

# Task It

## **Autores:**

Santiago Ávila

Mateo Bustos

Adrian Ramirez Gonzalez

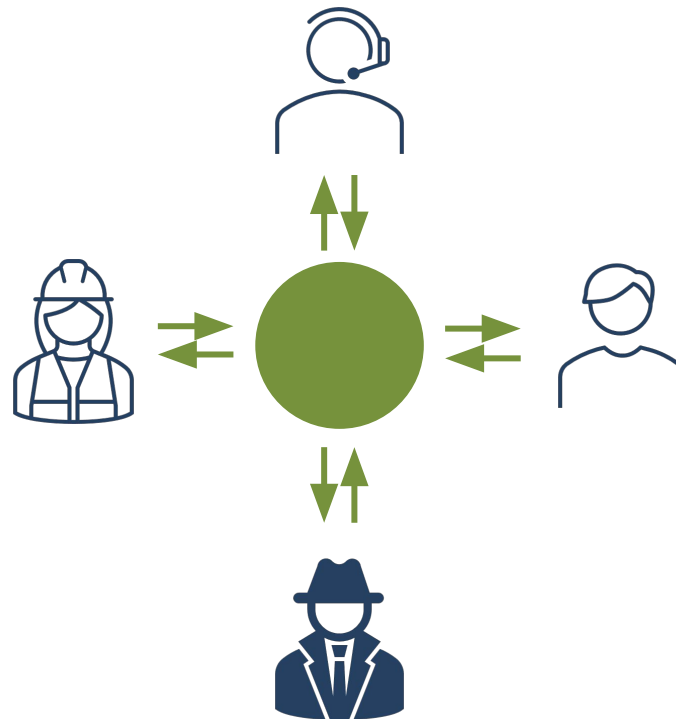


UNIVERSIDAD  
**NACIONAL**  
DE COLOMBIA  
SEDE BOGOTÁ



## Problema que se está resolviendo

Facilitación de la distribución de las tareas  
dentro de un grupo





## Requerimientos funcionales

### Organización de la información

El sistema tendrá una serie de subdivisiones que se guardarán en un AVL, donde cada una contará con:

1. Nombre.
2. Conjunto de tareas sin asignar.
3. Conjunto de tareas en progreso.
4. Conjunto de tareas finalizadas.

Cualquiera de las operaciones mencionadas más adelante se realizará sobre una subdivisión específica que indique el usuario.

### Inserción de nuevas tareas

El sistema permitirá que cualquier usuario pueda ingresar una nueva tarea siempre y cuando especifique una subdivisión en la que guardar la tarea, un nivel de prioridad del 1 al 10 para la tarea y una descripción válida de la misma. Si se cumplen las condiciones anteriores, el sistema guardará la tarea en la subdivisión correspondiente y se le asignará una identificación única.



## Requerimientos funcionales

### Asignar tareas

El programa podrá asignarle a cualquier usuario una tarea proveniente de una subdivisión específica. La tarea que se suministre será la que posea un nivel de prioridad más alto dentro de la subdivisión. Además, se le indicará al usuario la identificación que tiene la tarea.

### Notificación de la finalización de las tareas

Se podrá notificar la finalización de una tarea proporcionando únicamente la subdivisión a la que pertenece la tarea y una identificación válida.



## Requerimientos funcionales

### Visualizar el estado actual de las tareas de una subdivisión

El sistema podrá mostrar el estado de las tareas actuales de una subdivisión en la que esté interesado el usuario.

Para la visualización se tendrán tres tablas donde en cada una se mostrará las tareas que se encuentran en cada uno de los 3 estados posibles (Sin asignar, en progreso o finalizada).

### Permitir la creación de nuevas subdivisiones dentro del grupo

En caso de ser necesario se podrán crear nuevas subdivisiones indicando únicamente el nombre de esta. Sobre la subdivisión creada se podrán aplicar cualquiera de las anteriores funciones mencionadas.



## Implementación de nuevas estructuras de datos en la solución del problema planteado

### **Cola prioritaria:**

Guardado de tareas sin  
asignar en orden de  
prioridad

### **Pila:**

Guardado de tareas  
finalizadas

### **Arrays estáticos:**

Ingreso de datos a  
tablas

### **AVL:**

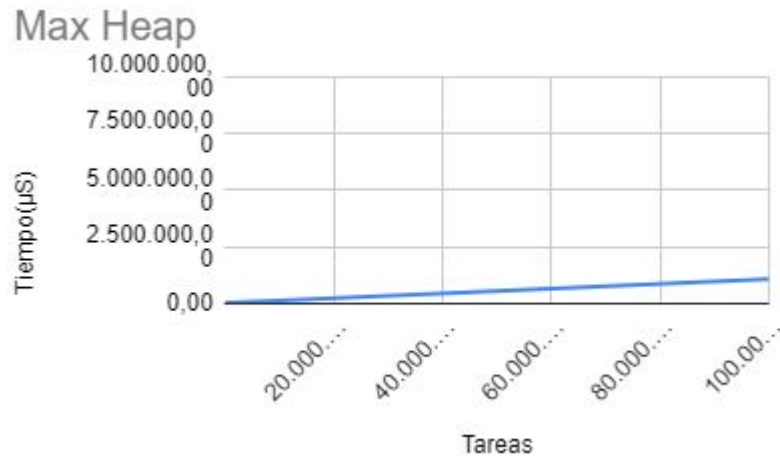
Guardado de tareas en progreso  
y de subdivisiones



## Ingresar tarea

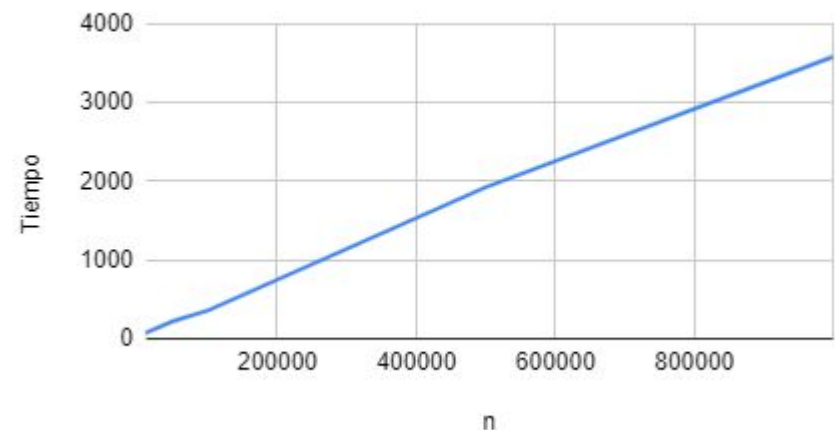
### Inserción a tareas sin asignar

#### Implementación con Max Heap



#### Implementación anterior: Lista simplemente enlazada con cola

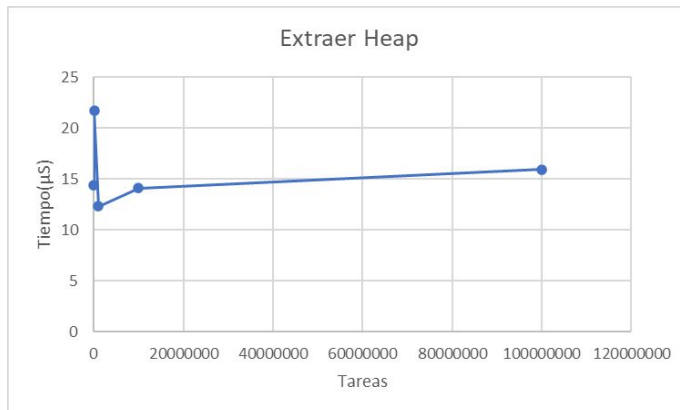
#### Lista Simplemente Enlazada



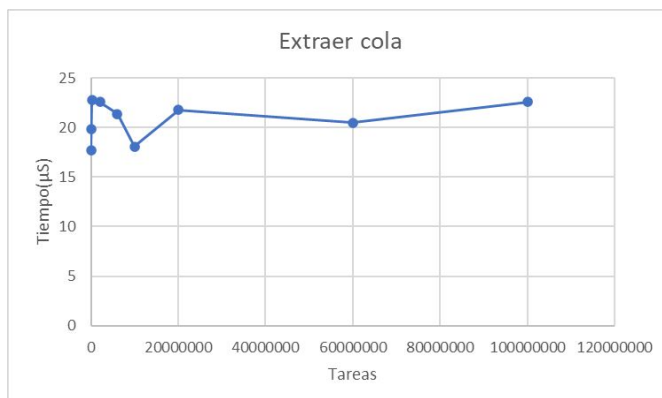
## Asignar tarea

### Eliminación de tareas si asignar

#### Implementación con Heap (cola prioritaria)

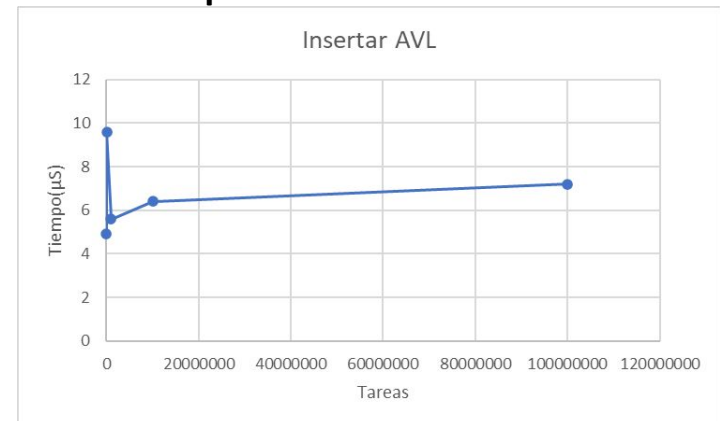


#### Implementación anterior: Desencolar



### Insertión a tareas en progreso

#### Implementación con AVL



#### Implementación anterior: Lista con cola

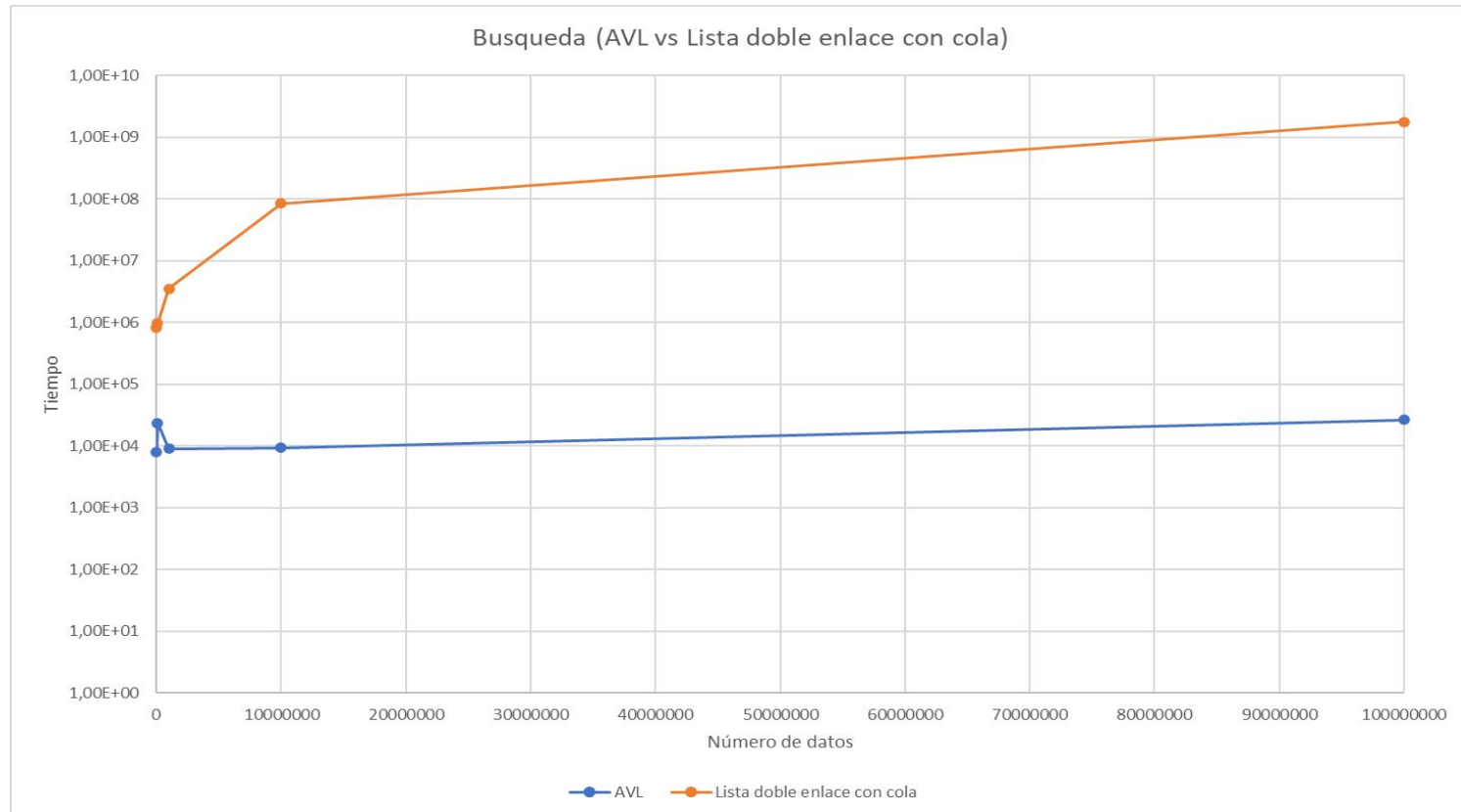






## Búsqueda de subdivisiones AVL vs Lista doblemente enlazada con cola

NOTA: El tiempo está escrito en escala logarítmica para visualizar mejor la gráfica





## Dificultades y lecciones aprendidas

### Dificultades

En comparación a las estructuras lineales, las estructuras no lineales tienen un grado de complejidad más alto en su implementación, principalmente ocasionado por el uso de la recursión, haciendo que la implementación de estas estructuras haya durado más tiempo de lo esperado.

### Lecciones

Tener un proyecto bien organizado desde un principio permite que las modificaciones que se hagan en versiones posteriores sean mucho más fáciles y rápidas de hacer.