

2.-

Para la búsqueda lineal el peor caso es que el elemento que se busca no se encuentre, por lo que hay que revisar todas las posiciones. Sólo necesita guardar en memoria el arreglo y el número a buscar. Es de complejidad en tiempo $O(n)$ y complejidad en espacio $O(1)$.

El peor caso para la búsqueda binaria es dividir el arreglo por 2 hasta que solo quede un elemento, por lo que su complejidad en tiempo es $O(\log(n))$. Para la iterativa se necesita guardar los límites, el elemento del medio, el elemento a buscar y el arreglo por lo tanto es de complejidad $O(1)$ en memoria. Para la recursiva el número de llamadas a la función queda acotado por $\log(n)$, por lo que es de complejidad $O(\log(n))$ en memoria.

3.-

La mejor implementación es la binaria iterativa, por que es la más acotada tanto en tiempo como espacio.