Lab Exercises - Arithmetic

CMPT333N

Problem 1

Determine whether a given integer number is prime.

```
Example:
?- is_prime(7).
Yes
```

Problem 2

Determine the greatest common divisor of two positive integer numbers.

```
Use Euclid's algorithm.
Example:
?- gcd(36, 63, G).
G = 9
Define gcd as an arithmetic function; so you can use it like this:
?- G is gcd(36,63).
G = 9
```

Problem 3

Determine whether two positive integer numbers are coprime.

```
Two numbers are coprime if their greatest common divisor equals 1.
Example:
?- coprime(35, 64).
Yes
```

Problem 4

Calculate Euler's totient function phi(m).

```
Euler's so-called totient function phi(m) is defined as the number of positive
integers r (1 <= r < m) that are coprime to m.
Example: m = 10: r = 1,3,7,9; thus phi(m) = 4. Note the special case: phi(1) = 1.

?- Phi is totient_phi(10).
Phi = 4
```

Find out what the value of phi(m) is if m is a prime number. Euler's totient function plays an important role in one of the most widely used public key cryptography methods (RSA). In this exercise you should use the most primitive method to calculate this function (there are smarter ways that we shall discuss later).

Problem 5

Determine the prime factors of a given positive integer.

```
Construct a flat list containing the prime factors in ascending order.
Example:
?- prime_factors(315, L).
L = [3,3,5,7]
```

Problem 6

Determine the prime factors of a given positive integer (2).

```
Construct a list containing the prime factors and their multiplicity.
Example:
?- prime_factors_mult(315, L).
L = [[3,2],[5,1],[7,1]]
```

Problem 7

Calculate Euler's totient function phi(m) (improved).

See problem P4 for the definition of Euler's totient function. If the list of the prime factors of a number m is known in the form of problem P6 then the function phi(m) can be efficiently calculated as follows: Let [[p1,m1],[p2,m2],[p3,m3],...] be the list of prime factors (and their multiplicities) of a given number m. Then phi(m) can be calculated with the following formula:

$$phi(m) = (p_1 - 1) * p_1^{m_1-1} * (p_2 - 1) * p_2^{m_2-1} * (p_3 - 1) * p_3^{m_3-1} ...$$

Problem 8

Compare the two methods of calculating Euler's totient function.

Use the solutions of problems 4 and 5 to compare the algorithms. Take the number of logical inferences as a measure for efficiency. Try to calculate phi(10090) as an example.

Problem 9

A list of prime numbers.

Given a range of integers by its lower and upper limit, construct a list of all prime numbers in that range.

Problem 10

Goldbach's conjecture.

Goldbach's conjecture says that every positive even number greater than 2 is the sum of two prime numbers. Example: 28 = 5 + 23. It is one of the most famous facts in number theory that has not been proved to be correct in the general case. It has been *numerically* confirmed up to very large numbers (much larger than we can go with our Prolog system). Write a predicate to find the two prime numbers that sum up to a given even integer.

```
Example:
?- goldbach(28, L).
L = [5,23]
```

Problem 11

A list of Goldbach compositions.

Given a range of integers by its lower and upper limit, print a list of all even numbers and their Goldbach composition.

```
Example:
?- goldbach_list(9,20).
10 = 3 + 7
12 = 5 + 7
14 = 3 + 11
16 = 3 + 13
18 = 5 + 13
20 = 3 + 17
```

In most cases, if an even number is written as the sum of two prime numbers, one of them is very small. Very rarely, the primes are both bigger than say 50. Try to find out how many such cases there are in the range 2..3000.

```
Example (for a print limit of 50):
?- goldbach_list(1,2000,50).
992 = 73 + 919
1382 = 61 + 1321
1856 = 67 + 1789
1928 = 61 + 1867
```