Directions: Solve the following problems. All written work must be your own. See the course syllabus for detailed rules.

1. Make a multiplication table for the unit group $\mathbb{Z}_9^*$.

2. Use the fast power algorithm to compute $2^{300} \pmod{1000}$. Show intermediate powers of 2.

3. Using a computer/calculator only for basic arithmetic operations (addition, subtraction, multiplication, and division), solve for x in $523x \equiv 211 \pmod{591}$. *Show your work.*

4. Let a, b, g, and m be integers such that $g^a \equiv 1 \pmod{m}$ and $g^b \equiv 1 \pmod{m}$. Prove that $g^{\gcd(a,b)} \equiv 1 \pmod{m}$.

5. Extended Euclidean Algorithm

   (a) Implement the Extended Euclidean Algorithm `EEA(a,b)` in a programming environment (such as python) that supports arbitrarily large integers. The function `EEA(a,b)` should compute three integers (d, u, v) such that $d = \gcd(a, b)$ and $au + bv = d$. Submit your code. (It should be fairly short.)

   (b) Let a and b be the following integers, printed with 70 digits per line.

$a =$
2025185487137959423540508380772562680927973512456311300849096944471748
5505766726772707930366549369422613411801280325983886891104728535909893
3801419072872779764693343761793224836462411923645494891165738965986387
9107316214146791748458147856209314460348398362605891740199807133532045
2617699014777368404297975378813885221405067529734456424079570500304517
2755760422978065138058849279208453857066742161325742568958532634610269
8398035254128974089635912788568550620877070262953630120327420284532115
0468512464911322239610529382349997239265386342588817797852386964955150
3527164998410156300500557670162741387427736645186698064 71

$b =$
1339300348729585555209286950897530524327055732306198908722346825714945
5422176226806777638639227377341493059265211131045036933580545130529193
1862856205212767871670892434675830616349424347432208438131751842346613
7524658406492957735167919734867656837096806158107930141867074972509046
9388096800321761480711435989167119769655987176566801234660338210660113
2116085293552473375257715388013350044629767692028057376633282332295356
6241447335340880208523672763456966367752953969399611082967141225940559
9214323596401633880110711650100894637685146090199413385825195891767738
9479503234383237867253101415548018448301542855816642627 83

Both a and b have 617 decimal digits; written in binary, a and b have 2048 and 2047 digits respectively. Using `EEA(a,b)`, find three integers (d, u, v) such that $d = \gcd(a, b)$ and $d = au + bv$. Hint: d is a prime number with 309 decimal digits.

Comment: The number

RSA-2048 =25195908475657893494027183240048398571429282126204032027777137836043662020707595556264018525880784406918290641249515082189298559149176184502808489120072844992687392807287776735971418347270261896375014971824691165077613379859095700097330459748808428401797429100642458691817195118746121515172654632282216869987549182422433637259085141865462043576798423387184774447920739934236584823824281198163815010674810451660377306056201619676256133844143603833904414952634432190114657544454178424020924616515723350778707749817125772467962926386356373289912154831438167899885040445364023527381951378636564391212010397122822120720357

also has 617 digits. Like our numbers a and b, the number RSA-2048 is the product of two large primes. Up until 2007, the security company RSA Laboratories offered $200,000 to anyone who could factor RSA-2048. An efficient method of factoring would break the RSA cryptosystem (which is still widely used). To date, no one has publicly announced a factorization of RSA-2048. Even though factoring large numbers like RSA-2048 seems difficult, finding the greatest common divisor of two large numbers is comparatively easy.