

CSE 301 Midterm-II Exam

I. QUESTION

Write a Java class named **Primes** which includes the following methods:

- **boolean isPrime(int N)**: Checks whether the number N is prime or not.
- **int numberOfPrimes(int N)**: Returns the number of prime numbers less than or equal to N .
- **int[] primeList(int N)**: Returns the prime numbers less than or equal to N in an array.
- **int gcd(int X, int Y)**: Returns the greatest common divisor of X and Y (Hint: First find the minimum of X and Y , second find the primes less than that minimum, and third find the prime numbers which divide both X and Y).
- **int primeFactorCount(int N)**: Returns the number of prime factors (not necessarily distinct) of N . Use `primeList` method to get the primes less than N .

Example:

```
isPrime(7) -> true
isPrime(12) -> false
numberOfPrimes(12) -> 5
numberOfPrimes(6) -> 3
primeList(12) -> {2, 3, 5, 7, 11}
primeList(6) -> {2, 3, 5}
primeFactorCount(6) -> 2
primeFactorCount(36) -> 4
primeFactorCount(1000) -> 6
```

II. QUESTION

Design a class named **Fraction** to represent rational numbers. The class contains:

- Two integer fields numerator and denominator that specify the numerator and denominator of a rational number. The default values are 1 for both.
- A no argument constructor that creates the default fraction.
- A constructor that creates a fraction with the specified numerator and denominator.
- A constructor that creates a fraction with the specified numerator only (denominator is 1).
- The accessor and mutator (get and set methods) for two fields.
- A method named `double getValue()` that returns the value of the fraction.
- A method named `void add(Fraction f)` that adds the fraction f to the current fraction.
- A method named `void negate()` that negates the fraction.
- A method named `void inverse()` that reverses the fraction (numerator becomes denominator, denominator becomes nominator).
- A method named `void multiply(Fraction f)` that multiplies the fraction f to the current fraction.