Assume we have an array $A[1..n]$ of $n$ distinct integers

1. Write an algorithm to check whether an integer $x$, exists in $A$. It returns the index of the element with value $x$, or 0 if not found. Do not make any assumptions regarding whether $A$ is sorted.

2. What is the complexity of the algorithm in 1?

3. What is the best case and worst case runtime?

4. Assume the array is sorted (in ascending order), write an algorithm to check whether an integer $x$ exists in an array $A$.

5. Will it make any difference if the array is sorted increasingly or decreasingly? Justify.

6. Make a search on the web and find the best sorting algorithm. What is the name of that algorithm? What is its complexity?

7. If we use the algorithm you found in 6 to sort $A$ then use the algorithm you wrote in 4, will the resulting algorithm be better or worth than the algorithm you wrote in 1? Justify.