Lab (3)

Number Theory

Name	Ranime Ahmed Elsayed Shehata.
ID	21010531.

Part 1: Prime Number Checker

➢ Problem Statement:

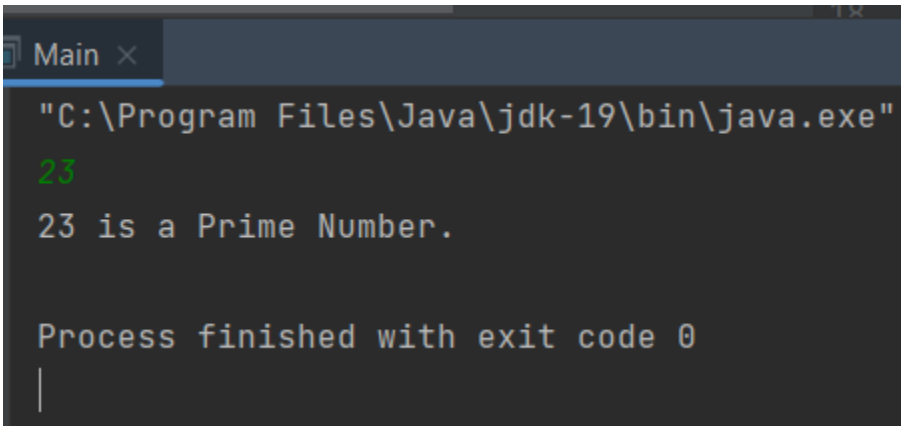
Implement a function that determines whether a given positive integer is a prime number or not using Sieve of Eratosthenes.

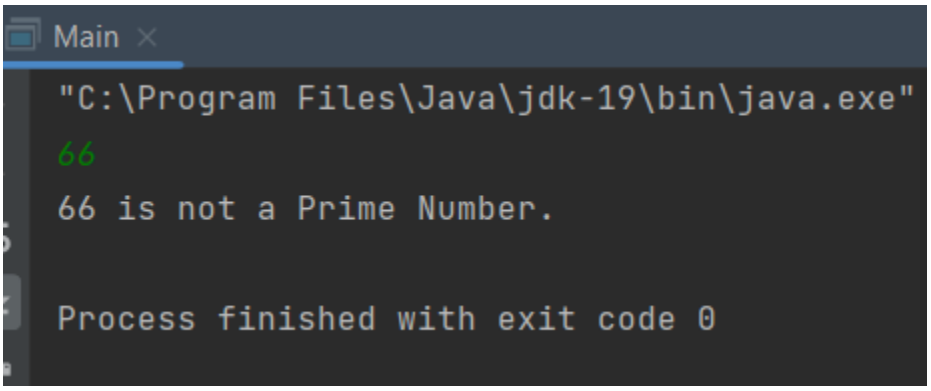
➢ Used data structures:

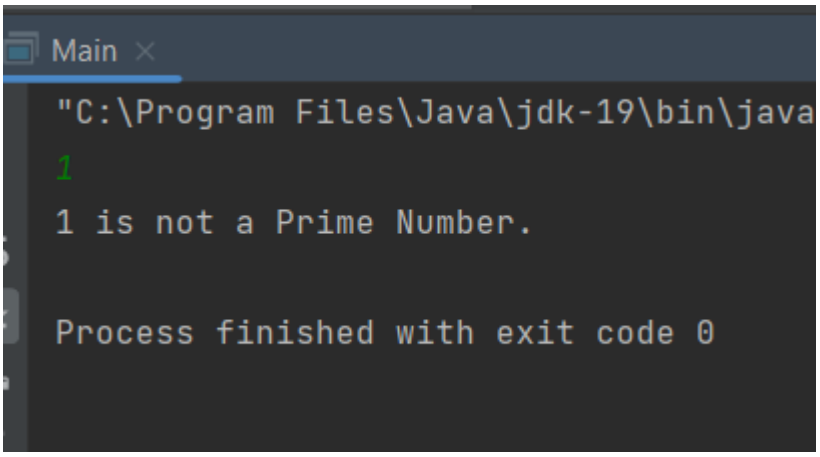
Arrays:

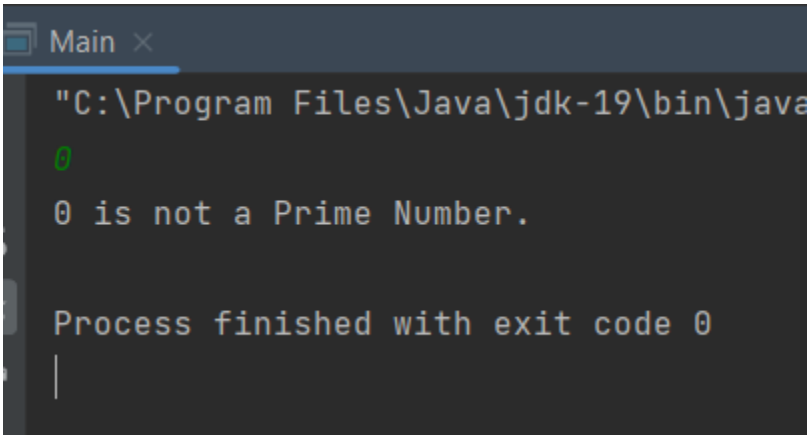
Name	Type
primes	boolean

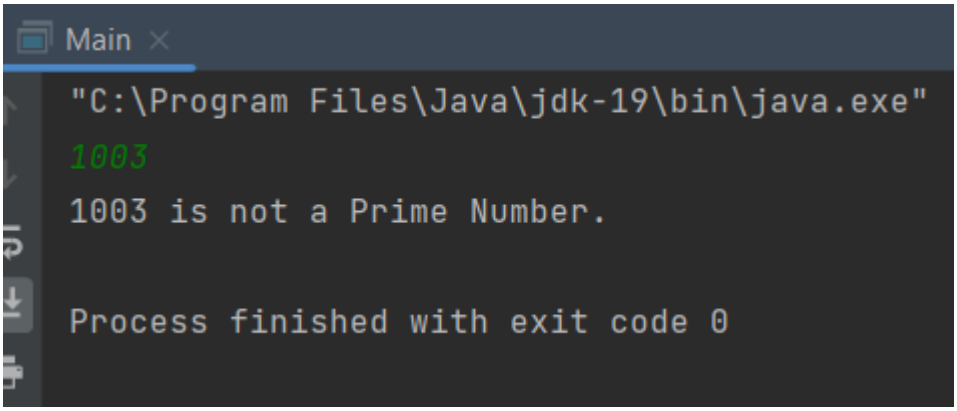
➢ Sample runs and different test cases:











No assumptions made.

Part 2: Prime Factorization

Problem Statement:

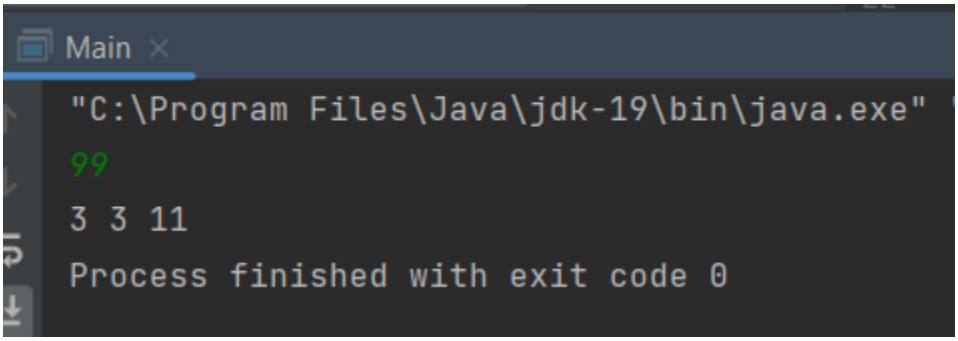
Create a function that computes the prime factors of a given integer.

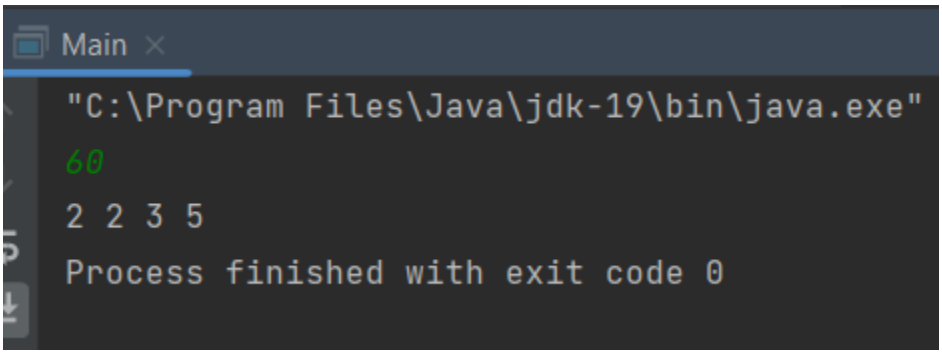
- **Used data structures:**

Arrays:

Name	Type
primes	boolean

- **Sample runs and different test cases:**





Assumptions:

All integers to be factorized are positive integers.

Part 3: GCD and LCM Computation

(a):

- **Problem Statement:**

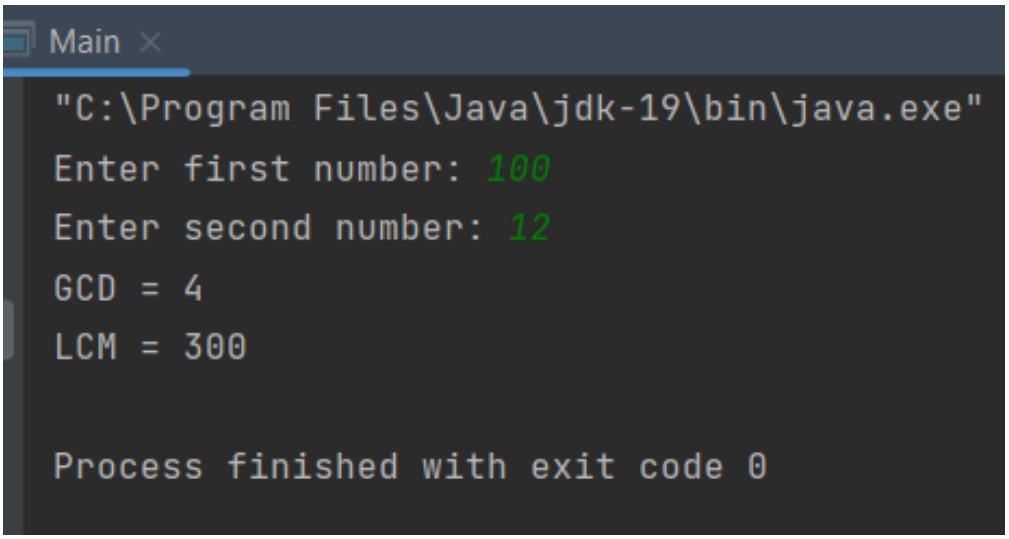
Implement functions to calculate the GCD and LCM of two positive integers.

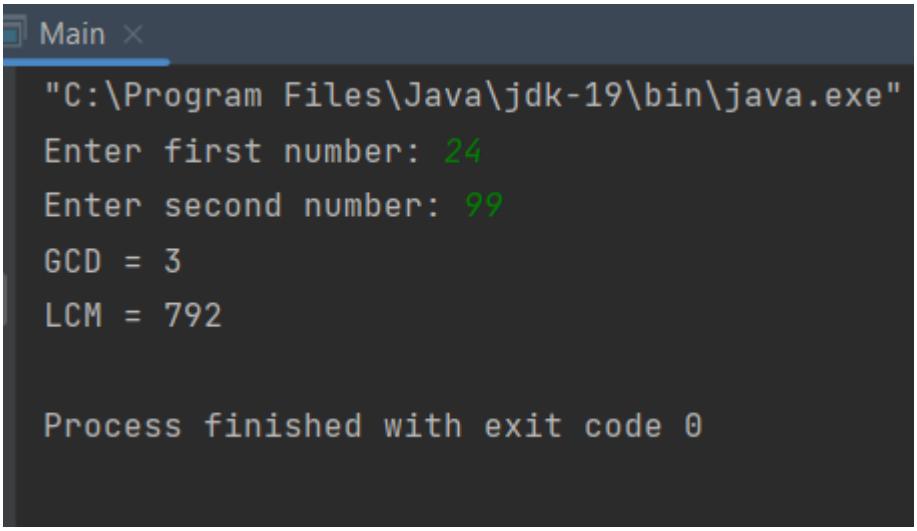
a) Using the Euclidean algorithm for GCD computation and the relationship between GCD and LCM to find the LCM.

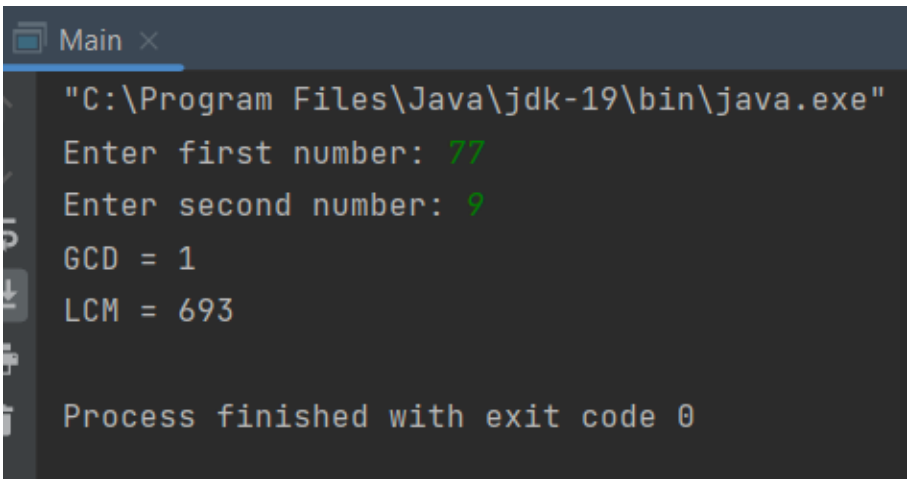
Used data Structures:

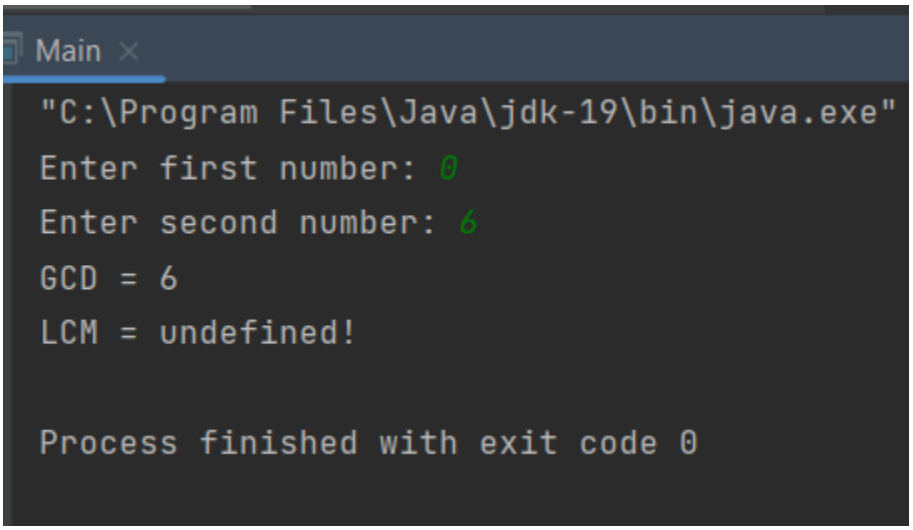
Nothing.

Sample runs and different test cases:









No assumptions made.

(b):

- **Problem Statement:**

Implement functions to calculate the GCD and LCM of two positive integers.

b) Using prime factorization.

- **Used Data Structures:**

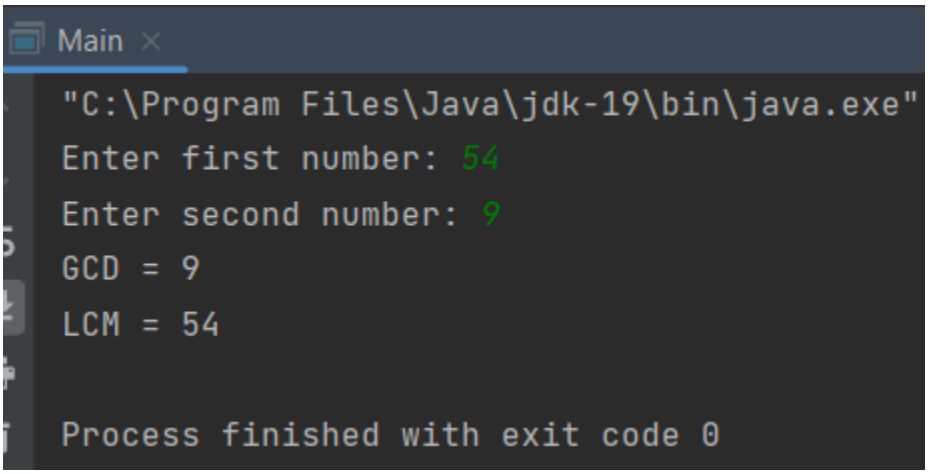
Arrays:

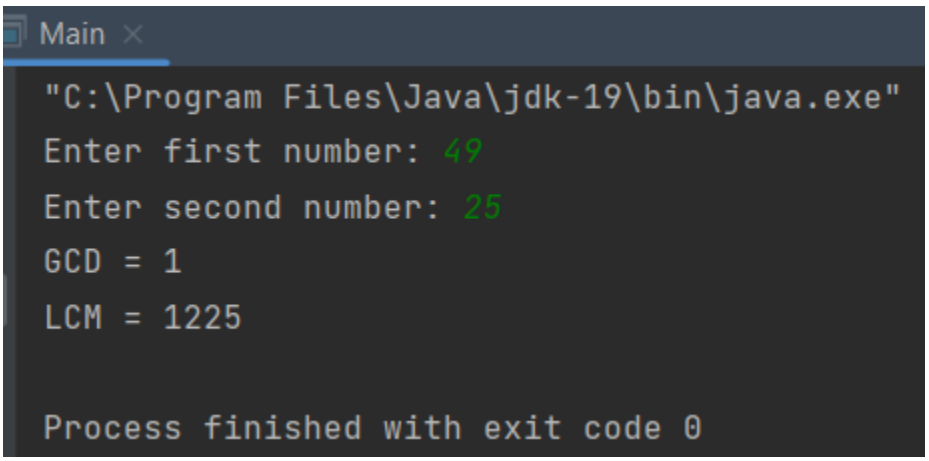
Name	Type
primes	boolean

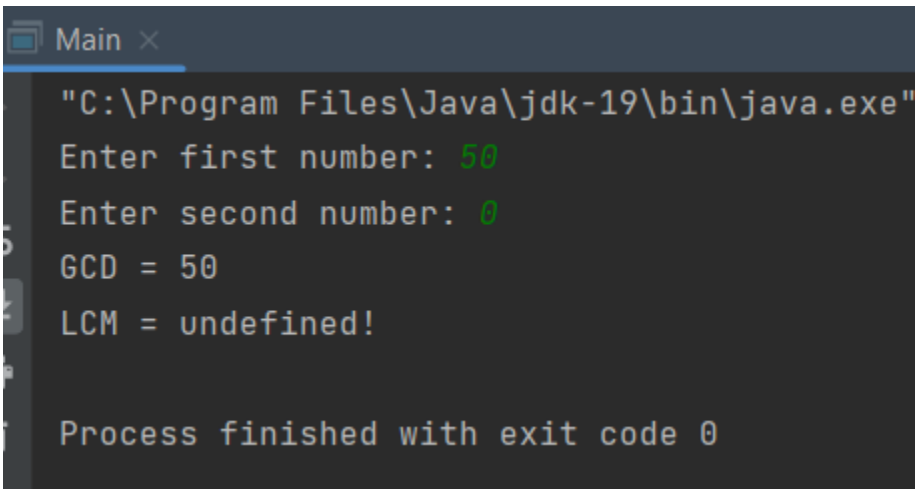
ArrayList:

Name	Type
primesList	Integer
gcd	Integer
lcm	Integer
primes1	Integer
primes2	Integer
factors1	Integer
factors2	Integer

- **Sample runs and different test cases:**







No assumptions made.

Part 4: Chinese Remainder Theorem

- **Problem Statement:**

Implement Chinese remainder theorem that takes as input m1, m2, m3, …., mn that are pairwise relatively prime and (a1, a2, …., an) and calculates x such that

x = a1 (mod m1)

x = a2 (mod m2)

…

x= an (mod mn)

- **Used data structures:**

Arrays:

Name	Type
a	Int
m	Int

- **Sample runs and different test cases:**

```
Enter the number of congruence relations: 3
Enter the values of m:
m1= 3
m2= 4
m3= 5
Enter the values of a:
a1= 2
a2= 3
a3= 1
x = 11

Process finished with exit code 0
```

```
Enter the number of congruence relations: 3
Enter the values of m:
m1= 3
m2= 5
m3= 7
Enter the values of a:
a1= 2
a2= 3
a3= 2
x = 23

Process finished with exit code 0
```

No assumptions made.