

## Auxiliar 5: Listas Recursivas

Todos los problemas deben ser resueltos en *Python*, utilizando estrictamente la Receta de Diseño entregada a lo largo del curso. Use nombres apropiados para funciones y variables, y testee cada vez que sea posible.

### 1. Listas de enteros

Como dice el nombre de esta sección, en esta parte trabajaremos sólo con listas de enteros. Ejemplo: `lista(3, lista(5, lista(12, lista(4, lista(5, None))))`

1. Recuerde/escriba las funciones `maximo: lista(any) any ->any` y `sacarElemento: lista(any) any ->lista(any)`
2. Piense en un algoritmo para ordenar decrecientemente (de mayor a menor) una lista (Hint no tan sutil: utilice las funciones que acaba de escribir 1313)
3. Implemente el algoritmo que diseñó en el paso anterior
4. Ahora, piense en cómo tendría que ser un algoritmo para ordenar crecientemente (menor a mayor) y adapte su código para que funcione correctamente (utilice funciones `lambda`)

### 2. Listas de estructuras

Suponga que en este momento existe una clasificatoria para no sé, digamos, un Mundial de Fútbol en Rusia el 2018 (sólo estamos suponiendo). Además, asuma que existe una estructura llamada `Pais`, que tiene dos campos: `nombre (str)` y `puntos (int)`.

1. Suponga que le dan una lista con todos los países que participan en la clasificatoria, ordenada alfabéticamente por el nombre del país. Ordene la lista de acuerdo a los puntos de los países, SIN ESCRIBIR NUEVAS FUNCIONES y sólo usando `lambda`.
2. Suponga ahora que existe una estructura llamada `PartidoChile`, que contiene los campos `pais` `estadio` `fecha`, y que le entregan una lista con los próximos partidos de Chile. Además, le entregan una lista de `strings` con los nombres de los estadios donde Chile es local. Con todo ello, escriba una función llamada `partidosLocal: lista(PartidoChile), lista(str) ->lista(PartidoChile)`, que retorne sólo aquellos partidos que serán jugados en un estadio donde Chile es local.  
Hint: sí, recibe dos listas, fíjese sobre cuál debe iterar.