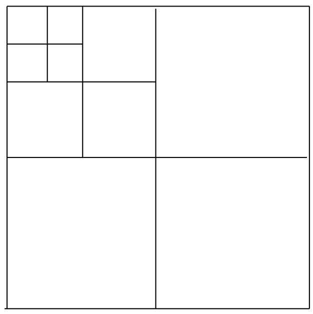
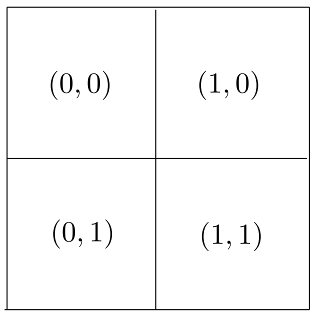
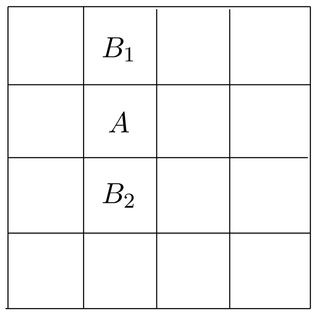
SOLUCIÓN.

Consideremos una imagen digital $\sum$ de orden $2^N \times 2^N$ constituida, por tanto, por 4^N pixeles. Estos pixeles se pueden obtener subdividiendo la imagen $\sum$ en cuatro cuadrantes, y estos cuadrantes a su vez en otros cuatro y así, sucesivamente hasta repetir este proceso recursivo N veces.



Esto procedimiento nos permite una codificación de cada pixel como una secuencia $(x_1, y_1)\dots(x_N, y_N)$ donde cada $(x_i, y_i) \in \{(0,0),(1,0),(0,1),(1,1)\}$ y donde $(0,0),(1,0),(0,1)\,y\,(1,1)$ representan los cuadrantes de una subdivisión.





$N = 2$,
B_1 es vecino interior de A según V,
B_2 es vecino exterior de A según V.

Dado un pixel A distinguiremos entre vecinos interiores y exteriores. Diremos que el pixel B es un vecino interior de A si se encuentran en un mismo cuadrante de $(N-1)$-subdivisión. Y en caso contrario, se dice que B es un vecino exterior de A. A partir de la codificación explicada anteriormente y dado un pixel $A = (x_1, y_1)\cdots(x_N, y_N)$, se pide:

1. Codificar los 4 pixeles que están situados en los extremos de la imagen $\sum$ y los cuatro del centro.

   Solución. Extremos de la imagen:

Sup. Izq:	$(0,0)^N$	Sup. Dcha:	$(1,0)^N$
Inf. Izq:	$(0,1)^N$	Inf. Dcha:	$(1,1)^N$

   Centro de la imagen:

Sup. Izq:	$(0,0)\,(1,1)^{N-1}$	Sup. Dcha:	$(1,0)\,(0,1)^{N-1}$
Inf. Izq:	$(0,1)\,(1,0)^{N-1}$	Inf. Dcha:	$(1,1)\,(0,0)^{N-1}$

2. Hallar el pixel vecino interior A según la dirección horizontal H y en la dirección vertical V.

   Solución. Dado $A = (x_1, y_1) \cdots (x_N, y_N)$,

   Vecino interno de A según H: $(x_1, y_1) \cdots (x_{N-1}, y_{N-1})\,(\overline{x_N}, y_N)$.

   Vecino interno de A según V: $(x_1, y_1) \cdots (x_{N-1}, y_{N-1})\,(x_N, \overline{y_N})$.

   Donde $\overline{1} = 0$, y $\overline{0} = 1$.

3. Dar una condición, en términos de la codificación de A, para que exista un pixel vecino exterior de A según la dirección H y V.

   Solución. Los píxeles que no tienen vecino exterior en la dirección horizontal son los que se encuentran en los bordes superior e inferior de la imagen.

   Los píxeles que no tienen vecino exterior en la dirección horizontal son los que se encuentran en los bordes izquierdo y derecho de la imagen.

   Dado $A = (x_1, y_1) \cdots (x_N, y_N)$ se tiene:

   - A tiene vecino exterior en la dirección horizontal si, y sólo si, existen $0 \leq i < j \leq N$ tal que $x_i \neq x_j$.
   - A tiene vecino exterior en la dirección vertical si, y sólo si, existen $0 \leq i < j \leq N$ tal que $y_i \neq y_j$.

4. En caso de que exista, hallar el único pixel vecino exterior de A según la dirección H y V.

   Solución. Dado $A = (x_1, y_1) \cdots (x_N, y_N)$

   Sea $k > 0$ tal que $x_n = x_{n-1} = \cdots = x_{n-k+1} \neq x_{n-k}$. Entonces el vecino exterior de A según H es:

   $$(x_1, y_1) \cdots (x_{n-k-1}, y_{n-k-1})\,(\overline{x_{n-k}}, y_{n-k}) \cdots (\overline{x_N}, y_N).$$

   Sea $l > 0$ tal que $y_n = y_{n-1} = \cdots = y_{n-l+1} \neq y_{n-l}$. Entonces el vecino exterior de A según V es:

   $$(x_1, y_1) \cdots (x_{n-l-1}, y_{n-l-1})\,(x_{n-l}, \overline{y_{n-l}}) \cdots (x_N, \overline{y_N})$$

SOLUCIÓN.

Definamos los $4'$-vecinos del pixel $P = (x, y)$ como los píxeles:

$$(x-1, y+1) \qquad (x+1, y+1)$$

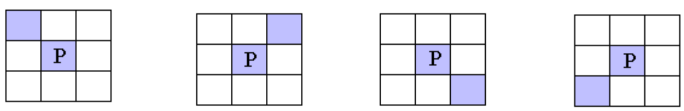
$$(x-1, y-1) \qquad (x+1, y-1)$$

Consideremos la adyacencia $(4', 4)$ (es decir, la $4'$-adyacencia para los píxeles negros y la 4-adyacencia para los blancos).

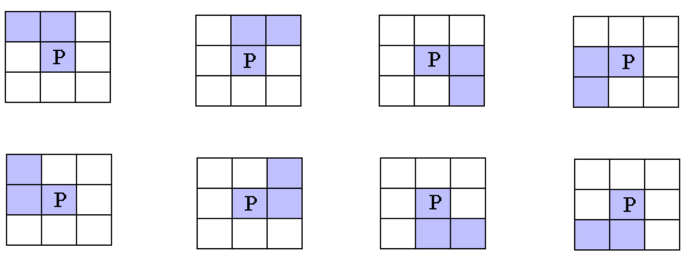
1. Decimos que un pixel P de color negro es un pixel *simple* si reemplazándolo por uno blanco no altera el número de componentes conexas blancas y negras dentro de su 8-vecindad. Con la topología $(4', 4)$, dar todas las configuraciones posibles de puntos simples que tengan a lo sumo dos píxeles negros en su 8-vecindad.

   Solución: Las configuraciones posibles son:

   - Las configuraciones posibles de puntos simples con un píxel negro en su 8–vecindad son:



   - Las configuraciones posibles de puntos simples con dos píxeles negros en su 8–vecindad son:



2. Explicar cómo sería el código de cadenas de una curva cerrada simple.

   Solución: Tomamos, por convenio, los siguientes valores para formar el código de cadenas:

3	2	1
4	P	0
5	6	7

   Entonces, el código de cadenas de una curva cerrada simple formada por píxeles negros considerando la $4'$–adyacencia sólo podría estar formada por los valores 1, 3, 5 y 7. Además el número de veces que aparece el valor 1 (resp. 3) es igual al número de veces que aparece el valor 5 (resp. 7). Esto es debido a que después de movernos en una dirección, tendremos que movernos en la dirección contraria el mismo número de veces para poder volver al punto inicial de la curva.

3. Dar un ejemplo de un objeto que en continuo sea conexo y que al digitalizarlo con esta topología cada pixel forme una componente conexa.

   Solución: Cualquier recta horizontal o vertical compuesta por píxeles negros, de un píxel de ancho, cumple esa condición.

4. Dar una condición para afirmar que un objeto que en continuo sea conexo al digitalizarlo con esta topología sea no conexo.

   Solución: Una condición suficiente para que el objeto digitalizado sea no conexo es que contenga dos píxeles $P(x_1, y_1)$ y $Q(x_2, y_2)$ tal que $x_1 + y_1 \neq x_2 + y_2 \mod 2$. Esto es debido a que si dos píxeles están conectados, entonces existiría un camino entre ellos considerando la $4'$–adyacencia. Observemos que un vecino (x, y) de un píxel (i, j) (con la $4'$–adyacencia) cumple que $x = i \pm 1$ e $y = j \pm 1$, por tanto, $x + y = i + j \mod 2$. Luego la suma de las coordenadas de todos los píxeles del camino deben tener la misma paridad.

5. Demostrar que no hay paradoja de la curva cerrada simple, es decir, una curva de este tipo siempre divide el plano digital en dos componentes conexas (una exterior y otra interior).

   Solución: Supongamos que hay paradoja, es decir, una curva cerrada simple no divide al plano en dos componentes conexa. Entonces considerando una orientación en la curva, algún píxel blanco situado a la derecha de la curva debe ser adyacente a algún píxel blanco situado a la izquierda. Pero esto nunca puede ocurrir con la $(4', 4)$–adyacencia, ya que los píxeles negros de la curva siempre van a estar situados en los puntos de 4–adyacencia de los píxeles blancos del borde de tal forma que evitan la vecindad de cualquier par de píxeles blancos situados a un lado y otro de la curva.