1.2 Problems NS-2

**Topic of this homework:**
Prime numbers, greatest common divisors, the continued fraction algorithm

**Prime numbers**

**Problem # 1:** Every integer may be written as a product of primes.

- 1.1: Write the numbers 1,000,000, 1,000,004, and 999,999 in the form $N = \prod_k \pi_k^{\beta_k}$. Hint: Use Matlab/Octave to find the prime factors.

**Ans:**

- 1.2: Give a generalized formula for the natural logarithm of a number $\ln(N)$ in terms of its primes $\pi_k$ and their multiplicities $\beta_k$. Express your answer as a sum of terms.

**Ans:**
Problem # 2: Using the computer

– 2.1: Explain why the following brief Matlab/Octave program returns the prime numbers \( \pi_k \) between 1 and 100.

\[
n=2:100;
k = \text{isprime}(n);
n(k)
\]

Ans:

– 2.2: How many primes are there between 2 and \( N = 100 \)?

Ans:

Problem # 3: Prime numbers may be identified using a sieve (See Figure).

– 3.1: By hand, complete the sieve of Eratosthenes for \( n = 1, \ldots, 49 \). Circle each prime \( p \), then cross out each number that is a multiple of \( p \).

Note: 1 should not be circled as it is not a prime.

– 3.2: What is the largest number you need to consider before only primes remain?

Ans:

– 3.3: Generalize: For \( n = 1, \ldots, N \), what is the largest number you need to consider before only the primes remain?

Ans:
– 3.4: Write each of these numbers as a product of primes: 22, 30, 34, 43, 44, 48, 49. **Ans:**

– 3.5: Find the largest prime \( \pi_k \leq 100 \). Do not use Matlab/Octave other than to check your answer. Hint: Write the numbers starting with 100 and count backward: 100, 99, 98, 97, . . . . Cross off the even numbers, leaving 99, 97, 95, . . . . Pull out a factor (only one is necessary to show that it is not prime). **Ans:**

– 3.6: Find the largest prime \( \pi_k \leq 1000 \). Do not use Matlab/Octave other than to check your answer. **Ans:**

– 3.7: Explain why \( \pi_k^{-s} = e^{-s \ln \pi_k} \). **Ans:**

**Greatest common divisors**

Consider using the Euclidean algorithm to find the greatest common divisor (i.e., GCD; the largest common prime factor) of two numbers. Note that this algorithm may be performed using one of two methods:

<table>
<thead>
<tr>
<th>Method</th>
<th>Division</th>
<th>Subtraction</th>
</tr>
</thead>
<tbody>
<tr>
<td>On each iteration</td>
<td>( a_{i+1} = b_i )</td>
<td>( a_{i+1} = \max(a_i, b_i) - \min(a_i, b_i) )</td>
</tr>
<tr>
<td></td>
<td>( b_{i+1} = a_i - b_i \cdot \floor(a_i/b_i) )</td>
<td>( b_{i+1} = \min(a_i, b_i) )</td>
</tr>
<tr>
<td>Terminates when</td>
<td>( b = 0 ) (GCD = ( a ))</td>
<td>( b = 0 ) (GCD = ( a ))</td>
</tr>
</tbody>
</table>

The division method (Eq. 2.1, Sec. 2.1.2, Ch. 2) is preferred because the subtraction method is much slower.

**Problem # 4: Understanding the Euclidean algorithm (GCD)**
- 4.1: Use the Octave/Matlab command `factor` to find the prime factors of $a = 85$ and $b = 15$.

\textbf{Ans:}

- 4.2: What is the greatest common prime factor of $a = 85$ and $b = 15$?

\textbf{Ans:}

- 4.3: By hand, perform the Euclidean algorithm for $a = 85$ and $b = 15$.

\textbf{Ans:}
- 4.4: By hand, perform the Euclidean algorithm for \( a = 75 \) and \( b = 25 \). Is the result a prime number?

\textbf{Ans:}

- 4.5: Consider the first step of the GCD division algorithm when \( a < b \) (e.g., \( a = 25 \) and \( b = 75 \)). What happens to \( a \) and \( b \) in the first step? Does it matter if you begin the algorithm with \( a < b \) rather than \( b < a \)?

\textbf{Ans:}

- 4.6: Describe in your own words how the GCD algorithm works. Try the algorithm using numbers that have already been divided into factors (e.g., \( a = 5 \cdot 3 \) and \( b = 7 \cdot 3 \)).

\textbf{Ans:}

- 4.7: Find the GCD of \( 2 \cdot \pi_{25} \) and \( 3 \cdot \pi_{25} \).

\textbf{Ans:}

\textbf{Problem # 5: Coprimes}

- 5.1: Define the term coprime.

\textbf{Ans:}
5.2: How can the Euclidean algorithm be used to identify coprimes?

Answer:

5.3: Give at least one application of the Euclidean algorithm.

Answer:

5.4: Write a Matlab function, \( \text{function } x = \text{my}_{-}\text{gcd}(a,b) \), that uses the Euclidean algorithm to find the GCD of any two inputs \( a \) and \( b \). Test your function on the \((a, b)\) combinations from the previous problem. Include a printout (or hand-write) your algorithm to turn in.

Hints and advice:

- Don’t give your variables the same names as Matlab functions! Since \( \text{gcd} \) is an existing Matlab/Octave function, if you use it as a variable or function name, you won’t be able to use \( \text{gcd} \) to check your \( \text{gcd}() \) function. Try \( \text{clear all} \) to recover from this problem.

- Try using a “while” loop for this exercise (see Matlab documentation for help).

- You may need to use some temporary variables for \( a \) and \( b \) in order to perform the algorithm.

Answer:

Continued fractions

Problem #6: Here we explore the continued fraction algorithm (CFA), discussed in Sec. 2.4.4. In its simplest form, the CFA starts with a real number, which we denote as \( \alpha \in \mathbb{R} \). Let us work with an irrational real number, \( \pi \in \mathbb{I} \), as an example because its CFA representation will be infinitely long. We can represent the CFA coefficients \( \alpha \) as a vector of integers \( n_k, k = 1, 2, \ldots, \infty \):

\[
\alpha = [n_1; n_2, n_3, n_4, \ldots] = n_1 + \frac{1}{n_2 + \frac{1}{n_3 + \frac{1}{n_4 + \cdots}}}
\]
As discussed in Sec. 2.4.3 (p. 27), the CFA is recursive, with three steps per iteration. For \( \alpha_1 = \pi \), \( n_1 = 3 \), \( r_1 = \pi - 3 \), and \( \alpha_2 \equiv 1/r_1 \).

\[
\alpha_2 = \frac{1}{0.1416} = 7.0625 \ldots
\]

\[
\alpha_1 = n_1 + \frac{1}{\alpha_2} = n_1 + \frac{1}{n_2 + \frac{1}{\alpha_3}} = \ldots
\]

In terms of a Matlab/Octave script,

```matlab
alpha0 = pi;
K=10;
n=zeros(1,K); alpha=zeros(1,K);
alpha(1)=alpha0;
for k=2:K %k=1 to K
    n(k)=round(alpha(k-1));
    %n(k)=fix(alpha(k-1));
    alpha(k)= 1/(alpha(k-1)-n(k));
    %disp([fix(k), round(n(k)), alpha(k)]); pause(1)
end
disp([n; alpha]);
%Now compare this to matlab’s rat() function
rat(alpha0,1e-20)
```

– 6.1: By hand (you may use Matlab/Octave as a calculator), find the first three values of \( n_k \) for \( \alpha = e^\pi \).

**Ans:**
6.2: For the proceeding question, what is the error (remainder) when you truncate the continued fraction after \( n_1, \ldots, n_3 \)? Give the absolute value of the error and the percentage error relative to the original \( \alpha \).

\textbf{Ans:}

6.3: Use the Matlab/Octave program provided to find the first 10 values of \( n_k \) for \( \alpha = e^\pi \), and verify your result using the Matlab/Octave command \texttt{rat}() .

\textbf{Ans:}

6.4: Discuss the similarities and differences between the Euclidean algorithm and the CFA.

\textbf{Ans:}

Problem # 7: CFA of ratios of large primes

6.1: (4pts) Expand 23/7 as a continued fraction. Express your answer in bracket notation (e.g., \( \pi = [3, 7, 16, \ldots] \)). Show your work. \textbf{Ans:}

6.2: Starting from the primes below 10^6, form the CFA of \( \pi_j/\pi_k \) with \( j = 78498 \) and \( k < j \).

\textbf{Ans:}
CHAPTER 1. NUMBER SYSTEMS

7.3: Look at other ratios of prime numbers and look for a pattern in the CFA of the ratios of large primes. What is the most obvious conclusion? **Ans:**

7.4: (1pts) Try the Matlab/Octave functions `rats(23/7)`, `rats(3.2857)`, and `rats(3.2856)`. What can you conclude? **Ans:**

7.5: (2pts) Can \( \sqrt{2} \) be represented as a finite continued fraction? Why or why not? **Ans:**

7.6: (2pts) What is the CFA for \( \sqrt{2} - 1 \)?

**Hint:** \[ \sqrt{2} + 1 = \frac{1}{\sqrt{2} - 1} = [2; 2, 2, \ldots] \]

**Ans:**

7.7: Show that

\[
\frac{1}{1 - \sqrt{a}} = a^\frac{1}{2} + a^\frac{3}{2} + a^2 + a^\frac{5}{2} + a^3 + \sqrt{a} + a^4 + a^3 + a^2 + a + 1 = 1 - a^6
\]

**Ans:**