1) Print 2nd largest and 2nd smallest elements from a given integer array 'arr' of size arr_len in a single loop. Method Signature void print2nd(int arr[], int arr_len) { //Your code here } Example 1 Input: arr[]: 5, 10, 0, 2, 3, 4 arr_size = 6 Output: 2, 5

```
package mypack;


public class ninetest
{
	public static void print(int arr[], int arr_len)
	{
		int first_min = Integer.MAX_VALUE;
		int second_min = Integer.MAX_VALUE;
		int first_max = Integer.MIN_VALUE;
		int second_max = Integer.MAX_VALUE;

		for(int i:arr)
		{
			if(i<first_min)
			{
				second_min = first_min;
				first_min = i;
			}
			else if(i<second_min && i!=first_min)
			{
				second_min = i;
			}

		}
		for(int i: arr)
```

```
            {
                if(i>first_max)
                {
                    second_max = first_max;
                    first_max = i;
                }
                else if(i>second_max && i!=first_max)
                {
                    second_max = i;
                }
            }
            System.out.println("Second largest is: " + second_max);
            System.out.println("Second smallest is: " + second_min);
        }

        public static void main(String[] args)
        {
            int arr[]= { 5, 10, 0, 2, 3, 4};
            int arr_size = 6;
            print(arr,arr_size);



        }

}
```

2) Given 2 sorted arrays of integers, print common elements between 2 arrays in single loop. Method Signature void printCommon(int[] arr1, int [] arr2, int arr1_len, int arr2_len) { }

```
package mypack;
```

```
public class ninetest
{
	public static void printCommon(int[] arr1, int [] arr2, int arr1_len, int arr2_len)
	{
		int i=0;
		int j=0;

		while(i<arr1_len && j<arr2_len)
		{
			if(arr1[i] == arr2[j])
			{
				System.out.print(arr1[i]);
				i++;
				j++;
			}
			else if(arr1[i]<arr2[j])
			{
				i++;
			}
			else
			{
				j++;
			}
		}

	}
```

```
	public static void main(String[] args)
	{
		int arr1[] = {1, 3, 4, 5, 7};
		int arr2[] = {2,3,5, 6};
		printCommon(arr1,arr2,5,4);




	}

}
```