A4

(1)
```
CINV (arr)
    n = arr.size
    if ( n < 2)
        return {0; arr}

    if ( n == 2)
        return {arr[0] < arr[1]; arr.sorted}

    m = n/2
    left = CINV(arr[0...m))
    right = CINV(arr[m...n))

    merged = combine(left.arr, right.arr)
    ans = left.ans + right.ans + merged.ans
    return {ans, merged.arr}
```

Divide:

Если длина массива < 2, то возвращаем 0 инверсий

Если длина массива = 2, то проверить условие

$$\begin{tikzcd} & 1 \\ (i<j \,\&\&\, a_i > a_j) \arrow[ur, "True"] \arrow[dr, "False"'] & \\ & 0 \end{tikzcd}$$

Если длина массива > 2, то делим массив напополам.

Conquer: Рекурсивно вызываем CINV для 2 частей массива

Combine: Сливаем 2 части в один массив, используя метод 2 указателей (параллельно считая инверсии)

$T(n) = 2T(\frac{n}{2}) + O(n+m)$ (два указателя)

Утверждение: $O(n+m) = O(n)$

Док-во: При делении массива четной длины $\overset{\text{длины}}{m} = \overset{\text{частей}}{n}$, иначе $m = n + \underset{\text{очень малый вклад}}{1}$

$\Rightarrow O(n+m) = O(n) = O(m)$

$\Rightarrow T(n) = 2T(n/2) + O(n) \overset{\text{Мастер теорема}}{\Longrightarrow} k=1,\ a=2,\ b=2 \rightarrow 1 = \log_2 2 \Rightarrow T(n) = O(n \cdot \log n)$

(2) Нужно добавить еще один цикл O(n), где будем проверять

left.arr[i] > 2·right[j] => count_inv = left.arr.size − i

В одном цикле будет сортировка

В другом цикле будет поиск инверсий

Сложность будет такой же $T(n) = O(n \log n)$

(т.к. от коэф. не зависит)