ASSIGNMENT 3: SOLUTIONS

Question 1.

Solution.
- Since $493 = 17 \cdot 29$, we find solutions mod 17 and mod 29 respectively and then build back up to solutions mod 493 using CRT giving $x \equiv 157, 191, 302, 336 \pmod{493}$.
- Reduce mod 13, find solutions and apply Hensel's Lemma. We get $x \equiv 1, 3, 9 \pmod{13}$. Check to see if we can apply Hensel's Lemma:
  - $f'(1) = 3(1)^2 \equiv 3 \pmod{13}$
  - $f'(3) = 3(3)^2 \equiv 1 \pmod{13}$
  - $f'(9) = 3(9)^2 \equiv 9 \pmod{13}$

  Repeatedly applying Hensel's Lemma then gives $x \equiv 1, 150432, 220860 \pmod{13^5}$.

□

Question 2.

Solution. This can be done by (clever or not so clever) process of elimination. For example, in the first case one obtains a primitive root r, say 3, by trial and error, and then uses it to find the rest of the primitive roots as $\left\{r^t \;\middle|\; t \nmid \phi(17) = 16\right\}$. A similar process works for $n = 27$.
- modulo 17:
$$3, 5, 6, 7, 10, 11, 12, 14$$
giving $\phi(\phi(17)) = 8$ primitive roots.
- modulo 27:
$$2, 5, 11, 14, 20, 23$$
giving $\phi(\phi(27)) = 6$ primitive roots.

□

Question 3.

Solution. From the previous question, $g = 3$ and the discrete log of 12 to the base 3 is $13 \pmod{16}$. □

Question 4. This question was not for marks as it wasn't really a question.

Solution. Just plug it in and see what happens. □

Question 5.

Solution. The problem is that a^{F_n} for $n \geq 5$ gets too big and we get an error. There are a couple reasonable ways to fix this: use binary exponentiation, or simply replace a in the Fermat test with `Mod(a, N)`.

To show that F_n is composite for $5 \leq n \leq 12$, choose some base a for which `ft(a, Fn)` is non-zero. For example, $a = 3$ will work and gives

$$\begin{aligned}
F_5 &: 497143883\\
F_6 &: 7810004532687335890\\
F_7 &: 142534992508324304154873225278075561694\\
F_8 &: 107\ldots102\\
F_9 &: 133\ldots639\\
F_{10} &: 808\ldots986\\
F_{11} &: 259\ldots161\\
F_{12} &: 847\ldots043.
\end{aligned}$$

□

Question 6.

Solution. Say, using your code from Exercise 5, you decided to check the result of the the Fermat test for $0 \leq n \leq 50$ with base 2. You notice that you get 0 every time. Based on this evidence you should conjecture that F_n is a Fermat (pseudo)prime to the base 2 for all n. Then try and prove (or disprove) it. It turns out that it is a straightforward exercise to show that F_n is a Fermat (pseudo)prime to the base 2 for all n.

One way to do this is to show that $2^{2^m} \equiv 1 \pmod{F_n}$ for $n < m$. Then setting $m = 2^n$ and multiplying by 2 gives the desired result. □

Question 7.

Solution. ($\Rightarrow$): Let $p^k \mid\mid n$, $(a,n)=1$. Then by assumption $a^{n-1} \equiv 1 \pmod{n}$, and so $a^{n-1} \equiv 1 \pmod{p}^k$. By choosing a to be a primitive root mod p^k, it follows that $\mathrm{ord}_p(a) = p^{k-1}(p-1) \mid n-1$. Since $p^k \mid n$, $p^{k-1} \mid n-1$ only if $k=1$. This proves that n has the desired factorization.

($\Leftarrow$): Let $(a,n)=1$. For each i such that $p_i \mid n$, $(a,p_i)=1$ so $\mathrm{ord}_{p_i}(a) \mid p_i - 1 \mid n-1$ and $a^{n-1} \equiv 1 \mod p_i$. Applying the CRT gives $a^{n-1} \equiv 1 \pmod{n}$. □

Question 8.

Solution. For $m > n > 0$, we have the that p-adic distance betwen a_m and a_n is

$$|a_m - a_n|_p = |p^{n+1}(1 + \ldots + p^{m-n-1})| = p^{-(n+1)}.$$

Thus, for any $\varepsilon > 0$, we can choose N large enough such that for $m > n > N$, $p^{-(n+1)} < \varepsilon$. This shows that (a_n) is a Cauchy sequence and hence converges in $\mathbb{Q}_p$.

Let $a = \lim_{n\to\infty} a_n$. Then

$$a(1-p) = (1 + p + \ldots) - (p + p^2 + \ldots) = 1$$

and $a = \frac{1}{1-p} \in \mathbb{Q}_p$.

Similarly, one shows that

$$|b_m - b_n|_p \leq p^{-(n+1)}$$

and so (b_n) is a Cauchy sequence. Write $b = \lim_{n\to\infty} b_n$. Then show that

$$b - 1 = (a-1)(1 + p + p^2 + \ldots) = a(a-1)$$

and $b = \frac{1-p+p^2}{(1-p)^2}$. □

Question 9. There were surprisingly few attempts to answer this question. Don't be intimidated by large numbers! This should have been pretty easy to do, given that it sounds like you covered it in class...

Solution. The first observation to make is that since we have $\phi(n)$, we can break RSA without having to factor n. (You should think about why this is).

Additionally, if we wanted to factor n, the knowledge of $\phi(n)$ will help us. The idea is as follows. We can use our knowledge of $\phi(n)$ to try and find a square root, call it t, of $1 \pmod{n}$. Then we have that $(t+1)(t-1) \equiv t^2 - 1 \equiv 0 \pmod{n}$, so both $t+1$ and $t-1$ divide n, and we can recover them via gcd calculations (as these gcd calculations are essentially an application of the Euclidean algorithm which is polynomial time in n, your computer will have no problem handling these large numbers).

Note that if we recover a trivial square root of 1, (*i.e.* $t \equiv \pm 1 \pmod{n}$), then we unfortunately recover the trivial factorization of n by this method.

As alluded to at the beginning, this method is based on the fact that we can use our knowledge of $\phi(n)$ to find square roots of 1. How do we do this? First, observe that $\phi(n)$ is necessarily even—say $2^m \mid\mid \phi(n)$ for some $m \geq 1$. Pick a base, a such that $(a,n)=1$. Then consider

$$\left(a^{\frac{\phi(n)}{2^m}}\right)^2, \left(a^{\frac{\phi(n)}{2^m}}\right)^{2^2}, \ldots, \left(a^{\frac{\phi(n)}{2^m}}\right)^{2^m} = a^{\phi(n)}$$

(of course, all $\pmod{n}$). Note that you can compute this by repeatedly squaring the first term. Do this until $\left(a^{\frac{\phi(n)}{2^m}}\right)^{2^i} \equiv 1 \pmod{n}$ (this process will terminate since $a^{\phi(n)} \equiv 1 \pmod{n}$). Necessarily $t = a^{\frac{\phi(n)}{2^{m-i+1}}}$ is a square root of $1 \pmod{n}$. Taking $\gcd(t+1,n)$ and $\gcd(t-1,n)$ will now return a factorization of n.

Assembled correctly, this method will give us a probabilistic factoring algorithm. The probabilistic part depends on the choice of base that we used, as some bases will recover a trivial factorization of n.

Let's finally apply this to our problem. In this case, note that $2^{32} \mid\mid \phi(n)$. Pick a base, say $a=2$ (easy choice since n is odd). We find that $t = a^{\frac{\phi(n)}{2^{32}}}$ and the above process returns a non-trivial factorization of n:

$$\begin{aligned} p = \gcd(t+1,n) =& 3783489102334656478591844213346155325437497471853216340862 19 \\ qr = \gcd(t-1,n) =& 6992607918276431076656636215550628566131511790062497678057 18/ \\ & 6333046868557225391899831205434145379119885761899420602984 85/ \\ & 0896721574517082063008260327. \end{aligned}$$

Here we're just guessing for the time being that p is one of desired primes since it's the smaller one. We have two obvious choices now. We can simply try to factor n again using a different base and hope that we get a different

factorization, or apply the above procedure to $n = qr$. Choosing the former, we repeat the above process with $a = 5$. Here, we get $t = a^{\frac{\phi(n)}{2^{30}}}$ and

$$\begin{aligned} q = \gcd(t+1, n) =& 185323955009471744507093833849366798683834244443114056794632807824057962331639 77 \\ pr = \gcd(t-1, n) =& 142757884993042259059608161461508419947368566792001810015130749079026773/ \\ & 826715530536524914274407123955035538493756929910554032966 9. \end{aligned}$$

Since we know that $n = pqr$ it follows that

$$\begin{aligned} p =& 378348910233465647859184421334615532543749747185321634086219 \\ q =& 185323955009471744507093833849366798683834244443114056794632807824057962331639 77 \\ r =& 377318081621938460678418953889955311049944229578257670222228038491755 1. \end{aligned}$$

□

Question 10. Note: This question wasn't marked. For those that did do it, I observed that many of you implicitly used unproven (and often wrong) claims about the $|a_n|_5$ (aka. divisibility properties of a_n by 5).

Solution. Important observation: if a_n is in lowest terms and 5 divides its numerator, then $a_n^2 + 1 \equiv 1 \pmod 5$. This is used in the proof that 5^n divides the numerator of $a_n^2 + 1$ when it is in lowest terms, and once we establish that $5^n \mid a_n^2 + 1$, it provides a lower bound on the 5-adic size of a_n, namely $|a_n|_5 \geq 1$. (As far as I observed no one did this!)

A straightforward induction proof (using the preceeding observation) shows that $5^n \mid b_n = a_n^2 + 1$. It is now easy to show that the sequence (b_n) and converges to 0. It follows that if we can show that (a_n) is Cauchy, then a_n converges to the square-root of -1 and we are done.

Now show that

$$|a_{n+1} - a_n| = \left| \frac{1 + a_n^2}{a_n} \right| = \frac{|b_n|}{|a_n|} \leq |b_n|$$

using $|a_n| \geq 1$. For $m > n$,

$$a_m - a_n = a_m - a_{m-1} + a_{m-1} - \ldots - a_{n+1} + a_{n+1} - a_n$$

and $|a_m - a_n| \leq \max\{|a_{i+1} - a_i| \mid i = n, \ldots, m-1\} \leq \max\{|b_i| \mid i = n \ldots, m-1\} \leq |b_n|$. Since we have shown (hypothetically) that $|b_n|$ converges to 0, (a_n) is Cauchy. □