BUBBLE SORT

```
class BubbleSort
{
	public static void main(String[] args)
	{
		int a[]=new int[args.length];
		for(int i=0;i<a.length;i++)
		{
			a[i]=Integer.parseInt(args[i]);
		}
		System.out.println("Array Size:"+a.length);
		System.out.println("Entered values");
		for(int i=0;i<a.length;i++)
			System.out.println(a[i]);
		//int b[]=bubbleSortMyLogic(a);
		int b[]=bubbleSort(a);
		System.out.println("Sorted values");
		for(int i=0;i<b.length;i++)
			System.out.println(b[i]);
	}
	static int[] bubbleSortMyLogic(int ar[])
	{
		int temp=0;
		for(int i=0;i<ar.length;i++)
		{
			for(int j=1;j<ar.length;j++)
			{
				if(ar[j-1]>ar[j])
				{
```

```
                    temp=ar[j-1];
                    ar[j-1]=ar[j];
                    ar[j]=temp;
                }
            }
        }
        return ar;
    }
    static int[] bubbleSort(int ar[])
    {
        int t=0;
        for(int i=0;i<ar.length-1;i++)
        {
            for(int j=i+1;j<ar.length;j++)
            {
                if(ar[i]>ar[j])
                {
                    t=ar[i];
                    ar[i]=ar[j];
                    ar[j]=t;
                }
            }
        }
        return ar;
    }
}
```

INSERTION SORT

```
class InsertionSort
{
	public static void main(String[] args)
	{
		int a[]=new int[args.length];
		for(int i=0;i<a.length;i++)
		{
			a[i]=Integer.parseInt(args[i]);
		}
		System.out.println("Array Size:"+a.length);
		System.out.println("Entered values");
		for(int i=0;i<a.length;i++)
			System.out.println(a[i]);
		insertionSort(a);
		//int b[]=insertionSort(a);
		System.out.println("Sorted values");
		for(int i=0;i<a.length;i++)
			System.out.println(a[i]);
	}
	static void insertionSort(int a[])
	{
		int ti=0;
		for(int i=1;i<a.length;i++)
		{
			int temp=a[i];
			int j=i;
			for( ; j>0; j--)
			{
```

```
                    if(temp>a[j-1])
                        a[j]=a[j-1];
                    else
                        break;
                }
                ti=i+j;
                a[j]=temp;
            }
            System.out.println("Number of iterations:"+ti);
            //return a;
        }
}
```

MERGE SORT

```
class MergeSort
{
	static void merge(int a[],int l,int m,int h)
	{
		int nL = m-l+1;
		int nR = h-m;
		int L[] = new int[nL];
		int R[] = new int[nR];
		for(int i=0;i<nL;i++)
			L[i]=a[l+i];
		for(int j=0;j<nR;j++)
			R[j]=a[m+1+j];
		int i=0,j=0,k=l;
		while(i<nL && j<nR)
		{
			if(L[i] <= R[j])
			{
				a[k] = L[i];
				i++;
			}
			else
			{
				a[k]=R[j];
				j++;
			}
			k++;
		}
		while(i<nL)
```

```
        {
            a[k] = L[i];
            i++; k++;
        }
        while(j<nR)
        {
            a[k] = R[j];
            j++; k++;
        }
    }
    static void sort(int a[],int l,int h)
    {
        if(l<h)
        {
            int m = (l+h)/2;
            sort(a,l,m);
            sort(a,m+1,h);
            merge(a,l,m,h);
        }
    }
    public static void main(String args[])
    {
        int a[]={5,4,3,2,1,6,7,8,9};
        sort(a,0,a.length-1);
        for(int i=0;i<a.length;i++)
            System.out.println(a[i]);
    }
}
```

QUICK SORT

```
class QuickSort
{
	static int partition(int a[],int l,int h)
	{
		int i = l-1;
		for(int j=l;j<h;j++)
		{
			if(a[j]<=a[h])
			{
				i++;
				int temp=a[i];
				a[i]=a[j];
				a[j]=temp;
			}
		}
		int temp=a[i+1];
		a[i+1]=a[h];
		a[h]=temp;
		return i+1;
	}
	static void quicksort(int[] a,int l,int h)
	{
		if(l<h)
		{
			int p=partition(a,l,h);
			quicksort(a,l,p-1);
			quicksort(a,p+1,h);
		}
```

```
	}
	public static void main(String args[])
	{
		int a[]={5,4,3,2,1,6,7,8,9};
		quicksort(a,0,a.length-1);
		for(int i=0;i<a.length;i++)
			System.out.println(a[i]);
	}
}
```

SELECTION SORT

```
class SelectionSort
{
	public static void main(String[] args)
	{
		int a[]=new int[args.length];
		for(int i=0;i<a.length;i++)
		{
			a[i]=Integer.parseInt(args[i]);
		}
		System.out.println("Array Size:"+a.length);
		System.out.println("Entered values");
		for(int i=0;i<a.length;i++)
			System.out.println(a[i]);
		//int b[]=bubbleSortMyLogic(a);
		int b[]=selectionSort(a);
		System.out.println("Sorted values");
		for(int i=0;i<b.length;i++)
			System.out.println(b[i]);
	}
```

```
    static int[] selectionSort(int ar[])
    {
        int min=0;
        for(int i=0;i<ar.length;i++)
        {
            min=ar[i];
            int k=i;
            for(int j=i+1;j<ar.length;j++)
            {
                if(ar[j]>min)
                {
                    min=ar[j];
                    k=j;
                }
            }
            ar[k]=ar[i];
            ar[i]=min;
        }
        return ar;
    }
}
```