```java
import java.util.Arrays;
import java.util.Scanner;
import java.text.DecimalFormat;
public class SieveOfEratosthenes {

    public static void main String[] args) {
        // TODO Auto-generated method stub
        System.out.println "Sieve of Eratosthenes Lab");
        Scanner stitch = new Scanner (System.in);
        for (int i = 0; i < 2; i++) {
        System.out.print "Enter the primes upper bound  ===>>  ");
        final int MAX = stitch.nextInt();
        boolean primes[] = new boolean[MAX];
        computePrimes(primes);
        displayPrimes(primes);
        System.out.println();
        }
       //System.out.println(Arrays.toString(primes));
      }

      public static void computePrimes(boolean primeArray[])
      { // This method will compute the prime numbers
        System.out.println "COMPUTING PRIME NUMBERS....");

        for (int i = 2; i < primeArray.length; i++) {
            primeArray [i] = true;
            //System.out.println(Arrays.toString(primeArray));
        }

        for (int i = 2;  i <= Math.sqrt(primeArray.length); i++) {
            if (primeArray i] == true) {
                for (int z = 2 * i; z < primeArray.length; z = z + i ) {
                    primeArray[z] = false;

                }
            }
        }
    }

      public static void displayPrimes(boolean primeArray[])
      { // This method will display the prime numbers
          System.out.println "PRIMES BETWEEN 1 AND " +(primeArray.length) +":");
          //System.out.println(Arrays.toString(primeArray));

          DecimalFormat tree = new DecimalFormat("0000");
          int c = 0;

          for (int j = 2; j < primeArray.length; j++){
              if (primeArray[j] == true) {
              System.out.print(tree.format(j)+ " ");

        //make a counter up to 16, then go to next line (print empty line)
            c++;
            if (c == 16) {
            System.out.println();
            c = 0;

```

```
58                 }
59             }
60         }
61     }
62 }
63
64
65
66
67
68
69
70
71
72
73 //boolean to numbers
74 //int counter = 0;
75 //int numberArray[] = new int [n];
76 // for (int i = 2; i < MAX-1; i++) {
77       //if (primeArray[i] == true) {
78           //numberArray[counter] = i;
79           //counter++;
80           //System.out.println(+counter);
81
82 //for (int i = 2; i < MAX; i ++) {
83 // if (primeArray[i] == true)
84     // n++;
85
86
```