

Informe final

EDP



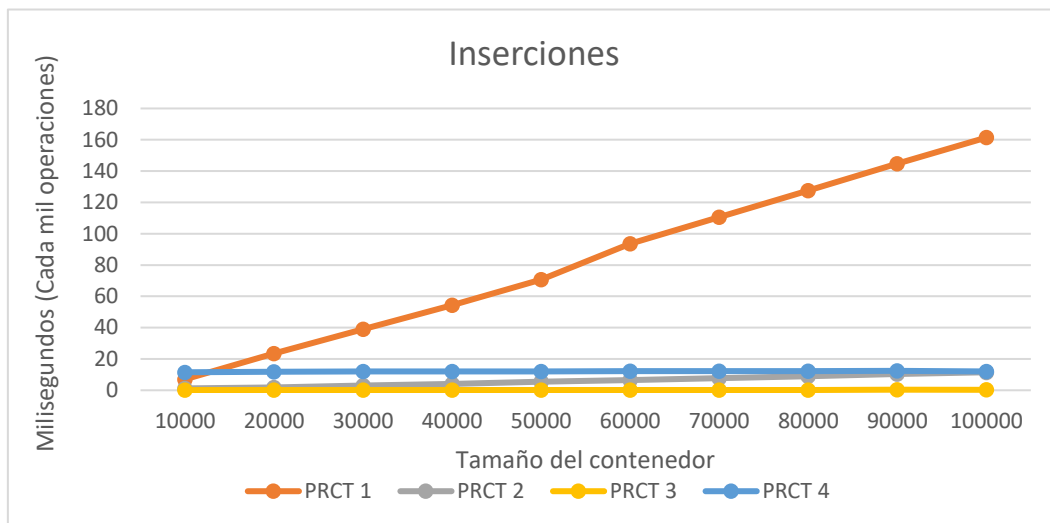
Grupo 01-44 Equipo 2

Cazorla Rodríguez, Sheila
Santana Acosta, Álvaro
Arzola Saavedra, Alejandro
David Rey Rodríguez,
Itamar
Suárez Navarro,
Amai
Medina Quintana, Pablo

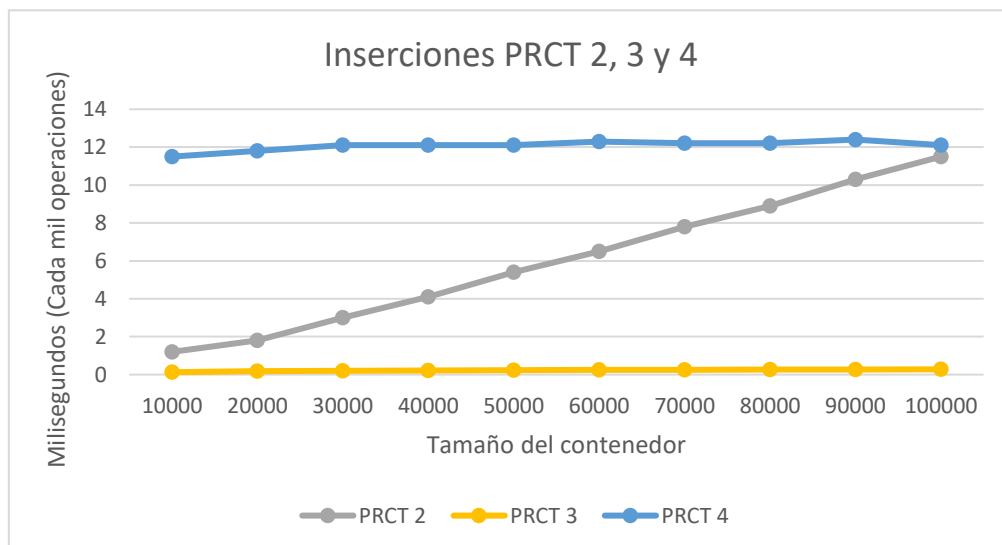
Inserciones

Observando las gráficas, la inserción en lista encadenada vemos que es el más lento. Debido a que las búsquedas del elemento que vamos a insertar se ejecutan comparando valor a valor de manera secuencial.

En el vector ordenado vemos una reducción considerable del tiempo, ya que en este caso las búsquedas a realizar son de manera dicotómica. Esta búsqueda es mucho más eficiente que la que usamos en la practica 1, porque necesita tratar menos elementos para hallar el que se está buscando. Sufre el inconveniente y necesidad de estar ordenado.



En el árbol binario de búsqueda encontramos que los tiempos son mucho más bajos que en la 2 práctica, ya que no tenemos la necesidad de realizar desplazamiento de los elementos produciendo un menor gasto computacional. Aunque cuidado porque al solo tener en cuenta menores a la izquierda y mayores a la derecha puede provocar que este contenedor degenera a una lista como la de la practica 1 porque no tiene operaciones de reequilibrado.

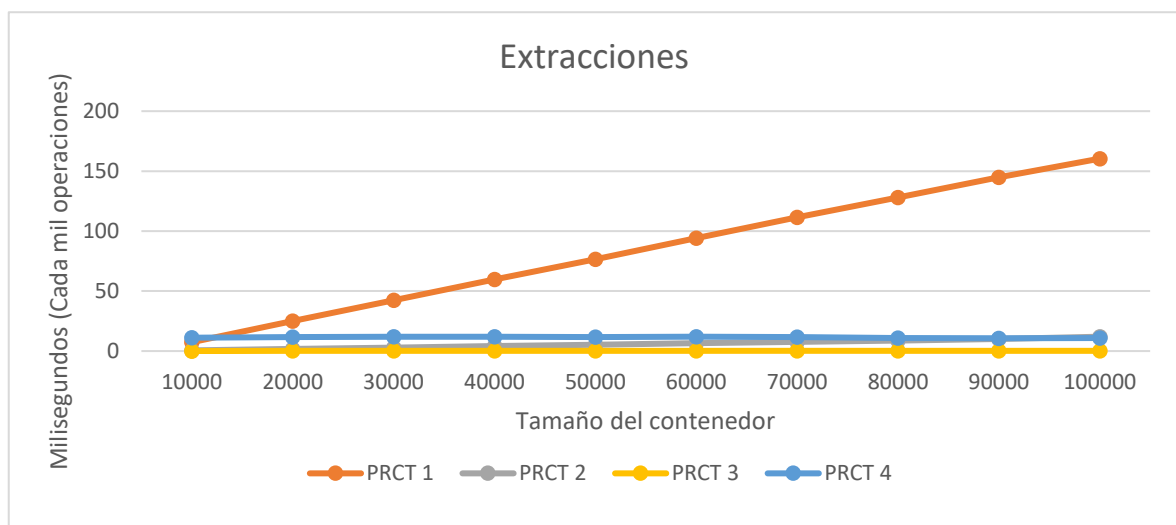


En el árbol B tenemos unos tiempos mayores que los de la practica anterior, ya que en esta práctica cada vez que se quiera insertar un elemento el programa tiene que tomar paginas e ir accediendo a memoria secundaria para leer y escribir dichos elementos, por ello el tiempo es mayor, aunque los tiempos pueden cambiar según el orden del árbol.

Sacamos de conclusiones que no tenemos un contenedor perfecto sino que depende del motivo para el que queramos usarlo. Solo lo practica 1 no necesitaba que el contenedor estuviera ordenado el resto de los contenedores estan ordenados los elementos. Podemos observar que en las insercciones en el arbol binario son las menos costosas.

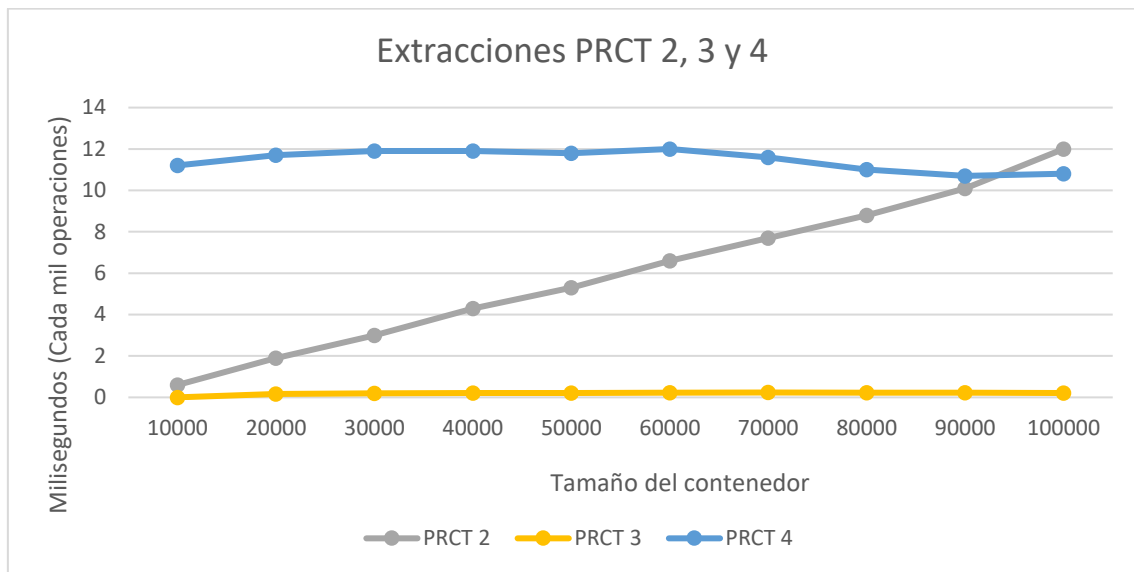
Extracciones

En cuanto a las extracciones encontramos similitudes con las explicaciones de las inserciones, ya que ambas tienen un comportamiento similar. Asimismo, con el árbol B, los tiempos se mantienen igual porque dependen de las paginas tomadas y del orden del árbol para escribir y leer, por sus particiones y rotaciones.



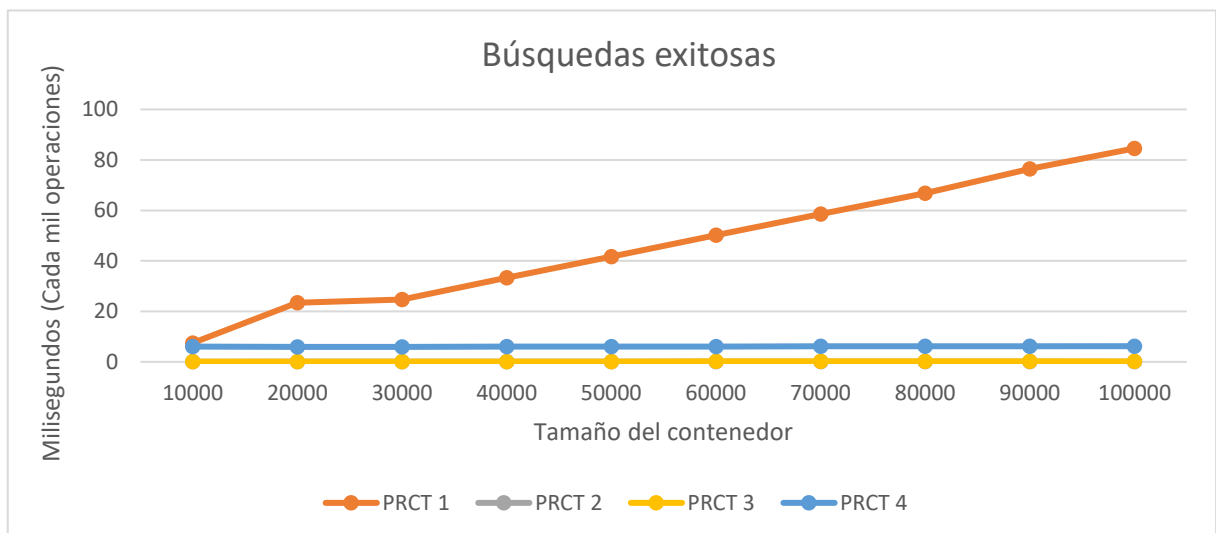
Por ende, en las extracciones, los tiempos se relacionan con las búsquedas exitosas. Cuando hablamos del contenedor lineal (como la lista encadenada) tenemos un promedio de $N/2$ comparaciones y N en el peor caso, pero cuando utilizamos la búsqueda dicotómica las comparaciones pasan al orden $\log_2 N$.

En la practica 2 utilizábamos un vector ordenado por lo que el acceso a los elementos es más rápido y en la practica 3 al utilizar un árbol tenemos que ir desde el nodo raíz hasta que encontremos el elemento.

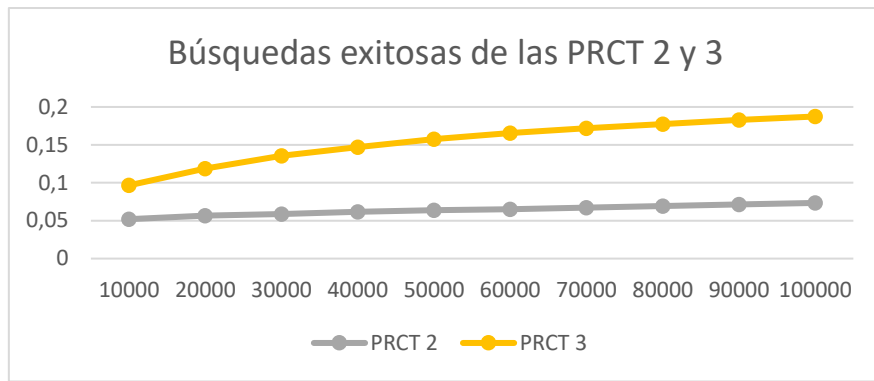


Búsquedas exitosas

En las búsquedas exitosas la lista encadenada presenta unos tiempos más altos en comparación al resto. Esto se debe a que si el elemento no se encuentra al inicio deberá recorrer todos los elementos que este tenga para encontrarlo.

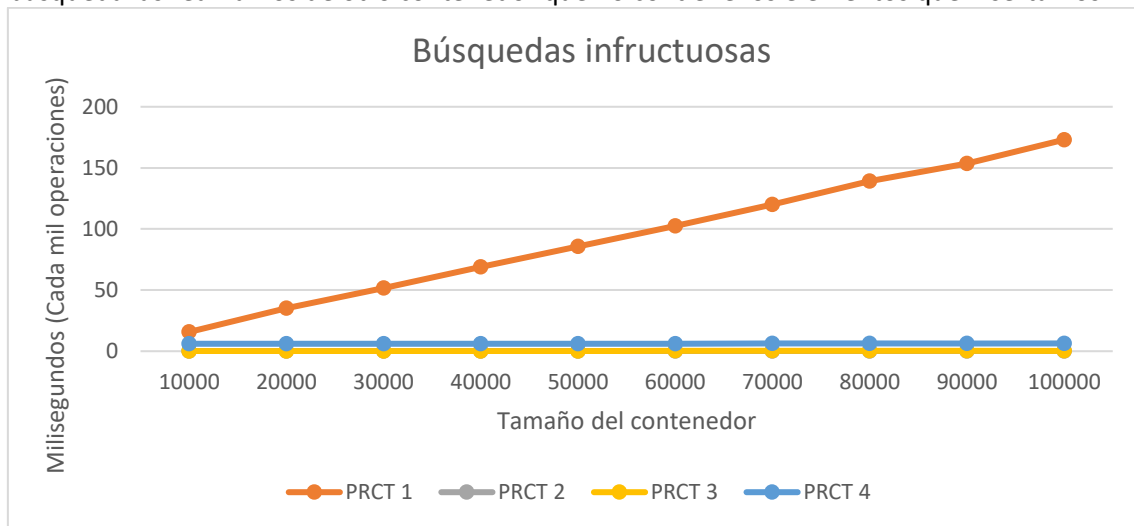


El resto de contenedores, como tenemos la búsqueda dicotómica, los tiempos se reducen bastante en comparación a la primera práctica, debido al cambio de $N/2$ a $\log_2 N$. Donde se reducen las comparaciones que tenemos que hacer para encontrar el elemento en cuestión.

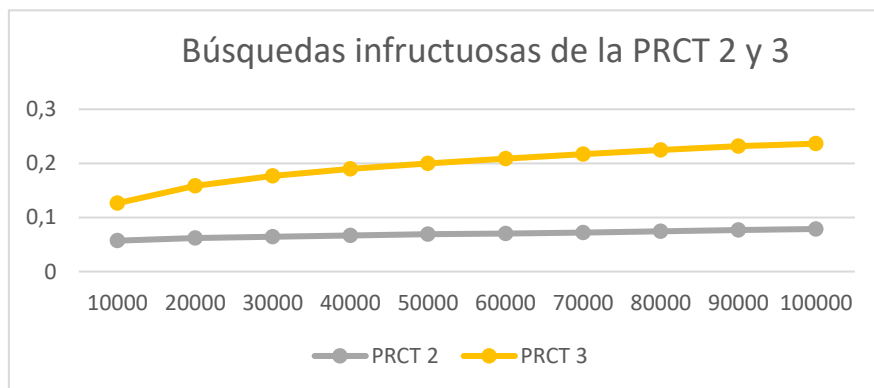


Búsquedas infructuosas

En el caso de las búsquedas infructuosas el contenedor para realizar las operaciones de búsqueda las realizamos de otro contenedor que no contiene los elementos que insertamos.

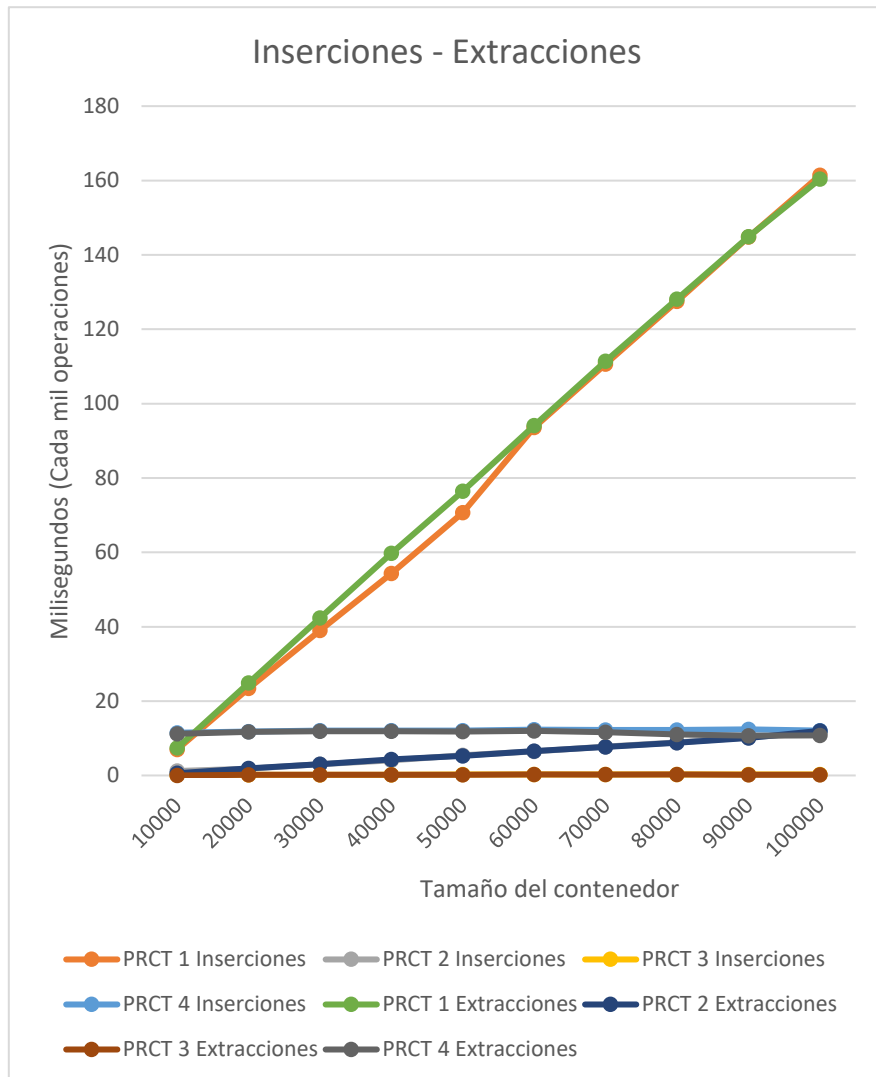


Lo que podemos observar es que los tiempos en este caso son un poco superiores a los de la búsqueda exitosa porque en la lista encadenada compara todos los elementos y no lo encontrara, y en el caso de árbol b hasta llegar a un nodo hoja y no encuentre el valor.



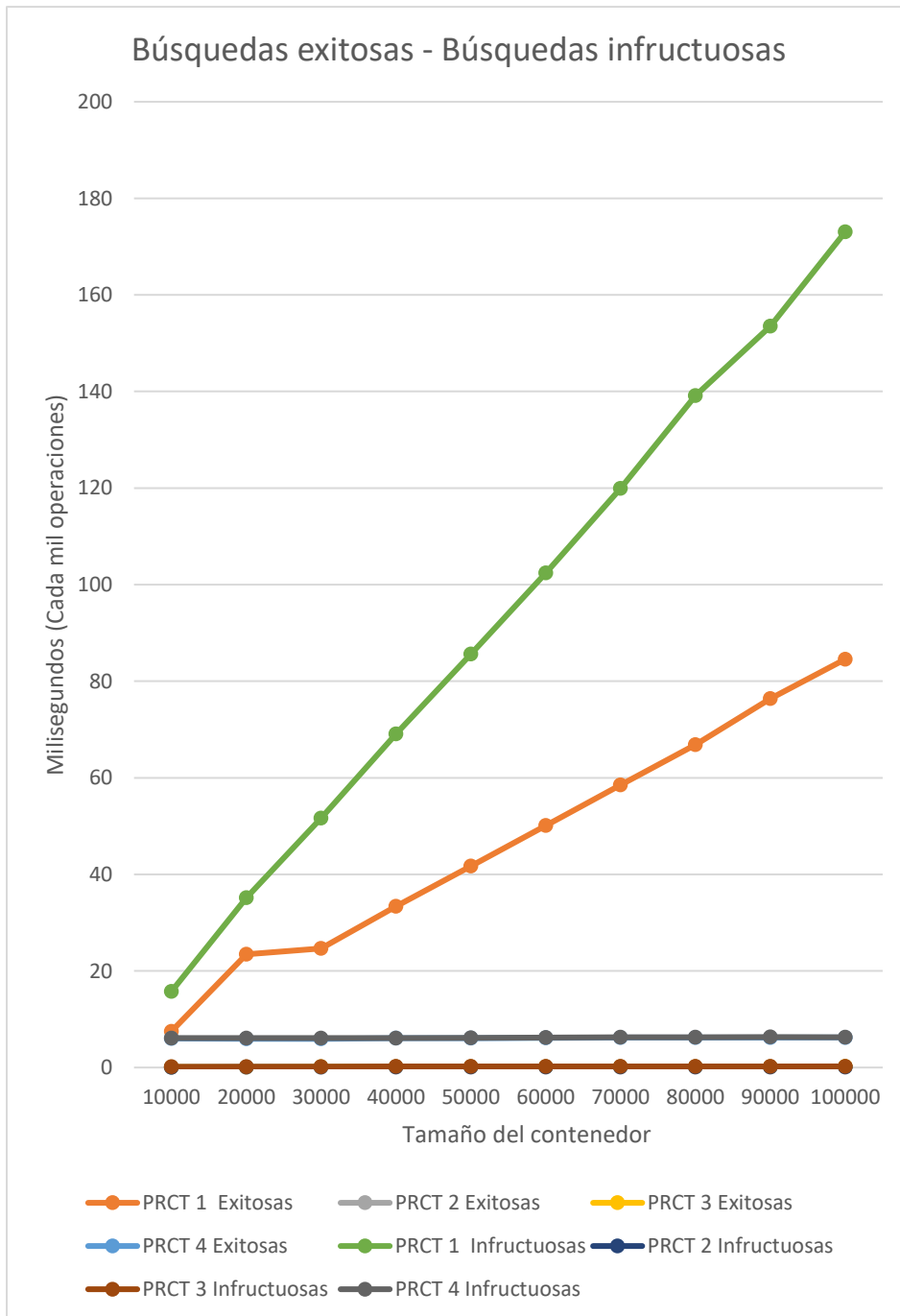
En el árbol B se realiza la búsqueda dicotómica, pero como aquí se tienen que leer los datos de memoria secundaria, los tiempos serán mucho más lentos.

Comparativa entre inserciones y extracciones



En la gráfica anterior podemos ver una comparativa entre extracciones e inserciones de todas las prácticas realizadas. En base a los resultados contemplamos que las extracciones e inserciones en la práctica 1 tiene un tiempo superior al resto. Asimismo el resto de practicas mantienen unos tiempos bastante parejos.

Comparativa entre búsquedas exitosas e infructuosas



Los datos mostrados en la gráfica anterior representan una comparativa entre las búsquedas exitosas y las búsquedas infructuosas como podemos observar es parecida a la anterior de las inserciones y las extracciones, tenemos que la practica 1 sigue teniendo los tiempos más altos que el resto. Como también vemos que los tiempos del resto de prácticas son bastante parejos.

Todas estas conclusiones se deben a que en la práctica 1 recorre todo elemento a elemento por lo que el tiempo aumenta a medida que el número de elementos es mayor

Comparativa del Árbol B en diferentes órdenes

A la vista de todos los resultados obtenidos hemos decidido que el contenedor del Árbol B (orden 55), es el más eficiente, ya que los tiempos son bajos. Como podemos contemplar en la gráfica de todos los órdenes del Árbol B observamos que el de orden 55 mantiene unos tiempos de ejecución inferiores al resto.

