Topics in CS: Problem Set 1

Due date: November 9, 2025.

Question 1. (40 points) Consider the sequence of numbers defined by $L_0 = 2$, $L_1 = 1$, and $L_n = L_{n-1} + L_{n-2}$ for every $n \geq 2$. In this question we will check whether this sequence has an analogue to Zeckendorf's theorem.

1. Getting to know this sequence.
   - Calculate by hand the first 15 numbers in this sequence and write them in table.
   - In another column, write the first 15 Fibonacci numbers.
   - Ignoring L_0 for now, can you identify a connection between Fibonacci numbers and this sequence? That is, can you write L_n as a function of a few Fibonacci numbers? (There is also a connection with L_0, but we will not discuss it for now.)
2. We will say that an integer $N \in \mathbb{N}^+$ (i.e., $N \in \mathbb{N}$ and $N \geq 1$) has a *valid representation* if there exists $\ell \in \mathbb{N}^+$ such that

   $$N = a_\ell L_\ell + a_{\ell-1} L_{\ell-1} + \ldots + a_0 L_0,$$

   where each $a_i \in \{0, 1\}$ with $a_\ell = 1$, and $a_i \cdot a_{i+1} = 0$ for every $i \in \{0, \ldots, \ell - 1\}$.

   Verify that every number in $\{1, \ldots, 20\}$ has a valid representation.
3. Prove that every integer $N \in \mathbb{N}^+$ has a valid representation.
4. Is this representation unique?

Question 3. (30 points)

1. Prove that the running time of Karatsuba's multiplication is $O(n^{\log_2 3})$.
2. Use your favorite programming language and implement Karatsuba's multiplication for inputs of arbitrary length.
3. Sample two integers of 4 bits and multiply them using your implementation.

   Output the steps of the computation together with the result.
4. Sample two integers of 512 bits and multiply them.

Question 3. (30 points) In this question, we you will compute Zeckendorf representation of a given number.

1. Describe an algorithm that given an integer N as input, outputs the largest Fibonacci number that is smaller than N.
2. Using the previous item, describe an algorithm that receives an integer N as input and outputs the Zeckendorf representation of N.

3. What is the running time of you algorithm?
4. Implement your algorithm using your favorite programming language.
5. Sample an 80-bit number and output its Zeckendorf representation

Good luck!