

Assignment 2 Problem 4

Let A be an array of n distinct integers. An *inversion* is a pair of indices (i, j) such that $i < j$ and $A[i] > A[j]$.

- a) Determine the maximum number and minimum number of inversions in an array of n distinct integers. Characterize what the arrays that attain these maxima and minima look like.
- b) Let A be an array of n distinct integers chosen at random. Given a pair of distinct indices $i < j$, show that the probability that (i, j) is an inversion is $1/2$.
- c) Let A be an array of n distinct integers chosen at random. Determine the expected number of inversions in an array of n distinct integers.
Note: You may use the result in part (b) even if you did not prove it.
- d) Suppose a sorting algorithm is only allowed to exchange adjacent elements $(i, i + 1)$. Show that its worst-case and average-case complexity is $\Omega(n^2)$.
Note: You may use the results from the previous parts even if you did not prove them.
Hint: How many inversions does an exchange of adjacent elements remove?

For parts (b-d), all permutations of the input-numbers are equally likely.

Solution:

- a) For the minimum case, the number of inversions is 0, in this case, array is already in the ascending order.
For the maximum case, the number of inversions is $\frac{n(n-1)}{2}$, in this case, array is already in the descending order, and all pairs are inversions.
- b) Given i and j (which can be arbitrary), we only focus on these two positions. It's obvious to notice that there exists a bijection between $A[i] = x, A[j] = y$ with $A[i] = y, A[j] = x$. The bijection can be expressed as only the entries on i th and j th position exchanges and others remain the same. Therefore, for all possibilities, half of them are inversion and another half are not. Therefore, the probability that (i, j) is an inversion is $\frac{1}{2}$.
- c) $E(\text{numbers of inversions}) = \frac{\text{number of inversion pairs}}{\text{total number of pairs}} * \text{number of pairs in the array}$
which implies $E(\#inversions) = \frac{1}{2} * \frac{n(n-1)}{2} = \frac{n(n-1)}{4}$ since the probability that (i, j) is an inversion is one half.
- d) Since the input are random and the permutations of the input-numbers are equally likely, we consider to convert the input in to the sorting integers $0, \dots, n-1$ which means the least integer becomes 0 and the largest integer becomes $n-1$. Consider position i and $i+1$. In the worst-case, they are obviously inversion. After we exchange them, in

the worst-case, which actually only remove 1 inversion. In the descending order array, by our assumption, the array should be $[n-1, n-2, \dots, 1, 0]$. Say i is not the last element, after we exchange $A[i]$ and $A[i+1]$, then the array becomes $[n-1, n-2, \dots, n-i, n-i-2, n-i-1, n-i-3, n-i-4, \dots, 1, 0]$. Then, we can find that, the elements from $A[0]$ to $A[i-1]$ are still greater than the elements in $A[i]$ and $A[i+1]$, similarly, the elements from $A[i+2]$ to $A[n-1]$ are still less than the elements in $A[i]$ and $A[i+1]$. Therefore, we need to do $\frac{n(n-1)}{2}$ times of exchanging. Which implies that the worst-case complexity is $\Omega(n^2)$. For the average case, similarly, each time of exchanging only removes one inversion, and since the expected value of the number of inversions are given in part b, therefore, we can conclude that the average number of exchange we need to do is $\frac{n(n-1)}{4}$, and thus, this implies that the average-case complexity is also $\Omega(n^2)$.