Carnegie Mellon University Instructor: Venkatesan Guruswami
15-252: More GITCS Term: Spring 2020

Problem Set 12

Due date: Thursday, April 23

Instructions

- You are allowed to collaborate with one other student taking the class, or do it solo.
- Collaboration is defined as discussion of the lecture material and solution approaches to the problems. Please note that *you are not allowed to share any written material and you must write up solutions on your own.* You must clearly acknowledge your collaborator in the write-up of your solutions.
- Solutions must be typeset in LaTeX and emailed to the TA.
- You should not search for solutions on the web. More generally, you should try and solve the problems without consulting any reference material other than what we cover in class and any provided notes.
- Please start working on the problem set early. Though it is short, the problem(s) might take some time to solve.

1. Imagine that you are the CEO of a promising new start-up, and the launch of an exciting new web product is impending. The product is set up so that its release requires a secret code $s \in \{0, 1, 2, \ldots, 15485862\}$[1] that you as CEO picked and only you know. Unfortunately, due to events beyond your control, you are forced to self-quarantine for two weeks around the planned launch date (and for security reasons you don't want to call in with or email the code).

   (a) You have a trusted team of 7 senior managers, but you are not willing to share the code with them lest one of them misuses it when the product is not fully ready for release. So you come up with a scheme to break the secret into pieces $m_1, m_2, \ldots, m_7$ (with m_i revealed privately to manager i) in the following controlled way:
   - If a majority (i.e., at least 4) of the senior managers agree on the release, then they can put their pieces together to correctly determine the code s and release the product.
   - On the other hand, if less than 4 managers collude and share their pieces, they will have no idea about the secret code s, in the sense that every possible value for s in $\{0, 1, 2, \ldots, 15485862\}$ will be consistent with their knowledge. (We assume that attempting a release with an incorrect code will be devastating so no one will attempt that.)

   Can you come up with a scheme to achieve these goals?

   (b) Upon further reflection, you decide that you should also involve a group of 15 distinguished engineers in this key decision. So now you would like a scheme to break your secret code into 7 pieces $m_1, m_2, \ldots, m_7$ for the managers and 15 pieces $e_1, e_2, \ldots, e_{15}$ for the engineers to ensure the following:

[1] 15485863 is the million'th prime number.

- If a majority (i.e., 4 or more) of managers *and* a majority (i.e., 8 or more) of engineers agree on the release, then they can put their pieces together to correctly determine the code s and release the product.
- In all other scenarios, namely if at most 3 managers favor the release or at most 7 engineers favor the release, then pooling the pieces of everyone favoring the release, they obtain no information about s (in the sense that every possible value for $s \in \{0, 1, \ldots, 15485862\}$ will be consistent with their knowledge).

What's your new scheme to achieve these goals?

(c) Upon even further reflection, you decide that you also want to allow the release of the product if *all* senior managers unanimously agree to do so, provided at least *two* engineers are also agreeable to the idea and share their pieces.

That is, the pieces of all senior managers and *any* two of the engineers suffices to recover s. But even with the pieces of all the managers and the piece of any single engineer, one should be able to learn nothing about s. Likewise, if even if just one manager doesn't share their piece, one gets no information about s unless a majority of the engineers share their pieces.

How will you modify the scheme from Part (b) to have this additional property (on top of the being able to recover the secret code when a majority of both managers and engineers share their pieces as in Part (b))?

For all parts, in addition to describing your scheme, you should provide a concise but clear explanation of why your proposed scheme has the stipulated properties.

Hint: Try to encode the secret as a coefficient of a polynomial, and the codes as the evaluations of the polynomial. Let f be a polynomial of degree d. If we know the value of f at $d+1$ points, then f is uniquely determined, and for every evaluation at $d+1$ points, there is a matching f.

For the second part, you can use two different polynomials.

For the final part, you can use two different schemes, one from the second part, and the other that works with the new combination.