5/13/2013

Task 3

Assignment 2

Tan Shi Terng Leon 4000602
UNIVERSITY OF WOLLONGONG

Design

How the program works

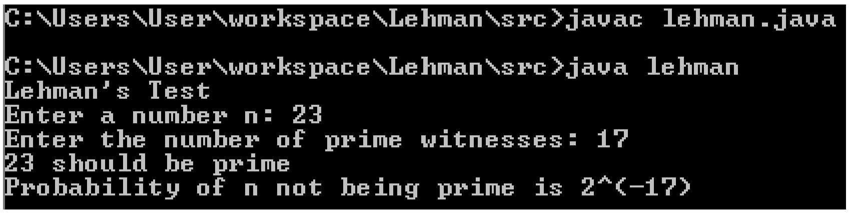
- The user inputs a number n and the number of required prime witnesses a
- The program does some error checking to ensure n is an odd number greater than 2 and the number of prime witnesses must be greater than one and smaller than n – 1
- For $i = 1\ to\ a$, a random number x between 1 and n – 1 inclusive is generated
- The program makes sure that the numbers generated are not repeated by keeping track of the numbers used
- If $\gcd(x, n) = 1$, the number is composite, the program prints the result to the screen and also writes the information to the file lehman-dump.txt and the program exits the loop
- Else the test value is computed

$$value = x^{(n-1)/2}\ mod\ n$$

- If the test value is not 1 or n – 1, the number n is composite. The program prints the result to the screen and also writes the information to the file lehman-dump.txt. The program then ends
- Or else it returns a prime witness, writes it to the file and continues generating another random number to test.
- If the program ends without finding any proof that the number is composite, it prints to the screen telling the user that n is probably prime. And also the probability that the test result is incorrect, which is 2^{-a}’

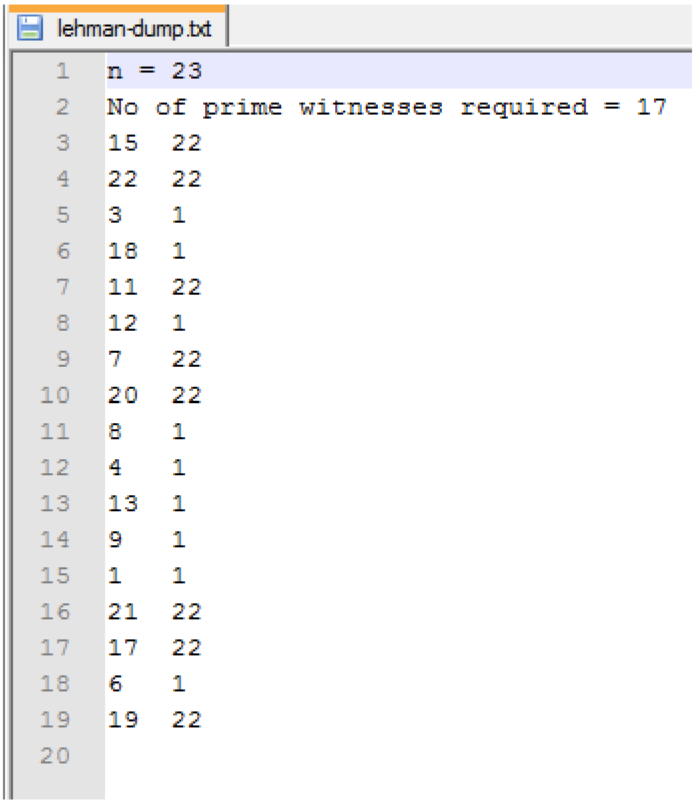
Examples

Testing number 23



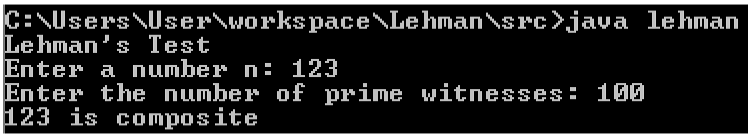
```
C:\Users\User\workspace\Lehman\src>javac lehman.java

C:\Users\User\workspace\Lehman\src>java lehman
Lehman's Test
Enter a number n: 23
Enter the number of prime witnesses: 17
23 should be prime
Probability of n not being prime is 2^(-17)
```

The test values return only 1 or 22



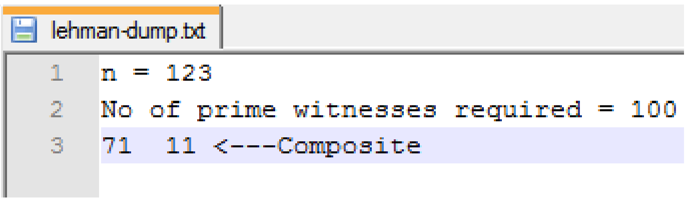
```
n = 23
No of prime witnesses required = 17
15   22
22   22
3    1
18   1
11   22
12   1
7    22
20   22
8    1
4    1
13   1
9    1
1    1
21   22
17   22
6    1
19   22
```

Testing if 123 is prime



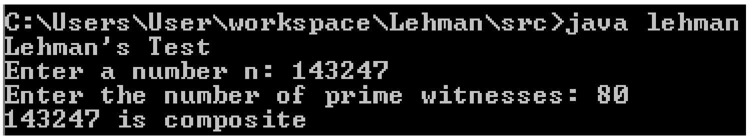
```
C:\Users\User\workspace\Lehman\src>java lehman
Lehman's Test
Enter a number n: 123
Enter the number of prime witnesses: 100
123 is composite
```

The test value of 71 returns 11 which is not 1 or 122



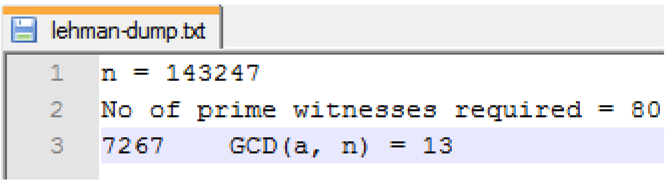
```
n = 123
No of prime witnesses required = 100
71   11 <---Composite
```

Testing 143247



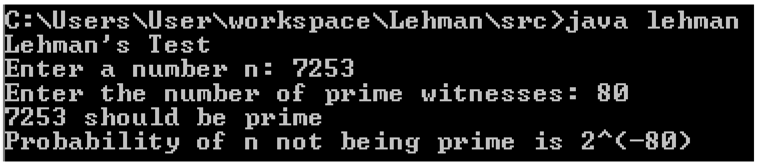
```
C:\Users\User\workspace\Lehman\src>java lehman
Lehman's Test
Enter a number n: 143247
Enter the number of prime witnesses: 80
143247 is composite
```

Since $gcd(7267,143247) = 13 \neq 1$, 143247 is composite



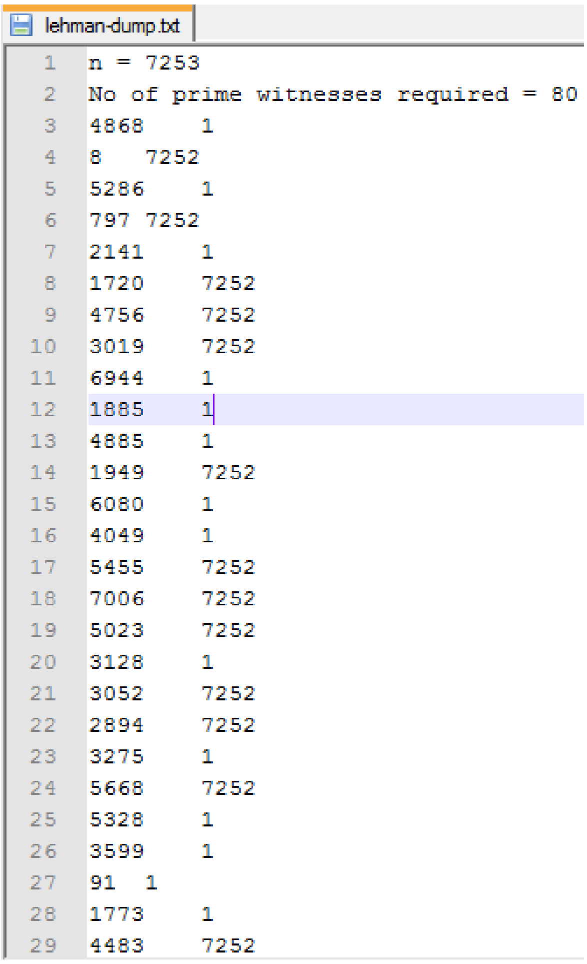
```
n = 143247
No of prime witnesses required = 80
7267    GCD(a, n) = 13
```

Testing 7253

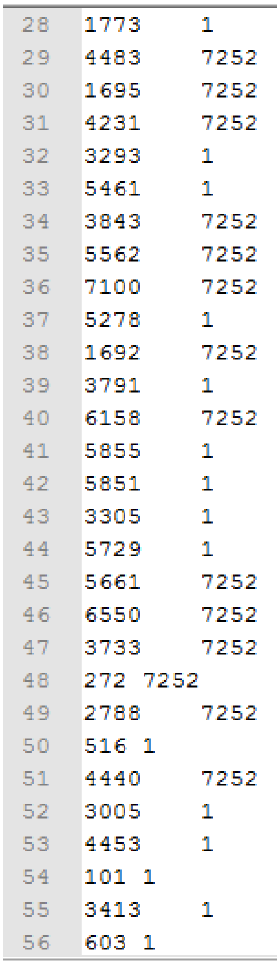


```
C:\Users\User\workspace\Lehman\src>java lehman
Lehman's Test
Enter a number n: 7253
Enter the number of prime witnesses: 80
7253 should be prime
Probability of n not being prime is 2^(-80)
```

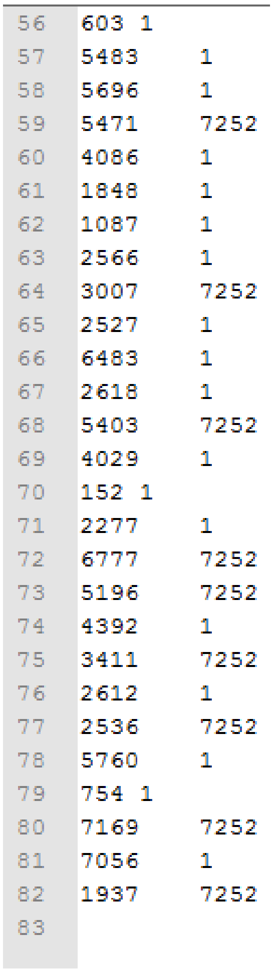
The test values only return 1 or 7252



```
n = 7253
No of prime witnesses required = 80
4868    1
8   7252
5286    1
797 7252
2141    1
1720    7252
4756    7252
3019    7252
6944    1
1885    1
4885    1
1949    7252
6080    1
4049    1
5455    7252
7006    7252
5023    7252
3128    1
3052    7252
2894    7252
3275    1
5668    7252
5328    1
3599    1
91  1
1773    1
4483    7252
```



```
1773    1
4483    7252
1695    7252
4231    7252
3293    1
5461    1
3843    7252
5562    7252
7100    7252
5278    1
1692    7252
3791    1
6158    7252
5855    1
5851    1
3305    1
5729    1
5661    7252
6550    7252
3733    7252
272 7252
2788    7252
516 1
4440    7252
3005    1
4453    1
101 1
3413    1
603 1
```



```
603 1
5483    1
5696    1
5471    7252
4086    1
1848    1
1087    1
2566    1
3007    7252
2527    1
6483    1
2618    1
5403    7252
4029    1
152 1
2277    1
6777    7252
5196    7252
4392    1
3411    7252
2612    1
2536    7252
5760    1
754 1
7169    7252
7056    1
1937    7252

```