

## Ejercicios – Recursión

### 1.- LAB1 – 2014 – 2. Reconociendo h-sequences

Una *h-sequence* está formada por un único 0 o por un 1 seguido de dos *h-sequences*. Una gramática de *h-sequences* podría ser la siguiente:

$$\langle h \rangle = 0|1\langle h \rangle\langle h \rangle$$

SON <i>h-sequences</i>	NO SON <i>h-sequences</i>
0	1
100	1001
10100	11010
110010100	1001110
110100100	1010101

Implemente una función **recursiva** en ANSI C que reciba como parámetro un arreglo de  $N$  elementos ( $N > 0$ ) y determine si es una *h-sequence* o no. Puede asumir que el arreglo solo contiene 0s y 1s.

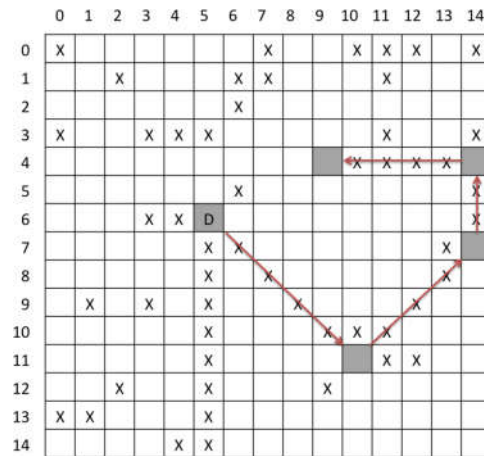
### 2.- LAB1 - 2014-2. D-Path

Se dispone de un tablero de caracteres de 15x15, en cuyos casilleros se pueden encontrar una 'X', un espacio en blanco ( ' ') o una 'D' tal como se muestra en la siguiente figura:

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
0	X							X			X	X	X		X
1			X				X	X				X			
2							X								
3	X			X	X	X						X			X
4											X	X	X	X	
5							X								X
6				X	X	D									X
7						X	X							X	
8						X		X						X	
9		X		X		X			X				X		
10						X				X	X	X			
11						X						X	X		
12			X			X				X					
13	X	X				X									
14					X	X									

Como se puede apreciar, la letra 'D' se encuentra rodeada de algunas letras 'X' y estas también pueden estar dispersas en otros casilleros del tablero.

Se desea elaborar un programa que permita determinar una ruta de desplazamiento de la letra 'D' (única en el tablero). La siguiente figura muestra la ruta que finalmente seguirá la letra 'D':

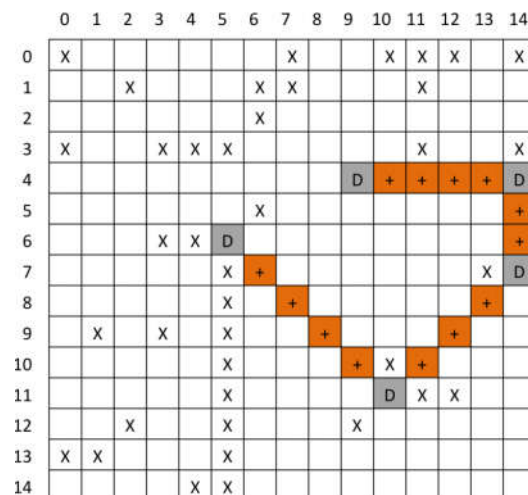


Para determinar esta ruta, el programa deberá seguir los siguientes pasos:

a) Determinar la dirección en la cual exista la **mayor** cantidad de 'X' adyacentes a una 'D' considerando, además, que al extremo de las 'X' debe haber un casillero en blanco. Para este paso **deberá usar, por lo menos, una función recursiva**. En el tablero que se da como ejemplo, la dirección será Diagonal Inferior Derecha. Observe que en la dirección Abajo hay más 'X' pero no se puede considerar esta dirección ya que no hay un casillero vacío al extremo.

b) Colocar la letra 'D' en el casillero en blanco y reemplazar todas las 'X' que se encuentran en la dirección seleccionada por '+' . En el ejemplo anterior, se coloca la letra 'D' en el casillero (11, 10), reemplazando por '+' las 'X' que se encuentran en los casilleros (7, 6), (8, 7), (9, 8) y (10, 9).

c) Repetir los pasos anteriores hasta que no haya posibilidad de mover la letra 'D'. Al final, el tablero quedaría como se muestra en la siguiente figura:



Implemente un programa en ANSI C que resuelva el problema descrito. Para ello, el tablero se leerá desde el archivo de texto *dpath.in* con redireccionamiento de la entrada estándar. En la primera línea de dicho archivo se encuentran las coordenadas de la letra 'D' en el tablero y luego, en las siguientes líneas, aparecen los caracteres que corresponden al tablero, siguiendo la distribución correspondiente de filas y columnas. Luego de encontrar el camino siguiendo el algoritmo planteado, deberá imprimir el tablero final en el archivo *dpath.out* con redireccionamiento de la salida estándar.