1. Este problema se solucionó con los conocimientos que nos impartió el profesor en clases sobre las listas, pero sobre todo con el operador de descomposición, ya que haciendo uso de este operador podemos obtener el segundo elemento de una lista. Sólo tuvimos que descomponer hasta el segundo elemento, y si 'X' era ese segundo elemento, entonces el programa debería decir que es true.
2. Este problema se solucionó utilizando el operador de descomposición, ya que aquí nos pide que debemos hacer un predicado que dado dos listas nos diga si son idénticos, pero con el primer y segundo elemento de las listas estén intercambiados, por lo tanto, usando ese operador podemos obtener esos dos elementos en concreto y decir si están intercambiados o no.
3. Este ejercicio se resolvió analizando como íbamos a traducir el arreglo y lo primero que vemos es que en el mandato nos dice que cuando la lista no está vacía sólo tenemos que traducir la cabeza de la lista y hacer una recursividad, pero con la cola de ésta, sin embargo en este problema nos pregunta ¿qué debemos hacer cuando la lista está vacía? Analizando esta pregunta se llegó a la conclusión de que sólo debemos de retornar la misma lista vacía para que unifique una vez que en la recursión se halla obtenido una lista vacía en el lado izquierdo, ya que si llegamos ahí el programa debería terminar y con la recursividad ir traduciendo la lista.
4. Este problema se solución respondiendo las preguntas que nos hace, sobre ¿qué tenemos que hacer con el caso base? y ¿qué debemos hacer con la cabeza de la lista en la parte de la recursión? En el caso base de la recursión es similar al ejercicio anterior porque es en la recursión que uno va realizando la operación del duplicado, por lo tanto, cuando llegue a la lista vacía, este termina porque ya no hay más que recorrer y se devuelve la misma lista vacía para que unifique. En el caso de la recursión es ir duplicando la cabeza de la lista del lado izquierdo e ir descomponiendo esta hasta llegar a la lista vacía.