

## CSDS 233 Spring Session 13

SI Leader: Jakob Danninger

4/4/2023

Disclosure: This is a supplement to class, not a replacement. This should not be your only study activity for exams, it should aid you in studying. I do not have access to the actual exam so questions here will differ from those on the exam.

### Session Objectives:

- 1) Understand how the time complexity of selection and insertion sort
- 2) Be able to implement selection and Insertion sort

### Questions

- 1) Draw the following array after using selection sort after every pass (small to big)

8	6	3	4
---	---	---	---

- 2) Draw the following array using selections sort after every pass, does selection sort keep going even if it is sorted? (small to big)

1	4	5	8
---	---	---	---

- 3) Draw the following array after Insertion sort at every pass

2	4	3	1	0
---	---	---	---	---

4) What happens if you use insertion sort but it is already sorted?

5) Fill in the table

	Worst Case	Best Case
Insertion		
Selecton		

6) Would it be better to use insertion or selection sort for a totally random array?

7) Would it be better to use insertion or selection sort for a sorted array?

8) Would it be better to use insertion or selection sort for a reverse order array?

9) Fill in the blanks

```
. public void insertionSort (int[] arr) {  
    for (int i = 1; i < arr.length; i++) {  
        int toInsert = arr[i]; int j;  
        for (j = i; j > 0 && toInsert  arr[j-1]; j--)  
            arr[j] = ;  
        arr[j] = toInsert; }  
    }
```

### Test Like question

Code a selection sort method that takes a given array of integers and returns the sorted array of integers. There are no special methods available.

```
Public int[] selectionSort(int[] arr){
```

```
}
```

Next session 11/15 covering bubble and quicksort

