

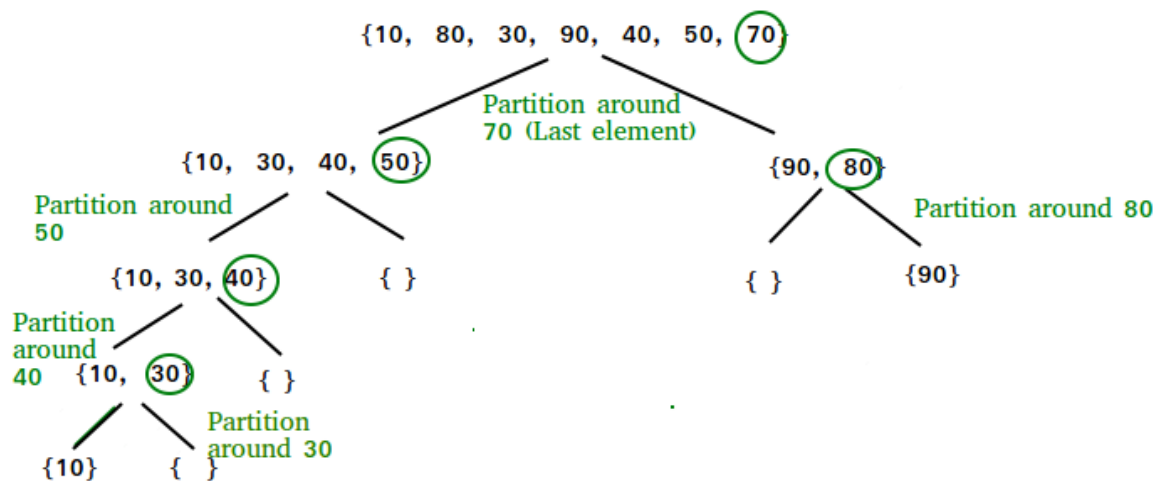
Act 1.3 - Actividad Integral de Conceptos Básicos y Algoritmos Fundamentales

En la actividad 1.3 se utilizó un algoritmo de búsqueda secuencial y para ordenamiento se utilizó el Quicksort.

Quicksort:

El algoritmo de ordenamiento Quicksort es un algoritmo que se basa en la técnica de dividir y conquistar el problema, selecciona un elemento como pivote y particiona el arreglo alrededor del pivote, este pivote se puede escoger de distintas maneras, nosotros escogimos el medio.

La parte central del algoritmo es la partición, que parte el algoritmo alrededor del pivote y acomoda a los elementos menores del pivote de un arreglo y a los elementos mayores en otro arreglo, después particiona las particiones y repite el proceso hasta que todo el arreglo esta ordenado.



La complejidad de este algoritmo es $O(n^2)$ en el peor de los casos y de $O(n \log n)$ en el caso promedio.

Utilizamos este algoritmo en lugar del merge sort no por su velocidad (que es parecida), si no porque su complejidad en espacio de memoria es $O(\log(n))$ y la bitácora tenía cerca de 16000 registros, por lo que se nos hizo una buena opción por esto mismo.

Búsqueda secuencial:

La búsqueda secuencial consiste en iterar a través de un arreglo elemento por elemento hasta encontrar el elemento buscado en caso de no ser encontrado el elemento se retorna un -1, por lo que la complejidad del algoritmo es de $O(n)$.

Búsqueda binaria:

El algoritmo de búsqueda binaria solo funciona con arreglos ordenados y consiste en seleccionar el punto medio del arreglo que llamaremos pivote y revisar 2 condiciones:

- ¿Es el elemento que buscamos? Si es el caso, lo regresamos
- ¿Es mayor o menor al elemento que buscamos?

Si el pivote es menor quiere decir que el elemento que buscamos está del lado derecho de este mismo, así que nos deshacemos de la mitad menor y seleccionamos el punto medio del arreglo de elementos mayores al pivote, si el pivote es mayor hacemos lo mismo, pero con el lado menor de los elementos, repetimos esto hasta que el pivote sea el elemento que buscamos.

La complejidad de este algoritmo es de $O(\log n)$

¿Por qué no utilizamos la búsqueda binaria?

La razón por la que no utilizamos la búsqueda binaria es porque la búsqueda binaria busca un valor exacto en todo el arreglo y la bitácora tiene distintos registros por día, por lo que la búsqueda binaria nos regresaba el índice cuando encontraba cualquier registro del día indicado, por lo que decidimos utilizar una búsqueda secuencial en su lugar, un poco modificada, pero el principio de la búsqueda secuencial es el mismo.

Conclusión:

Con esta actividad pudimos observar la funcionalidad que tiene el combinar distintos algoritmos para resolver un mismo problema, esto porque un problema puede ser dividido en más problemas pequeños, y cada algoritmo se encarga de resolver uno de estos problemas pequeños que al final forman el problema completo.