
Problem Set 4 Solutions

Submit this problem set in PostScript, PDF, or MS Word format to `6857-submit@csail.mit.edu` before lecture on the due date. We have provided templates for L^AT_EX and Microsoft Word on the course website. Each solution must appear on separate sheets of paper. Mark the top of each sheet with your name(s), the course number (6.857), the problem set number and question, and the date.

You are to work in groups of three or four people and should submit a single set of solutions for all problems parts designated [**Group**]. You should turn in a separate, individual solution to any problems designated [**Individual**].

We may distribute our favorite solution to each problem as the “official” solution. If you do not wish for your homework to be used as an official solution, or if you wish that it only be used anonymously, please note this on your homework.

Problem 4-1. MoogTMle Want Ads [Group]

Flush with cash after its recent IPO, MoogTMle has launched a campaign to hire top computer science students. To attract talented students, MoogTMle posts the following puzzle on a billboard:

{ First 20-digit Sophie Germain Prime in e }.com

A *Sophie Germain prime* is a prime number p such that $2p + 1$ is also prime. Similarly, what is referred to as a *safe prime* is a prime q such that $(q - 1)/2$ is a prime. We have provided Python code for basic number theory functions at <http://crypto.csail.mit.edu/classes/6.857/code/numbthy.py>. Code to generate a fixed number of digits of e is available at <http://crypto.csail.mit.edu/classes/6.857/code/e.py>.

- (a) What URL would MoogTMle’s billboard lead to?

Solution: The prime $p = 18491463140934317381$ has position 850 and length 20 and $2p + 1 = 36982926281868634763$ is prime.

- (b) Find the longest Sophie Germain prime you can that occurs as a sequence of consecutive digits in the digits of $e = 2.718281828459045235\dots$. Turn in the length of the prime (in digits), the value of the prime, and its position in the decimal expansion of e . For example the Sophie Germain prime 23 has length 2 and is at position 17, and the Sophie Germain prime 6059563073 has length 10 and is at position 150.

Solution: Individual solutions differed by how much computation time they spent.

- (c) Estimate the length of the longest Sophie Germain prime p you would expect to find among the first million digits of e , using the known density of primes as a guide.

Solution:

This problem is deceptively hard. In our envisioned solution, we approximated that a k digit number x would be prime with probability $1/\ln(x) = 1/(k \ln(10))$ by the prime number theorem. If it were independent, $2x + 1$ would be prime with probability $1/((k + 1) \ln(10))$, so we’d approximate that a number would be a SG prime with probability $1/(k \ln(10))^2 = c/k^2$, where $c \approx .188$. If we are searching the first n digits of a number for primes of length k , there would be $n - k + 1$ trials.

We thought to look for a k where the expected number of primes is 1. This turns out that for $n = 1000000$, this is $k \approx 867$. This is *very wrong*.

The fact is, there are a lot of trials between lengths $k + 1$ and n – about $O((n - k)^2)$. We need k to be large enough so that the probability that *any* one of these candidates lengths contains a SG prime is still low. If we make a rough over-estimate that each of the candidates has a $1/k^2$ chance of being prime, our upper-bound of k would be a constant fraction of n . We can approximate the expected number of SG primes of length greater than k as follows:

$$C * \sum_{j=K+1}^N \frac{N - j + 1}{j^2}$$

Unfortunately, this does not have an easily solvable closed form. If we solve this numerically for $n = 100000$ and search for a k such that this sum is less than .5, we find that we actually need a k of size approximately 225,000. Essentially, what we solved for was: "What is largest k such that you can expect to find at least one SG prime of that length in the first million digits of e ?", rather than the stated answer.

Min Wu, Philip Rha, Saba Gul, Yuran Lu pointed out our erroneous thinking and gave this solution:

"It is widely believed that there are approximately $d(N) = 2C_2N/(\ln N)^2$ Sophie Germain primes less than N , where C_2 is the twin prime constant (approximately 0.6601618158). We can model the first 1000000 digits of e as a set of $1000000 - k$ random k -digit numbers. There are a total of 10^k different k -digit numbers, so the probability that an arbitrary k -digit number is a Sophie Germain prime is $f(k) = d(10^k)/10^k$. The probability $g(k)$ that there is at least one Sophie Germain prime of length k in the first million digits of e is one minus the probability that there are none. Thus, $g(k) = 1 - (1 - f(k))^{1000000 - k}$. The probability $h(k)$ that the largest Sophie Germain prime in the first million digits of e is of length k is the probability that we can find Sophie Germain prime of length k , times the probability that we can't find any longer ones. Thus, $h(k) = g(k) \cdot \prod_{i=k+1}^{1000000} (1 - g(i))$. Now, we can calculate the expected largest Sophie Germain prime in the first million digits of e by taking the expectation, $E = \sum_1^{1000000} (i \cdot g(i))$. Plugging this into MapleTM, we get can get the exact value of E . Unfortunately, MapleTM's math engine crashes when even trying to calculate a single value for $h(k)$, so we don't have an exact value. However, looking at some values MapleTM returns for $g(k)$, we see that $g(60000)$ and $g(70000)$ are both around $1/10000$. This makes it reasonably probably that for some k between 60000 and 70000, there is a Sophie Germain prime of length k in the first million digits of e . I expect the longest prime to be of length around 100000."

- (d) Now find the longest safe prime p' you can in the digits of e such that 3 is a generator modulo p' . That is, find the biggest p' you can in e such that $p' = 2p + 1$ and $\langle 3 \rangle = \mathbb{Z}_{p'}^*$.

Solution: Oops! The number 3 is a generator only for the safe prime 7.

We accepted either "7" or the largest safe prime you could find such that 5 was its generator.

- (e) Estimate the length of the longest safe prime $p' = 2p + 1$ you would expect to find among the first million digits of e such that 3 is a generator modulo p' .

Solution: There is only one prime with 3 as its generator.

If you tried 5, note that for safe primes $p = 2q + 1$, $(q-1)$ elements are generators, since $\Phi(p - 1) = \Phi(2q) = q - 1$ elements are generators, so there is about a $1/2$ chance that 5 is a generator of a safe prime, assuming that 5 is "random". (As we learned from our experience with 3, this isn't always a good assumption.)

Problem 4-2. ElGamal Signatures [Individual]

Recall the ElGamal public-key cryptosystem presented in class. Given system parameters prime p and generator g , individuals will generate a public key y by choosing a random secret key $x \in \mathbb{Z}_p^*$ and computing $y = g^x \bmod p$. Given a public key y , message m is encrypted by selecting a random $r \in \mathbb{Z}_p^*$ and computing $E(m, r, y) \rightarrow (g^r \bmod p, y^r m \bmod p)$. Someone who knows y 's corresponding secret key x can decrypt a ciphertext (u, v) by computing $D(u, v, x) \rightarrow \frac{v}{u^x} \bmod p$.

- (a) Ben Bitdiddle gets lazy and reuses the same r to encrypt multiple messages with ElGamal. Suppose Alyssa P. Hacker happens to know the plaintext message corresponding to one of Ben's ciphertexts. Explain how Alyssa is able to read all of Ben's messages.

Solution: Let Ben's ciphertexts be of the form $c_j = (g^r, y^r m_j)$. Suppose Alyssa knows m_j . She can then divide this value out of c_j to obtain (g^r, y^r) . Now that she knows y^r , she can trivially recover all the other messages.

One can also compute digital signatures in the ElGamal cryptosystem. To sign a message m , someone who knows the private key x first selects a random $r \in \mathbb{Z}_p^*$ and computes the signature:

$$\begin{aligned} S(m, r, x) &: s \leftarrow g^r \bmod p \\ & t \leftarrow \left(\frac{m - sx}{r} \right) \bmod (p - 1) \\ & \text{Return signature } (s, t) \end{aligned}$$

A signature (s, t) on message m is verified as follows:

$$\begin{aligned} V(m, s, t) &: V_1 \leftarrow s^t y^s \bmod p \\ & V_2 \leftarrow g^m \bmod p \\ & \text{Accept signature if } V_1 = V_2 \end{aligned}$$

- (b) Ben gets lazy and stops generating new random values r for each signature. He signs two messages m_1 and m_2 using the same random value r . This produces two signatures (s_1, t_1) and (s_2, t_2) . Show how given m_1 , m_2 and Ben's signatures, Alyssa can recover Ben's secret key x . Assume that $(t_1 - t_2)$ and s_1 are relatively prime to $(p - 1)$.

Solution:

$$\begin{aligned} (s_1, t_1) &= \left(g^r, \left(\frac{m_1 - s_1 x}{r} \right) \bmod (p - 1) \right) \\ (s_2, t_2) &= \left(s_1, \left(\frac{m_2 - s_1 x}{r} \right) \bmod (p - 1) \right) \\ t_1 - t_2 &= \frac{m_1 - m_2}{r} \bmod (p - 1) \end{aligned}$$

Because $d = (t_1 - t_2)$ generates \mathbb{Z}_{p-1} , it must have some inverse d^{-1} , which we can efficiently find knowing p and q . We can then compute $(m_1 - m_2)d^{-1} = r$. We can now solve for $m_1 - s_1 x \bmod (p - 1)$. Since we assume that s_1 is relatively prime to $(p - 1)$, we can solve for x .

Problem 4-3. Voter-Verifiable Elections [Group]

- (a) In Chaum's system presented in lecture, the voter gets two ballots at the polling place. One is used to actually vote; the other is used to test that the (hidden) mappings on the bulletin board are correct. Why is this more desirable than having precinct officials test a bunch of random ballots initially and then only give a single ballot to each voter?

Solution: The fact that the voter chooses what to test is why Chaum calls the system "voter-verifiable elections". If a precinct official is corrupt, they can collude with the trustees to "test" ballots that will pass while actual votes are processed incorrectly in the mixnet. If the voter has the power to choose which ballot is tested, the voter retains the ability to verify for himself that the mixnet is operating correctly.

- (b) Recall that the "bulletin board" in Chaum's system as presented in lecture contains two separate batches of envelopes and a batch of unobscured ballots. The first trustee reveals the mapping between half of the ballots in the first batch and the corresponding half in the second batch; the second trustee reveals the mapping between the *other* half of the second batch and the corresponding unobscured ballots.

In his paper linked from the course website, however, Chaum describes a system that uses many trustees in the chain; each reveals half of the mappings they know in such a way that a complete path from the first envelope to the unobscured ballot is never revealed. What is benefit of using more than two levels of trustees?

Solution: If the two trustees collaborate, they can violate voter privacy. Indeed, even if one of the two is corrupt, he can reveal how half of the voters voted. With multiple trustees, the trust is distributed such that even if one trustee reveals *all* of his mappings (publicly, or to a third party like the Mafia), a full path is not revealed.

- (c) What is one security property of an electronic voting system that is *not* addressed by Chaum's system?

Solution: One possible answer is that of voter authentication or preventing a voter from voting multiple times.