Q15. Child Nodes in Binary Tree (20 marks):

A **Binary Tree (BT)** is a non-linear data structure where nodes, each containing a distinct positive integer value, are distributed over x levels, as illustrated in the Figure Q15. For the examples shown, there are two binary trees, BT1 and BT2, where BT1 has 3 levels of nodes and BT2 has 4 levels of nodes.

At Level 1 of a BT, there is one and only one node, and it can be connected to at most two child nodes at Level 2. Recursively, for $2 \leq i \leq x$, a Level-i node must be connected to only one parent node at Level $i - 1$, and at most two child nodes at Level $i + 1$.

Binary Tree Structure	Array Representation

Figure Q15. Two Binary Trees (BT1 and BT2) and their array representations

A BT can be represented using an array in a programme as shown in Figure Q15. In the array representation, we need to create an array with size $2^x - 1$, assuming every node at Level $x - 1$ and above has exactly 2 children, left child node and right child node. Since some of the nodes in the original BT can have less than 2 children, we need to use value -1 to represent the empty child in the array.

For example, in BT2, node 6 only has a left child (i.e., node 2) and the right child is empty. Thus, you must enter -1 for the right child (green colour box in array) value in node 6. The left and right children (blue colour boxes in array) for this empty right child in node 6 must also be set to -1. Note that since Level x is the last level, for the nodes at Level x (e.g.: nodes 3, 15 and 18 in BT2) will have no further child nodes by definition, therefore it is not necessary to store the value for their child nodes.

In this question, you are required to write a programme to read an array and the value of a selected parent node, then print the values of the child nodes of this parent node. For example, in BT2, if the selected parent node is the node with value 13, then the left child value is 12 and right child value is 17; etc.

Write a programme to

Input, in sequence

x, the number of levels of the BT, where $1 \leq x \leq 6$; then,

p, the value of the selected parent node; and finally,

$(2^x - 1)$ integers that represent the nodes in the BT stored in an array.

Note that the value of each node will be different in the range from 1 to 100 if it is not an empty node. If a node is empty, then -1 will be read for this node.

Output, in sequence, two integers that are the left child value and the right child value, respectively, of the selected parent node, p. If any of the child value is empty, then print "NULL".

试题 15. 二元树的子节点（20 分）：

二元树 **（Binary Tree，BT）** 是一种非线性资料结构。如图 Q15 所示，二元树中每一个节点（node）包含了特定的正整数，且这些节点分布在 x 层（levels）中。例如，图 Q15 中有两个二元树，BT1 以及 BT2，而 BT1 有 3 层的节点，BT2 则有 4 层的节点。

在二元树的第一层中，只有一个唯一的节点，而这节点最多能链接到两个在第二层的子节点（child nodes）。假设 $2 \leq i \leq x$，递归地，一个在 i 层的节点必须链接到一个在 $i-1$ 层的母节点（parent node），同时只能拥有最多两个在 $i+1$ 层的子节点。

二元树（Binary Tree）的结构	数组（Array）的表示法

图 Q15. 两个二元树 (BT1 及 BT2) 和它们的数组示意图

请再参考图 Q15，一个二元树在程式中可以用数组（array）的型态来表示。在数组表示法中，我们必须创建一个大小为 $2^x - 1$ 的数组，同时假设所有分布在 $x - 1$ 层或以上的节点皆有两个子节点，即左子节点和右子节点。由于原有的二元树中，有的节点可能有少过两个的子节点，这样的话，在数组中我们就必须用-1 来表示空置的子节点。

例如上图中，BT2 的 节点 6 只有一个左子节点（即节点 2），但右子节点是空的。此时，我们必须以-1 来表示节点 6 的右子节点（即在数组中绿色的方格）。由于这个子节点是空置的，自然的它往下的左、右子节点也都应该是空的，所以它们也必须用-1来表示（即在数组中的两格蓝色方格）。要注意的是，由于 x 是最底层，所以在 x 层的节点（即在 BT2 中的节点 3，15 和 18）是肯定没有子节点的，所以也无需继续再储存其空置的子节点。

在这道题中，你需要编写一个程序以读取一个数组以及一个选定的母节点的值。然后输出其子节点的值。例如，在 BT2，假设选定的母节点为节点 13，则其左子节点的值为 12，以及右子节点的值为 17，等等。

试写一程式以

依序输入

x，在二元树中的层数，已知 $1 \le x \le 6$；然后，

p，选定的母节点的值；最后，

($2^x - 1$) 个整数，分别代表了此二元树在数组表示法里的节点值。

注意的是，只要节点不是空置的，它们之间的值将各不相同，且在 1 到 100 的范围之内。但若节点是空置的，则其代表值将是-1。

依序输出，两个分别代表了该母节点 p 的左字节点和右字节点的整数值。当相应的子节点为空置时，则输出 "NULL"来表示。

Example (例子)

Input (输入)	Output (输出)
3 7 13 7 20 15 8 2 9	15 8
4 2 8 6 13 2 -1 12 17 -1 3 -1 -1 -1 -1 15 18	NULL 3
5 70 30 20 55 11 23 48 70 8 12 -1 26 37 51 -1 -1 -1 -1 -1 17 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1	NULL NULL