A. Let $m = 1022117$. Assume that m is a product of two distinct primes. Suppose that, using Shor's algorithm, you compute the following periods in $(\mathbb{Z}/m\mathbb{Z})^*$:

1. The period of $k = 966244$ is $p = 7084$.
2. The period of $k = 713912$ is $p = 7728$.
3. The period of $k = 788451$ is $p = 255024$.

Is this enough information to factor m? Execute the factoring algorithm.

The naive way to factor a large integer m is *trial division*: Divide m by 2, then 3, then 4, then 5, and so on, until you find a factor of m. This is slow, but Grover's algorithm can speed it up, if we can apply Grover's algorithm to the following f.

B. Suppose that you are given m such that $m = ab$, where a and b are distinct primes. Find an n and a classical function $f : \{0,1\}^n \to \{0,1\}$ such that $\sum_\alpha f(\alpha) = 1$ and $f(\alpha) = 1$ if and only if the corresponding integer a is a factor of m. (This requires some care. You might want to write f in pseudocode or Python.)

The running time of algorithms for factoring m are often expressed in the notation

$$L[x, y] = e^{(y+o(1))(\log m)^x (\log \log m)^{1-x}},$$

where $x \in [0, 1]$ (and log means the natural logarithm — base e). The value of x is more important than the value of y, so we are often vague about y. The quadratic sieve has running time $L[1/2, y]$ for some y. The number field sieve and function field sieve have running times $L[1/3, y]$ (assuming that the generalized Riemann hypothesis is true).

C. For each of the following questions, give the answer in terms of $n = \log_2 m$, which is the number of bits needed to represent m.

1. What is $L[1, y]$? Plug in 1 for x, simplify, and re-express in terms of n.
2. Similarly, what is $L[0, y]$?
3. Later this term we will see that Shor's algorithm has complexity $\mathcal{O}(n^2)$. (Here, $n = 2 \log_2 m$, but the extra factor of 2 doesn't matter.) So where does it fall on the L scale?
4. Where does trial division fall on the L scale?
5. Assuming that we can apply Grover's algorithm (with known $k = 1$) to your f from Problem B, we get a factoring algorithm. Where does this algorithm fall on the L scale?