

TPE PROGRAMACIÓN 3
1era Entrega

Bianco Manuel
Corengia Bernardo

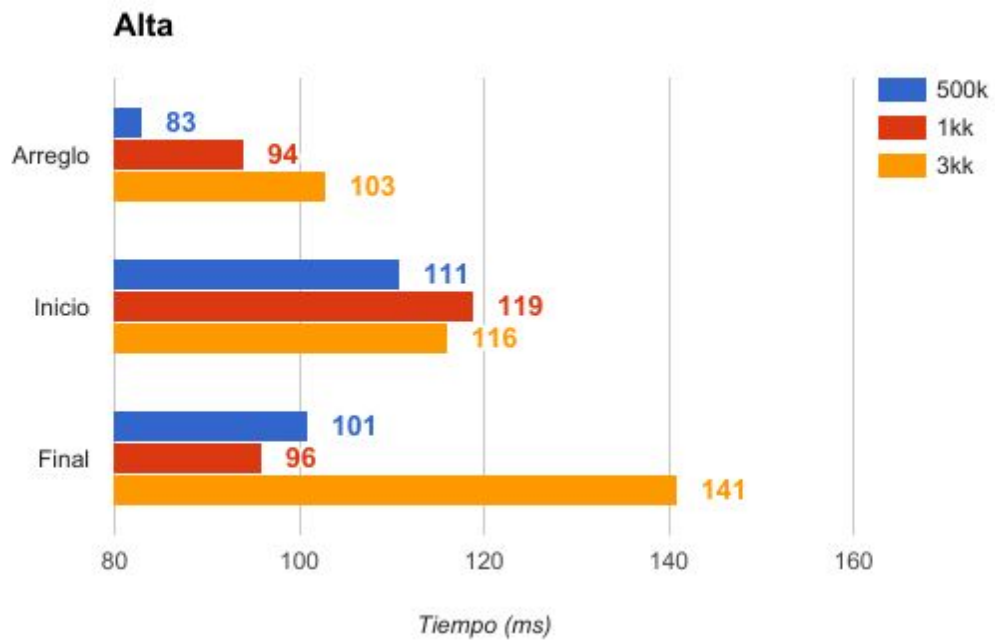
Para llevar a cabo la consigna, realizamos 3 aplicaciones, utilizando una estructura diferente para guardar los usuarios:

- Arreglo de usuarios
- Lista que inserta al principio
- Lista que inserta al final

Cada aplicación posee 4 clases:

- **Arreglo_string:** Se utiliza principalmente para los deportes.
- **Arreglo_usuarios / Lista (agrega al principio) / Lista (agrega al final):**
Clase de la estructura en donde se almacenan los usuarios (arreglo, lista que agrega al principio, o lista que agrega al final). Posee entre otras cosas funcionalidad para agregar y eliminar un usuario, como así también comprobar la existencia del mismo en la estructura dada. **La lista que agrega al final, guarda también el nodo final (aparte del inicial) por eso los tiempos son muy similares a la lista que inserta al principio.**
- **Usuario:** Guarda los datos de un usuario. Posee entre otras cosas funcionalidad para crear un usuario a partir de un String[] y devolver un string con los datos del usuario separados por “;”.
- **Main:** Implementa 3 métodos:
 - **Lector(String path):** Carga a partir de un archivo los usuarios a la estructura elegida (Arreglo o lista).
 - **Alta(Arreglo/Lista usuarios):** Lee un archivo (“dataset_insert_10000.csv”), agrega los usuarios leídos en el mismo a la lista/arreglo, crea un nuevo archivo, y guarda en este los datos del usuario agregado (sin repetir los deportes practicados) y con el tiempo que tardó la operación, separados por (“;”).
 - **Busqueda(Arreglo/Lista usuarios):** Lee un archivo (“dataset_busqueda_10000.csv”), y crea uno nuevo con los dni de los usuarios del archivo (sin sus deportes) un boolean (true o false) si encontró el usuario leído en la estructura existente (arreglo o lista) y el tiempo que tardó la operación, separados por (“;”).

Representación de los tiempos de ejecución:



Conclusiones:

En conclusión, se puede apreciar, que tanto los tiempos de inserción de listán al principio, como al final, son muy similares. Esto se debe a que en estas últimas, seteamos una variable con el último nodo, así, a la hora de insertar en el último, solo tenemos que consultar un solo valor y no recorrer toda la lista.

En general, los 3 tipos de listas tuvieron tiempos de alta similares, aunque las listas fueron más rápidas a la hora de dar de alta, en comparación contra los arreglos, debido a que, si el arreglo estaba lleno, tenía que generar uno nuevo con el doble del tamaño, y copiar los datos del viejo en el nuevo.

Aun así, tardaron un poco más a la hora de la búsqueda.

Los valores reales máximos obtenidos en los archivos generados se omitieron, porque no son representativos, ya que algunos valores eran de muchos órdenes mayores, por “cuelgues” en la ejecución. (23323 nanosegundos vs 3586399 nanosegundos).