

- 1) Dadas las siguientes estructuras hacer la primitiva de insertarUltimo. (3 puntos)

<pre> Lista{     INodo primero;     int tam; } </pre>	<pre> Nodo{     int dato     INodo siguiente;     INodo anterior; } </pre>
---	--

- 2) Realizar la búsqueda secuencial, para un array estático y para una lista Dinámica (mismas estructuras del ejercicio 1). ¿Cuál de los dos métodos es más eficiente? ¿En qué momentos? (2 puntos)
- 3) Dado estos dos algoritmos, y suponiendo que resuelven lo mismo, ¿cuál es más eficiente? ¿Por qué? (1 punto)

<pre> public int funcion1(List&lt;Integer&gt; lista) {     int conteo = 0;     boolean enBloque = false;      for (int num : lista) {         if (num &gt; 0) {             if (!enBloque) {                 conteo++;                 enBloque = true;             }         } else {             enBloque = false;         }     }      return conteo; } </pre>	<pre> public int funcion2 (List&lt;Integer&gt; lista) {     int conteo = 0;      for (int i = 0; i &lt; lista.size(); i++) {         if (lista.get(i) &lt;= 0) continue;         for (int j = i; j &lt; lista.size(); j++) {             if (lista.get(j) &lt;= 0) break;             if (j == lista.size() - 1    lista.get(j + 1) &lt;= 0) {                 conteo++;                 i = j;                 break;             }         }     }      return conteo; } </pre>
---	---

- 4) Insertar estos números en un árbol binario de búsqueda, y recorrer el árbol post-order, hacer el gráfico.  
11,3,4,5,12,32,43,2,1,11,13,43,12 (1 puntos)
- 5) Realizar el algoritmo pre-order para un árbol binario de int, bajo el paradigma de TDA. (1 punto)
- 6) Insertar en un árbol AVL los números: 11,3,4,5,12,32,43,2,1,11,13,43,12, mostrar como queda el árbol. Justificar cada paso. (2 puntos)

Notas:

6	7	8	9	10	< 6
4	5	7	9	10	Desaprobado