MATH 173A, SAMPLE FINAL

2 August 2023

Name: ______________________________

Student ID: ___________________________

Instructions: This 60 minute exam will be closed book, closed note, and proctored using Zoom proctoring (your webcam must be enabled). This exam must be hand-written on paper (not on a tablet). No checking cell phones during the exam. Do not type on the computer unless it is to ask the instructor a question through the Zoom chat. Your answers will be graded based on clarity as well as correctness; a correct answer will not receive full credit if the reasoning is difficult to follow. Good luck.

Question	Score	Maximum
0		1
1		
Total		40

One point out of 40 on the final will be assigned for your pdf upload being organized and easy to read (for example, the question numbers should be clearly indicated and should be in consecutive order 1, 2, ...).

1. Assume Bob's RSA modulus is $N = 119 = 7 \cdot 17$ and his encryption exponent is $e = 91$.
   a. Compute Bob's secret decryption exponent d using the Extended Euclidean Algorithm. Show your work. (Express your final answer d as a positive number.)
   b. How many steps will it take for Bob to recover Alice's secret message using the Fast Powering Algorithm, if computing a product modulo N counts as one step?

2. Assume Eve knows an RSA modulus N and an encryption, decryption pair e and d. Describe in as much detail as possible the procedure for using this information to efficiently factor N.

3. For each of the tasks below, identify the computation it requires as well as the fast algorithm used for performing that computation. Some computations and algorithms may be used more than once, and some may not be used. No justification is necessary.

   Steps in the RSA algorithm.

   1. Bob chooses e relatively prime to $\varphi(N)$. (Assume Bob knows $\varphi(N)$ already.)
   2. Alice computes $c = x^e \bmod N$.
   3. Bob computes d such that $e \cdot d \equiv 1 \bmod \varphi(N)$. (Assume Bob knows $\varphi(N)$ already.)
   4. Bob computes x from d and $x^e \bmod N$.

   Computation.

   a. $\gcd$
   b. factorization into primes
   c. e-th root modulo m (for some modulus m)
   d. modular exponentiation
   e. discrete logarithm
   f. multiplicative inverse modulo m (for some modulus m)

   Algorithm.

   i. Fast Powering Algorithm (also called Square-and-Multiply)
   ii. Euclidean Algorithm
   iii. Extended Euclidean Algorithm
   iv. No known fast algorithm

4. Give three examples of computations (like what is described in the "Computations" list above) that are relevant to Math 173A and for which there is no known fast algorithm. (For example, for which we would not expect a modern computer to be able to perform that calculation on a

general 500-bit input.) For each of your three examples, say in a few words how it is relevant to Math 173A.

5. We would like to determine whether or not the number 7039 is prime. Some data is given in the table below.

i	1	2	879	880	1759	1760	3519	3520	5399	5400	7037	7038	8797	8798
$7^i \mod 7039$	7	49	2440	3002	4320	2084	7038	7032	4733	4975	5028	1	4320	2084

   a. If we use the Fermat primality test with base 7, what do we learn? Briefly explain your answer.

   b. If we use the Miller-Rabin primality test with base 7, what do we learn? Briefly explain your answer.

6. The basic idea behind many factorization algorithms is to use the *difference of squares* formula $x^2 - y^2 = (x+y)(x-y)$. There is also a *difference of cubes* formula
$$x^3 - y^3 = (x-y)(x^2 + xy + y^2).$$

   What GCD computation can be performed in the hope that it yields a prime factor of 42281, given the following data? (You do not need to make the GCD computations.)

$$\begin{aligned}
119^3 &\equiv 2^9 \cdot 5^2 && \mathrm{mod} 42881\\
157^3 &\equiv 23 \cdot 461 && \mathrm{mod} 42881\\
316^3 &\equiv 23 \cdot 1607 && \mathrm{mod} 42881\\
332^3 &\equiv 3^3 \cdot 5^4 && \mathrm{mod} 42881\\
407^3 &\equiv 10211 && \mathrm{mod} 42881\\
438^3 &\equiv 3 \cdot 7 \cdot 11 \cdot 103 && \mathrm{mod} 42881\\
471^3 &\equiv 3 \cdot 5 \cdot 1933 && \mathrm{mod} 42881
\end{aligned}$$

7. In the context of RSA, which one of the following is typically easy to compute (meaning it could be completed quickly on a computer)? Briefly explain why that computation is easy.

   i. Given N (but not its factorization $N = p \cdot q$), find $\varphi(N)$, where φ denotes the Euler phi-function.

   ii. Given N (but not its factorization $N = p \cdot q$), find e such that $\gcd(e, \varphi(N)) = 1$.

   iii. Given N (but not its factorization $N = p \cdot q$), e, and x, find $x^e \mod N$.

   iv. Given N (but not its factorization $N = p \cdot q$) and e, find d such that $ed \equiv 1 \mod \varphi(N)$.

8. Assume $N = pq$ is an RSA modulus, assume $a^2 \equiv b^2 \mod N$, and assume $a \equiv -b \mod N$. What can you say about $\gcd(a+b, N)$? Explain your answer.

9. Imagine a version of RSA that uses as its modulus $N = p$ (where p is a prime) rather than $n = p \cdot q$, but all other details are kept the same. Explain why this would not be a secure cryptosystem.

10. What is the primary feature that distinguishes public key cryptosystems (like Elgamal and RSA) from private key cryptosystems?

11. Imagine the United States government has posted a list of one billion prime numbers p which are of a suitable size to produce an RSA modulus. Eve knows that Bob has created his RSA modulus N from these prime numbers, so she begins multiplying pairs of the primes, hoping to find N. So for example, she computes $p_1 \cdot p_2$, then $p_1 \cdot p_3$, ..., then $p_1 \cdot p_{1000000000}$, then $p_2 \cdot p_3$. Before going on to $p_2 \cdot p_4$, she notices it has taken her computer an hour to get this far, and she is only one billionth of the way done. What advice would you give to Eve in her goal of factoring N?