

Lógica y Matemática Computacional
Licenciatura en Sistemas de Información

ARBOLES

Ing. JULIO C. ACOSTA

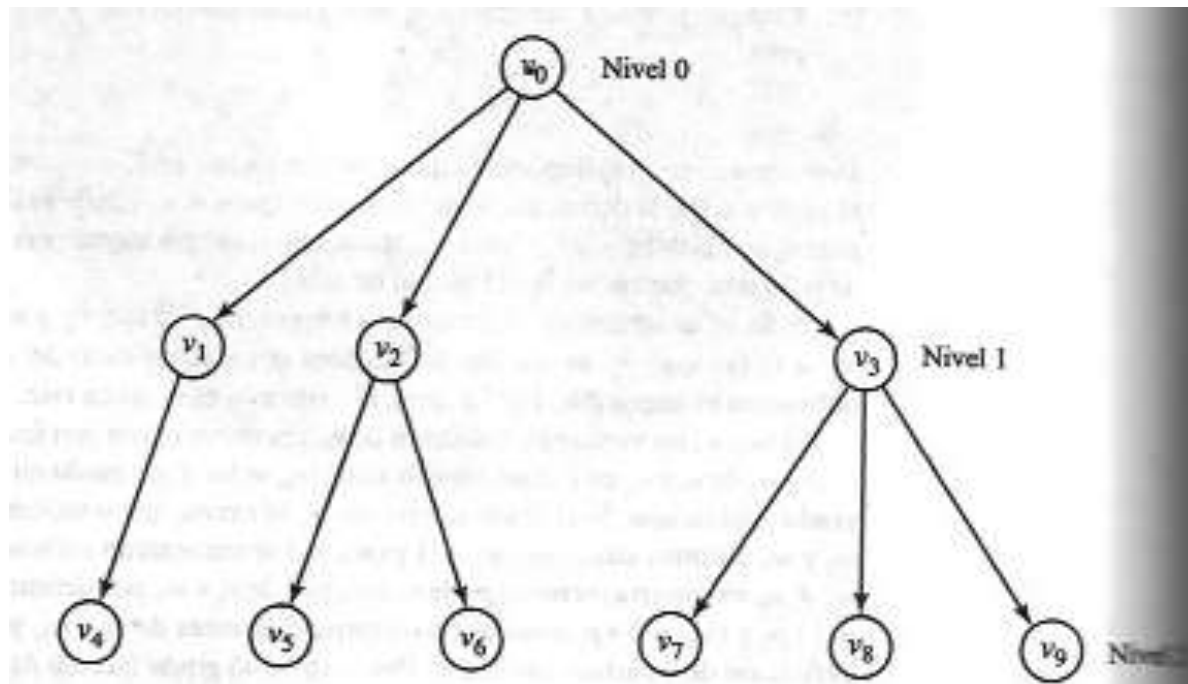
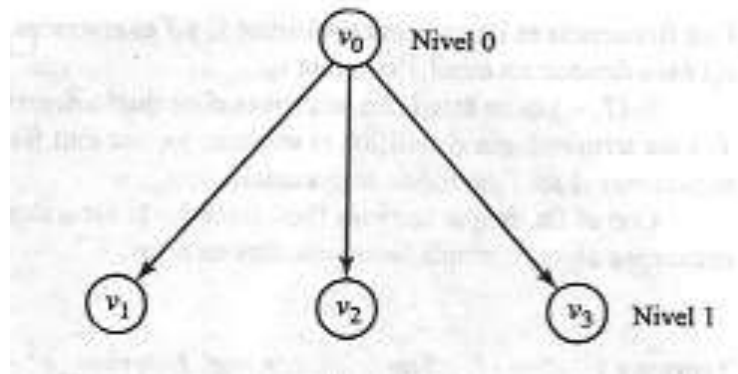
Facultad de Ciencias Exactas y Naturales y Agrimensura - UNNE

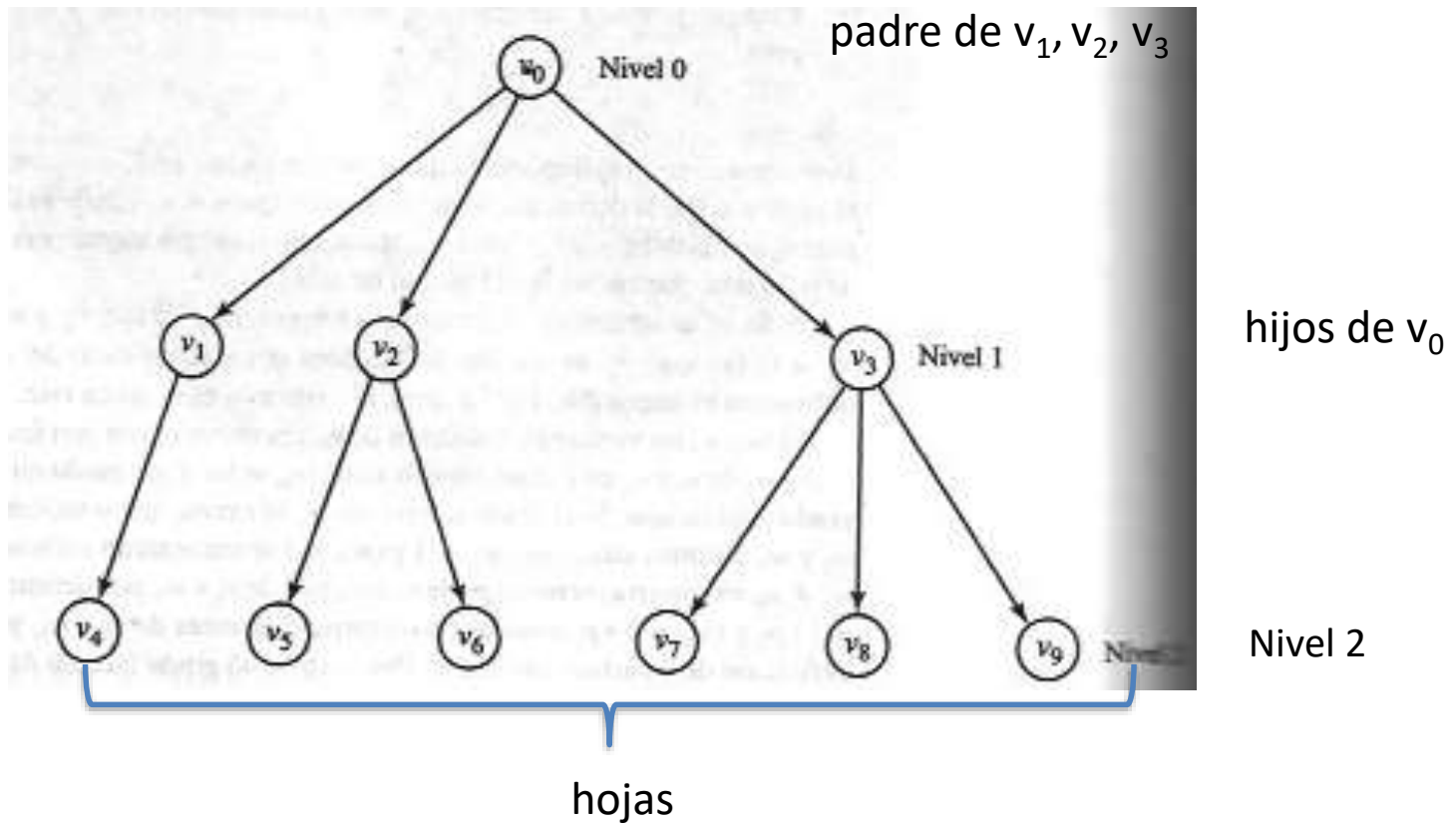
- Árboles con raíz.
- Árboles etiquetados.
- Árboles binarios.
- Búsquedas en árboles.
- Árboles no dirigidos.
- Árboles generadores o de expansión.
- Arbol de expansión mínima.
- Algoritmo de Prim.
- Algoritmo de Kruskal.
- Algoritmo de árboles de deducción de una fórmula de la lógica proposicional.

Sea un conjunto A y T una **relación** definida en A

T es un *árbol* (T, v_0) en A , si existe un vértice v_0 en A con la propiedad que:

- 1) Existe una única trayectoria en T de v_0 a cualquier otro vértice v en A
- 2) No existe trayectoria de v_0 a v_0
- 3) v_0 es único y es llamado *raíz del árbol* T
- 4) Si escribimos (T, v_0) , designamos el árbol T con raíz v_0 sobre un conjunto A ; un elemento v de A , es un *vértice en* T .





Ancestros Nodos Altura

Descendiente Raíz

Determine en cada caso si R definida en A es un árbol.

$$A = \{ t, u, v, w, x, y, z \}$$

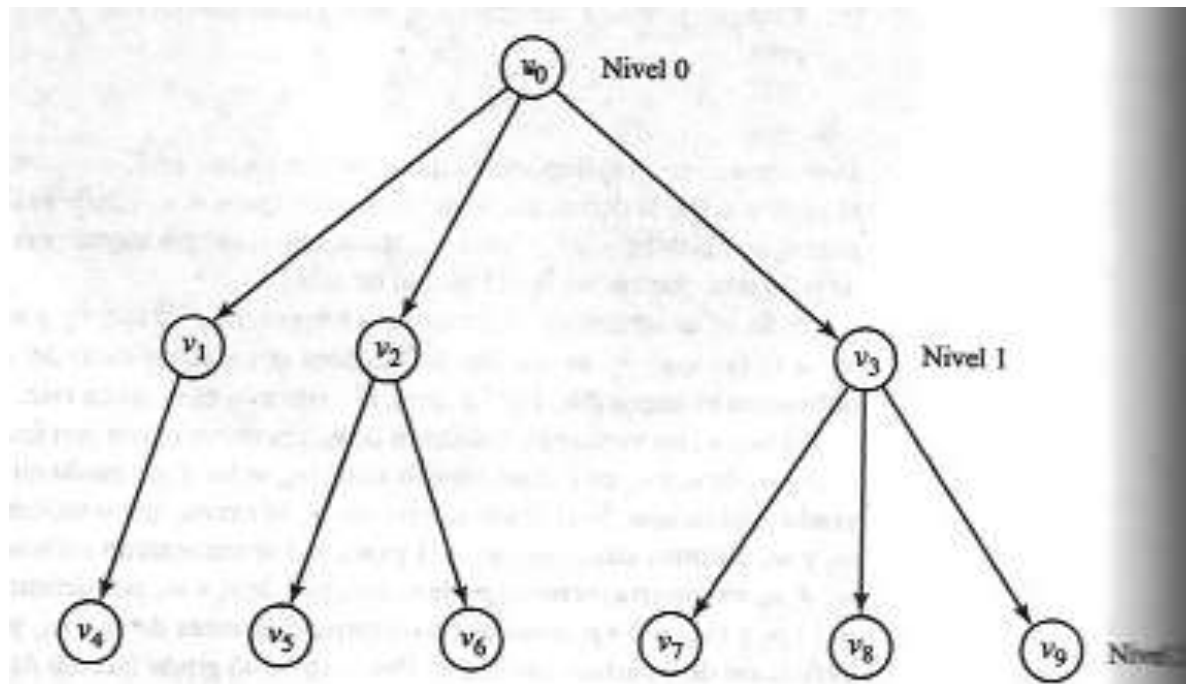
$$R = \{ (t,u); (u,w); (u,x); (u,v); (v,z); (v,y) \}$$

$$A = \{ a, b, c, d, e, f \}$$

$$R = \{ (a,b); (c,e); (f,a); (f,c); (f,d) \}$$

TEROREMA 1: Sea (T, v_0) un árbol con raíz.
Entonces

- (a) No existen ciclos en T .
- (b) v_0 es la única raíz en T .
- (c) Cada vértice en T distinto de v_0 tiene grado interno 1, y v_0 tiene grado interno 0



DEMOSTRACION: TEROREMA 1: Sea (T, v_0) un árbol con raíz. Entonces:

(a) No existen ciclos en T .

Suponga que existe un ciclo q en T que comienza y termina en v

Sabemos por definición que: $v \neq v_0$

Debe existir una trayectoria p de v_0 a v

Entonces:

$q \circ p$ es una trayectoria de v_0 a v diferente de p

Lo que contradice la definición de árbol

Por tanto, NO existen ciclos en T

TEROREMA 1: Sea (T, v_0) un árbol con raíz.

Entonces:

(b) v_0 es la única raíz en T .

Si v_0' es otra raíz de T , existe una trayectoria p que va de v_0 a v_0'
y una trayectoria q que va de v_0' a v_0

Entonces:

$q \circ p$ es un ciclo que va de trayectoria de v_0 a v_0

Lo que contradice la definición de árbol

Por tanto, v_0 es raíz única

TEROREMA 1: Sea (T, v_0) un árbol con raíz.

(c) Cada vértice en T distinto de v_0 tiene grado interno 1, y v_0 tiene grado interno 0

Sea w_1 un vértice en T , distinto de v_0

Entonces existe una trayectoria v_0, \dots, v_k, w_1 en T . $(v_k, w_1) \in T$
 w_1 tiene grado interno al menos 1.

Si w_1 tiene grado interno mayor que 1, deben existir vértices w_2 y w_3

$(w_2, w_1) \in T$ $(w_3, w_1) \in T$

$w_2 \neq v_0$ $w_3 \neq v_0$

Existen trayectorias p_2 de v_0 a w_2

p_3 de v_0 a w_3

$$(w_2, w_1) \circ p_2$$

$$(w_3, w_1) \circ p_3$$

Son trayectorias diferentes de v_0 a w_1

Lo que contradice la definición de árbol con raíz en v_0

Por tanto en grado interno de w_1 es uno

Ejercicio: Argumente que v_0 tiene grado cero

TEOREMA2: Sea (T, v_0) un árbol con raíz sobre u conjunto A.
Entonces

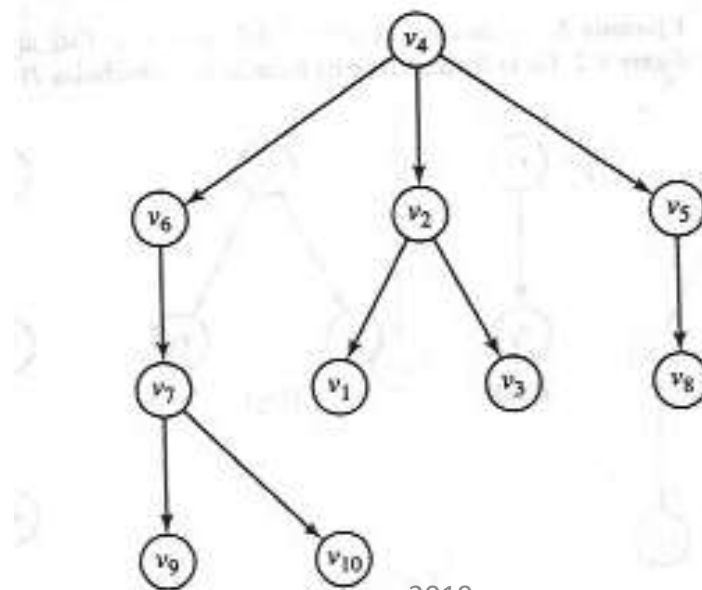
(a) T es **Arreflexiva**

(b) T es **Asimétrica**

(c) T si $(a T b) \wedge (b T c)$ entonces $(a \nmid c)$ **Atransitiva**

Ejemplo: Sean $A = \{ v_1, v_2, v_3, v_4, v_5, v_6, v_7, v_8, v_9, v_{10} \}$

$T = \{ (v_2, v_3); (v_2, v_1) (v_4, v_5) (v_4, v_6) (v_5, v_8) (v_6, v_7) (v_4, v_2) (v_7, v_9) (v_7, v_{10}) \}$

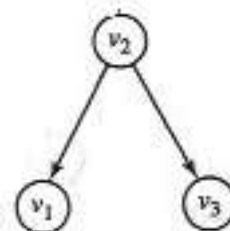
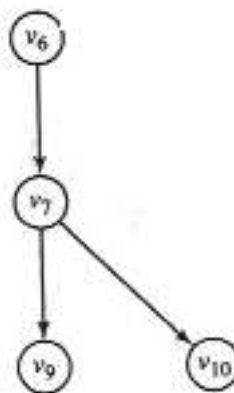
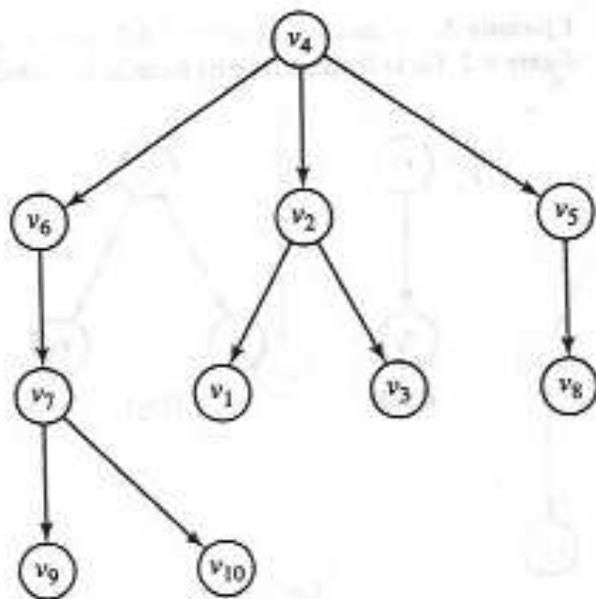


TEOREMA 3: Si (T, v_0) es un árbol con raíz v_0 y v pertenece a T , entonces:

$T(v)$ también es un árbol con raíz en v .

$T(v)$ es el subárbol que comienza en v

Ejemplo: El siguiente árbol tiene raíz en v_4 ,
con subárboles, por ejemplo, $T(v_6)$; $T(v_2)$ y $T(v_5)$



DEMOSTRACION: Si (T, v_0) es un árbol con raíz v_0 y v pertenece a T , entonces:

$T(v)$ también es un árbol con raíz en v .

$T(v)$ es el subárbol que comienza en v

Existe una trayectoria de v a cualquier otro vértice en $T(v)$
(por definición)

Si existe un vértice w en $T(v)$ tal que:

existen dos trayectorias distintas q y q' de v a w y

p es la trayectoria en T de v_0 a v

Entonces:

$$q \circ p \qquad q' \circ p$$

serían dos trayectorias distintas en T de v_0 a w .

Dos trayectorias distintas en T de v_0 a w es IMPOSIBLE

T es un árbol con raíz en v_0

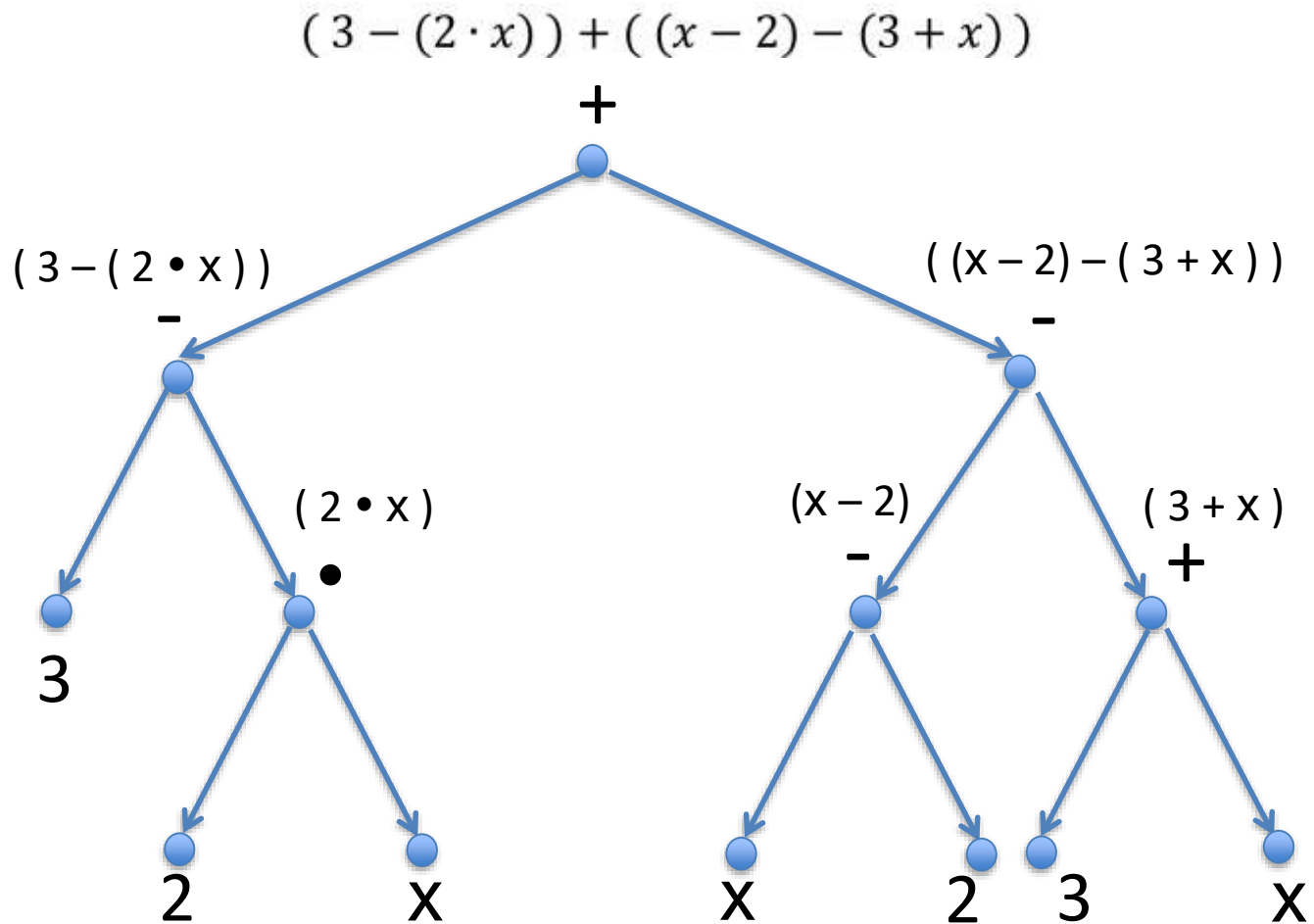
Cada trayectoria desde v a w en $T(v)$ debe ser única

Si q es un ciclo en v en $T(v)$; q es un ciclo también en T

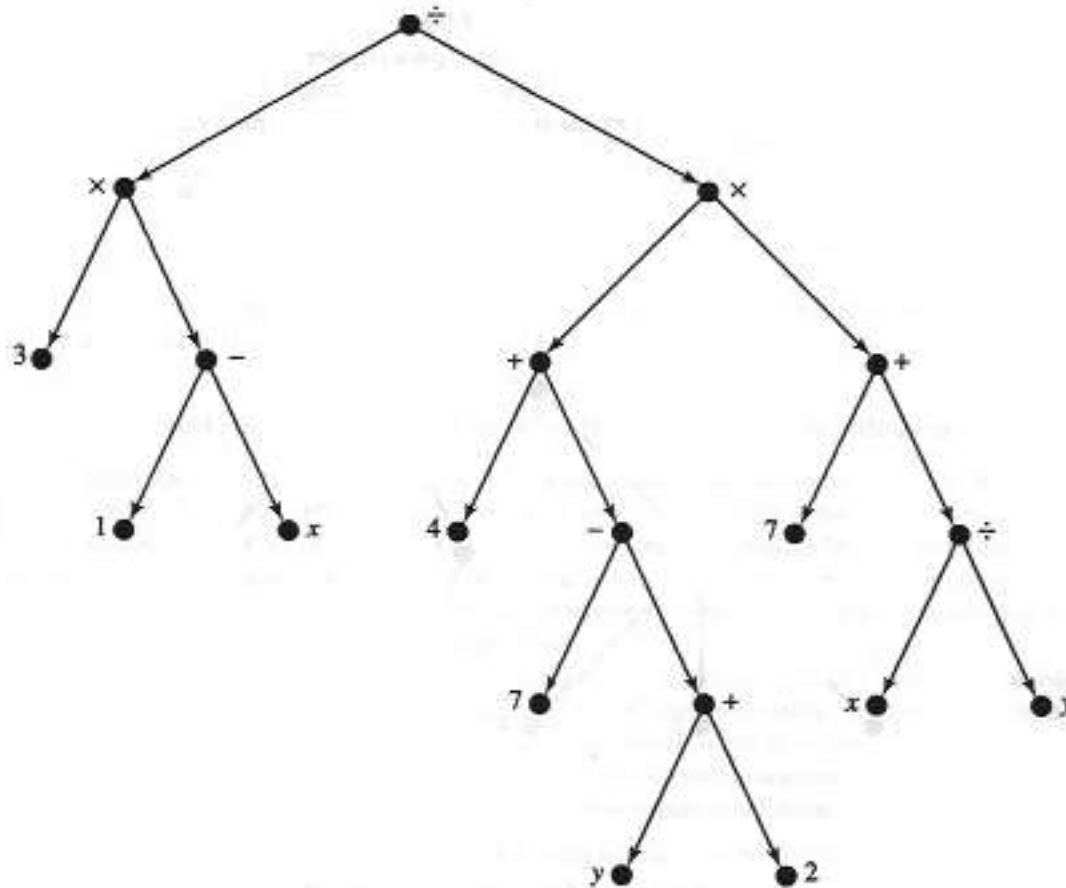
Esto contradice el Teorema 1 (a); por tanto q NO EXISTE

Esto implica que $T(v)$ es un árbol con raíz en v

Arboles etiquetados

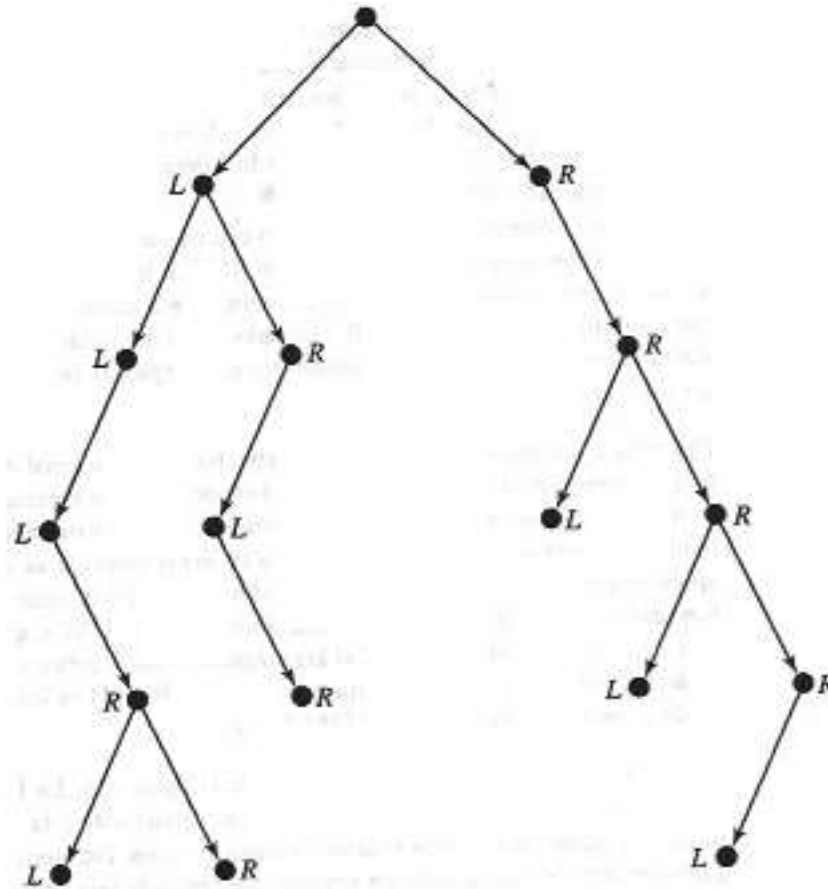


Componga la expresión que se corresponde con el árbol:



$$(3 \times (1 - x)) / ((4 + (7 - (y + 2))) \times (7 + (x / y)))$$

ARBOL BINARIO POSICIONAL



CODIGO ASCII II

Lista completa de caracteres, letras, signos y símbolos del código ASCII :

Caracteres de control ASCII no imprimibles :

codigo ascii 00 = **NUL** (Carácter nulo)
codigo ascii 01 = **SOH** (Inicio de encabezado)
codigo ascii 02 = **STX** (Inicio de texto)
codigo ascii 03 = **ETX** (Fin de texto, pero conson tarjetas inglesas de poker)
codigo ascii 04 = **EOT** (Fin de transmisión, pero diamonds tarjetas de poker)
codigo ascii 05 = **ENQ** (Consulta, pero trabosas tarjetas inglesas de poker)
codigo ascii 06 = **ACK** (Reconocimiento, pero pizas cartas de poker)
codigo ascii 07 = **BEL** (Trine)
codigo ascii 08 = **BS** (Retroceso)
codigo ascii 09 = **HT** (Tabulador horizontal)
codigo ascii 10 = **LF** (Nueva línea - salto de línea)
codigo ascii 11 = **VT** (Tabulador vertical)
codigo ascii 12 = **FF** (Nueva página - salto de página)
codigo ascii 13 = **CR** (ENTER - retorno de carro)
codigo ascii 14 = **SO** (Desplazamiento hacia afuera)
codigo ascii 15 = **SI** (Desplazamiento hacia adentro)
codigo ascii 16 = **DLE** (Escape de vínculo de datos)
codigo ascii 17 = **DC1** (Control dispositivo 1)
codigo ascii 18 = **DC2** (Control dispositivo 2)
codigo ascii 19 = **DC3** (Control dispositivo 3)
codigo ascii 20 = **DC4** (Control dispositivo 4)
codigo ascii 21 = **NAK** (Confirmación negativa)
codigo ascii 22 = **SYN** (Inactividad sincronica)
codigo ascii 23 = **ETB** (Fin del bloque de transmisión)
codigo ascii 24 = **CAN** (Cancelar)
codigo ascii 25 = **EM** (Fin del medio)
codigo ascii 26 = **SUB** (Sustitución)
codigo ascii 27 = **ESC** (Esc - escape)
codigo ascii 28 = **FS** (Separador de archivos)
codigo ascii 29 = **GS** (Separador de grupos)
codigo ascii 30 = **RS** (Separador de registros)
codigo ascii 31 = **US** (Separador de unidades)
codigo ascii 127 = **DEL** (DEL - Suprime, borra, eliminar)

Caracteres ASCII alfanumericos imprimibles :

codigo ascii 32 = **espacio** (Espacio en blanco)
codigo ascii 48 = **0** (Letra 0 mayúscula)
codigo ascii 49 = **1** (Letra 1 mayúscula)
codigo ascii 50 = **2** (Letra 2 mayúscula)
codigo ascii 51 = **3** (Letra 3 mayúscula)
codigo ascii 52 = **4** (Letra 4 mayúscula)
codigo ascii 53 = **5** (Letra 5 mayúscula)
codigo ascii 54 = **6** (Letra 6 mayúscula)
codigo ascii 55 = **7** (Letra 7 mayúscula)
codigo ascii 56 = **8** (Letra 8 mayúscula)
codigo ascii 57 = **9** (Letra 9 mayúscula)
codigo ascii 58 = **:** (Letra 0 mayúscula)
codigo ascii 59 = **;** (Letra 1 mayúscula)
codigo ascii 60 = **<** (Letra 2 mayúscula)
codigo ascii 61 = **=** (Letra 3 mayúscula)
codigo ascii 62 = **>** (Letra 4 mayúscula)
codigo ascii 63 = **?** (Letra 5 mayúscula)
codigo ascii 64 = **@** (Letra 6 mayúscula)
codigo ascii 65 = **A** (Letra 7 mayúscula)
codigo ascii 66 = **B** (Letra 8 mayúscula)
codigo ascii 67 = **C** (Letra 9 mayúscula)
codigo ascii 68 = **D** (Letra 0 mayúscula)
codigo ascii 69 = **E** (Letra 1 mayúscula)
codigo ascii 70 = **F** (Letra 2 mayúscula)
codigo ascii 71 = **G** (Letra 3 mayúscula)
codigo ascii 72 = **H** (Letra 4 mayúscula)
codigo ascii 73 = **I** (Letra 5 mayúscula)
codigo ascii 74 = **J** (Letra 6 mayúscula)
codigo ascii 75 = **K** (Letra 7 mayúscula)
codigo ascii 76 = **L** (Letra 8 mayúscula)
codigo ascii 77 = **M** (Letra 9 mayúscula)
codigo ascii 78 = **N** (Letra 0 mayúscula)
codigo ascii 79 = **O** (Letra 1 mayúscula)
codigo ascii 80 = **P** (Letra 2 mayúscula)
codigo ascii 81 = **Q** (Letra 3 mayúscula)
codigo ascii 82 = **R** (Letra 4 mayúscula)
codigo ascii 83 = **S** (Letra 5 mayúscula)
codigo ascii 84 = **T** (Letra 6 mayúscula)
codigo ascii 85 = **U** (Letra 7 mayúscula)
codigo ascii 86 = **V** (Letra 8 mayúscula)
codigo ascii 87 = **W** (Letra 9 mayúscula)
codigo ascii 88 = **X** (Letra 0 mayúscula)
codigo ascii 89 = **Y** (Letra 1 mayúscula)
codigo ascii 90 = **Z** (Letra 2 mayúscula)
codigo ascii 91 = **[** (Abre corchete)
codigo ascii 92 = **** (Barra invertida, contrabarra, barra inversa)
codigo ascii 93 = **]** (Cierra corchete)
codigo ascii 94 = **^** (Intersección - acento circunflejo)
codigo ascii 95 = **_** (Guion bajo, subrayado, subrayar)
codigo ascii 96 = **`** (Acento grave)
codigo ascii 97 = **a** (Letra a minúscula)
codigo ascii 98 = **b** (Letra b minúscula)
codigo ascii 99 = **c** (Letra c minúscula)
codigo ascii 100 = **d** (Letra d minúscula)
codigo ascii 101 = **e** (Letra e minúscula)
codigo ascii 102 = **f** (Letra f minúscula)
codigo ascii 103 = **g** (Letra g minúscula)

Caracteres ASCII extendidos imprimibles :

codigo ascii 128 = **Ç** (Letra C cedilla mayúscula)
codigo ascii 129 = **À** (Letra a minúscula con diéresis)
codigo ascii 130 = **Á** (Letra a minúscula con acento agudo)
codigo ascii 131 = **Â** (Letra a minúscula con acento circunflejo)
codigo ascii 132 = **Ã** (Letra a minúscula con diéresis)
codigo ascii 133 = **Ä** (Letra a minúscula con acento grave)
codigo ascii 134 = **Å** (Letra a minúscula con anillo)
codigo ascii 135 = **ç** (Letra c cedilla minúscula)
codigo ascii 136 = **À** (Letra a minúscula con acento circunflejo)
codigo ascii 137 = **Á** (Letra a minúscula con diéresis)
codigo ascii 138 = **Â** (Letra a minúscula con acento grave)
codigo ascii 139 = **Ã** (Letra a minúscula con diéresis)
codigo ascii 140 = **Ä** (Letra a minúscula con acento circunflejo)
codigo ascii 141 = **Å** (Letra a minúscula con acento grave)
codigo ascii 142 = **Ä** (Letra A mayúscula con diéresis)
codigo ascii 143 = **Å** (Letra A mayúscula con anillo)
codigo ascii 144 = **Ê** (Letra E mayúscula con acento agudo)
codigo ascii 145 = **æ** (Distinguo latino ae minúscula)
codigo ascii 146 = **Æ** (Distinguo latino AE mayúscula)
codigo ascii 147 = **ø** (Letra o minúscula con acento circunflejo)
codigo ascii 148 = **Ö** (Letra o minúscula con diéresis)
codigo ascii 149 = **ø** (Letra o minúscula con acento grave)
codigo ascii 150 = **ö** (Letra u minúscula con acento circunflejo)
codigo ascii 151 = **ø** (Letra u minúscula con acento grave)
codigo ascii 152 = **ÿ** (Letra y minúscula con diéresis)
codigo ascii 153 = **Û** (Letra O mayúscula con diéresis)
codigo ascii 154 = **Ü** (Letra U mayúscula con diéresis)
codigo ascii 155 = **ü** (Letra o minúscula con barra inclinada)
codigo ascii 156 = **Ë** (Signo Letra Esarlina)
codigo ascii 157 = **ß** (Letra O mayúscula con barra inclinada)
codigo ascii 158 = **×** (Signo de multiplicación)
codigo ascii 159 = **ƒ** (Símbolo de función, forin neerlandés)
codigo ascii 160 = **¸** (Letra a minúscula con acento agudo)
codigo ascii 161 = **í** (Letra i minúscula con acento agudo)
codigo ascii 162 = **ï** (Letra i minúscula con acento agudo)
codigo ascii 163 = **î** (Letra i minúscula con acento agudo)
codigo ascii 164 = **ï** (Letra i minúscula con acento agudo)
codigo ascii 165 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 166 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 167 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 168 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 169 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 170 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 171 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 172 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 173 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 174 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 175 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 176 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 177 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 178 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 179 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 180 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 181 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 182 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 183 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 184 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 185 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 186 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 187 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 188 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 189 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 190 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 191 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 192 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 193 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 194 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 195 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 196 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 197 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 198 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 199 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 200 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 201 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 202 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 203 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 204 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 205 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 206 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 207 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 208 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 209 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 210 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 211 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 212 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 213 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 214 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 215 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 216 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 217 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 218 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 219 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 220 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 221 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 222 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 223 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 224 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 225 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 226 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 227 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 228 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 229 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 230 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 231 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 232 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 233 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 234 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 235 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 236 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 237 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 238 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 239 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 240 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 241 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 242 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 243 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 244 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 245 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 246 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 247 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 248 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 249 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 250 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 251 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 252 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 253 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 254 = **ÿ** (Letra y minúscula con acento agudo)
codigo ascii 255 = **ÿ** (Letra y minúscula con acento agudo)

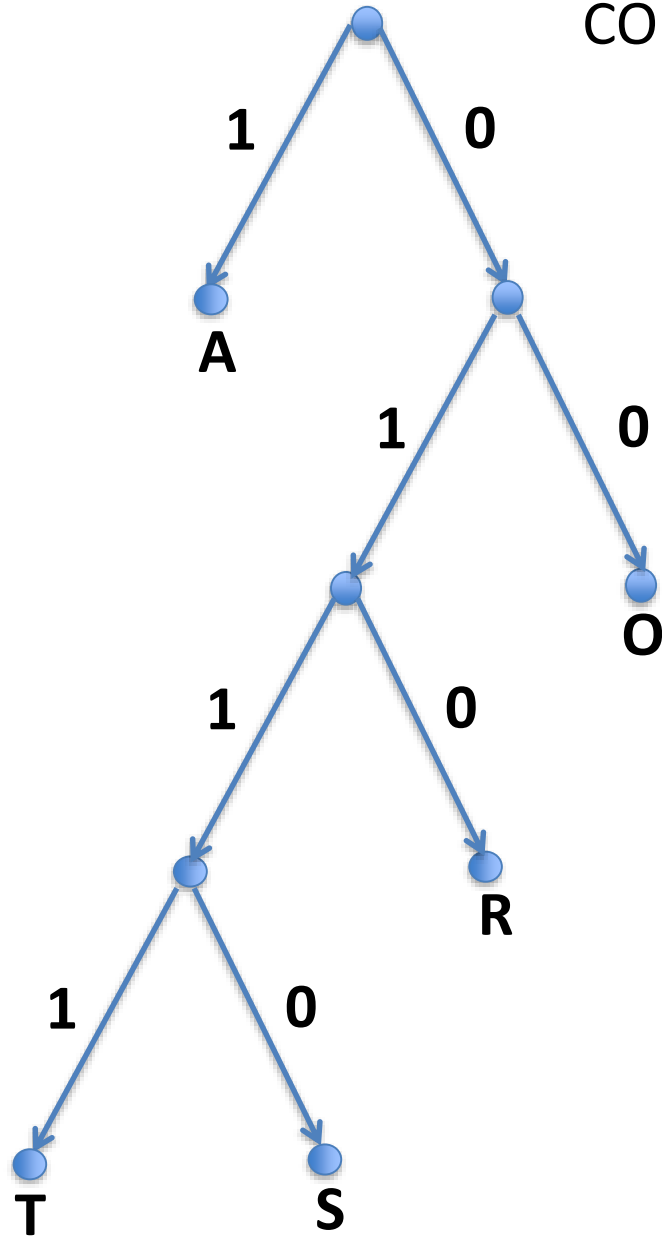
codigo ascii 33 = **!** (Signo de exclamación, signo de admiración)
codigo ascii 34 = **"** (Comillas dobles, comillas altas o inglesas)
codigo ascii 35 = **#** (Signo numeral o arrobas)
codigo ascii 36 = **\$** (Signo peso)
codigo ascii 37 = **%** (Signo de porcentaje - por ciento)
codigo ascii 38 = **&** (Y - ampersand - el letra)
codigo ascii 39 = **'** (Comillas simples, apóstrofe)
codigo ascii 40 = **(** (Abre paréntesis)
codigo ascii 41 = **)** (Cierra paréntesis)
codigo ascii 42 = ***** (Asterisco)
codigo ascii 43 = **+** (Signo mas, suma, positivo)
codigo ascii 44 = **,** (Coma)
codigo ascii 45 = **-** (Signo menos, resta, negativo, guión medio)
codigo ascii 46 = **.** (Punto)
codigo ascii 47 = **/** (Barra inclinada, división, operador cociente)
codigo ascii 48 = **0** (Número cero)
codigo ascii 49 = **1** (Número uno)
codigo ascii 50 = **2** (Número dos)
codigo ascii 51 = **3** (Número tres)
codigo ascii 52 = **4** (Número cuatro)
codigo ascii 53 = **5** (Número cinco)
codigo ascii 54 = **6** (Número seis)
codigo ascii 55 = **7** (Número siete)
codigo ascii 56 = **8** (Número ocho)
codigo ascii 57 = **9** (Número nueve)
codigo ascii 58 = **:** (Dos puntos)
codigo ascii 59 = **:** (Punto y coma)
codigo ascii 60 = **<** (Menor que)
codigo ascii 61 = **=** (Signo igual, igualdad, igual que)
codigo ascii 62 = **>** (Mayor que)
codigo ascii 63 = **?** (Cierra signo interrogación)
codigo ascii 64 = **@** (Arriba)
codigo ascii 65 = **A** (Letra A mayúscula)
codigo ascii 66 = **B** (Letra B mayúscula)
codigo ascii 67 = **C** (Letra C mayúscula)
codigo ascii 68 = **D** (Letra D mayúscula)
codigo ascii 69 = **E** (Letra E mayúscula)
codigo ascii 70 = **F** (Letra F mayúscula)
codigo ascii 71 = **G** (Letra G mayúscula)
codigo ascii 72 = **H** (Letra H mayúscula)

codigo ascii 104 = **H** (Letra h minúscula - letra h con tilde - eta)
codigo ascii 105 = **I** (Letra ERE mayúscula - letra N con tilde - ENE)
codigo ascii 106 = **J** (Ordinal femenino, indicador de género femenino)
codigo ascii 107 = **K** (Ordinal masculino, indicador de género masculino)
codigo ascii 108 = **L** (Abre signo interrogación)
codigo ascii 109 = **M** (Símbolo de Maria Regenerata)
codigo ascii 110 = **N** (Signo de negación)
codigo ascii 111 = **O** (Un medio, mitad, fracción)
codigo ascii 112 = **P** (Un cuarto, cuarta parte, fracción)
codigo ascii 113 = **Q** (Abre signo de exclamación, signo de admiración)
codigo ascii 114 = **R** (Abre corchete bajos, angulares, letinas o espasmos)
codigo ascii 115 = **S** (Cierra corchete bajos, angulares, letinas o espasmos)
codigo ascii 116 = **T** (Bloque color llamado densidad baja, carácter gráfico)
codigo ascii 117 = **U** (Bloque color llamado densidad media, gráfico)
codigo ascii 118 = **V** (Bloque color llamado densidad alta, carácter gráfico)
codigo ascii 119 = **W** (Línea simple vertical de recuadro gráfico)
codigo ascii 120 = **X** (Línea vertical con empalme de recuadro gráfico)
codigo ascii 121 = **Y** (Letra a mayúscula con acento agudo)
codigo ascii 122 = **Z** (Letra A mayúscula con acento circunflejo)
codigo ascii 123 = **A** (Letra A mayúscula con acento grave)
codigo ascii 124 = **B** (Símbolo Copyright, bajo derecho de autor)
codigo ascii 125 = **C** (Doble línea vertical empalme izquierdo, gráfico)
codigo ascii 126 = **D** (Línea doble vertical de recuadro gráfico, verticales)
codigo ascii 127 = **E** (Línea doble superior derecha de recuadro)
codigo ascii 128 = **F** (Línea doble superior inferior derecha de recuadro)
codigo ascii 129 = **G** (Signo centavo, centimo o centésimo)
codigo ascii 130 = **H** (Signo monetario YEN japonés, YUAN chino)
codigo ascii 131 = **I** (Línea simple superior de recuadro gráfico)
codigo ascii 132 = **J** (Línea simple superior de recuadro gráfico)
codigo ascii 133 = **K** (Línea horizontal con empalme de recuadro gráfico)
codigo ascii 134 = **L** (Línea horizontal con empalme de recuadro gráfico)
codigo ascii 135 = **M** (Línea vertical con empalme de recuadro gráfico)
codigo ascii 136 = **N** (Línea simple horizontal de recuadro gráfico)
codigo ascii 137 = **O** (Línea simple empalme de recuadro gráfico)
codigo ascii 138 = **P** (Letra a minúscula con tilde)
codigo ascii 139 = **Q** (Letra a minúscula con acento agudo)
codigo ascii 140 = **R** (Línea doble superior izquierda de recuadro)
codigo ascii 141 = **S** (Línea doble superior superior izquierda de recuadro)
codigo ascii 142 = **T** (Doble línea horizontal empalme arriba, recuadro)
codigo ascii 143 = **U** (Doble línea horizontal empalme abajo, recuadro)

codigo ascii 104 = **H** (Letra h minúscula)
codigo ascii 105 = **I** (Letra i minúscula)
codigo ascii 106 = **J** (Letra j minúscula)
codigo ascii 107 = **K** (Letra k minúscula)
codigo ascii 108 = **L** (Letra l minúscula)
codigo ascii 109 = **M** (Letra m minúscula)
codigo ascii 110 = **N** (Letra n minúscula)
codigo ascii 111 = **O** (Letra o minúscula)
codigo ascii 112 = **P** (Letra p minúscula)
codigo ascii 113 = **Q** (Letra q minúscula)
codigo ascii 114 = **R** (Letra r minúscula)
codigo ascii 115 = **S** (Letra s minúscula)
codigo ascii 116 = **T** (Letra t minúscula)
codigo ascii 117 = **U** (Letra u minúscula)
codigo ascii 118 = **V** (Letra v minúscula)
codigo ascii 119 = **W** (Letra w minúscula)
codigo ascii 120 = **X** (Letra x minúscula)
codigo ascii 121 = **Y** (Letra y minúscula)
codigo ascii 122 = **Z** (Letra z minúscula)
codigo ascii 123 = **{** (Abre llave curva - llaves curvas)
codigo ascii 124 = **|** (Barra vertical, pieza, línea vertical)
codigo ascii 125 = **}** (Cierra llave - llaves curvas)
codigo ascii 126 = **~** (Signo de equivalencia, tilde o virgula de la h)

codigo ascii 104 = **H** (Letra h minúscula con acento agudo)
codigo ascii 105 = **I** (Letra i minúscula con acento agudo)
codigo ascii 106 = **J** (Letra j minúscula con acento agudo)
codigo ascii 107 = **K** (Letra k minúscula con acento agudo)
codigo ascii 108 = **L** (Letra l minúscula con acento agudo)
codigo ascii 109 = **M** (Símbolo matemático de congruencia, equivalencia)
codigo ascii 110 = **N** (Signo mas menos)
codigo ascii 111 = **O** (ASCE 242)
codigo ascii 112 = **P** (Tres cuartos, fracción)
codigo ascii 113 = **Q** (Fin de párrafo - signo de caído)
codigo ascii 114 = **R** (Signo de sección)
codigo ascii 115 = **S** (Signo de división)
codigo ascii 116 = **T** (Cedilla, virgula baja)
codigo ascii 117 = **U** (Signo de grado, ardo)
codigo ascii 118 = **V** (Diéresis)
codigo ascii 119 = **W** (Punto centésimo, punto medio, coma gascona)
codigo ascii 120 = **X** (Superíndice uno)
codigo ascii 121 = **Y** (Superíndice tres, potencia tres, al cubo)
codigo ascii 122 = **Z** (Superíndice dos, al cuadrado)
codigo ascii 123 = **{** (Cuadrado negro, carácter gráfico)
codigo ascii 124 = **|** (Signo de separación - non breaking space)

CODIGO DE HUFMAN

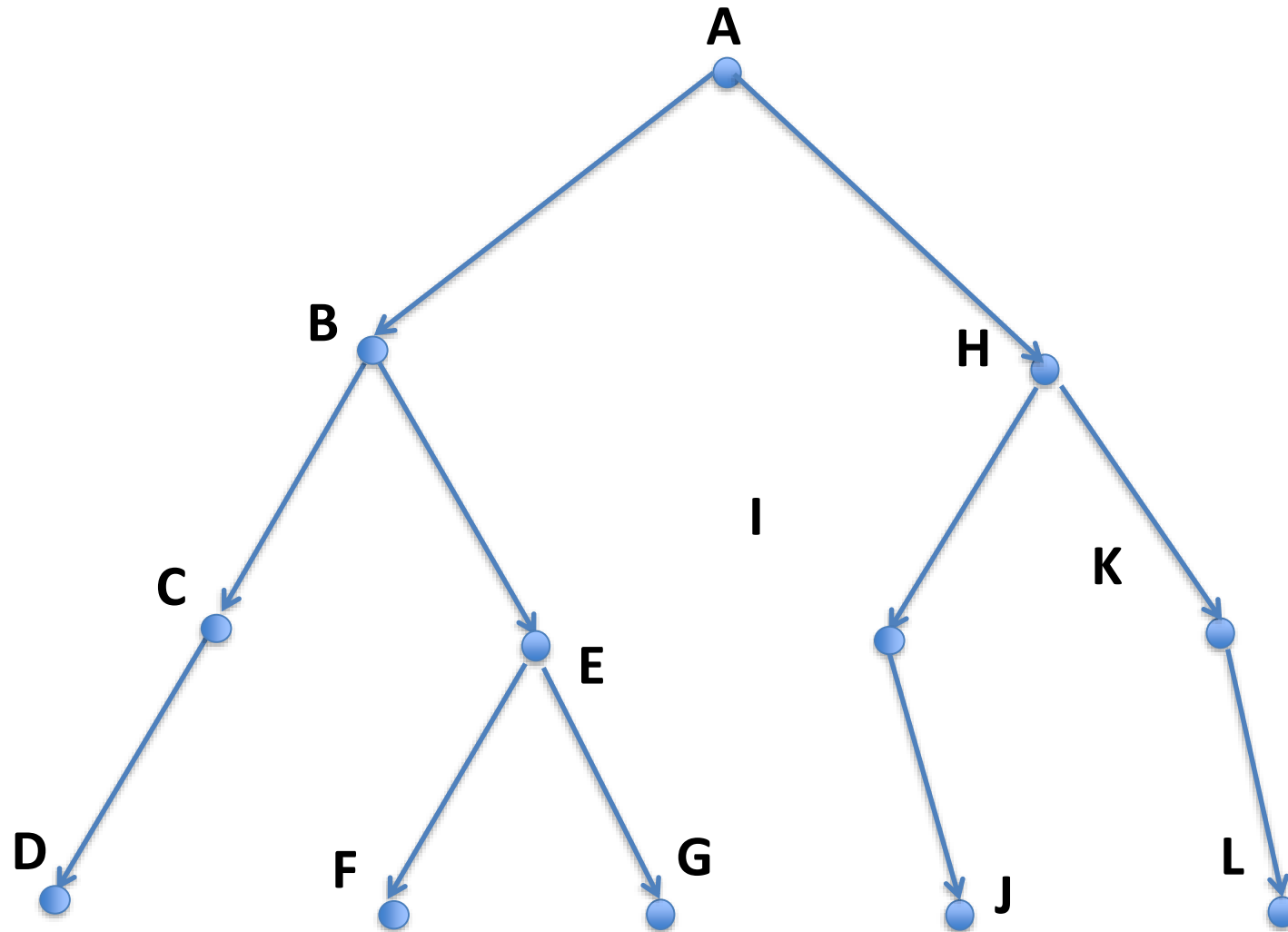


Carácter		
A	100	0001
B	100	0010
C	100	0011
1	011	0001
2	011	0010
!	010	0001
*	010	1010

RAT 0 1 0 1 0 1 1 1

RATO 0 1 0 1 0 1 1 1 0 0

BUSQUEDA EN ARBOLES



Algoritmo PREORDEN

PASO 1: Visite v

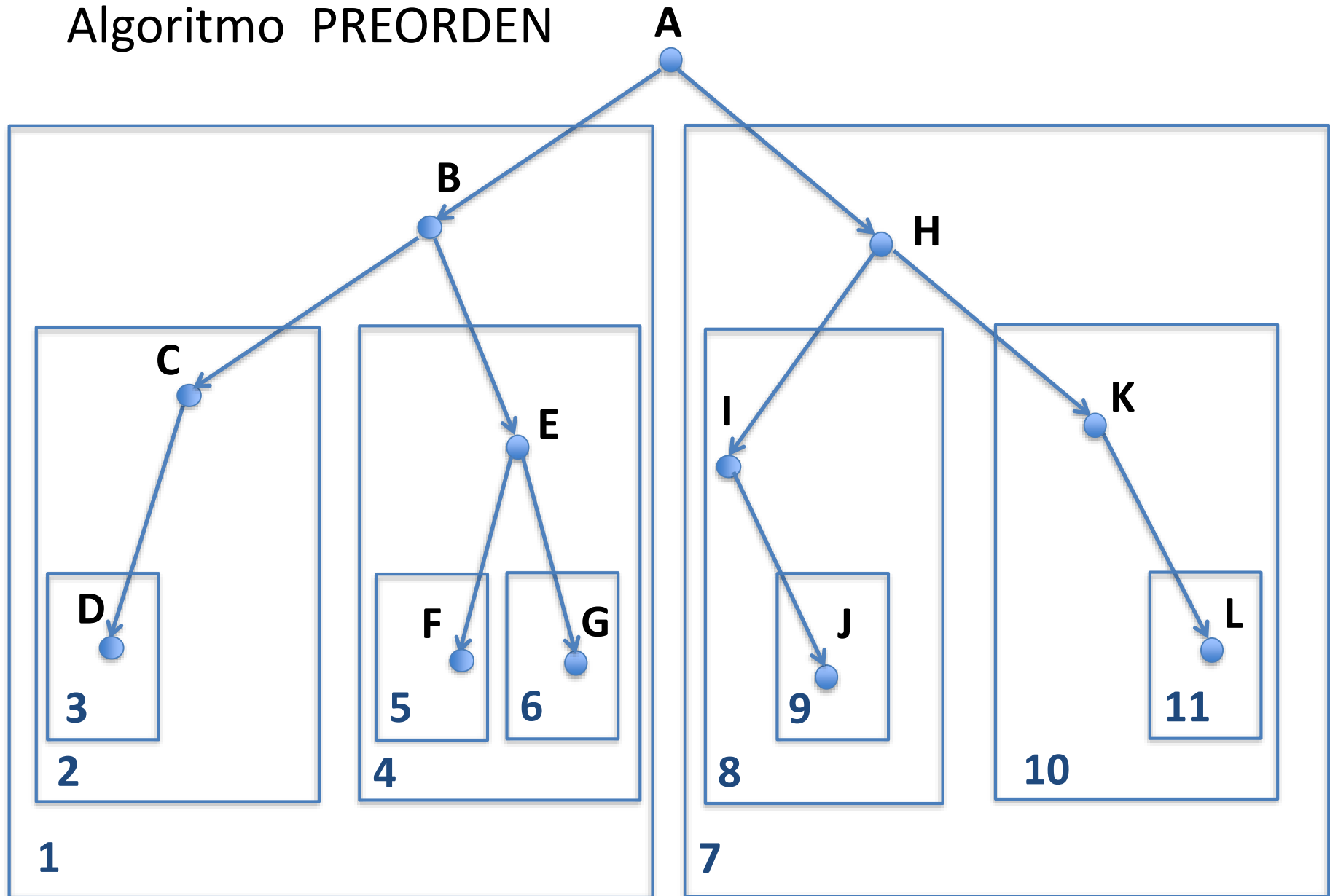
PASO 2: Si existe v_L , entonces aplique este algoritmo
a $(T(v_L), v_L)$

PASO 3: Si existe v_R , entonces aplique este algoritmo
a $(T(v_R), v_R)$

Fin del algoritmo

-
1. Visite la raíz.
 2. Busque en el sub árbol izquierdo, si existe
 3. Busque en el sub árbol derecho, si existe

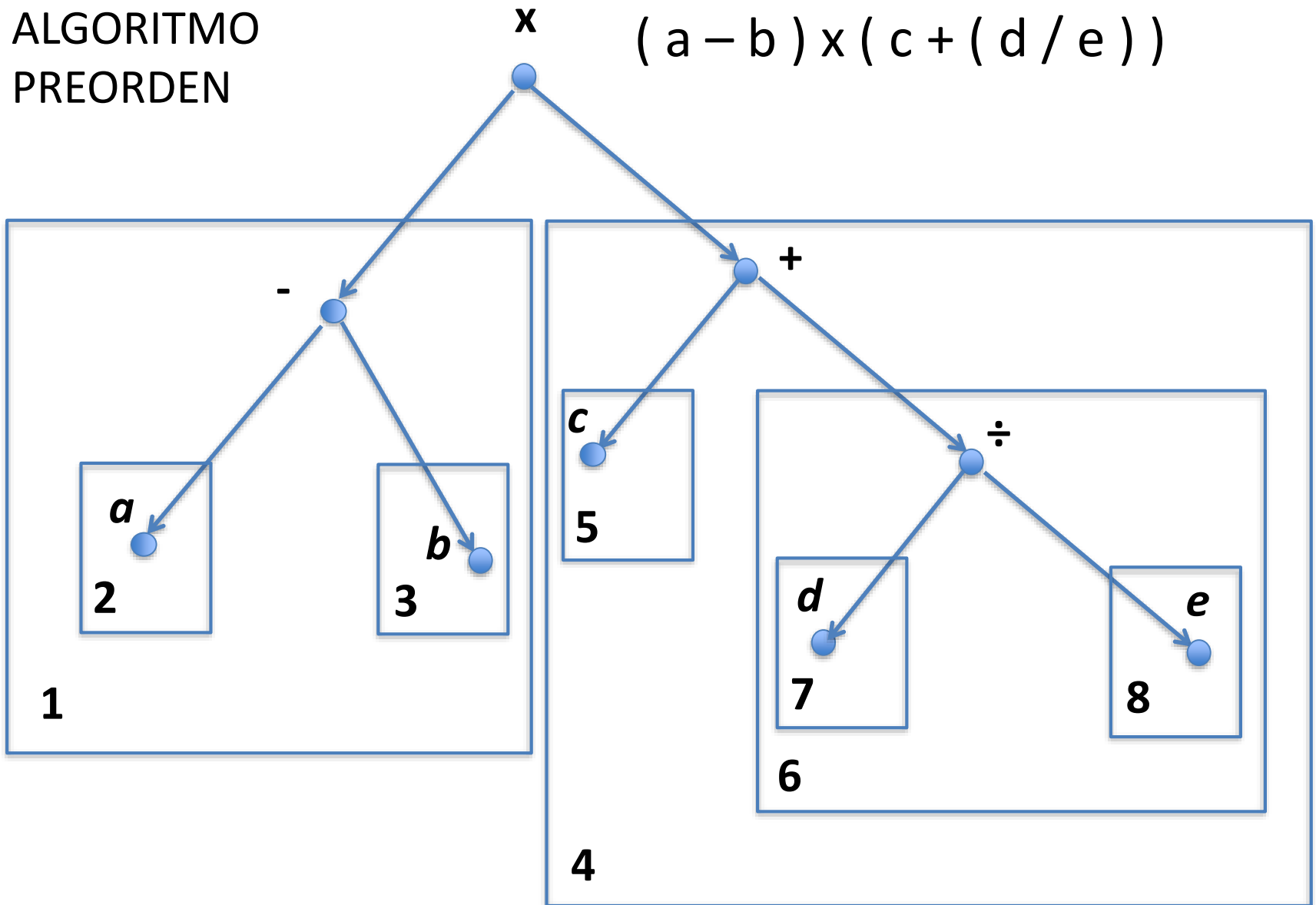
Algoritmo PREORDEN



A B C D E F G H I J K L

ALGORITMO
PREORDEN

$$(a - b) \times (c + (d / e))$$



$x - a b + c \div d e$

Escritura polaca
2019

ALGORITMO PREORDEN

$$(a - b) \times (c + (d / e))$$

$$a = 6$$

$$b = 4$$

$$x - a \quad b + c / d \quad e$$

$$c = 5$$

$$d = 2$$

$$e = 2$$

$$x - 6 \quad 4 + 5 \quad / \quad 2 \quad 2$$

$$x - 6 \quad 4 + 5 \quad 1$$

$$x \quad 2 \quad 6$$

$$12$$

Algoritmo ENTREORDEN

PASO 1: Si existe v_L , entonces aplique este algoritmo a $(T(v_L), v_L)$

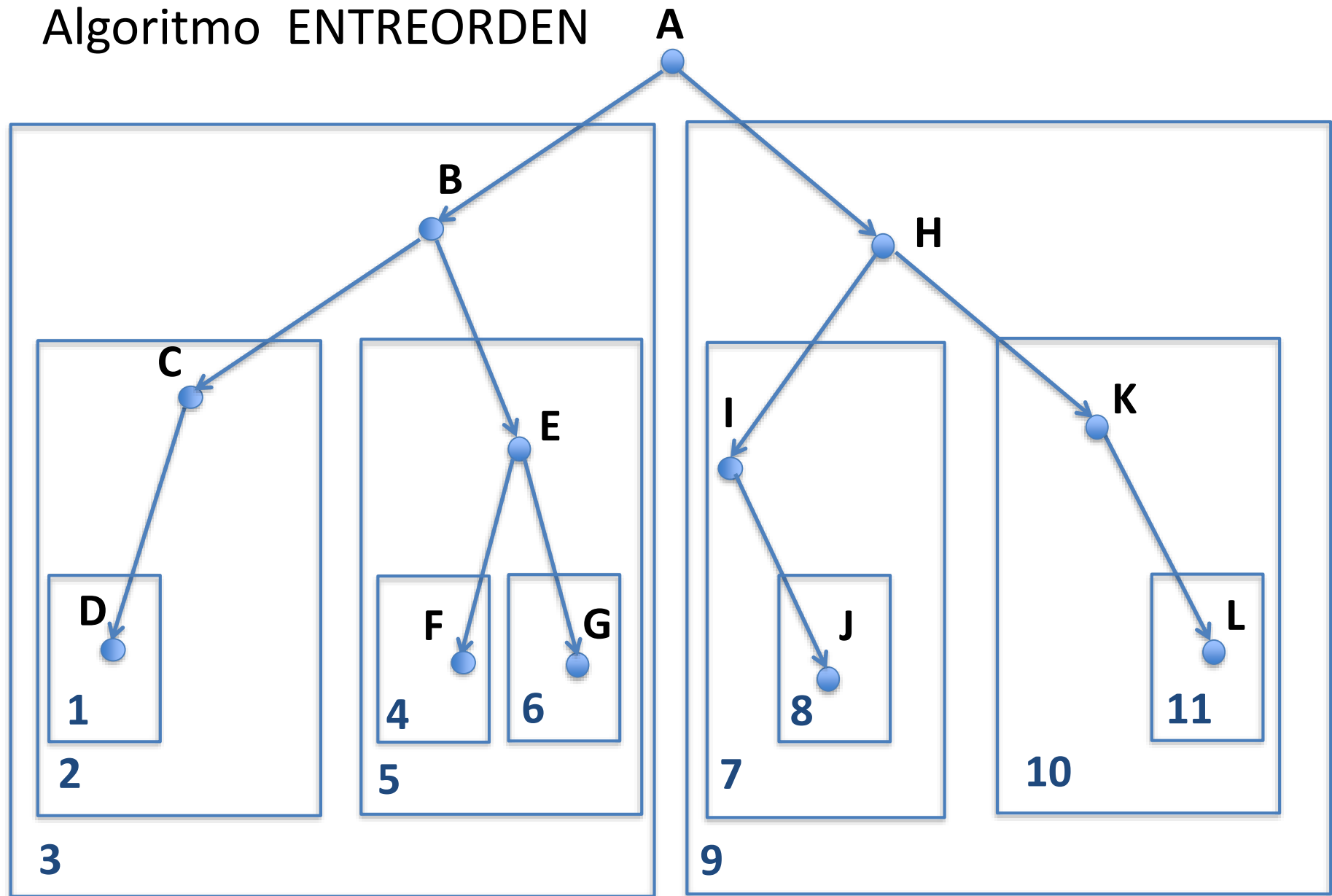
PASO 2: Visite v

PASO 3: Si existe v_R , entonces aplique este algoritmo a $(T(v_R), v_R)$

Fin del algoritmo

-
1. Busque en el sub árbol izquierdo, si existe
 2. Visite la raíz.
 3. Busque en el sub árbol derecho, si existe

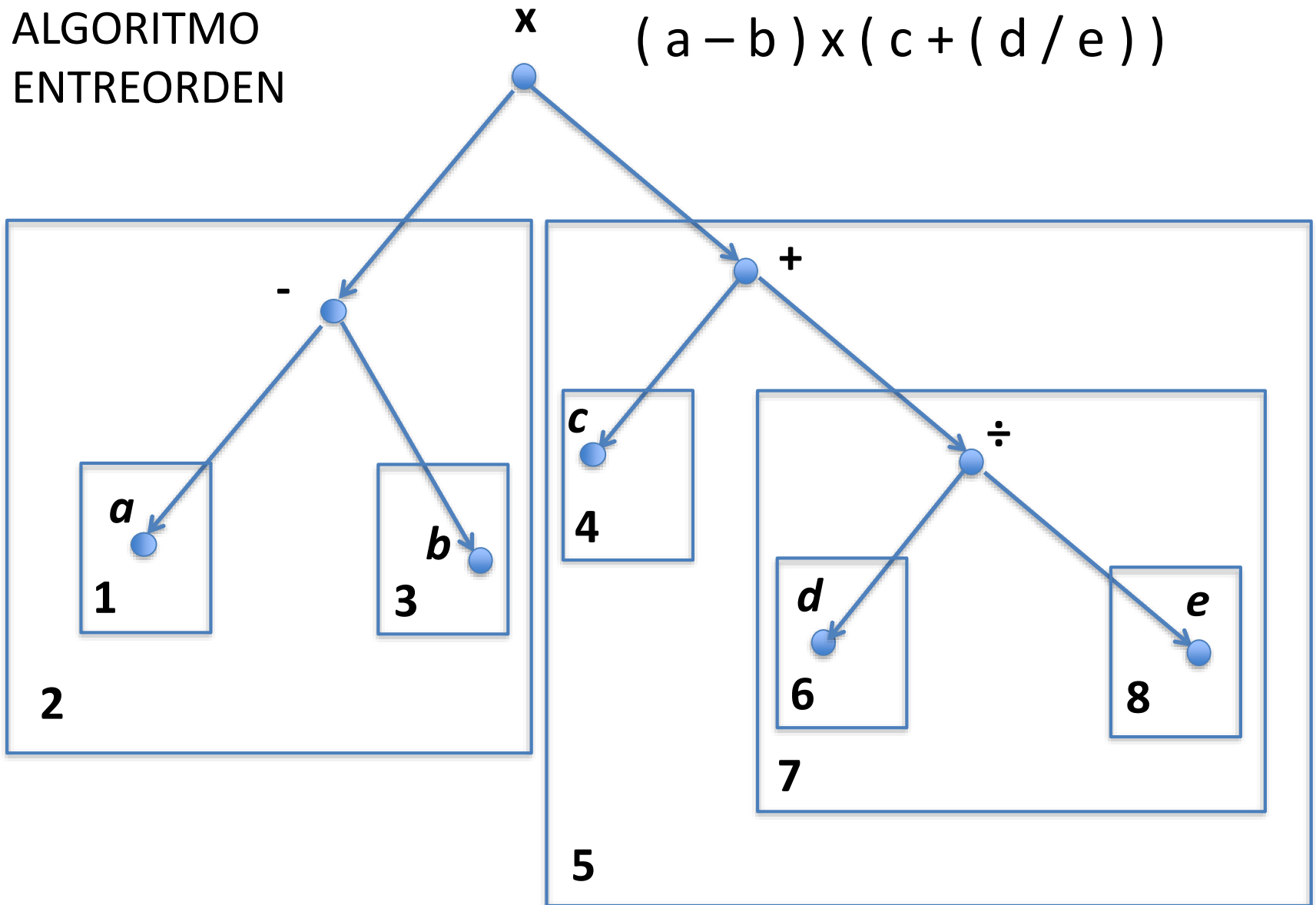
Algoritmo ENTREORDEN



D C B F E G A I J H K L

ALGORITMO
ENTREORDEN

$$(a - b) \times (c + (d / e))$$



$$a - b \times c + d \div e$$

ALGORITMO ENTREORDEN

$$(a - b) \times (c + (d / e))$$

$a - b \times c + d / e$

$$\underbrace{6 - 4}_2 \times 5 + \underbrace{2 / 2}_1$$

$$2 \times \underbrace{5 + 1}_6$$

$$\underbrace{2 \times 6}_{12}$$

12

$a = 6$

$b = 4$

$c = 5$

$d = 2$

$e = 2$

$$6 - \underbrace{4 \times 5}_{20} + \underbrace{2 / 2}_1$$

$$\underbrace{6 - 20}_{-14} + 1$$

$$- 14 + 1 = - 13$$

ambiguo

Algoritmo POSORDEN

PASO 1: Si existe v_L , entonces aplique este algoritmo a $(T(v_L), v_L)$

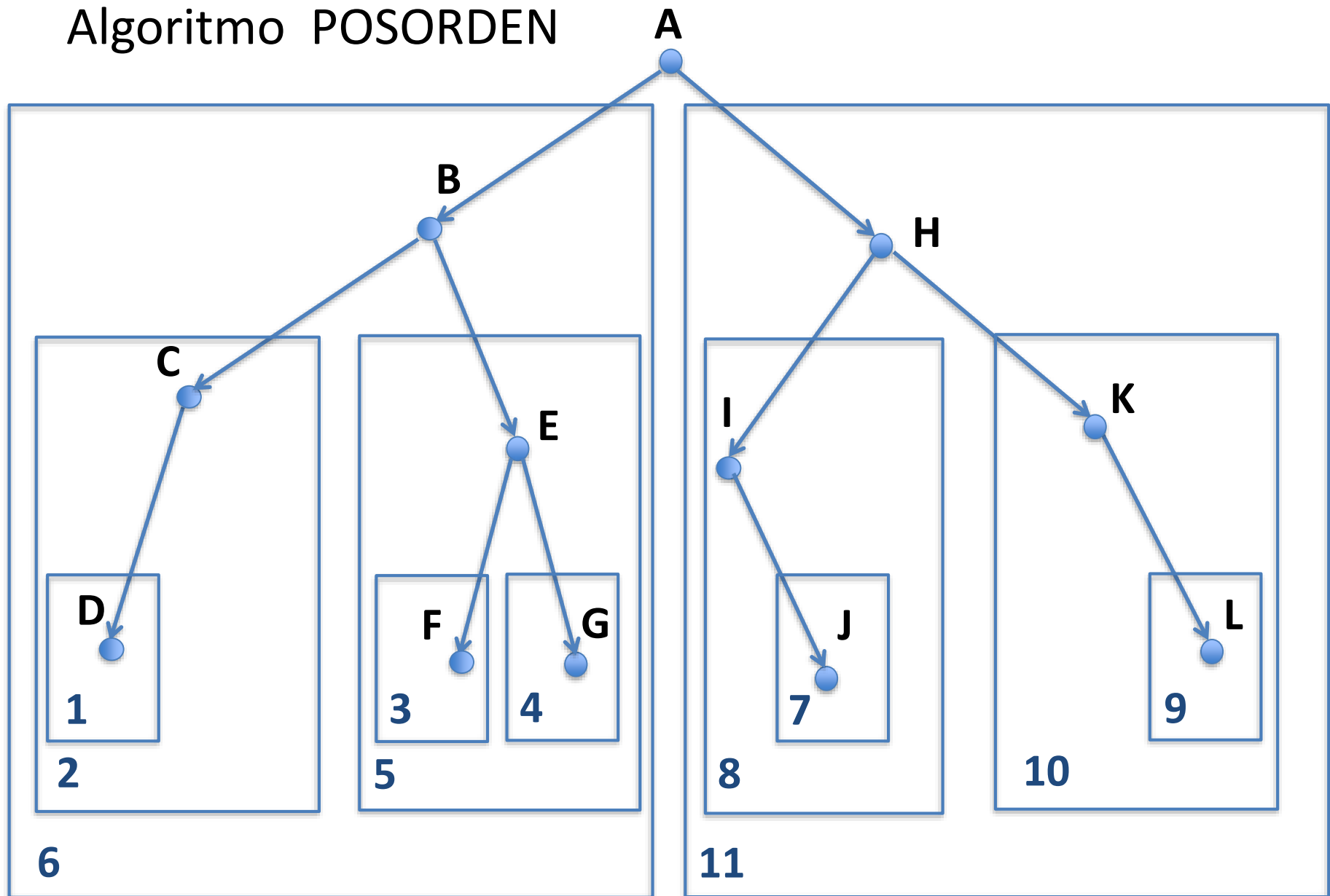
PASO 2: Si existe v_R , entonces aplique este algoritmo a $(T(v_R), v_R)$

PASO 3: Visite v

Fin del algoritmo

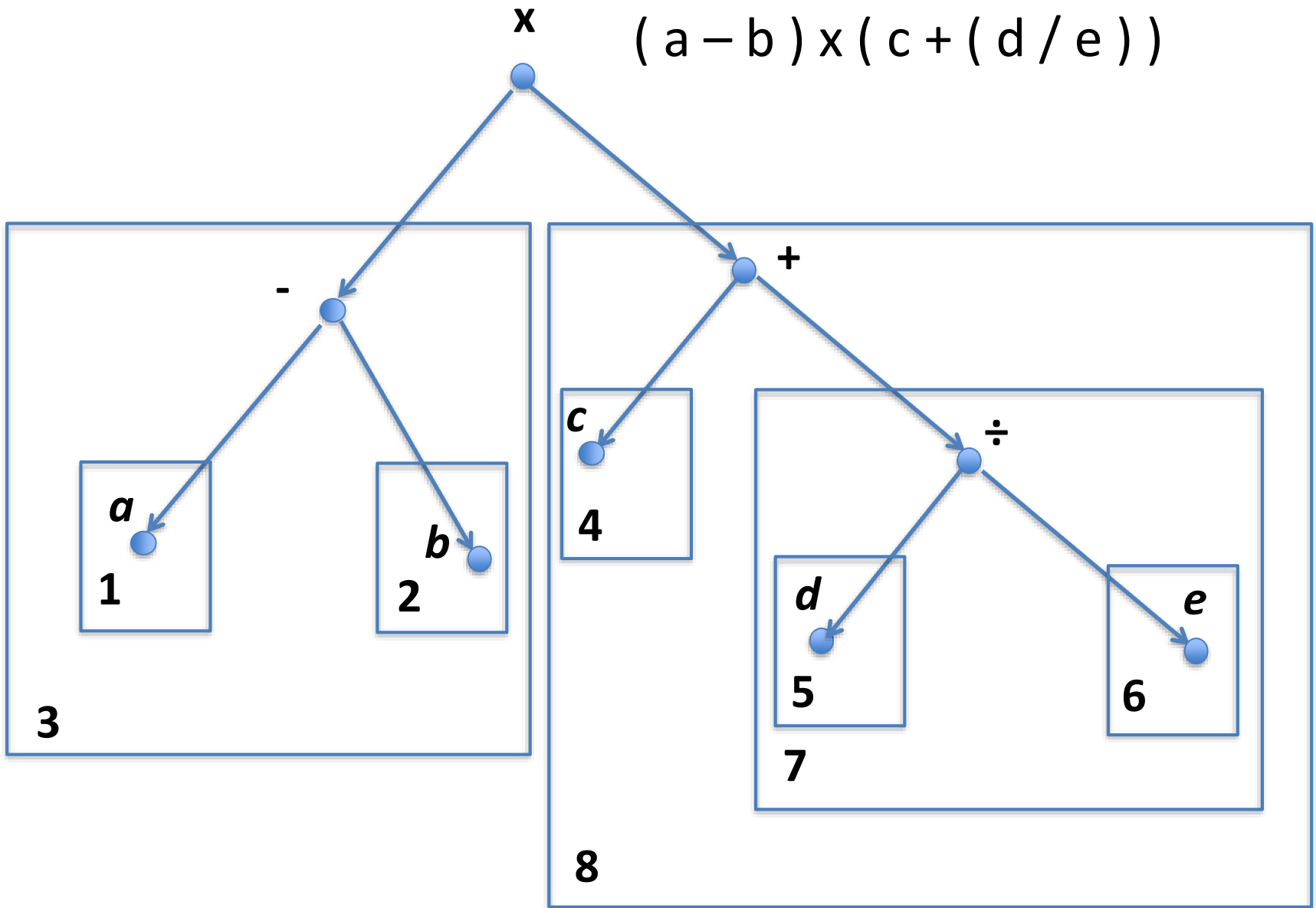
-
1. Busque en el sub árbol izquierdo, si existe
 2. Busque en el sub árbol derecho, si existe
 3. Visite la raíz.

Algoritmo POSORDEN



D C F G E B J I L K H A

$$(a - b) \times (c + (d / e))$$



$a \ b \ - \ c \ d \ e \ \div \ + \ x$

Escritura polaca inversa
2019

Algoritmo POSORDEN

$$(a - b) \times (c + (d / e))$$

a b - c d e / + x

$$\underbrace{6 \ 4 \ -}_{2} \quad \underbrace{5 \ 2 \ 2 \ /}_{1} \ + \ x$$

$$2 \quad \underbrace{5 \ 1 \ +}_{6} \ x$$

$$\underbrace{2 \quad 6 \quad \times}_{12}$$

$$a = 6$$

$$b = 4$$

$$c = 5$$

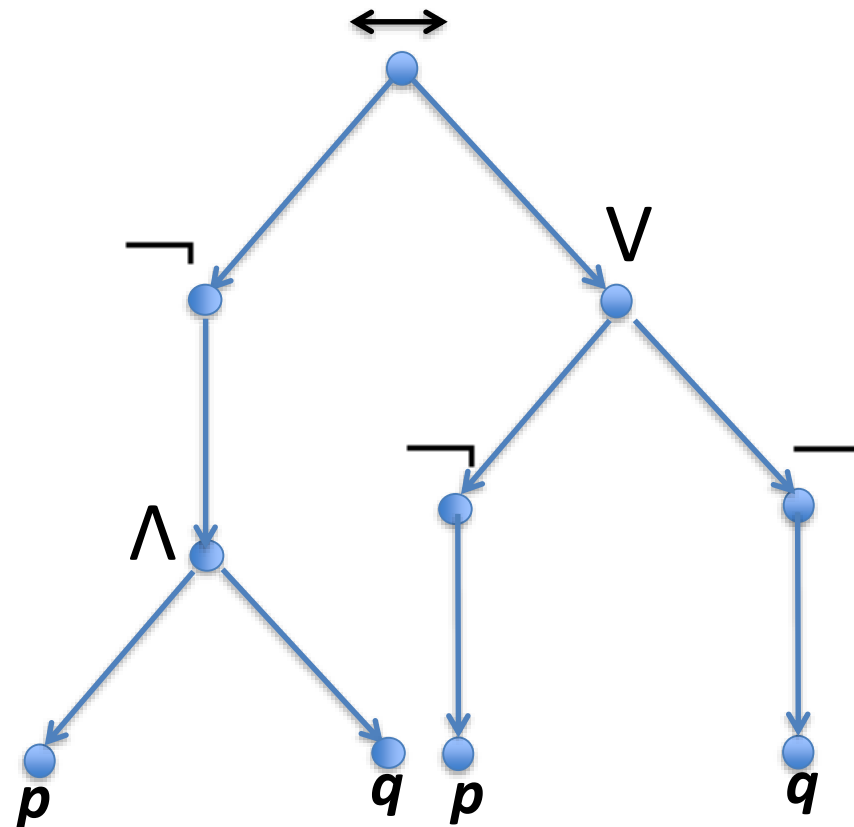
$$d = 2$$

$$e = 2$$

12

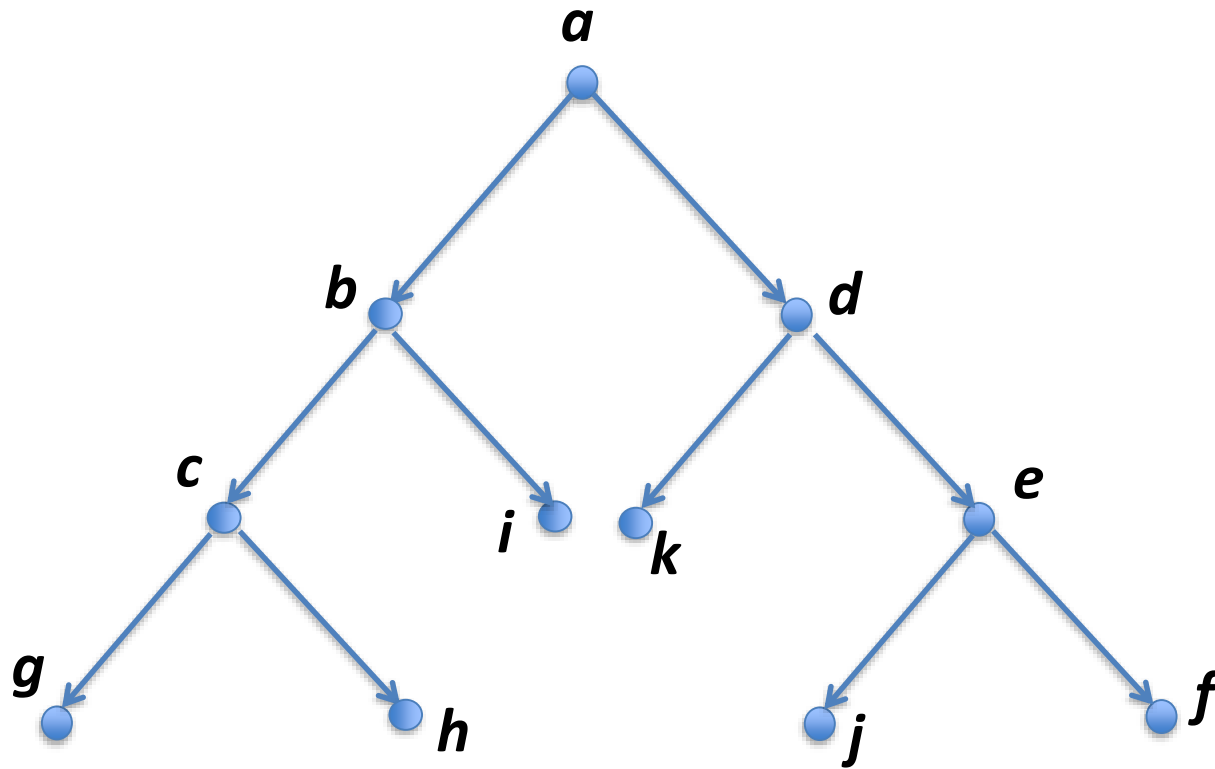
Ejercicios: Construya un árbol para la siguiente operación lógica e implemente su recorrido en los tres sentidos

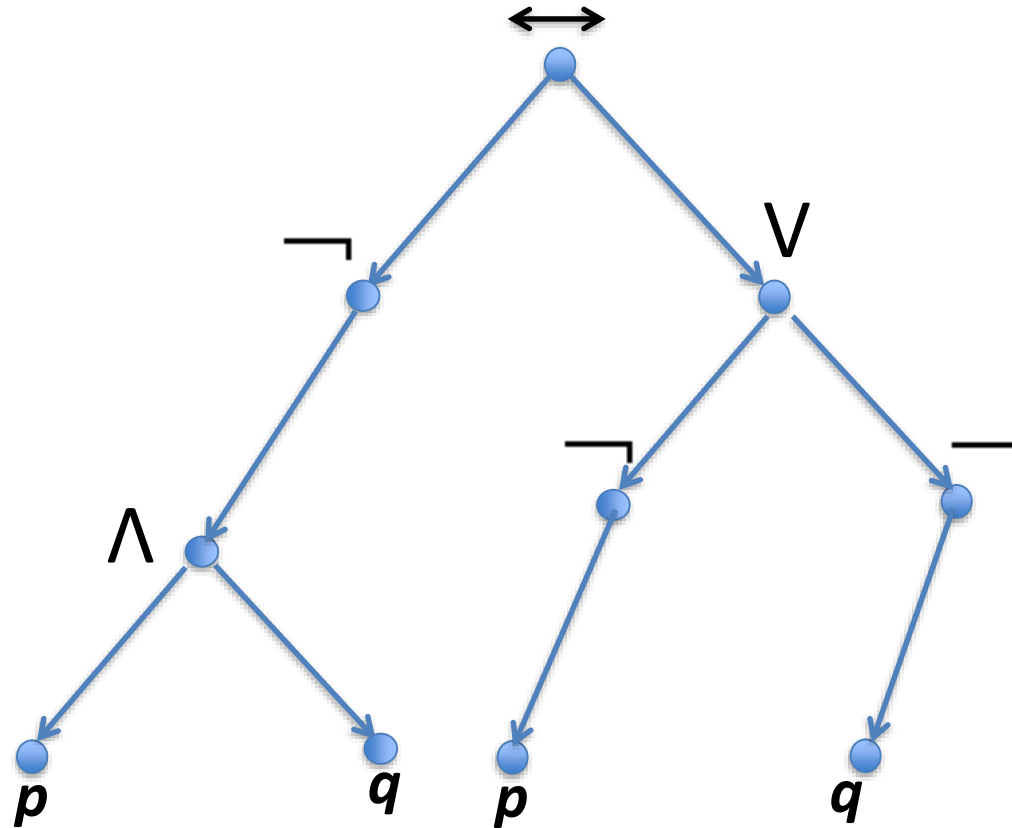
$$(\neg(p \wedge q)) \leftrightarrow (\neg p \vee \neg q)$$



Ejercicios:

Efectúe búsqueda PREORDEN,
ENTREORDEN Y POSORDEN en el
árbol presentado





$\leftrightarrow \neg \wedge p q V \neg p \neg q$

PREORDEN

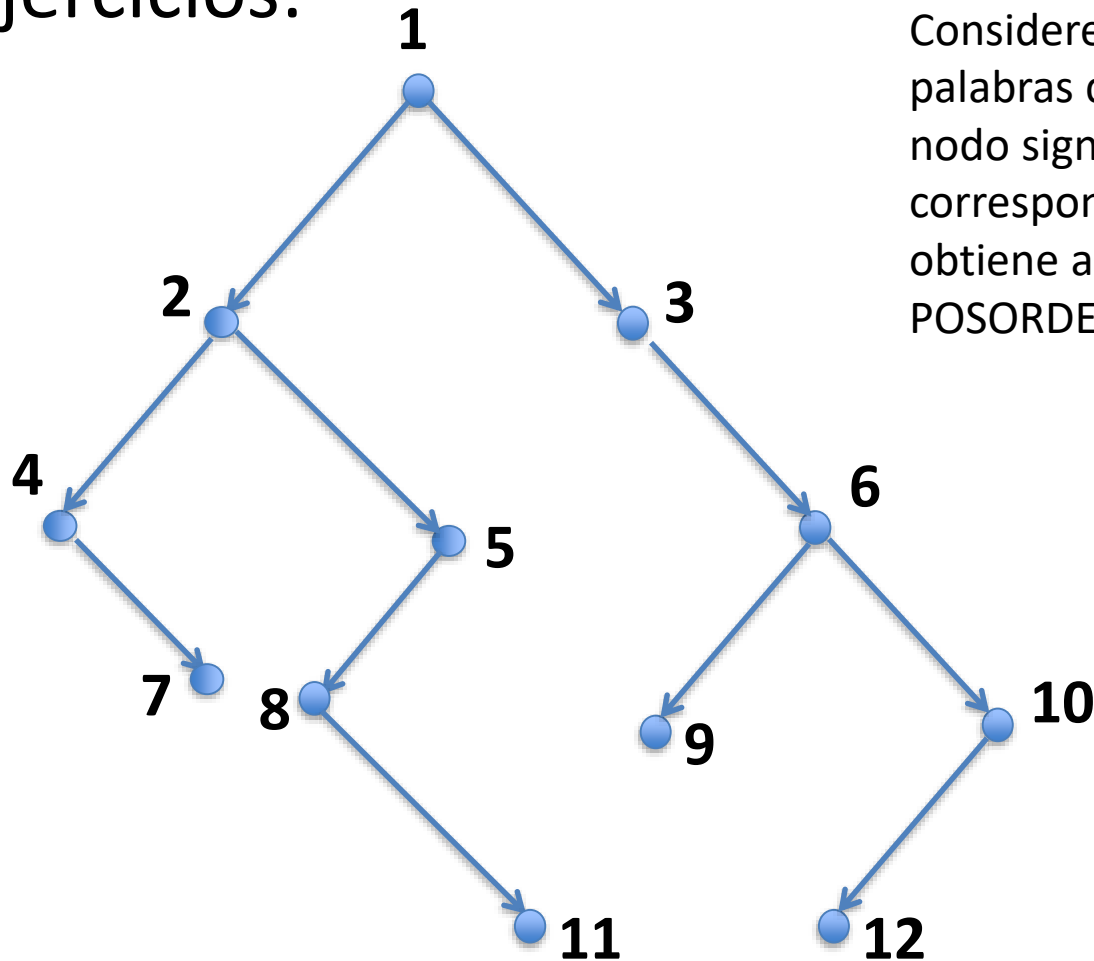
$p q \wedge \neg p \neg q \neg V \leftrightarrow$

POSORDEN

$\neg p \wedge q \leftrightarrow \neg p V \neg q$

ENTREORDEN (ambigua)

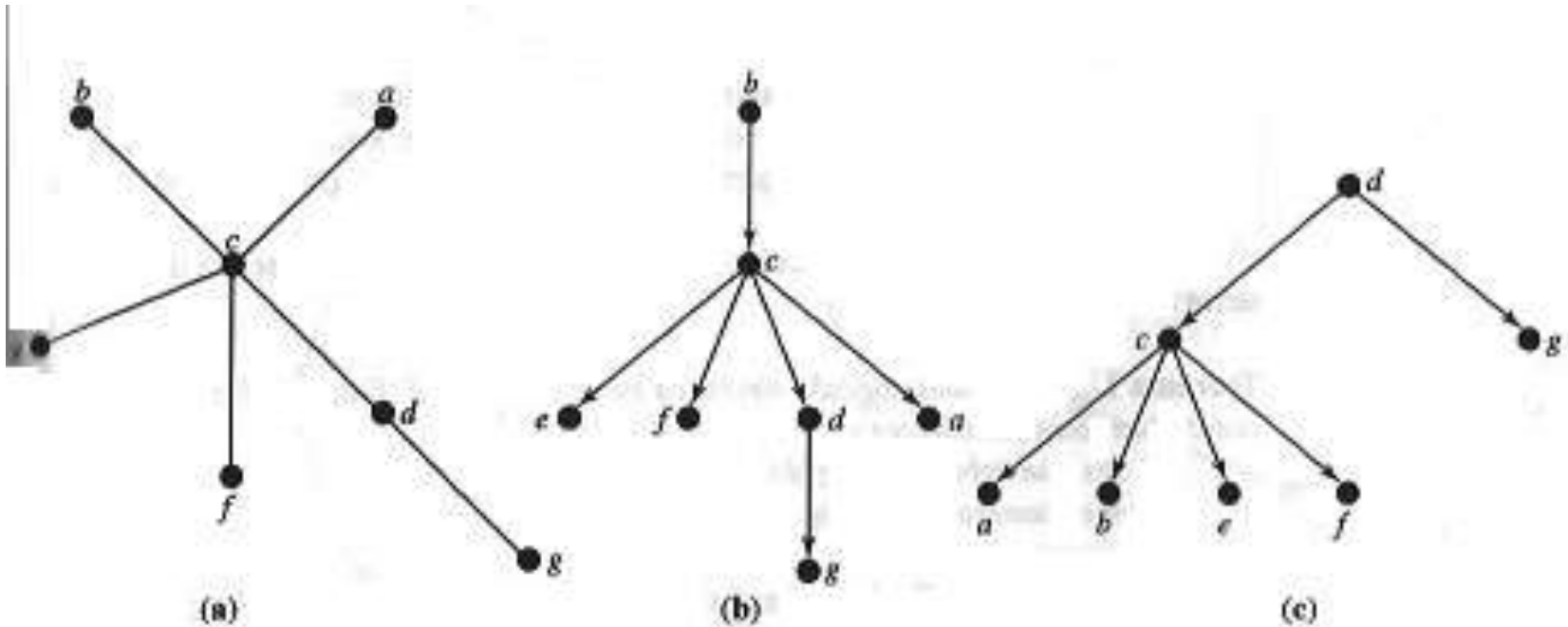
Ejercicios:

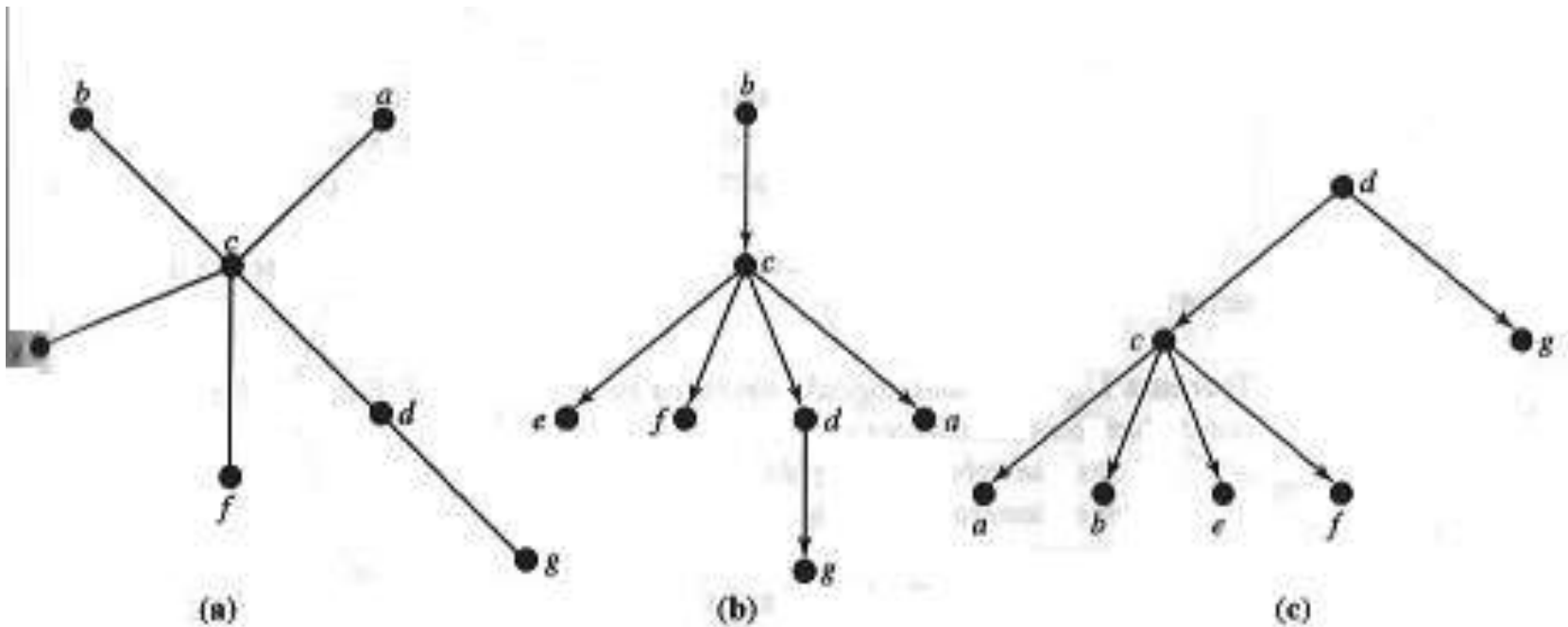


Considere el árbol de la figura y la lista de palabras dadas. Suponga que la visita a un nodo significa imprimir la palabra correspondiente. Imprima la frase que se obtiene al realizar una búsqueda en POSORDEN en el árbol

- | | |
|------------|-----------|
| 1. UNA | 7. YO |
| 2. PURPURA | 8. UNA |
| 3. VEA | 9. ESPERO |
| 4. NUNCA | 10. YO |
| 5. VACA | 11. VI |
| 6. NUNCA | 12. QUE |

ARBOLES NO DIRIGIDOS



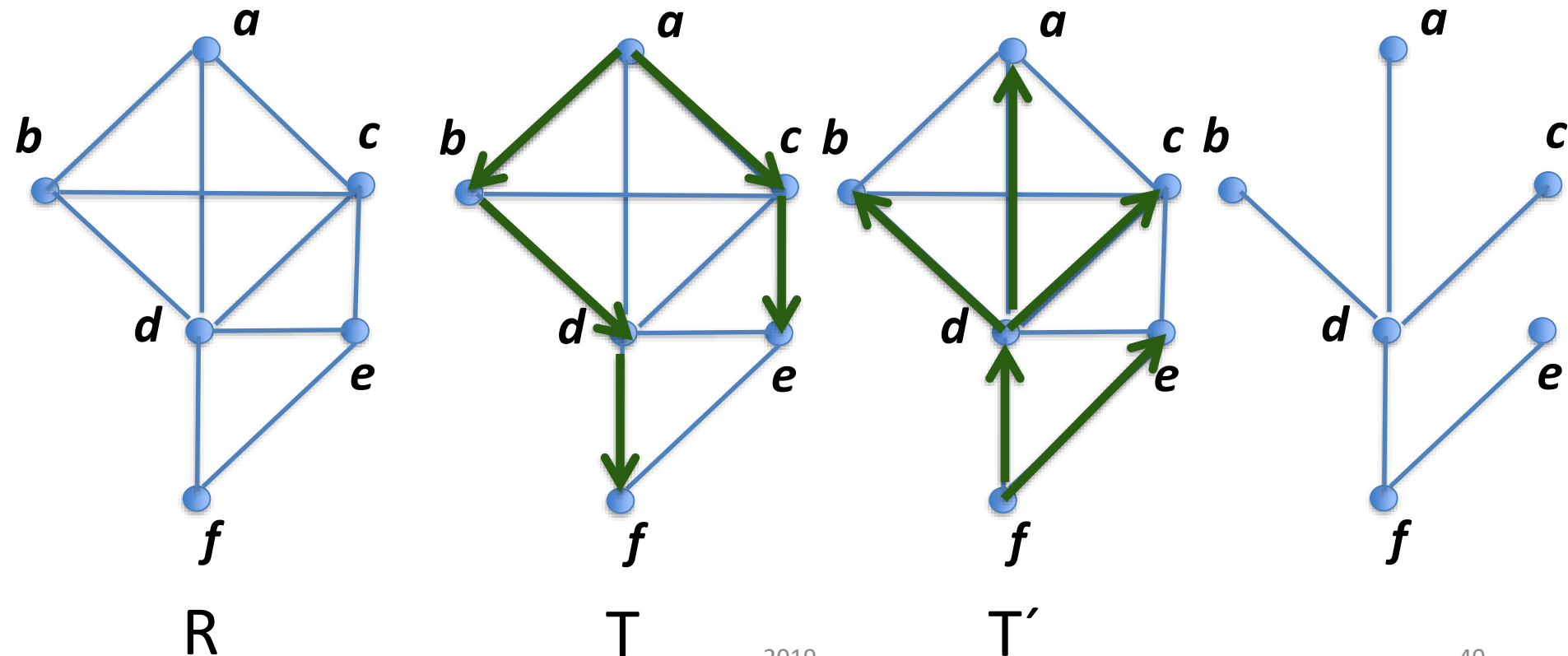


TEOREMA 1: Sea R una relación simétrica en un conjunto A .
Entonces las siguientes proposiciones son equivalentes:

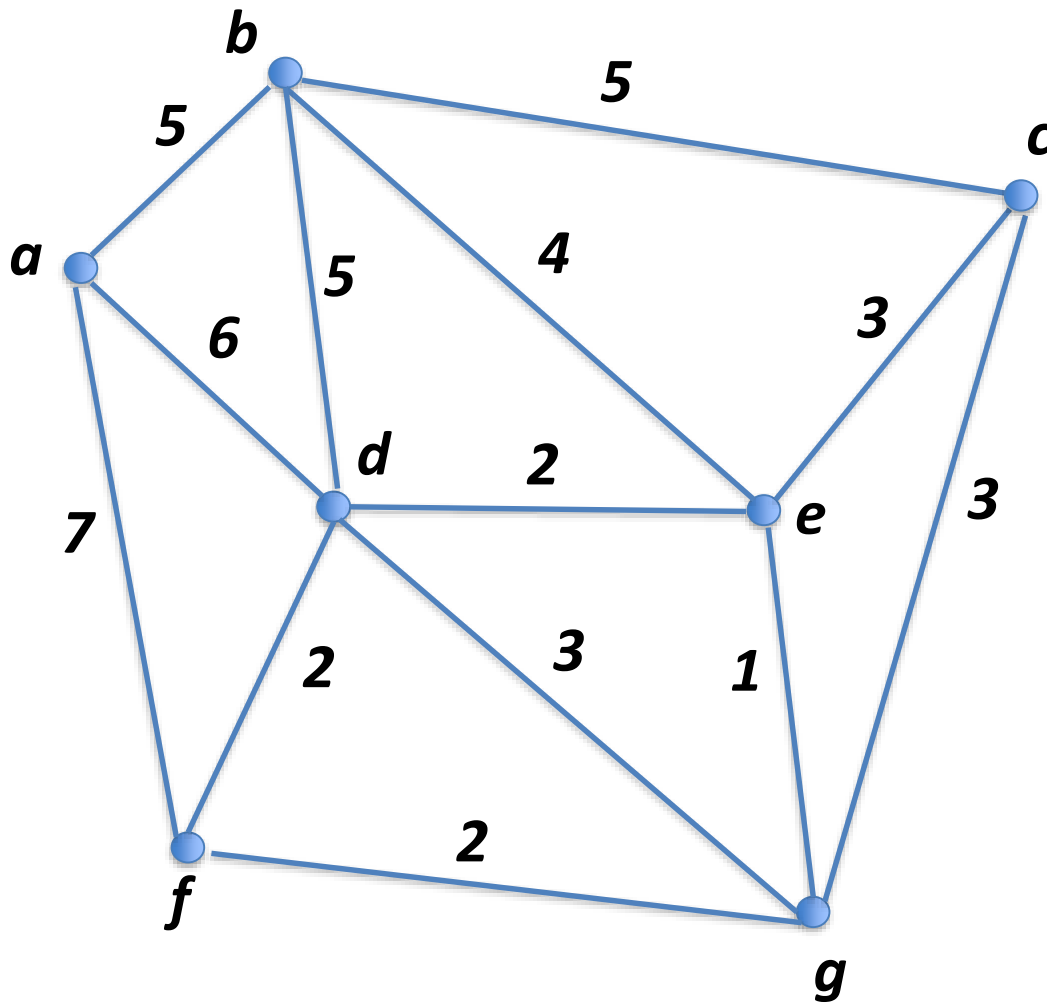
- (a) R es un árbol no dirigido
- (b) R es conexo y acíclico

ARBOLES DE EXPANSION DE RELACIONES CONEXAS

Si R es una relación simétrica conexa sobre un conjunto A , un árbol T en A es un árbol de expansión para R si T es un árbol con exactamente los mismos vértices que R y se puede obtener T eliminando algunas aristas de R



ARBOLES DE EXPANSION MINIMA



Sea el grafo $G=(V,E)$

- Ponderado
- No dirigido
- Conexo
- Sin lazos

ALGORITMO DE PRIM

PASO 1

Hacemos el contador $i = 1$

Colocamos un vértice arbitrario $v_1 \in V$ en el conjunto P .

Definimos $N = V - \{v_1\}$ y $T = \emptyset$

PASO 2

Para $1 \leq i \leq n - 1$, donde $|V| = n$, sean $P = \{v_1, v_2, \dots, v_i\}$
 $T = \{e_1, e_2, \dots, e_{i-1}\}$ y $N = V - P$.

Añadimos a T la arista mas corta (la de peso mínimo) de G que conecta un vértice x en P con un vértice $y (=v_{i+1})$ en N .

Colocamos y en P y lo eliminamos de N

PASO 3

Hacemos $i = i + 1$

Si $i = n$, el subgrafo de G dado por las aristas e_1, e_2, \dots, e_{n-1} es conexo, con n vértices, $n - 1$ aristas y es un árbol óptimo para G .

Si $i < n$, regresamos al paso 2.

ALGORITMO DE PRIM

Inicialización: $i = 1; P = \{a\}; N = \{b, c, d, e, f, g\}; T = \emptyset$

Primera iteración: $T = \{\{a, b\}\}; P = \{a, b\}; N = \{c, d, e, f, g\}; i = 2$

Segunda iteración: $T = \{\{a, b\}, \{b, e\}\}; P = \{a, b, e\}; N = \{c, d, f, g\}$
 $i = 3$

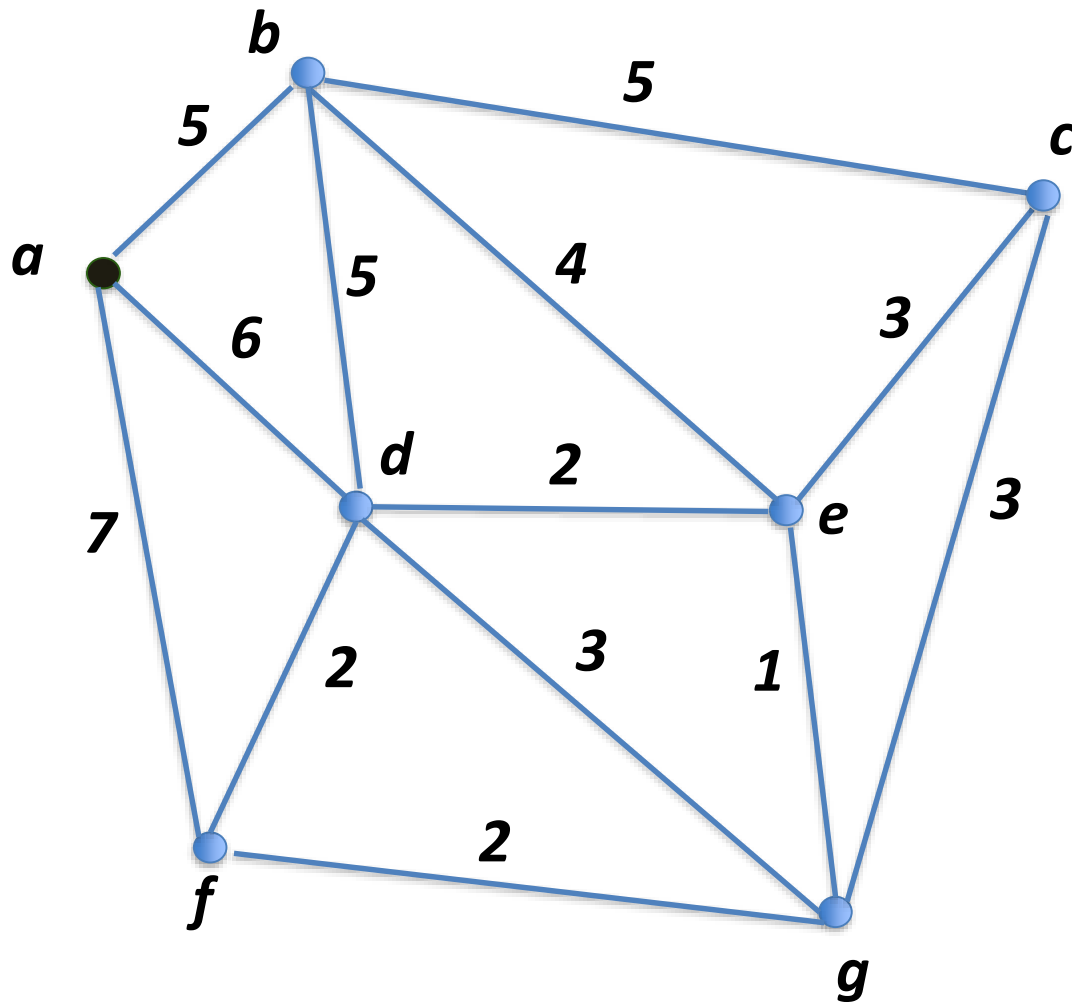
Tercera iteración: $T = \{\{a, b\}, \{b, e\}, \{e, g\}\}; P = \{a, b, e, g\}$
 $N = \{c, d, f\}; i = 4$

Cuarta iteración: $T = \{\{a, b\}, \{b, e\}, \{e, g\}, \{d, e\}\}$
 $P = \{a, b, e, g, d\}; N = \{c, f\}; i = 5$

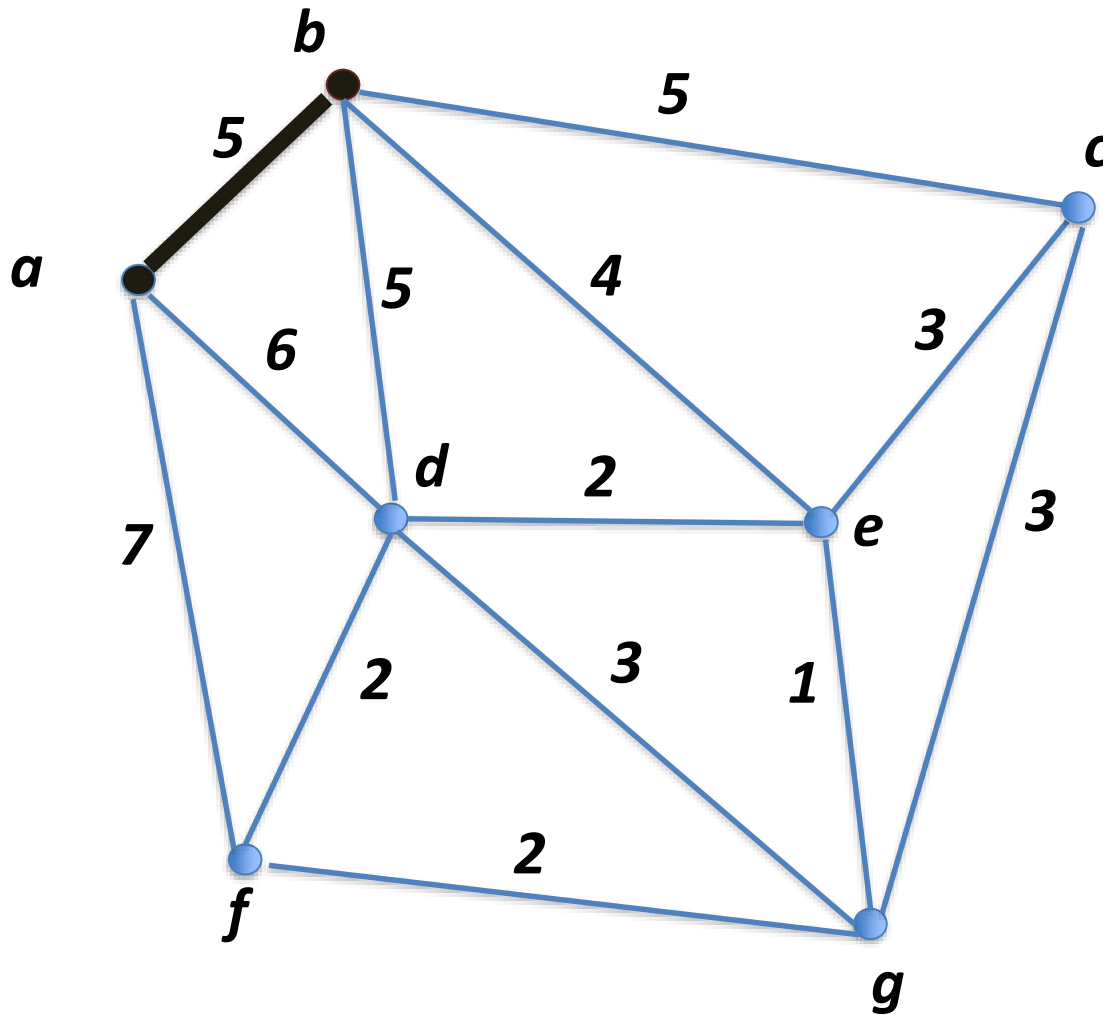
Quinta iteración: $T = \{\{a, b\}, \{b, e\}, \{e, g\}, \{d, e\}, \{f, g\}\}$
 $P = \{a, b, e, g, d, f\}; N = \{c\}; i = 6$

Sexta iteración: $T = \{\{a, b\}, \{b, e\}, \{e, g\}, \{d, e\}, \{f, g\}, \{c, g\}\}$
 $P = \{a, b, e, g, d, f, c\} = V; N = \emptyset; i = 7 = |V|$

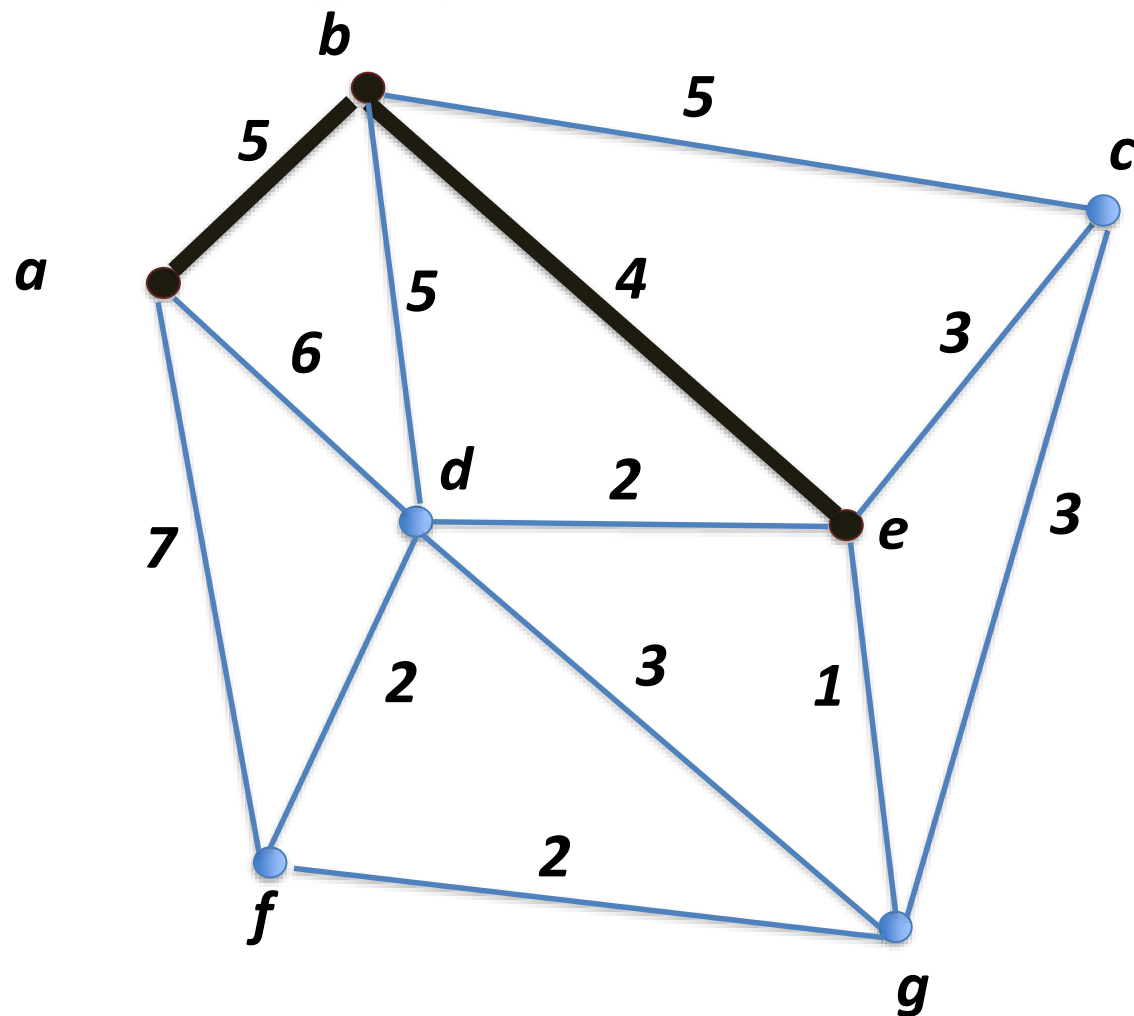
Inicialización: $i = 1$; $P = \{a\}$; $N = \{b, c, d, e, f, g\}$; $T = \emptyset$



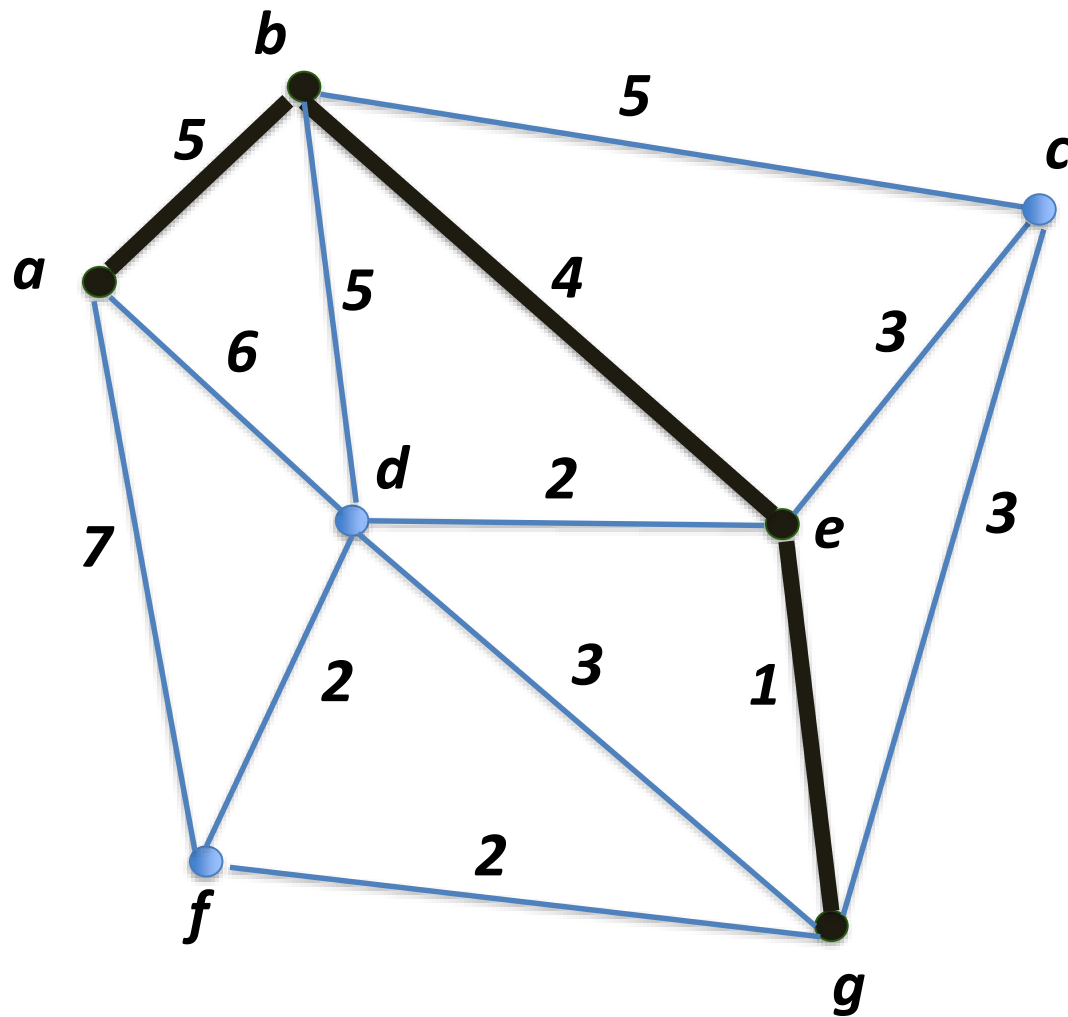
Primera iteración: $T = \{\{a, b\}\}; P = \{a, b\}; N = \{c, d, e, f, g\}; i = 2$



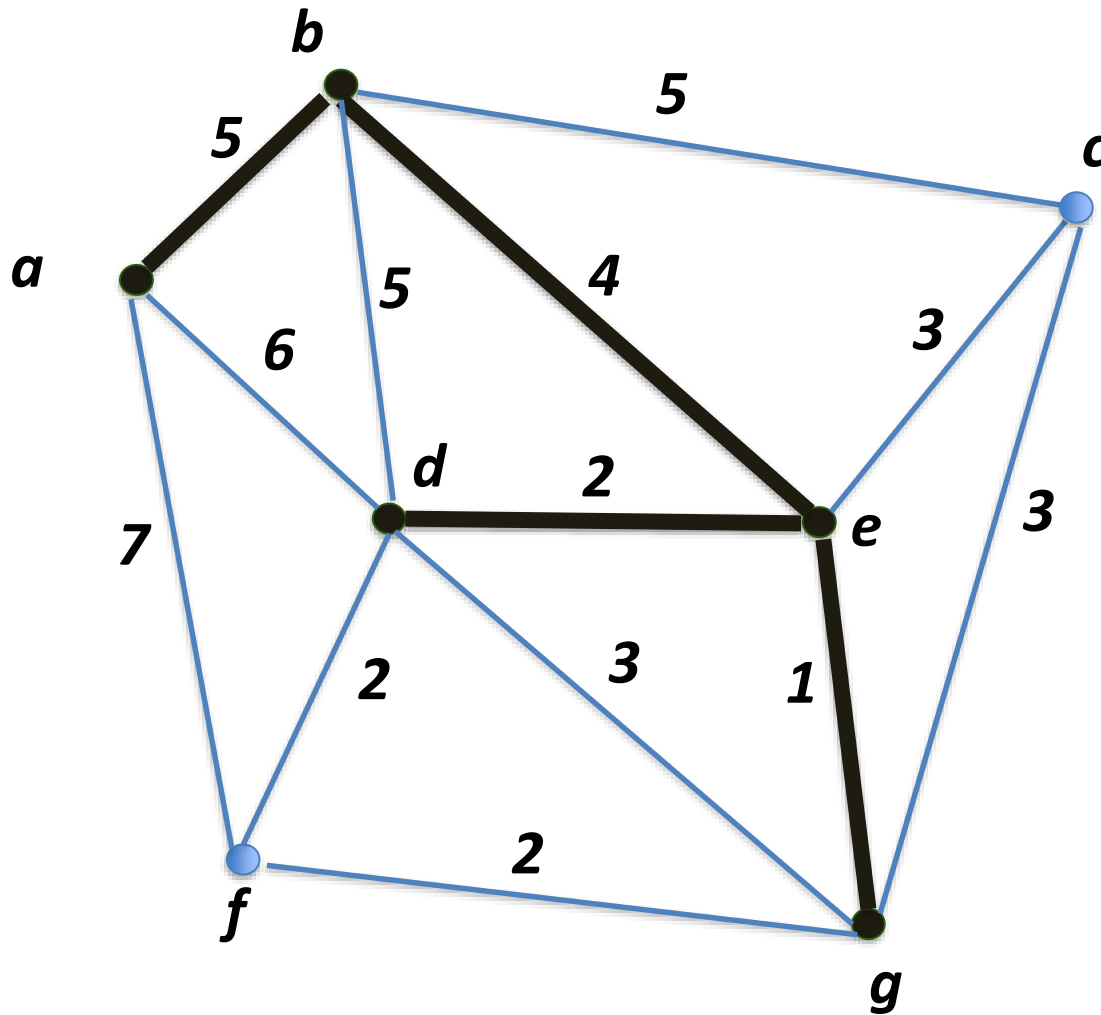
Segunda iteración: $T = \{\{a, b\}, \{b, e\}\}; P = \{a, b, e\}; N = \{c, d, f, g\}$
 $i = 3$



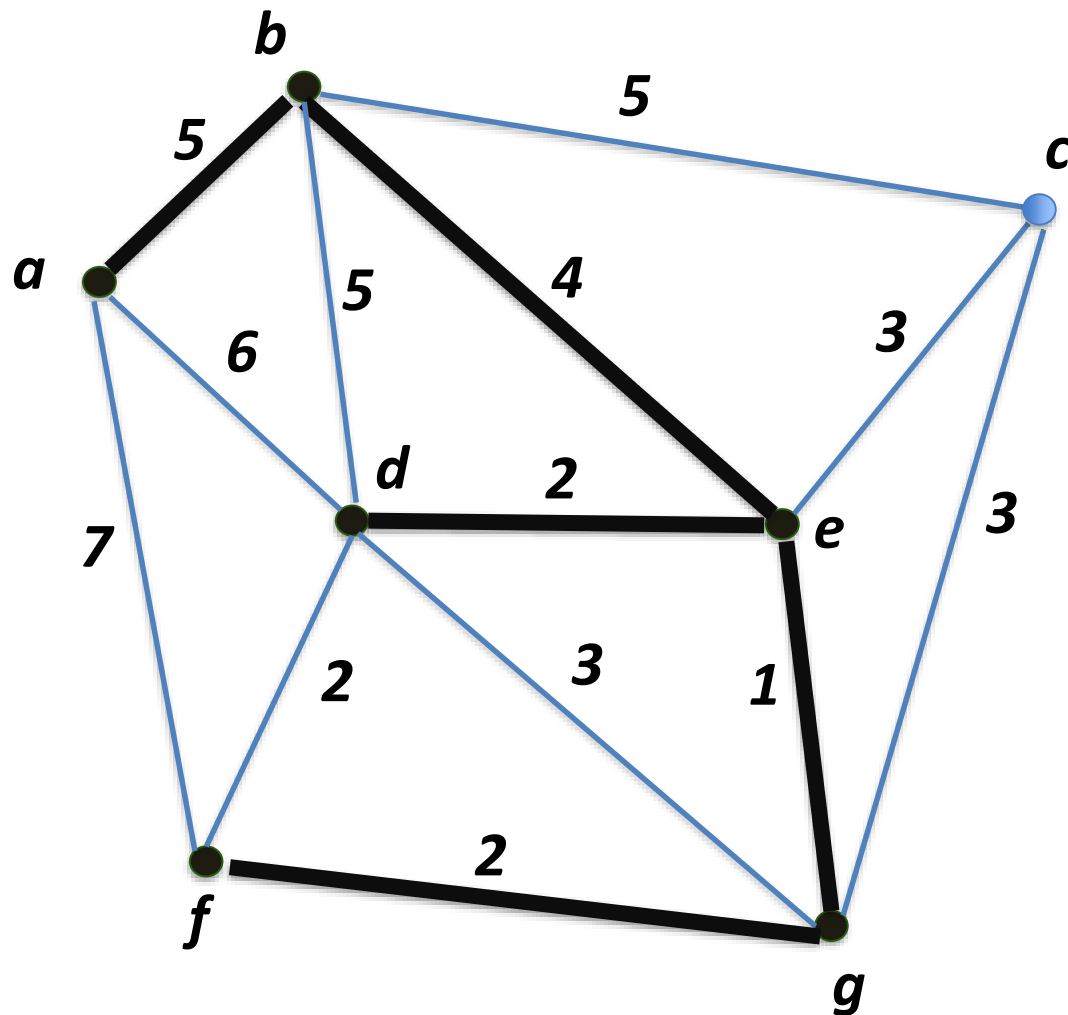
Tercera iteración: $T = \{\{a, b\}, \{b, e\}, \{e, g\}\}; P = \{a, b, e, g\}$
 $N = \{c, d, f\}; i = 4$



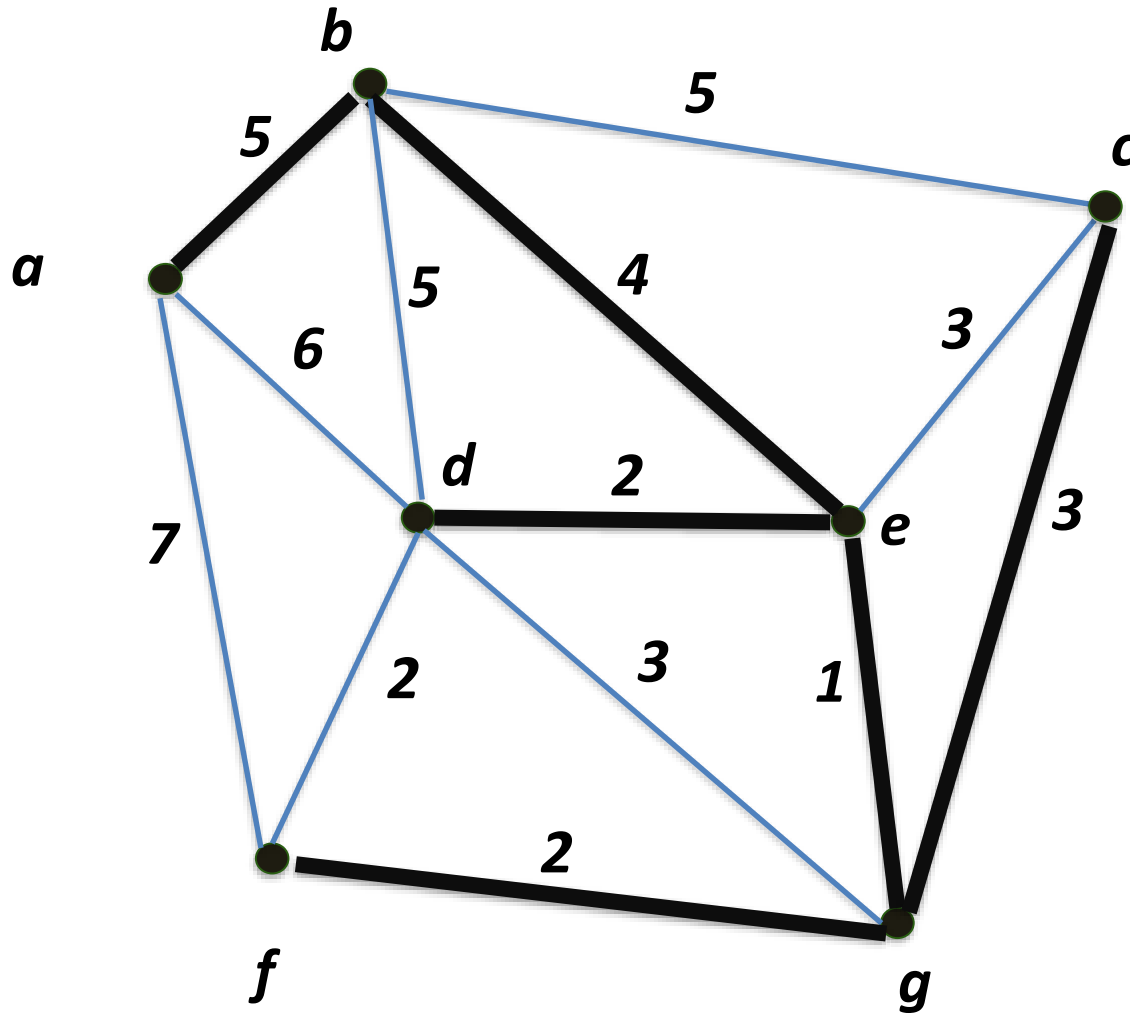
Cuarta iteración: $T = \{\{a, b\}, \{b, e\}, \{e, g\}, \{d, e\}\}$
 $P = \{a, b, e, g, d\}$; $N = \{c, f\}$; $i = 5$

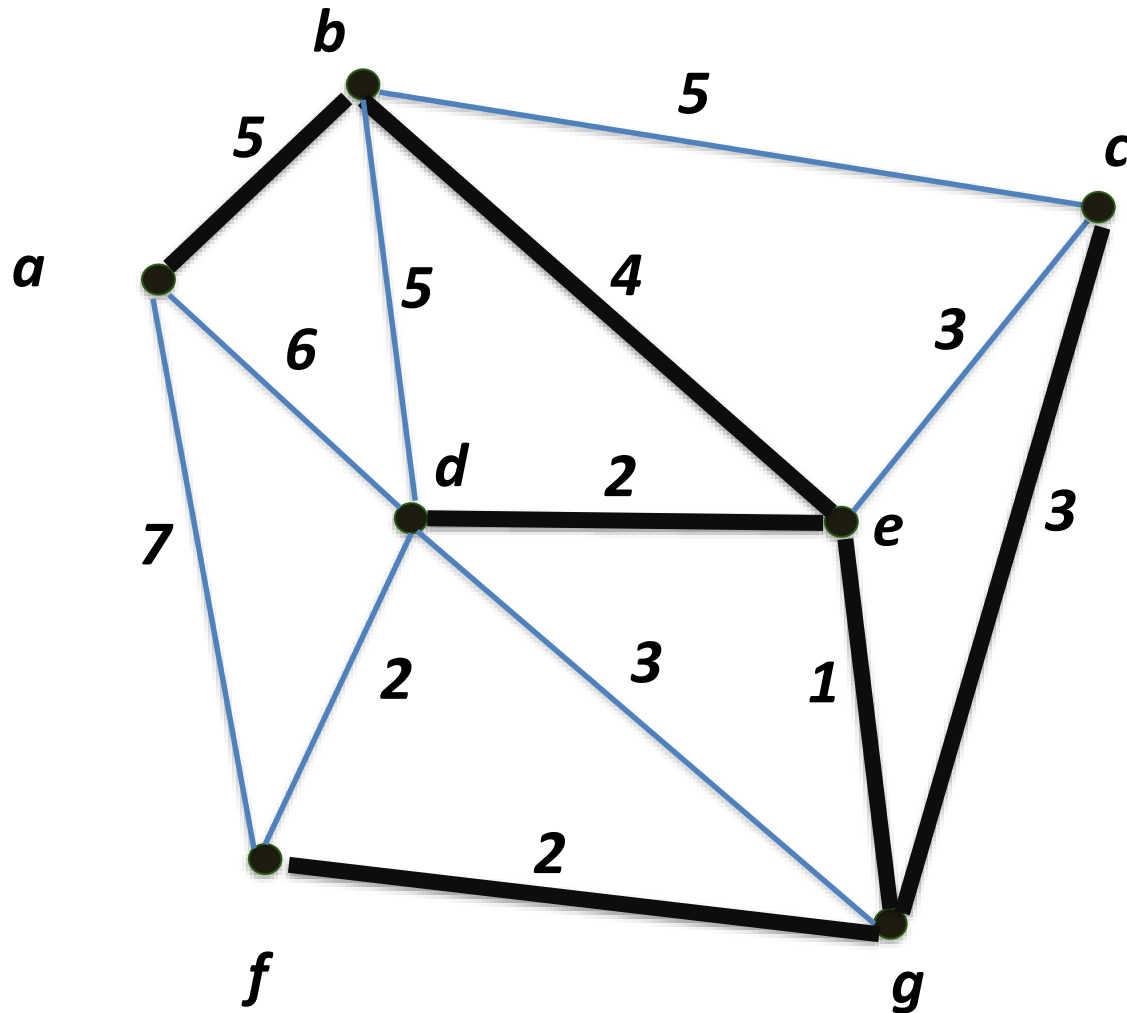


Quinta iteración: $T = \{\{a, b\}, \{b, e\}, \{e, g\}, \{d, e\}, \{f, g\}\}$
 $P = \{a, b, e, g, d, f\}$; $N = \{c\}$; $i = 6$



Sexta iteración: $T = \{\{a, b\}, \{b, e\}, \{e, g\}, \{d, e\}, \{f, g\}, \{c, g\}\}$
 $P = \{a, b, e, g, d, f, c\} = V$; $N = \emptyset$; $i = 7 = |V|$





T es un árbol de expansión mínimo de peso 17 para G

ALGORITMO DE KRUSKAL

PASO 1. Hacemos el contador $i = 1$

Seleccionamos una arista e_1 en G , tal que $p(e_1)$ séalo mas pequeño posible.

PASO 2. Para $1 \leq i \leq n - 2$, si hemos seleccionado las aristas e_1, e_2, \dots, e_i , entonces seleccionamos la arista e_{i+1} de las aristas restantes de G , de modo que:

- a) $p(e_{i+1})$ sea lo mas pequeño posible.
- b) El subgrafo de G determinado por las aristas $e_1, e_2, \dots, e_i, e_{i+1}$ (y los vértices incidentes) no contenga ciclos.

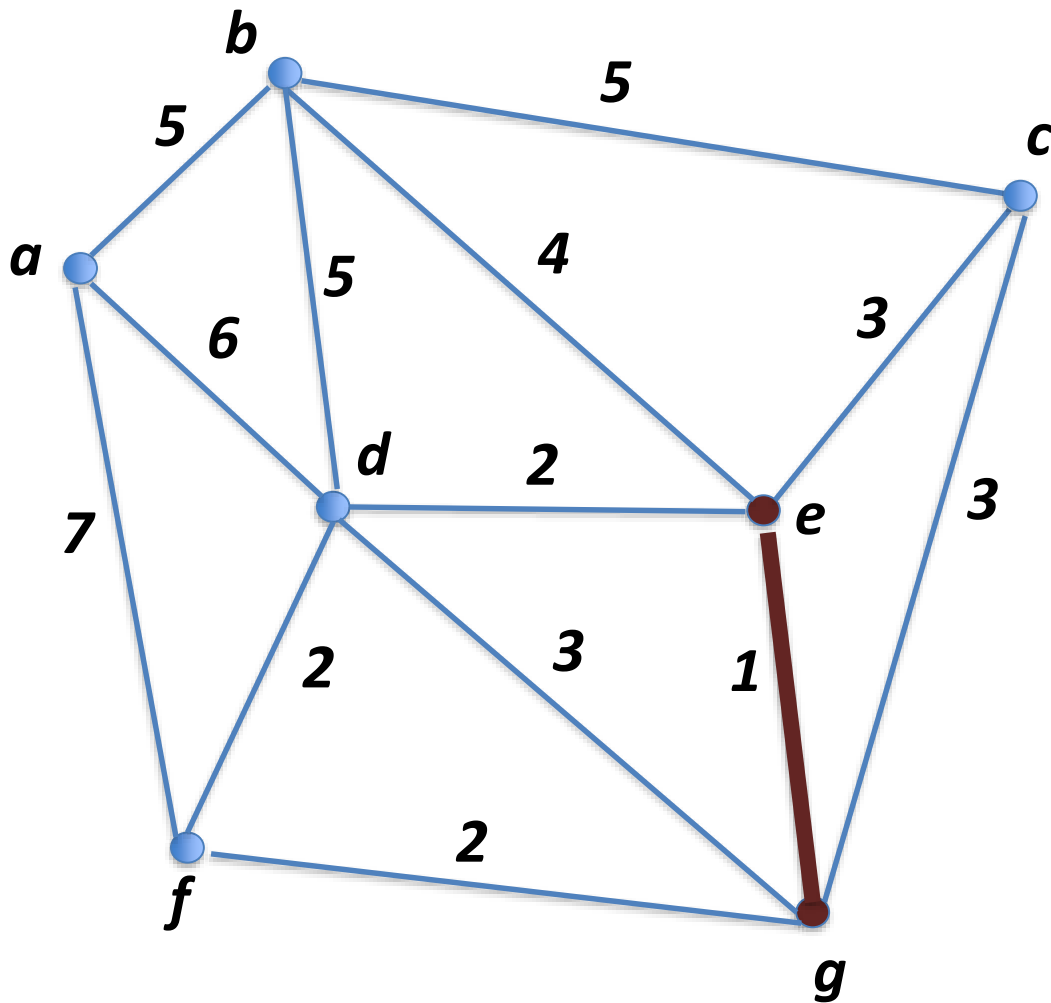
PASO 3. Hacemos $i = i + 1$.

Si $i = n - 1$, el subgrafo de G determinado por las aristas e_1, e_2, \dots, e_{n-1} es conexo, con n vértices y $n - 1$ aristas, yes un árbol de expansión mínimo para G .

Si $i < n - 1$, regresamos al paso 2.

INICIALIZACION: $i = 1$

Seleccionamos la arista de menor peso $\{e, g\}$

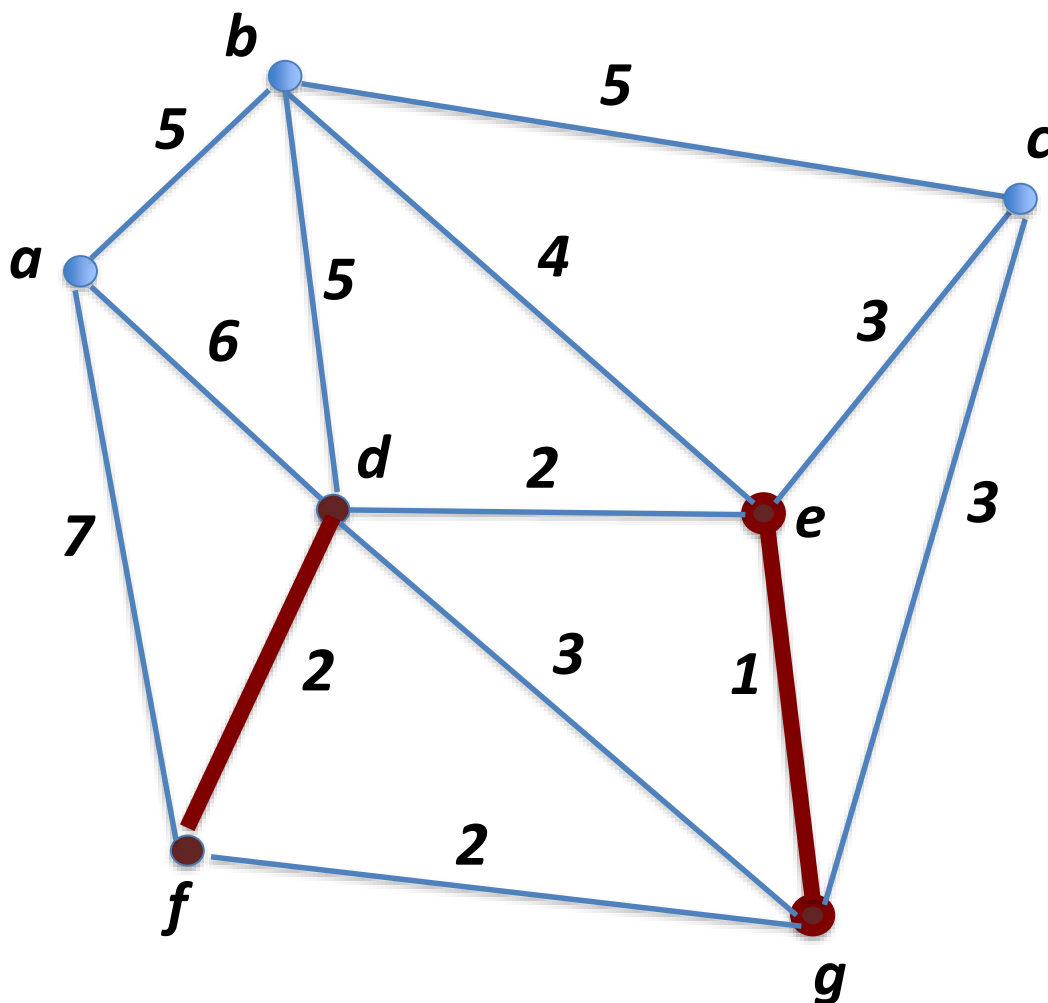


PRIMERA ITERACION:

Entre las aristas restantes tres tienen peso 2, seleccionamos $\{d, f\}$

$T = \{\{e, g\}, \{d, f\}\}$ Hacemos $i = i + 1 = 2$

$i = 2 < 6$ volvemos al paso 2.

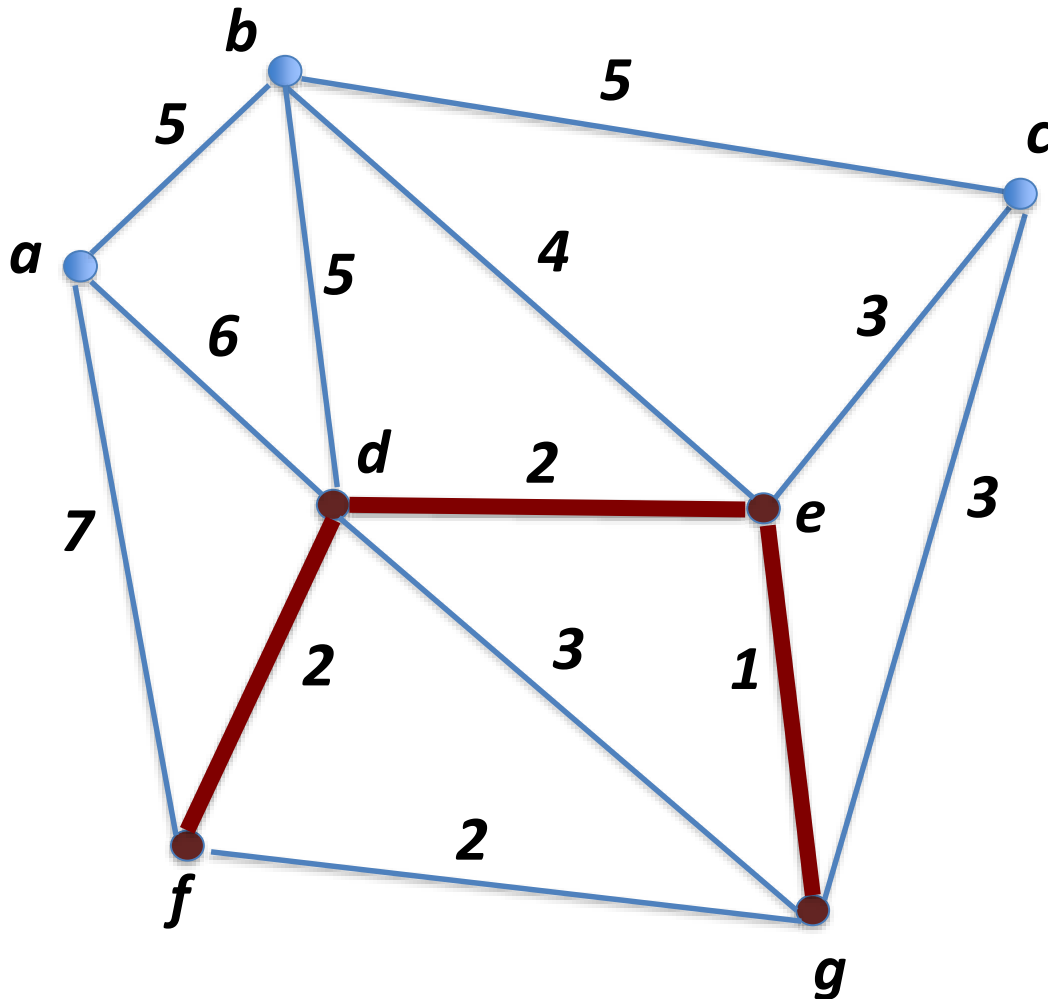


SEGUNDA ITERACION:

Dos de las aristas restantes tienen peso 2. Seleccionamos $\{d, e\}$

$T = \{\{e, g\}, \{d, f\}, \{d, e\}\}$ Hacemos $i = i + 1 = 3$

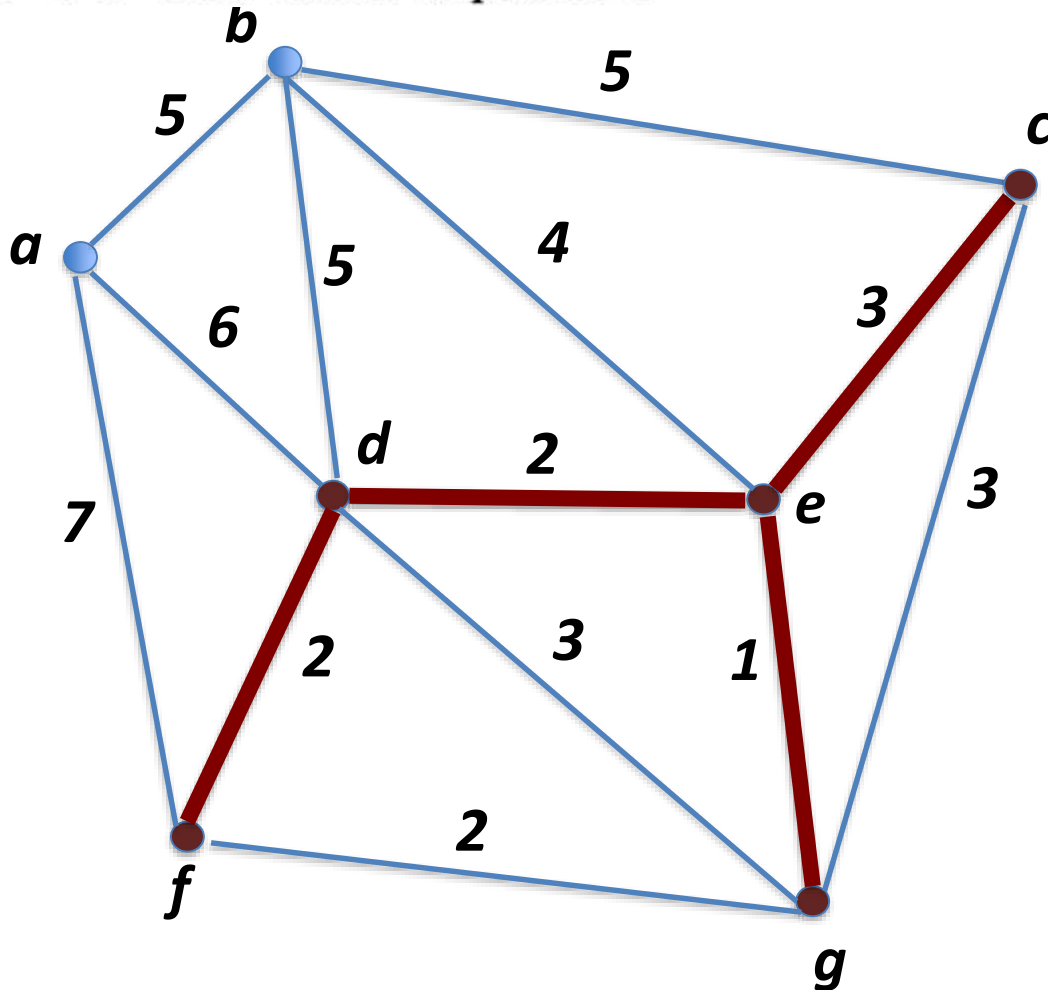
$i = 3 < 6$ volvemos al paso 2.



TERCERA ITERACION:

De las aristas restantes $\{f, g\}$ tiene el peso mínimo, pero produce un ciclo, en consecuencia es descartada y se elige $\{c, e\}$.

$T = \{\{e, g\}, \{d, f\}, \{d, e\}, \{c, e\}\}$ Hacemos $i = i + 1 = 4$
 $i = 4 < 6$ volvemos al paso 2.

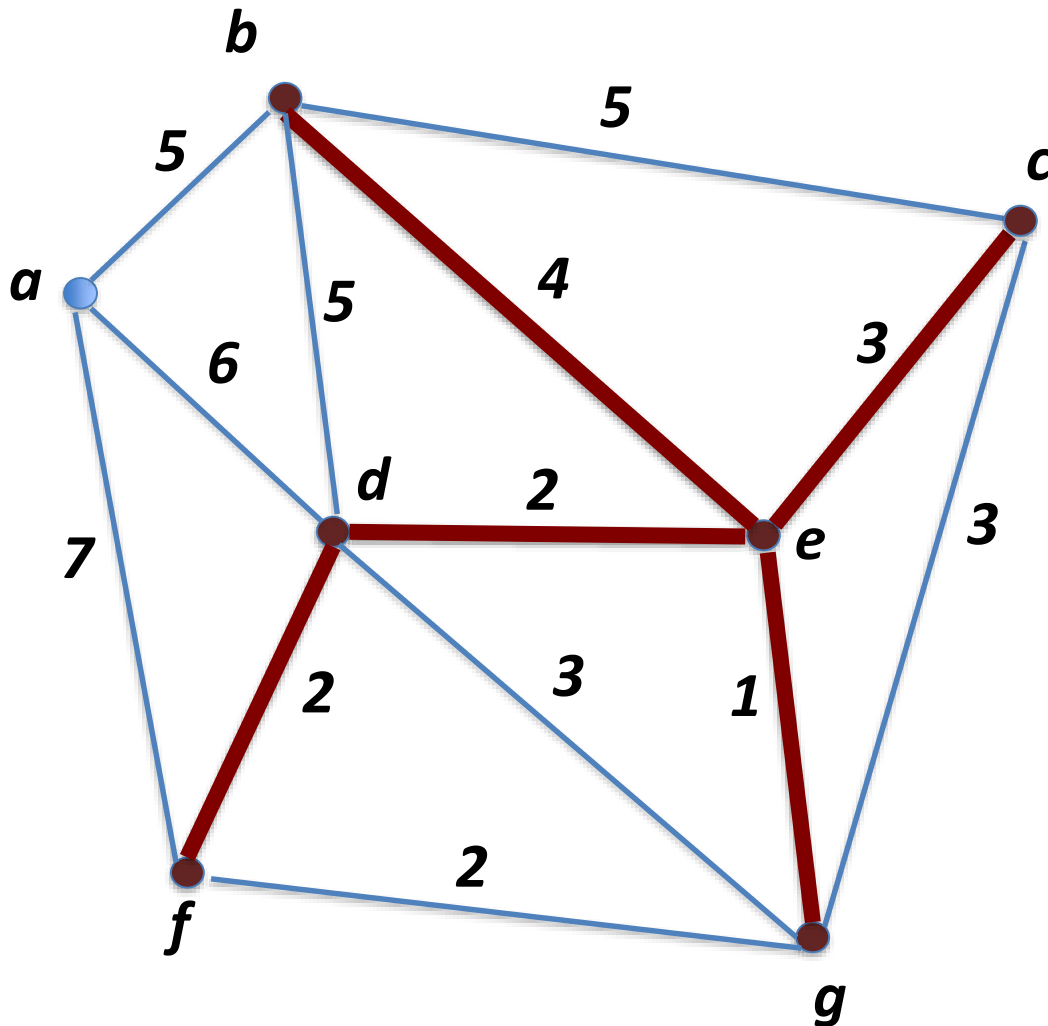


CUARTA ITERACION:

Se elige $\{b, e\}$.

$T = \{\{e, g\}, \{d, f\}, \{d, e\}, \{c, e\}, \{b, e\}\}$ Hacemos $i = i + 1 = 5$

$i = 5 < 6$ volvemos al paso 2.



QUINTA ITERACION:

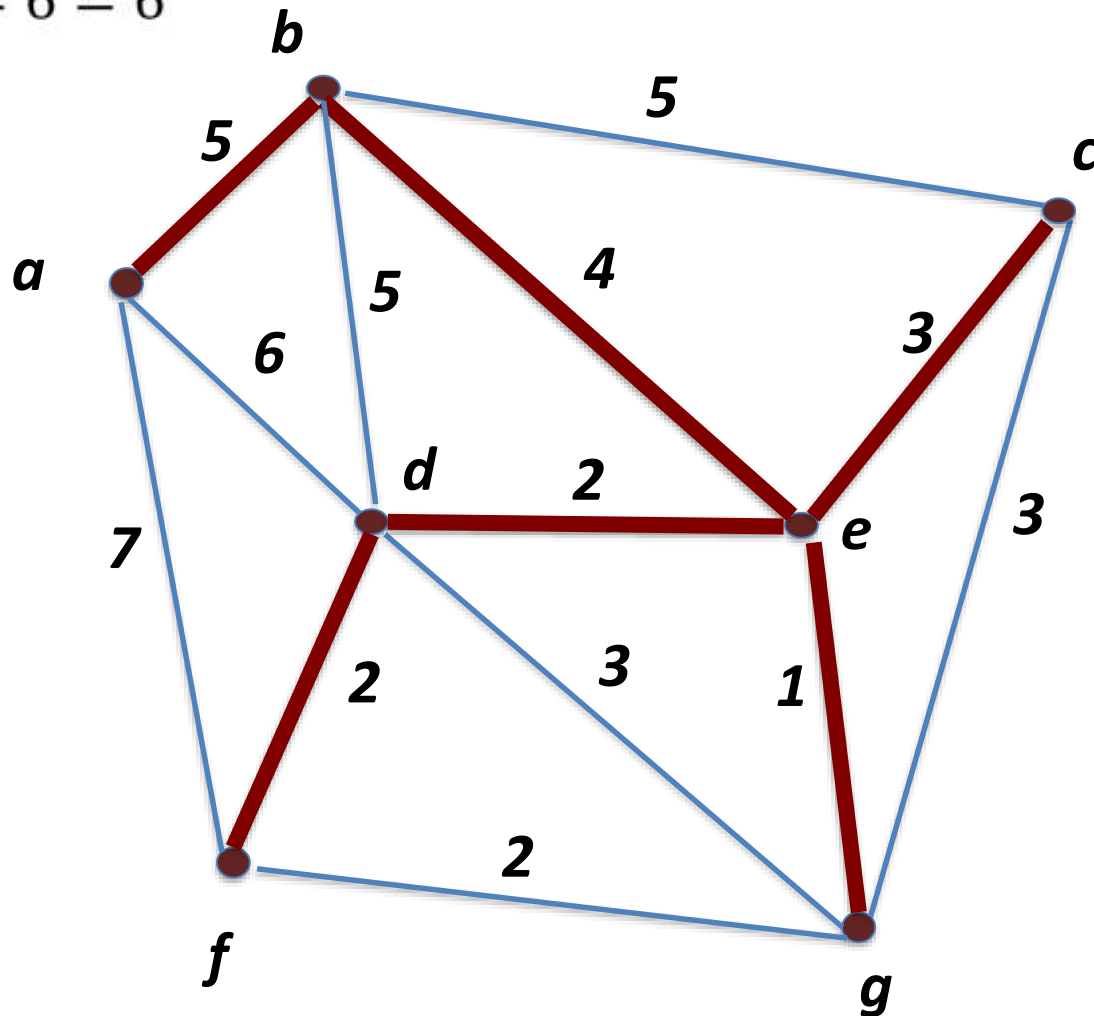
Se elige $\{a, b\}$.

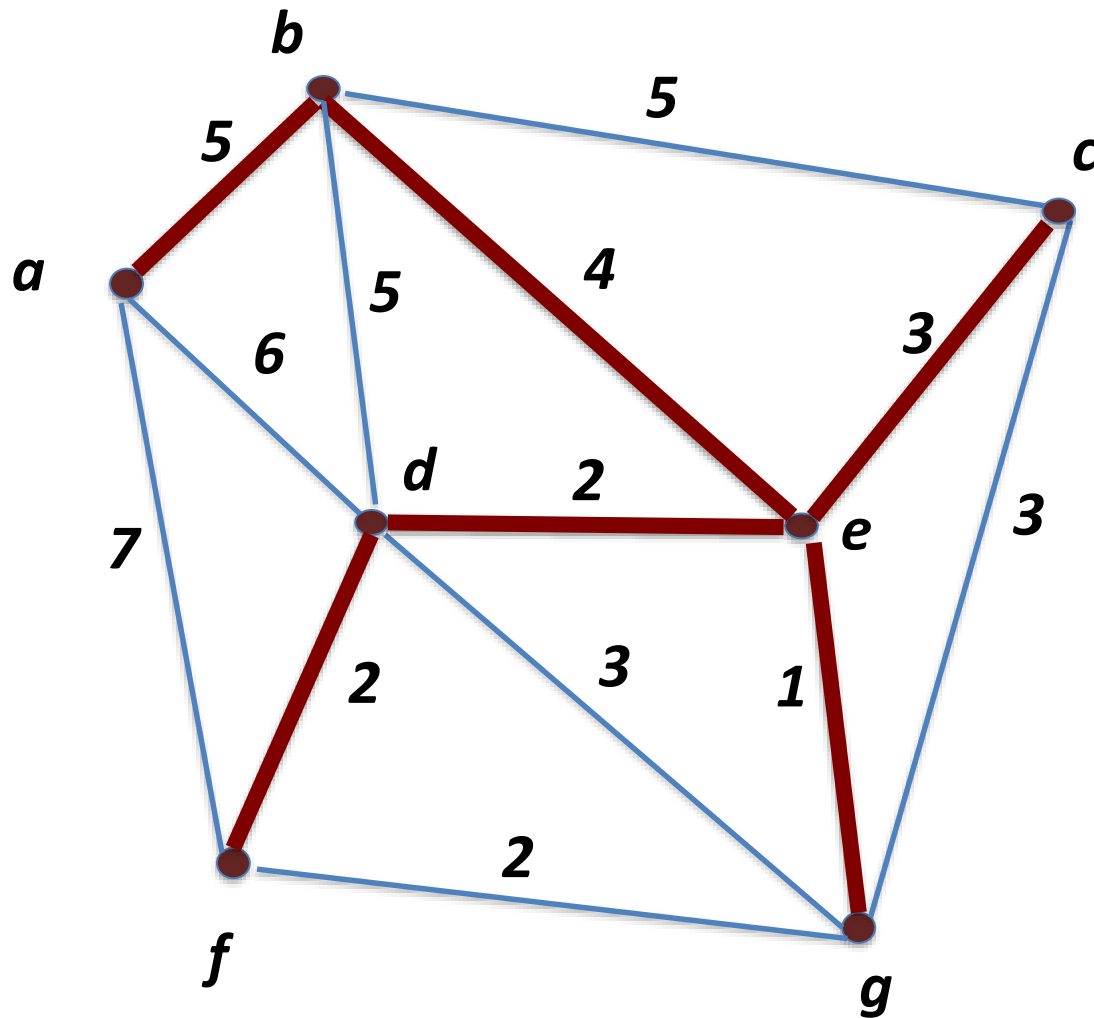
$$T = \{\{e, g\}, \{d, f\}, \{d, e\}, \{c, e\}, \{b, e\}, \{a, b\}\}$$

Hacemos $i = i + 1 = 6$

$$i = 6 = 6$$

FIN





T es un árbol de expansión mínimo de peso 17 para G

