Cryptology Programming Assignement
Due February 9, 2023

Write a function (Python recommended)
def factor(n: int) -> list:

that finds the prime factorization of the integer n. As discussed in class, this should work for any positive integer that fits in 64 bits. Use the naive algorithm (trying small primes), and then, if the number is not completely factored, i.e. it is not a tabulated prime and Miller-Rabin says it is definitely not a prime, use Pollard's Rho Algorithm to factor n. Remember that it is known that Miller-Rabin is infallible for 64-bit integers if the first twelve primes are used!

Your output should be a list whose even-indexed elements are the prime factors, and whose odd-indexed elements are the exponents of those factors. Thus

factor(60)

should return
[2, 2, 3, 1, 5, 1].

You are encouraged to use auxiliary functions to modularize your code!