

Segundo trabajo para evaluación

Reutilizando el fichero del trabajo anterior, diseñar y verificar los siguientes *método iterativos* y el método `split` (fichero `split.dfy`):

- 1.- método que obtenga la lista que resulta de concatenar dos listas dadas
- 2.- método que dada una lista obtenga su inversa.
- 3.- método que dadas dos listas ordenadas (no-decrecientes) de enteros, obtenga la lista que resulta de hacer la mezcla ordenada de ambas.

Par especificar este método deberás

- a) definir un predicado recursivo que decida si una lista de enteros es ordenada no-decreciente
- b) definir una función “`multiset_of`” que dada una lista obtenga el multiset (conjunto con repeticiones) de elementos que contiene (Dafny tiene el tipo predefinido `multiset<T>`)

Para demostrar la corrección de este método deberás:

- c) probar un lema que afirme que el `multiset_of` del `append` de dos listas es la unión de los `multiset_of` de ambas listas
- d) definir un predicado que decida cuando un elemento pertenece a una lista
- e) añadir invariantes afirmando que todos los elementos de algunas listas auxiliares son mayores o iguales que los de la lista en la que se va construyendo la mezcla
- f) probar un lema que afirme que si dos listas `xs` e `ys` son ordenadas y todos los elementos de `ys` son mayores o iguales que los de `xs`, entonces `append(xs,ys)` es ordenada.
- g) otros lemas o `asserts` adicionales a usar son:
 - Un elemento pertenece al `append` de dos listas si y solo si pertenece a alguna de ellas.
 - Si una lista es no-vacía y ordenada su cabeza es menor o igual que cualquier otro elemento.
 - Un elemento `x` pertenece al `Cons` de otro elemento y con `Nil` si y solo si ambos (`x` e `y`) son iguales.
 - Si una lista `xs` es no-vacía todo elemento de `xs.tail` es también elemento de `xs`.