**TALLER 2 ACI**

**ANÁLISIS DE COMPLEJIDAD:**

Complejidad del algoritmo del punto b:

* La inicialización de los recursos consume 6 instrucciones en total.
* Para ordenar y mostrar el arreglo se necesitan = 6 + (29 \* 20) – 7
* La función guardar toma en total 14 instrucciones en ejecutarse.
* Para mostrar el valor más grande actual y su posición se requieren 18 instrucciones.
* Para borrar el elemento mayor del primer arreglo son necesarias 6 instrucciones.
* Luego, para mostrar el arreglo ordenado se utilizan = 26 instrucciones + 2 para terminar el programa

Por lo tanto, para ejecutar el programa con el arreglo dado, se requieren un total de *651 instrucciones*.

Para ordenar un vector que esté completamente ordenado como el siguiente:  
[20, 19, 18, 17, 16, 15, 14, 13, 12, 11, 10, 9, 8, 7, 6, 5, 4, 3, 2, 1]

El programa debería hacer algunas iteraciones extra, en total las instrucciones requeridas para ordenar y mostrar el arreglo ascenderían a = 6 + (37 \* 20) – 1.

O sea que, sumando el resto de las instrucciones, quedarían un total de = 817 instrucciones necesarias para ordenar un arreglo de 20 elementos de mayor a menor en el peor caso.

Para ordenar un vector de 100 elementos (en el peor caso) se necesitarían aproximadamente = *4,085 instrucciones.*

Para un vector de 1000 elementos = *40, 850 instrucciones.*

Para un vector de 1,000,000 elementos = *40,850,000 instrucciones.*