CS5010 E1

1. Selection Sort

   source code:

```
public static class SelectionSort{
    public void selectionSort(int[] arr){
        int i, j;
        int minIndex = 0;
        for (i = 0; i < arr.length; i++){
            minIndex = i;
            for (j = i; j < arr.length; j++)
{
                if (arr[j] < arr[minIndex])
{minIndex = j;}
            }
            if (minIndex != i){
                swap(arr, i, minIndex);
            }
        }
    }

    private static void swap(int[] arr, int
indx1, int indx2) {
        int temp = arr[indx1];
        arr[indx1] = arr[indx2];
        arr[indx2] = temp;
    }
}

```

```
 0  1  2  3  4
[2, 5, 3, 1, 4]
```

only the first iteration is detailed explained

list	i	j	minIndex	i==minIndex
[2, 5, 3, 1, 4]	0	0	0	true
[2, 5, 3, 1, 4]	0	1	0	true
[2, 5, 3, 1, 4]	0	2	0	true
[2, 5, 3, 1, 4]	0	3	3	false
[2, 5, 3, 1, 4]	0	4	3	false
[1, 5, 3, 2, 4] swapped	1	1	1	true
...			...	...
[1, 5, 3, 2, 4]	1	3	3	false
[1, 2, 3, 5, 4] swapped	2	2	2	true
...			...	...
[1, 2, 3, 5, 4] no swap	3	3	3	true
[1, 2, 3, 5, 4]	3	4	4	false

list	i	j	minIndex	i==minIndex
[1, 2, 3, 4, 5] sorted	4	4	4	true

2. Insertion Sort

```
public static class InsertionSort{
    public void insertionSort(int[] arr){
        int i, j;
        for (i = 0; i < arr.length; i++){
            for (j = i; j > 0; j--){
                if (arr[j] < arr[j-1])
{this.swap(arr, j, j-1);}
            }
        }
    }

    private static void swap(int[] arr, int
indx1, int indx2) {
        int temp = arr[indx1];
        arr[indx1] = arr[indx2];
        arr[indx2] = temp;
    }
}
```

```
[2, 5, 3, 1, 4]
```

list	i	j
[2, 5, 3, 1, 4]	0	0
[2, 5, 3, 1, 4]	1	1
[2, 5, 3, 1, 4]	1	0
[2, 5, 3, 1, 4]	2	2
[2, 5, 3, 1, 4]	2	1
[2, 3, 5, 1, 4] swapped	2	0
[2, 3, 5, 1, 4]	3	3
[2, 3, 5, 1, 4]	3	2
[2, 3, 1, 5, 4] swapped	3	1
[2, 1, 3, 5, 4] swapped	3	0
[1, 2, 3, 5, 4] swapped	4	4
[1, 2, 3, 4, 5] swapped	4	3