

**Instituto Tecnológico de Costa Rica (ITCR)**  
**Ingeniería en Computación, Alajuela (IC-Alajuela)**  
**Tarea corta**

**Tema:** Programación funcional - Racket

**Entrega:** Un archivo zip que incluya el código fuente de Racket y documentación en un archivo pdf **a través del TEC-digital**.

**Modo de trabajo:** Grupos de máximo dos personas.

**Objetivo:** implementar funcionalidad para el manejo de un árbol n-ario utilizando programación funcional en Racket.

**Descripción:**

La tarea consiste en crear un programa en Racket que maneje toda la funcionalidad de un árbol n-ario. Un árbol es una estructura de datos sin ciclos, formada por nodos o vértices y aristas.

**Se debe implementar las siguiente operaciones para la estructura de datos árbol:**

- Crear el árbol (create-tree). Los nodos del árbol deben contener un identificador que puede ser numérico.
- Listar todos los nodos (list-all-nodes): recibe una estructura de tipo árbol y despliega el contenido completo de esta (nodos y aristas). El despliegue puede ser como las o los estudiantes deseen.
- Buscar un nodo (find-node): devuelve todos los datos asociados a un nodo y el nodo padre. Recibe el árbol y el nodo a buscar.
- Insertar un nuevo nodo en una posición determinada en el árbol (insert-node). Recibe el árbol, el nodo padre y el identificador del nodo a insertar.
- Eliminar un nodo (delete-node). Recibe el árbol y el nodo a eliminar. La función retorna el nuevo árbol.
- Buscar el ancestro (ancestor). Recibe un árbol y el nodo para el cual se desea buscar el ancestro.

**Implementación:**

- La funcionalidad debe tener la capacidad de manejar árboles n-arios.
- El programa debe tener documentación breve que incluya una descripción del problema, descripción de la solución (estructura de datos utilizada, descripción de las funciones clave, entre otros), explicación de cómo se corre y cómo se utiliza (incluyendo capturas de pantalla).

- Debe programarse en Racket (la tarea no será aceptada en otro lenguaje de programación).
- En el código: **no pueden usar iteraciones por ejemplo “for”, “for\*” o “for/list”.** Pueden usar variables globales pero no más de 2.
- El programa debe venir acompañado de un vídeo que muestra la ejecución de toda la funcionalidad.

## Rúbrica

	Puntos
<b>Documentación (breve)</b>	
Presenta descripción del sistema	1
Describe la solución (estructura de datos utilizada, descripción de las funciones clave, entre otros), explicación de cómo se corre y cómo se utiliza (incluyendo capturas de pantalla).	3
Se incluyen capturas de pantallas representativas de una corrida.	2
La redacción y ortografía es muy buena en todas las secciones del documento. El documento está escrito en tercera persona y no tiene más de tres faltas ortográficas o de redacción.	3
<b>Implementación</b>	
La funcionalidad debe tener la capacidad de manejar árboles n-arios.	10
En el código: <b>no pueden usar iteraciones por ejemplo “for”, “for*” o “for/list”.</b> Pueden usar variables globales pero no más de 2.	3
El programa debe venir acompañado de un vídeo que muestra la ejecución de la funcionalidad.	5
<b>Funcionalidad del árbol:</b>	
<ul style="list-style-type: none"> <li>• Crear el árbol (create-tree). Los nodos del árbol deben contener un identificador que puede ser numérico.</li> </ul>	3

<ul style="list-style-type: none"> <li>Listar todos los nodos (list-all-nodes): recibe una estructura de tipo árbol y despliega el contenido completo de esta (nodos y aristas). El despliegue puede ser como las o los estudiantes deseen.</li> </ul>	2
<ul style="list-style-type: none"> <li>Buscar un nodo (find-node): devuelve todos los datos asociados a un nodo. Recibe el árbol y el nodo a buscar.</li> </ul>	3
<ul style="list-style-type: none"> <li>Insertar un nuevo nodo en una posición determinada en el árbol (insert-node). Recibe el árbol, el nodo padre y el identificador del nodo a insertar.</li> </ul>	3
<ul style="list-style-type: none"> <li>Eliminar un nodo (delete-node). Recibe el árbol y el nodo a eliminar. La función retorna el nuevo árbol.</li> </ul>	2
<ul style="list-style-type: none"> <li>Buscar el ancestro (ancestor). Recibe un árbol y el nodo para el cual se desea buscar el ancestro.</li> </ul>	2