1

When the array is in descending order, all the number pairs are reverse order. Therefore, it has the maximum mixed-up-ness score, which is equal to the number of pairs. If the size of an array is 16, the number of pair is $C_{16}^{2} = 120$.

2

In worst case, the runtime is $O(n^2)$. In each iteration of the outer for loop, the inner for loop iterates n-i times to check whether two items are in reverse order. And the running time of the outer for loop is n. Hence, the worst-case running time of this brute-force algorithm is $O(n^2)$.

3

$$T(n) = 2T\left(\frac{n}{2}\right) + O(n)$$

4

$$T(n) = 2T\left(\frac{n}{2}\right) + cn = 2\left(2T\left(\frac{n}{4}\right) + \frac{cn}{2}\right) + cn$$

$$= 4T\left(\frac{n}{4}\right) + 2cn = 4\left(2T\left(\frac{n}{8}\right) + \frac{cn}{4}\right) + 2cn$$

$$= 8T\left(\frac{n}{8}\right) + 3cn = \ldots\ldots = 2^k T\left(\frac{n}{2^k}\right) + kcn$$

$$When\ n = 2^k, k = \log(n).\ \ T\left(\frac{n}{2^k}\right) = T(1)\ is\ a\ constant.$$

$$T(n) = nT(1) + cnlog(n) = cnlog(n) + cn$$

$$Ignoring\ the\ lower - order\ term, \qquad T(n) = O(nlog(n))$$

5

$$T(n) = aT\left(\frac{n}{b}\right) + O\left(n^d\right),$$

$$T(n) = \{O\left(n^d\right) \qquad\qquad if\ d > log_b(a)$$

$$O\left(n^d \log(n)\right)\ \ if\ d = log_b(a)$$

$$O\left(n^{log_b(a)}\right)\ \ if\ d < log_b(a)$$

$$a = 2, b = 2, d = 1, \qquad log_b(a) = 1 = d,$$

$$Therefore, \qquad T(n) = O\left(n^d log(n)\right) = O(nlog(n))$$