DATA STRUCTURES AND ALGORITHMS

ASSIGNMENT NO. 4

QUESTION NO. 1

In Place Sorting Algorithms-

An in-place algorithm is an algorithm that does not need an extra space and produces an output in the same memory that contains the data by transforming the input 'in-place'. However, a small constant extra space used for variables is allowed.

CONTINUED….

Out Place Sorting Algorithms-

An algorithm that is not in-place is called a not-in-place or out-of-place algorithm. Unlike an in-place algorithm, the extra space used by an out-of-place algorithm depends on the input size.

The standard merge sort algorithm is an example of out-of-place algorithm as it requires O(n) extra space for merging. The merging can be done in-place, but it increases the time complexity of the sorting routine.

DIFFERENCE BETWEEN IN PLACE AND OUT PLACE ALGORITHMS

An in-place sorting algorithm sorts the elements in place: that is, it needs only O(1) extra space. An out-of-place sorting algorithm needs extra space to put the elements in as it's sorting them. Usually this means O(n) extra space.

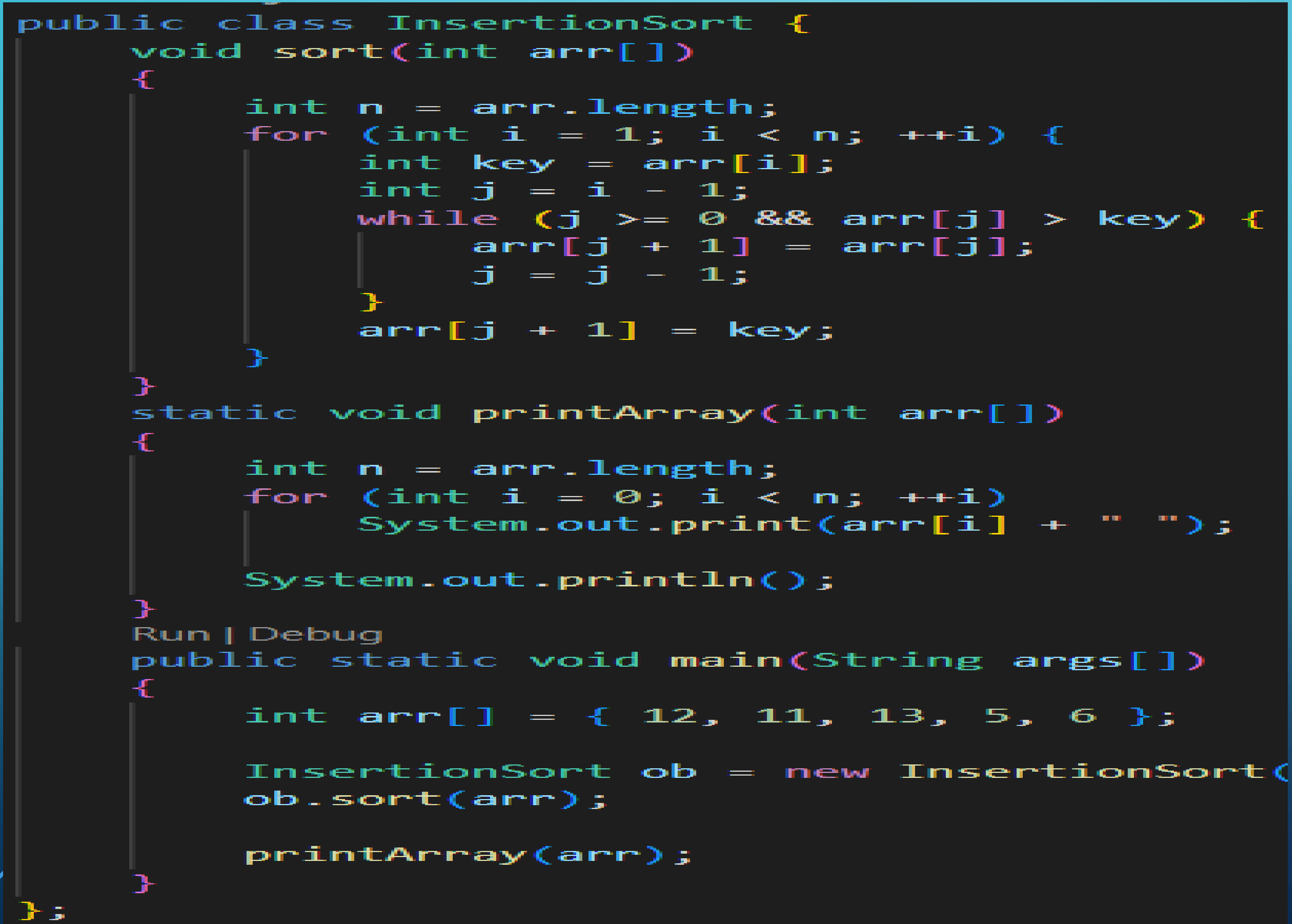
in some sorting algorithms, the program requires space which is more than or equal to the elements being sorted. **Sorting which uses equal or more space is called not-in-place sorting**

QUESTION NO. 2

Implementation of Insertion Sort in both (In Place and Out Place Manner)

At each iteration, insertion sort removes one element from the input data, finds the location it belongs within the sorted list, and inserts it there. It repeats until no input elements remain. **Sorting is typically done in-place, by iterating up the array, growing the sorted list behind it.**

CODE FOR THE SAME



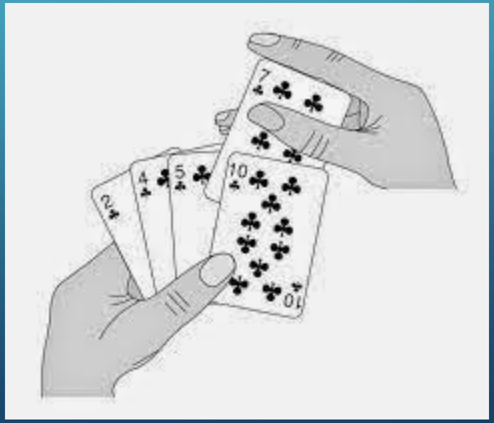
```
public class InsertionSort {
    void sort(int arr[])
    {
        int n = arr.length;
        for (int i = 1; i < n; ++i) {
            int key = arr[i];
            int j = i - 1;
            while (j >= 0 && arr[j] > key) {
                arr[j + 1] = arr[j];
                j = j - 1;
            }
            arr[j + 1] = key;
        }
    }
    static void printArray(int arr[])
    {
        int n = arr.length;
        for (int i = 0; i < n; ++i)
            System.out.print(arr[i] + " ");

        System.out.println();
    }
    Run | Debug
    public static void main(String args[])
    {
        int arr[] = { 12, 11, 13, 5, 6 };

        InsertionSort ob = new InsertionSort(
        ob.sort(arr);

        printArray(arr);
    }
};
```

QUESTION NO. 3

Practical Applications of In-Place and Out-Place Sorting Algorithms-

These are **simple sorting algorithms that works similar to the way you sort playing cards in your hands**. The array is virtually split into a sorted and an unsorted part. Values from the unsorted part are picked and placed at the correct position in the sorted part.

One more real-world example of these algorithms is how **tailors arrange shirts in a cupboard**, they always keep them in sorted order of size and thus insert new shirts at the right position very quickly by moving other shirts forward to keep the right place for a new shirt.



CONTINUED….

The contact list in your phone is sorted, which means you can easily access your desired contact from your phone since the data is arranged in that manner for you. In other words, “it is sorted”.

While shopping on flip kart or amazon, you sort items based on your choice, that is, price low to high or high to low.

The app docker on your phone is an easily relate-able example.

THANK YOU….

Efforts By-

Devank garg

SID-21104098