Temporal and Spatial Complexity Analysis

Selection Sort

```
Public void sort(int arr[])    {
        int n = arr.length;
        for (int i = 0; i < n-1; i++)        {
            int min_idx = i;
            for (int j = i+1; j < n; j++) {
                if (arr[j] < arr[min_idx]) {
                    min_idx = j;
        }
}
            int temp = arr[min_idx];
            arr[min_idx] = arr[i];
            arr[i] = temp;
        }
    }
```

Temporal Complexity

# Line	*# Times it repeats*
2	1
3	N
4	N-1
5	((N(N+1))/2)+N
6	((N(N+1))/2)
7	((N(N+1))/2)
10	N-1
11	N-1
12	N-1

$$T(A) = C2 + C3N + C4(N-1) + C10(N-1) + C11(N-1) + C12(N-1) + C5\left(\frac{N(N+1)}{2} + N\right) + C6\left(\frac{N(N+1)}{2}\right) + C7\left(\frac{N(N+1)}{2}\right)$$

$$T(A) = C2 + C3N + C4N - C4 + C10N - C10 + C11N - C11 + C12N - C12 + C5\left(\frac{N(N+1)}{2} + N\right) + (C6 + C7)\left(\frac{N(N+1)}{2}\right)$$

$$T(A) = -(-C2 + C4 + C10 + C11 + C12) + N(C3 + C4 + C10 + C11 + C12) + C5\left(\frac{N(N+1)}{2} + N\right) + (C6 + C7)\left(\frac{N(N+1)}{2}\right)$$

$$T(A) = -A + N(B) + C\left(\frac{N(N+1)}{2} + N\right) + D\left(\frac{N(N+1)}{2}\right)$$

$$T(A) = C\left(\frac{N^2 + 3N}{2}\right) + D\left(\frac{N^2 + N}{2}\right) - A + NB$$

$$T(A) = \frac{N^2 + 3N}{2} + \frac{N^2 + N}{2}$$

$$T(A) = \frac{2N^2 + 4N}{2}$$
$$T(A) = N^2 + 2N$$

Temporal Complexity O (N^2)

Spatial Complexity

Type	Variable	Size of 1 Atomic Value	Number of Atomic Values
Input	Int[] arr	32 bits	N
Tempt	Int n	32 bits	1
	Int i	32 bits	1
	Int min_indx	32 bits	1
	Int j	32 bits	1
	Int temp	32 bits	1

Total spatial complexity = Input+Tempt+Output= n+5= θ(n)
Auxiliary spatial complexity =1+1+1+1+1= (1)
Auxiliary spatial complexity tempt+ output =1+1+1+1+1= (1)

Insertion Sort

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

Temporal Complexity

# Line	***# Times it repeats***
2	1
3	N
4	N-1
5	N-1
6	((N(N+1))/2)+N
7	((N(N+1))/2)
8	((N(N+1))/2)
10	N-1

$$T(A) = C2 + C3N + C4(N-1) + C5(N-1) + C10(N-1) + C6\left(\frac{N(N+1)}{2} + N\right) + C7\left(\frac{N(N+1)}{2}\right) + C8\left(\frac{N(N+1)}{2}\right)$$

$$T(A) = C2 + C3N + C4N - C4 + C5N - C5 + C10N - C10 + C6\left(\frac{N(N+1)}{2} + N\right) + (C7 + C8)\left(\frac{N(N+1)}{2}\right)$$

$$T(A) = -(-C2 + C4 + C5 + C10) + N(C3 + C4 + C5 + C10) + C6\left(\frac{N(N+1)}{2} + N\right) + (C7 + C8)\left(\frac{N(N+1)}{2}\right)$$

$$T(A) = -A + N(B) + C\left(\frac{N(N+1)}{2} + N\right) + D\left(\frac{N(N+1)}{2}\right)$$

$$T(A) = C\left(\frac{N^2+3N}{2}\right) + D\left(\frac{N^2+N}{2}\right) - A + NB$$

$$T(A) = \frac{N^2 + 3N}{2} + \frac{N^2 + N}{2}$$
$$T(A) = \frac{2N^2 + 4N}{2}$$
$$T(A) = N^2 + 2N$$

Temporal Complexity O (N^2)

Spatial Complexity

Type	Variable	Size of 1 Atomic Value	Number of Atomic Values
Input	Int[] arr	32 bits	N
Tempt	Int n	32 bits	1
	Int i	32 bits	1
	Int key	32 bits	1
	Int j	32 bits	1

Total spatial complexity = Input+Tempt+Output= n+4= θ(n)
Auxiliary spatial complexity =1+1+1+1= (1)
Auxiliary spatial complexity tempt+ output =1+1+1+1= (1)