```java
import java.util.*;


public class ProblemSet1 {
    public static void main(String[] args) {
        System.out.println("The answer to Multiples of 3 and 5:" + multiple(1000));
        System.out.println("The answer to Even Fibonacci Numbers:" + fib(4000000));
        Long num = 600851475143L;
        System.out.println("The answer to largest prime factor is:" + factor(num));
        System.out.println("The answer to Largest Palindrome Product is:" + palindrome());
        System.out.println("The answer to Smallest Multiple:" + divisibleByAll(2520,20));
        System.out.println("The answer to Sum Square Difference:" + sumSquareDifference(100));
    }

    static int multiple(int n) {//problem 1
        int sum = 0;
        for(int i = 0; i< n; i++) {
            if(i % 3 ==0 || i %5 == 0) {
                sum += i;
            }
        }
        return sum;
    }

```

```
26      static int fib(int max) {//problem 2
27          int sum = 0;
28          int val1 = 1;
29          int val2 = 2;
30          int val = 1;
31          while(val2 < max) {
32              val = val1+val2;
33              if(val % 2 ==0) {
34                  sum+=val;
35              }
36              val1 = val2;
37              val2 = val;
38          }
39          return sum+=2;
40      }
41
42
43      static int factor(long n) {//Problem 3
44          int result = 0;
45          int count = 2;
46          while(n > 1) {
47              while(n%count ==0) {
48                  result = Math.max(result, count);
49                  n/=count;
50              }
51              count++;
52          }
53          return result;
54      }
55
```

```
56      static int palindrome() {//problem 4
57          int result = 0;
58          for(int i = 999; i > 99; i--) {
59              for(int j = 999; j > 99; j--) {
60                  int val = i * j;
61                  if(isPalindrome(val)) {
62                      result = Math.max(result, val);
63                  }
64              }
65          }
66          return result;
67      }
68
69      static boolean isPalindrome(int n) {//problem 4
    cont'd
70          String s = Integer.toString(n);
71          boolean isTrue = true;
72          for(int i = 0; i < s.length() / 2; i++) {
73              if(s.charAt(i) !=
    s.charAt(s.length()-1-i)) {
74                  isTrue =  false;
75              }
76          }
77          return isTrue;
78      }
79
80      static int divisibleByAll(int start,int n)
    {//problem 5
81          while(!isDivisibleByAll(start, n)) {
82              start+=2;
```

```
83          }
84          return start;
85      }
86
87      static boolean isDivisibleByAll(int val, int n)
    {//problem 5 cont'd
88          for(int i = 2; i <= n; i++) {
89              if(val % i != 0) {
90                  return false;
91              }
92          }
93          return true;
94      }
95
96
97      static int sumSquareDifference(int n)
    {//problem 6
98          int sumOfSquares = 0;
99          for(int i = 1; i < n+1;i++) {
100             sumOfSquares+= (i*i);
101         }
102         int squareOfSum = ((n * (n+1))/2) * ((n *
    (n+1))/2);
103         return squareOfSum-sumOfSquares;
104     }
105
106 }
107
```