

1) La provincia de Santa Fe tiene información sobre las ciudades de la provincia en listas enlazadas simples (en cada lista vienen varias ciudades del mismo departamento). En cada nodo de una lista enlazada con ciudades, la información almacenada es:

- Código Departamento (entero en 1..20)
- Nombre de la ciudad (30 caracteres)
- Cantidad de habitantes (entero sin signo)
- Sector (0:Sur, 1:Centro, 2: Norte)

a) Declarar las estructuras de datos necesarias para manejar una lista Ciudades con estas características, y una variable *Start*, que pueda utilizarse para apuntar al comienzo de la misma.

b) Se dispone de varias listas de Ciudades (una lista para cada Departamento) ya cargadas con información, y ordenadas por cantidad de habitantes de menor a mayor.

Se precisa diseñar la función *ItemB()* que recibe dos listas de Ciudades, un valor entero *X*, y un valor entero *Cant* y debe cumplir con el siguiente requerimiento:

Retornar una lista de Ciudades con la información de las *X* ciudades de las dos listas, que tengan una cantidad de habitantes mayor que *Cant*. Esta lista debe estar ordenada por cantidad de habitantes de mayor a menor. (se puede asumir que las dos listas recibidos no se encuentran vacías).

Explique brevemente con sus palabras -sin codificar-, dos formas diferentes de cumplir con este requerimiento.

c) Codificar la función *ItemC()* respetando las siguientes restricciones, esta función tiene que retornar una lista resultado, con la misma información que la retornada en la función *ItemB()*, pero:

- Se reciben dos listas de ciudades como entrada (la lista A se encuentra ordenada por cantidad de habitantes de menor a mayor, la lista B no tiene ningún orden). Ninguna de las listas recibidas puede modificarse y no está permitido utilizar algoritmos de ordenamiento sobre listas.
- La lista resultado debe pasarse como parámetro.
- Como resultado, la función debe retornar la cantidad de habitantes por cada sector de la provincia.

Se puede utilizar (siempre acompañadas del código correspondiente) funciones auxiliares que ayuden a descomponer ó facilitar la solución. Se puede codificar tanto funciones iterativas como recursivas.
