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

1. [JJJ 1.36] Compute the value of $2^{(p-1)/2} \pmod{p}$ for every prime $3 \le p < 20$. (You do not need to show the details of your computation.) Make a conjecture as to the possible values of $2^{(p-1)/2} \pmod{p}$ and prove that your conjecture is correct.

2. [JJJ 1.41] Consider the affine cipher with key $k = (\alpha, \beta)$ whose encryption and decryption functions are given by

$$e_k(m) \equiv \alpha m + \beta \pmod{p}$$
$$d_k(c) \equiv \alpha^{-1}(c - \beta) \pmod{p}$$

   (a) Let $p = 541$ and let $k = (34, 71)$. Encrypt the message $m = 204$. Decrypt the ciphertext $c = 431$.

   (b) Assuming that p is public knowledge, explain why the affine cipher is vulnerable to a chosen plaintext attack. How many plaintext/ciphertext pairs are likely to be needed to recover the private key?

   (c) Alice and Bob decide to use the prime $p = 601$ for their affine cipher. The value of p is public knowledge. Eve intercepts the ciphertexts $c_1 = 324$ and $c_2 = 381$, and she also manages to find the corresponding plaintexts are $m_1 = 387$ and $m_2 = 491$. Determine the private key (α, β) and then use it to encrypt the message $m_3 = 173$.

3. [JJJ 1.43] Let n be a large integer and let $\mathcal{K} = \mathcal{M} = \mathcal{C} = \mathbb{Z}_n$. For each of the functions below, answer the following questions.

   - Is e an encryption function? In other words, is e an injective function?
   - If e is an encryption function, what is the associated decryption function d?
   - If e is not an encryption function, can you make it into an encryption function by restricting the set of keys $\mathcal{K}$ to a smaller, but still reasonably large subset?

   (a) $e_k(m) \equiv k - m \pmod{n}$

   (b) $e_k(m) \equiv k \cdot m \pmod{n}$

   (c) $e_k(m) \equiv (k + m)^2 \pmod{n}$

4. Fast Power Algorithm

   (a) Implement the fast power algorithm `fast_power`(g, a, m) that computes $g^a \pmod{m}$. A recursive implementation will probably not work due to limited stack space provided by most programming environments, so an iterative implementation is recommended. Print out your code for hard-copy submission with this assignment.

(b) Use your code to compute $3^a \pmod{p}$ where

$$
\begin{aligned}
a = \ & 8210362893450651574131722296757356605200830169356578785813400124279769 \\
& 1494399109783730964536462425805314596635511606535459103343485526667825 \\
& 0438301548529598352882812656428385418093139636082570658299829788458938 \\
& 9083806978918503471627935113458406773943290254539587711017833207101432 \\
& 5550216588266041278200122901497676684219641814803583019462296990591112 \\
& 6993897921967321817986478442195134063060064678359754030334303960856670 \\
& 3483740856368972704219205926958570459413034458778737766317296872902209 \\
& 6773871939461088592535234193912878536049772231013533830752722286864466 \\
& 4552070651137382023448891804352986044611267798726544229245 1
\end{aligned}
$$

$$
\begin{aligned}
p = \ & 8651150043557325511450175101264786208775439422974363414691402687392683 \\
& 8617465807737218060864732534779835429286856745443958175654305684571482 \\
& 1187406006409811660901887625785757970044918073643563547474989534274434 \\
& 9444036680156887905621835235579495131134575217730594952389382011952799 \\
& 2513893144681242141885337139933240910034594095241655333780810035436287 \\
& 1109951870215818680246819107214492903711323010930843097199754799801451 \\
& 3126712689350813090877776469762068452734380642840997344165805371355959 \\
& 3568737196797196120707485389647731118058940157480809918125993307434175 \\
& 6543771264188237265473464737563681021550920284059941660272 9
\end{aligned}
$$

Hint: to check your work, the answer begins “860...” and the sum of the digits is 2765.