

For in-place sorting algorithm that uses $O(1)$ extra space and has worst-case time complexity of $O(N^2)$

```
for (int i=0; i<size+1; i++){           // first for loop

    for (int j=0; j<size+1; i++) {      // second for loop

        if (a[i] > a[j+1]){              //create temp for in-place
sorting algorithm
            int temp =a[j];
            a[j] = j+1;

        }
    }
}
```

$O(1)$ is unstable because the order is not in linear order and recursive, the program is in worst time complexity.

Best-case time complexity of algorithm would be $O(n)$ for in-place sorting algorithm since the loops would not be done twice and would single and linear.

Worst case time-complexity would be $O(N^2)$ as it requires 2 loops, more order checking and un-stability

For exam sort function the array highlights shows the temp part as well for being the bases of the sorted and size of the array.