* Lista simple enlazada ordenada (LSEO):
  + Es una lista enlazada a la que se le añade la propiedad de ordenación de sus datos.
* Pueden realizarse múltiples operaciones, al igual que en las LSE, sobre este tipo de estructuras:
  + Inserción:
    - Puede insertarse elementos, al principio, en le medio o al final de la lista.
    - En un principio la lista esta vacía, y se van agregando los elementos y que siempre cumplan con el ordenamiento de una manera creciente.
    - Generalmente implica un recorrido sobre la lista, con el final de localizar la posición exacta en donde se debe agregar el elemento, “engancharse” el nuevo nodo, respetando el ordenamiento, para ello realizaremos el recorrido e iremos contrastando la información de los elementos de la lista y la información del elemento que se desea insertar.
    - A medida que recorramos la lista debemos ir almacenando en el nodo **ant**, la dirección del nodo ya visitado, antes del pasar al siguiente nodo.
    - Para insertar, según la ubicación dentro de la LSEO, tenemos:
      1. Insertar primero (lista vacía):
         * Debemos crear nuestro nodo x;
         * Luego debe ser referenciado, este nodo, por **list**, para esto almacenaremos la referencia que “señala” a nuestro nodo, por ejemplo, x, en list, y de este modo se vincula el nuevo nodo a la lista.
      2. Insertar primero (lista con elementos):
         * Debemos crear nuestro nodo x;
         * Verificar que el valor del nodo que queremos añadir a la lista es mejor que la información que almacena el primer nodo de la lista;
         * Haremos que el campo **ps** del nuevo nodo almacene **list**, esto producirá que “señale” al mismo nodo;
         * Luego almacenaremos la dirección del nodo x en la referencia **list**, de esta manera el nodo se engancha en la lista de manera ordenada.
      3. Insertar final:
         * Suponemos que recorrimos por completo la lista almacenando en nuestro nodo **ant** la dirección del nodo que abandonábamos;
         * Identificamos que llegamos al final de la lista porque la referencia **p** es igual a **null**, y el último nodo de la lista esta referenciado por **ant**;
         * Esto significa que la información que se almacena en el nodo x, en el campo información, que queremos insertar, posee un valor mayor que cualquier información almacenada en cualquier nodo de la lista;
         * Debemos almacenar el valor **null** en el campo **ps** del nuevo nodo, y luego almacenar en el **ps** del nodo anterior la dirección del nuevo nodo x, de esta manera el nuevo nodo queda “enganchado” al final de la lista.
      4. Insertar medio:
         * Cuando recorremos la lista, y mediante la comparación sucesiva de la información que se almacena en el nodo x que queremos insertar con los elementos de la lista, resulta que es menor que alguna información almacenada, entonces encontramos el punto de inserción que nos interesa para no alterar el orden de la estructura;
         * En esta situación, tendremos el “puntero” **ant** “señalando” al nodo que visitamos previamente;
         * Debemos hacer que el “puntero” del nodo x “señale” al nodo p, y a continuación, el “puntero” del nodo **ant**, debe “señalar” al nodo x, esto producirá que el nodo **x** quede “enganchado” en la lista respetando el ordenamiento.
  + Eliminación:
    - Al igual que en el caso de la inserción, se realiza un recorrido sobre la lista haciendo una constatación de la información que queremos eliminar y la información contenida en los nodos que estamos “visitando” durante el recorrido, con el objetivo de reconocer aquel nodo que queremos quitar de la lista.
    - En el caso que encontremos un campo información cuyo valor es mayor que el valor que estamos buscando, podemos concluir que el dato no se encuentra en la lista.
      1. Eliminar principio: