1. Искомый элемент под индексом от i до j не включительно. j = i + 1 ко входу в цикл, значит minindex = i действительно наименьший среди одного элемента i. На каждой итерации проверяется arr[j] и обновляется minindex, если arr[j] меньше текущего минимума => arr[minindex] - минимум от i до j и инвариант перед следующей итерацией вновь выполняется.

После выхода из цикла имеем подмассив от i до j (n), а в minindex за счет условия лежит индекс искомого минимума.

2. Так же, до входа в цикл i = 0 и первые 0 элементов образуют массив из 0 минимальных. На i = k во внутреннем цикле находится k-ый минимум, который меняется местами с i-ым, образуя подмассив из k элементов минимумов, отсортированных за счет того, что остальные были выбраны из большего числа элементов, чем k-ый. В конце весь список отсортирован и i = n элементов в нём.

3. Проверка выполнена во время объяснения инвариантов выше, с описанием состояния в каждой из итераций и фаз цикла.