

Algoritmos y Estructura de datos**Segundo Parcial 2022**

Nombre y apellido: _____

Ejercicios				Nota
1 (2 pts)	2 (3 pts)	3 (3 pts)	4 (2 pts)	

CONSIDERACIONES: RESOLVER CADA EJERCICIO EN UNA HOJA DIFERENTE. LOS EJERCICIOS QUE NO ESTÉN CORRECTOS EN UN 50% NO SUMARÁN PUNTOS PARA LA NOTA FINAL. PARA CADA EJERCICIO DEBE **DEFINIR EL TIPO DE DATO** DE CADA TDA UTILIZADO..

Ejercicio 1: Dado un árbol binario, **devolver una cola** (TDA *queue*) que en su contenido se encuentren **los punteros a los valores** de cada nodo del árbol **ordenados por nivel**, donde los niveles más bajos queden al final de la cola y la raíz (nivel 0) al principio. Ejemplo:



Encabezado de la función: `queue* enqueue_tree (btn* root);`

Se debe tener en cuenta:

- Los valores del ejemplo son orientativos, debe funcionar para cualquier tipo de dato en los nodos del árbol.
- Debe desarrollar todas las funciones y estructuras que utilice, excepto las pertenecientes a los TDA.

Ejercicio 2: Escribir una función para un **File System** que reciba un **id de un directorio** y un **id de usuario** como entrada y devuelva una **lista** (TDA *list*) con los nombres de los archivos que tienen como atributo de grupo (*group_ID*) uno los grupos a los que pertenece el usuario. **Debe utilizar el TDA de FileSystem diseñado.** En la solución se debe incluir como mínimo los siguientes algoritmos:

- Encontrar el directorio en la estructura del *File System*.
- Obtener los grupos del usuario dado.
- Recorrer el directorio y por cada archivo encontrado validar la condición para agregarlo a la lista.

Ejercicio 3: Dados dos **digrafos ponderados**, el primero representado por su **matriz de adyacencia** y el segundo representado por su **lista de adyacencia**, desarrollar una **función booleana** que indique si el segundo grafo (lista) es un **subgrafo** del primero (matriz). Definir los **tipos de datos** de ambos grafos sin usar ningún TDA. Asumir que los vértices están identificados por sus índices en la estructura de datos.

Encabezado de la función: `int is_subgraph (graph_matrix* g1, graph_list* g2);`

Ejercicio 4: Suponiendo la siguiente secuencia de valores:

12, 45, 78, 89, 56, 23, 21, 54, 87, 98, 65

- Represente gráficamente el **árbol B+**, **de grado 4**, que se obtiene ante la inserción de tales valores en dicho orden.
- Del árbol obtenido en el punto anterior, elimine los valores 54, 23 y 98. En cada paso grafique el árbol resultante.