2) $sub(\neg\varphi) = \{\neg\varphi\} \cup sub(\varphi)$
3) $sub(\varphi \wedge \psi) = \{\varphi \vee \psi\} \cup sub(\varphi) \cup sub(\psi)$
4) $sub(\varphi \wedge \psi) = \{\varphi \wedge \psi\}\ sub(\varphi) \cup sub(\psi)$
5) $sub(\varphi \to \psi) = \{\varphi \to \psi\} \cup sub(\varphi) \cup sub(\psi)$
6) $sub(\varphi \leftrightarrow \psi) = \{\varphi \leftrightarrow \psi\} \cup sub(\varphi) \cup sub(\psi)$
7) $sub(\forall x \varphi) = \{\forall x \varphi\} \cup sub(\varphi)$
8) $sub(\exists x \varphi) = \{\exists x \varphi\} \cup sub(\varphi)$

¿Qué es una fórmula?:

1) Toda fórmula atómica es una fórmula
2) φ y ψ fórmulas $\Rightarrow \neg\varphi, \varphi \vee \psi, \varphi \wedge \psi, \varphi \to \psi, \varphi \leftrightarrow \psi$ son fórmulas
3) si φ fórmula y x variable $\Rightarrow \forall x \varphi$ y $\exists x \varphi$ fórmulas

Subfórmulas $= \{R(x,y),\ \forall y R(x,y),\ \neg \forall y R(x,y),\ R(x,y) \wedge \neg \forall y R(x,y),\ \forall x (R(x,y) \wedge \neg \forall y R(x,y))\}$

¿Qué es una ocurrencia?: Una ocurrencia de una variable x en una fórmula φ es cada una de las veces en las que x aparece en una fórmula atómica de φ.

¿Qué es una ocurrencia ligada? Dada una ocurrencia de una variable x en una fórmula φ. La ocurrencia de x es ligada si está en el ámbito de un cuantificador de la forma $\forall x$ o $\exists x$. Las ocurrencias de x que no son ligadas se llaman simples.

¿Qué es una variable ligada?: La que tiene ocurrencias ligadas en φ.

¿Qué es una variable libre?: La que tiene ocurrencias libres en φ.

¿Qué es una sentencia?: Dada una fórmula F, F es sentencia si $Lib(F) = \emptyset$, variables libres de la fórmula.

Variables ligadas: x, variables libres: y (1 ocurrencia), no hay sentencias.

Ejercicios forma clausulada, prenexa y de Skolem:

¿Qué es una forma normal prenexa? : una fórmula φ es una forma prenexa si $\varphi = C_1x_1 \dots C_nx_n \Phi$, donde $C_i = \forall$ o $\exists$ y Φ es una fórmula sin cuantificadores (expresiones lingüísticas).

¿Qué es una forma normal de Skolem? : una fórmula φ es de Skolem si es de la forma $\varphi = \forall x_1 \forall x_2 \dots \forall x_n \Phi$, donde Φ es una fórmula sin cuantificadores

La forma clausulada es como la del tema anterior.