# Universidad Interamericana de Panamá Facultad de Ingeniería, arquitectura y diseño Escuela de Ingeniería y Sistemas.

Materia: Estructura de Datos II Cuatrimestre: IIQ\_2019

Código: 301-00040 Facilitador: Leonardo Esqueda

Nombre completo: Diego Alexander Martínez A.

Carrera: Ing. En Sistemas Computacionales

# Parcial #1

ID: 8-878-1209

## Instrucciones: La prueba parcial está enfocada en dos (2) pates: Teórico y Práctico; constando con 13 problemas. Lea detenidamente antes de responder.

1. **Teoría:**
   1. ¿Qué son las Estructuras de Datos? 5ptos

*Son diferentes formas de organizar información para manipular, buscar e insertar estos datos de manera eficiente.*

* 1. ¿De dónde vienen las estructuras de datos? 5ptos

*Dispositivos, supermercados, de todas partes.*

* 1. ¿Mencione las dos (2) ramas de la estructura de datos interna? 5ptos

*Estáticas y Dinámicas.*

* 1. ¿Cómo interactúan las estructuras de datos internas y externas? 5ptos

*La interacción que tiene una estructura de datos interna con una externa se da a que se complementan al realizar un proceso. Es decir, con las estructuras externas obtenemos los datos que se requiere analizar, pero son gracias a las estructuras internas que podemos realizar cálculos, resolución de problemas haciendo uso o extracción de la data obtenida de las estructuras de datos externas.*

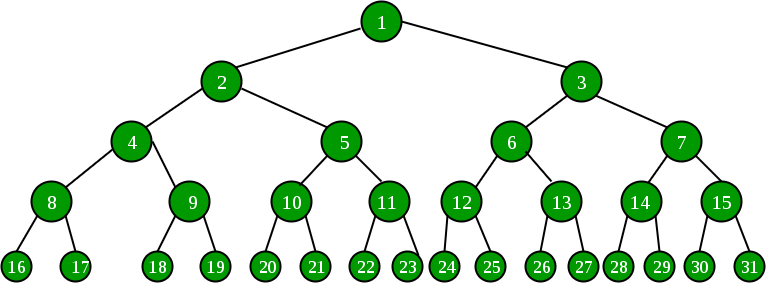
* 1. ¿Cómo funcionan los árboles Binarios? 5ptos

*La inserción de sus elementos se realizan siguiendo un criterio, si el ítem del nodo a insertar es menor a su nodo padre, la inserción se realiza por la izquierda y en caso contrario la inserción se realiza por el lado derecho y como consecuencia de esto nuestro árbol siempre estará organizado de tal manera que los hijos izquierdos de cada nodo serán menores a él y los derechos serán mayores, lo cual nos permite realizar búsquedas muy eficientes debido a la organización de esta información donde para buscar un elemento solo es necesario ir comparando el valor a buscar con el valor del nodo actual y así tomar la decisión si ir por la izquierda o por la derecha, ahorrándonos las búsquedas por el otro lado del árbol.*

* 1. Describa un caso en donde usted podría realizar y utilizar un árbol binario. 5ptos

*Un excelente caso para utilizar un árbol binario sería para administrar las consultas o búsquedas en mis bases de datos. El sistema responde más rápido porque no tiene que hacer validaciones por cada elemento, sino más bien por bloques, si el criterio especificado entra en el rango de este nodo corresponde entonces miro adentro para sustraerlo. Ganaría eficiencia en mis búsquedas y procesos.*

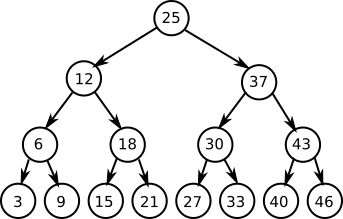
* 1. Mencione y explique 5 propiedades de los métodos de ordenamiento de los árboles binarios. 5ptos
  + *Tiene un buen rendimiento.*
  + *Es estable (no cambia el orden relativo de elementos iguales).*
  + *No requiere espacio de almacenamiento extra.*
  + *Puede ordenar listas tal cual las recibe.*
  + *Eficiencia en consultas y búsquedas*
  1. Distribuya los nodos -5,2,-11,4,-13,5,3,-14,1,6,10,-12,8 en un árbol binario y determine ¿Cuáles son los nodos hoja? ¿Cuáles son nodos de 2 hijos? 5ptos
  2. Realice el recorrido preorden, inorden y postorden de los árboles siguientes: 5ptos



## PreOrden: 1,2,4,3,8,5,6,7,16,9,10,11,12,13,14,15,17,18,19,20,21,22,23,23,25,26,27,28,29,31,31

## InOrden: 1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30,31

## PostOrden: 3,7,6,5,15,14,13,12,11,10,9,31,30,29,28,27,26,25,24,23,22,21,20,19,18,17,16,8,4,2,1

1. 

**PreOrden:** 25,12,6,3,9,18,15,21,37,30,37,33,43,40,46

**InOrden:** 3,6,9,12,15,18,21,25,27,30,33,37,40,43,46

**PostOrden:** 3,9,6,15,18,12,27,33,30,40,46,43,37,25

* 1. Describa un problema del mundo real y que usted considere que pueda ser solucionado utilizando

las estructuras de datos. Mencione qué modelo estructura de datos utilizaría, porqué y desarrolle la solución 10ptos

## Práctica:

* 1. Realice un algoritmo lógico el cual explique y demuestre el proceso de inserción de los elementos del problema teórico número dos (2) en un árbol binario. 15ptos
  2. Desarrolle un programa en el lenguaje C++ o python en donde cargue por defecto los elementos del árbol: 9 – A y realice los recorridos: preorden, inorden y postorden 15ptos



* 1. Las Torres de Hanoi es un juego matemático que consiste en tres varillas verticales y un número indeterminado de discos que determinarán la complejidad de la solución. No hay dos discos iguales, están colocados de mayor a menor en una varilla ascendentemente, y no se puede colocar ningún disco mayor sobre uno menor a él en ningún momento. El juego consiste en pasar todos los discos a otra varilla colocados de mayor a menor ascendentemente.

Leyenda: Dios al crear el mundo, colocó tres varillas de diamante con 64 discos en la primera. También creó un monasterio con monjes, los cuales tienen la tarea de resolver esta Torre de Hanoi divina. El día que estos monjes consigan terminar el juego, el mundo acabará. El mínimo número de movimientos que se necesita para resolver este problema es de 264-1. Si los monjes hicieran un movimiento por segundo, los 64 discos estarían en la tercera varilla en poco menos de 585 mil millones de años. Como comparación para ver la magnitud de esta cifra, la Tierra tiene como 5 mil millones de años, y el Universo entre 15 y 20 mil millones de años de antigüedad, sólo una pequeña fracción de esa cifra.

Resolución: el problema de las Torres de Hanoi es curioso porque su solución es muy rápida de calcular, pero el número de pasos para resolverlo crece exponencialmente conforme aumenta el número de discos. Para obtener la solución más corta, es necesario mover el disco más pequeño en todos los pasos impares, mientras que en los pasos pares sólo existe un movimiento posible que no lo incluye. El problema se reduce a decidir en cada paso impar a cuál de las dos pilas posibles se desplazará el disco pequeño:

El algoritmo en cuestión depende del número de discos del problema.

Si inicialmente se tiene un número impar de discos, el primer movimiento debe ser colocar el disco más pequeño en la pila destino, y en cada paso impar se le mueve a la siguiente pila a su izquierda (o a la pila destino, si está en la pila origen).

La secuencia será DESTINOt3, AUXILIARt2, ORIGENt1, DESTINO, AUXILIAR, ORIGEN, etc.

Si se tiene inicialmente un número par de discos, el primer movimiento debe ser colocar el disco más pequeño en la pila auxiliar, y en cada paso impar se le mueve a la siguiente pila a su derecha (o a la pila origen, si está en la pila destino).

La secuencia será AUXILIAR, DESTINO, ORIGEN, AUXILIAR, DESTINO, ORIGEN, etc. 15 ptos

