Suppose that you are observing the Internet traffic of an RSA user. You would like to be able to read messages sent to this user. You know his n and e, because these are public. In each problem A-C, explain how you could read his messages, if you possessed a fast algorithm to solve the problem described.

A. Integer factoring: Given $n \geq 2$, find all of the prime factors of n.

B. The Euler ϕ-function: Given $n \geq 1$, compute $\phi(n)$.

C. The *logarithm* problem in $\mathbb{Z}/m\mathbb{Z}$: Given $a, b \in \mathbb{Z}/m\mathbb{Z}$, determine whether a $k \in \mathbb{Z}$ exists such that $a^k = b$, and compute k if it exists.

D. Problem C glosses over a potential issue: We do not know that k is unique if it exists. If there are multiple answers for k, then are there any relationships among them, and what do they imply about Problem C?

E. What follows is the skeleton of a function definition in Python. Complete the definition, so that the function accomplishes its task using recursion, and using no other functions. My solution uses only four more lines of code. If you like, you may rewrite the entire function in another language.

```
# Returns c and d such that a c + b d == gcd(a, b).
# Input: Integer >= 0. Integer >= 0. Assumes that not both are 0.
# Output: List of two integers.
def gcdCombination(a, b):
    if b == 0:

    elif a == 0:

    else:
```