

```
1
2  /**
3   A PROGRAM TO IMPLEMENT MOBIUS FUNCTION.
4   WHICH IS A FUNCTION WHICH RETURNS -> 1 FOR THE NATURAL NUMBER 1.
5   WHICH IS A FUNCTION WHICH RETURNS -> 0 FOR A NATURAL NUMBER WITH RE
6   PEATING PRIME FACTORS.
7   WHICH IS A FUNCTION WHICH RETURNS ->  $(-1)^p$  FOR A NATURAL NUMBER WI
8   TH p DISTINCT PRIME FACTORS.
9  */
10 //SUCHIT TE XII A
11 import java.util.*;
12 public class Mobiusfn
13 {
14     static int num;
15     String checkPrime(int receive)
16     {
17         int ctr=0;
18         for(int i=1;i<=receive;i++)
19         {
20             if(receive%i==0)
21             {
22                 ctr++;
23             }
24             else
25             {
26                 continue;
27             }
28         }
29         if(ctr==2)
30         {
31             String reply="PRIME NUMBER";
32             return reply;
33         }
34         else
35         {
36             String reply="NOT PRIME NUMBER";
37             return reply;
38         }
39     }
40     int primeFactors()
41     {
42         int temp=0;
43         Mobiusfn obj1 = new Mobiusfn();
44         int ctr=0;;
45         System.out.println("THE PRIME FACTORS ARE");
46         for(int i=2;i<=num;i++)
47         {
48             String check1=obj1.checkPrime(i);
49             if(check1=="PRIME NUMBER")
50             {
51                 if(num%i==0)
52                 {
53                     temp=num/i;
54                     System.out.println(i);
55                 }
56             }
57         }
58     }
59 }
```

```

53         ctr++;
54         String check2=obj1.checkPrime(temp);
55         num=temp;
56         if(check2=="PRIME NUMBER")
57         {
58             System.out.println(temp);
59             ctr++;
60             break;
61         }
62         else if(temp%i==0)
63         {
64             ctr=0;
65             break;
66         }
67         else
68         {
69             continue;
70         }
71     }
72 }
73 else if(check1=="NOT PRIME NUMBER")
74 {
75     continue;
76 }
77 }
78 return ctr;
79 }
80 public static void main()
81 {
82     Scanner sc=new Scanner(System.in);
83     System.out.println("THIS PROGRAM IS AIMED AT IMPLEMENTING MOBI
US FUNCTION.");
84     System.out.println("WHICH IS A FUNCTION WHICH RETURNS -> 1 FOR
THE NATURAL NUMBER 1");
85     System.out.println("WHICH IS A FUNCTION WHICH RETURNS -> 0 FOR
A NATURAL NUMBER WITH REPEATING PRIME FACTORS");
86     System.out.println("WHICH IS A FUNCTION WHICH RETURNS -> (-1)^
p FOR A NATURAL NUMBER WITH p DISTINCT PRIME FACTORS");
87     System.out.println("PLEASE ENTER A NATURAL NUMBER");
88     num = sc.nextInt();
89     int number=num;
90     Mobiusfn obj = new Mobiusfn();
91     int hold=0;
92     if(num!=1)
93     {
94         hold=obj.primeFactors();
95     }
96     else if(num==1)
97     {
98         System.out.println("THE NUMBER ENTERED : "+number);
99         System.out.println("MOBIUS FUNCTION RESPONSE : 1");
100        System.exit(0);
101    }
102    if(hold==0)

```

```
103         {
104             System.out.println("THE NUMBER ENTERED :"+number);
105             System.out.println("MOBIUS FUNCTION RESPONSE : 0");
106         }
107         else
108         {
109             System.out.println("THE NUMBER ENTERED :"+number);
110             System.out.println("MOBIUS FUNCTION RESPONSE : "+Math.pow(
-1,hold));
111         }
112     }
113 }
114
```