



UNIVERSIDAD DE CARABOBO  
Facultad Experimental de Ciencias y Tecnología  
Departamento de Computación  
Unidad Académica de Algoritmos y Programación  
CAO403: Programación II



**Profesor:** Álvaro Espinoza

**Fecha:** 16/04/2025

**Taller II**  
**Estructuras Jerárquicas: Árboles**  
**(Valor: 33.33%)**

**1. Instrucciones**

- a. El ejercicio propuesto requiere la lectura de datos de la entrada estándar (standard input) y la escritura de resultados en la salida estándar (standard output).
- b. Para el desarrollo del taller se debe utilizar el lenguaje de programación C++, junto con sus librerías estándar. Para la compilación de sus códigos fuentes, debe realizarla por medio de un archivo makefile, el cual deberá entregar junto con sus códigos fuentes. De no poseer makefile la entrega, el taller no será revisado.
- c. Su código debe estar debidamente comentado.
- d. Debe utilizar lowerCamelCase para las variables, y UpperCamelCase para la creación de tipos de datos (en caso de necesitarlos).
- e. SI se permite el uso de las bibliotecas de STL que proporciona C++.
- f. Debe utilizar sus propias librerías de Árboles Binarios, N Arios y Binarios de Búsqueda creadas por usted.
- g. Debe modularizar su código de tal forma que se evite el “código spaghetti”.
- h. El archivo makefile debe generar un ejecutable con el nombre “arboles”, con el fin de facilitar las labores de corrección de su taller, el no hacerlo generará puntos menos en su nota final del taller.
- i. La modalidad del taller es VIRTUAL y se permite el material de apoyo.
- j. La fecha tope de entrega es el día 16/04/2025 a las 11:59pm.
- k. La entrega debe hacerse al siguiente correo: [alvaroeed@gmail.com](mailto:alvaroeed@gmail.com)
- l. Su entrega debe incluir: `archivo.zip` con su código fuente. `Makefile`, `archivo.txt` con su nombre y cédula. Y en caso de compartir bibliotecas con algún compañero, agregar un archivo `integrantes.txt` con las cédulas y nombres de quienes la están usando (máximo 3 personas), cualquier persona con el uso de la misma librería que no haya sido reportada, será considerado como plagio y reprobado con la nota mínima.

## 2. Enunciado

“No todos los árboles dan frutos. Algunos dan... dolor.”



shutterstock.com · 466393721

El profesor de Programación 2 ha alcanzado un nivel de leyenda urbana en los pasillos del campus. Algunos dicen que no duerme, solo compila. Otros aseguran que vio el apocalipsis venir... y lo modeló con un `struct`. Pero lo único que todos saben con certeza es que le encanta hacer sufrir a sus estudiantes con árboles binarios. Y justo cuando creían que nada podía ser peor que los punteros dobles... llegó el taller.

Este taller.

Sí, este que estás leyendo.

Y sí, es *él* quien lo hizo.

Como parte de su plan maestro para evaluar la resistencia emocional de sus grupos, el profe ha creado una métrica 100% innecesaria, pero dolorosamente efectiva: la MUP (Métrica Universal de Padecimiento). El grupo de estudiantes que obtenga la mayor MUP será recompensado con... bueno, *no ser ignorado en clase*.

Para un conjunto de estudiantes seleccionados  $S = \{a, b, c, \dots\}$  de un árbol binario (que representa la sección, y están ordenados según sus notas), la **MUP** se calcula así:

$$MUP(S) = \sum_{(x,y)} (D(x, y) \times N_d(x, y))$$

Donde:

$D(x, y)$  Es la distancia en NODOS entre x e y (sin contar x e y).

$N_d(x, y)$  Es la distancia en NIVELES entre x e y. (Si dos nodos están en el mismo nivel la distancia es cero).

### Tu tarea

Recibirás varios árboles (uno por cada grupo de estudiantes sometidos al profesor), junto con un subconjunto de alumnos que merecen ser evaluados por la MUP. Tu misión es determinar **cuál árbol (grupo)** tiene la mayor MUP. Para ello debes calcular el MUP para todas las combinaciones de pares del subconjunto dado. No porque eso los haga mejores... sino porque así lo quiso el profe.

*“No es el taller que querías. Es el taller que merecías por no repasar árboles binarios.”*

### 3. Formato de Entrada

Un número  $t$  indicando cuántos casos de prueba hay.

Para cada caso:

- a. Un número  $n$ : cantidad de grupos de estudiantes (es decir, árboles).
- b. Por cada grupo:
  - i. Dos líneas que representan recorridos del árbol (uno siempre es **INORDEN**, el otro puede ser **PREORDEN** o **POSTORDEN**).
  - ii. Una línea con los  $m$  identificadores de los estudiantes a calcular el MUP

### 4. Ejemplo de Entrada

```
1
2
INORDEN juan camila santiago mariana pedro
PREORDEN santiago camila juan mariana pedro
juan mariana pedro
POSTORDEN laura jose fernando esteban carla
INORDEN laura fernando jose carla esteban
laura jose carla
```

## 5. Formato de Salida

Por cada caso, imprime una línea con el número del árbol cuyo subconjunto tiene la mayor MUP. Si hay empate, gana el grupo que aparece primero en la entrada. No insistas.

## 6. Ejemplo de Salida

2

Nota: el número de arriba es solo un ejemplo, no necesariamente es el valor correcto del caso ejemplo. Es tu deber analizar el problema y ver cuál es el valor correcto a retornar.

## 7. Playlist Recomendada Para la Resolución del Taller

- Metal Cristiano: [YouTube Metal Cristiano Mix](#)
- Rock No Cristiano: [YouTube Ghost - Mary On A Cross \(Official Audio\)](#)
- Cyberpunk: [YouTube Cyberpunk 2077 Breathtaking Mix | by Extra Terra](#)



*"La inteligencia artificial no reemplazará al ser humano, pero aquellos que la dominen reemplazarán a quienes no lo hagan." - Anónimo*