Question-1

We can use decreas-by-factor algorithm for this problem. Solve this problem by dividing size by 2 each time
For example $n=16$ then $n=8$, $n=4$, ...

Complexity Analyze

$T_{Best}(n) = \Theta(1)$

$T_{worst}(n) = T(n/2) + 1$

Master theorem
$a=1, b=2, f(n)=1, d=0 \rightarrow a = b^d \rightarrow T_w(n) \in \Theta(n^0 \cdot \log n)$
$T_w(n) \in \Theta(\log n)$

$T(n) \in O(\log n)$

Question-2

I used quick sort for Question-2 and quick select for Question-3 and both algorith use lomoto partition.
In this part quick sort using for divide and conquer solution and first element will be worst last element will be best in result array.

Complexity Analyze

$T_{best}(n) = 2T(n/2) + n$
$a=2, b=2, f(n)=n, d=1 \rightarrow a=b^d \rightarrow T_b(n) \in \Theta(n^1 \cdot \log n)$

$T_{worst}(n) = T(n-1) + n$

$T(n) = T(n-1) + n$
$T(n-1) = T(n-2) + n-1$
$T(n-2) = T(n-3) + n-2$

$T(2) = 2 + T(1)$
$T(1) = 0$

$n + n-1 + n-2 \dots + 2 + 0$
$= \frac{n(n)}{2} = n^2 \rightarrow T_w(n) = \Theta(n^2)$

$T_{best} = \Theta(n \cdot \log n)$
$T_{worst} = \Theta(n^2)$
$T(n) = O(n^2)$

Question - 3

In problem definiton, we couldn't make sorting therefore I used quick select algorithm for solution with using the definiton covered in the lesson. It find success rate of the first meaningful k^{th} experiment.

Complexity Analyze

$T_{best} = T_{worst} = T(n-1) + n$

$T(n) \in \Theta(n^2) \rightarrow$ I solved in Question-2

Question-4

One of the topics we covered in the course was counting inversions. I thought I could do this using a similar operation to merge sort and using this algorithm I found inversion pairs.

Complexity Analyte

$T(n) = 2T(n/2) + n$

Master Theorem

$a=2, b=2, f(n)=n, d=1 \rightarrow a=b^d \rightarrow T(n) \in \Theta(n^d \cdot \log n) \in \Theta(n \log n)$

Question-5

For brute force algorithm = Multiply base by exponent using loop

$$T(n) = \sum_{i=1}^{n} 1 = n \rightarrow T(n) \in \Theta(n)$$

For divide and conquer = Divide the exponent recursively and create sub problems and multipy base each time.

$T(n) = T(n/2) + 1$

Master theorem

$a=1, b=2, f(n)=1, d=0 \rightarrow a=b^d \rightarrow T(n) = \Theta(n^0 \log n) = \Theta(\log n)$

$T(n) \in \Theta(\log n)$