HW 8 CMSC 456. DUE Nov 12
SOLUTIONS
NOTE- THE HW IS FIVE PAGES LONG

1. (0 points) READ the syllabus- Content and Policy. What is your name? Write it clearly. What is the day of the final?

GOTO NEXT PAGE

2. (Read the slides on LWE Diffie-Hellman) (WARNING: this problem continues onto the next page. Write the following programs. I highly suggest using python's numpy library to implement this easier.)

   (a) $GENMATRIX(n, p)$: gen a rand $n \times n$ matrix of elements of $\{0, \ldots, p-1\}$. We view entries as elements of $\mathbb{Z}_p$.
   (See numpy.randint - it can generate random integer arrays. This can be done with one line)

   (b) $GENERR(n, p)$: gen a rand n-vector of elements of $\{0, 1, p-1\}$.

   - Prob of a 0 is $\frac{n-2}{n}$
   - Prob of a 1 is $\frac{1}{n}$
   - Prob of a $p-1$ is $\frac{1}{n}$

   We view entries as elements of $\mathbb{Z}_p$.
   (You can generate a random floating-point number in the range [0,1) with numpy.random)

   (c) $GENDATA(n, p)$: (This is pseudocode, NOT actual python code — you will have to translate this into actual code)

   i. $A := GENMATRIX(n, p)$
   ii. $\vec{y} := GENERR(1/n)$
   iii. $\vec{e_y} := GENERR(1/n)$
   iv. $\vec{x} := GENERR(1/n)$
   v. $\vec{e_x} := GENERR(1/n)$
   vi. $a = \vec{y}A\vec{x} + (\vec{y} \cdot \vec{e_x})$
   (numpy.mod(numpy.dot($\vec{y}, A$), p) can be used to preform a dot product over modulo p)
   vii. $b = \vec{y}A\vec{x} + (\vec{x} \cdot \vec{e_y})$
   viii. if $a \in \{0, \ldots, \lfloor p/4 \rfloor\} \cup \{\lfloor 3p/4 \rfloor, \ldots, p-1\}$ $\hat{a} = 0$, else $\hat{a} = 1$.
   ix. if $b \in \{0, \ldots, \lfloor p/4 \rfloor\} \cup \{\lfloor 3p/4 \rfloor, \ldots, p-1\}$, $\hat{b} = 0$, else $\hat{b} = 1$.
   x. the variable `agree` is YES if $\hat{a} = \hat{b}$ and NO otherwise.
   xi. Your code will output a tuple or an array of $[a, b, \hat{a}, \hat{b}, agree]$

GOTO NEXT PAGE

xii. Here is a sample of printing your output:
OUTPUT STARTS HERE
$n = 5$, $p = 17$, $N = 5$.

a	b	$\hat{a}$	$\hat{b}$	`agree`
3	2	0	0	YES
10	12	1	0	NO
7	9	1	1	YES
1	0	0	0	YES
5	6	0	0	YES

$\hat{a}$ and $\hat{b}$ agree 80% of the time.
END OF OUTPUT
*****Note that $N = 5$ and there are five lines.*****

GOTO NEXT PAGE

NOTE- For the above problems no points are given but submit anyway to help us grade the problems below which ARE for points.

NOTE- THIS IS STILL PROBLEM TWO:

(a) (0 points- But do it to check your program. Do not give us the output). Run program $GENDATA$ with the following inputs.

i. $n = 4$, $p = 19$

ii. $n = 10$, $p = 23$

(b) Make a method to take $[n, p, N]$ as input and output (1) the percent of agreement (called peragree) (2) the percent of the time they agree and the bit is 0 (called peragree0) (3) the percent of the time they agree and the bit is 1 (called peragree1). Call this method $GENDATA2(n, p, N)$.

(c) Make a method to take a LIST of $[n, p, N]$ inputs and output a table of the n, p, N and peragree, peragree0, peragree1. Call this method $GENDATA3(n, p, N)$. A sample output is:

n	p	N	`peragree`	`peragree0`	`peragree1`
5	17	5	80	45	55
6	19	5	75	48	52
7	23	10	90	49	51

This can be generated by printing

"n\tp\tN\tperagree\tperagree0\tperagree1"

If you follow this format for the entries of the table, your results should line up.

(d) (25 points) Run $GENDATA3$ using all $5 \leq n \leq 100$ where $n \equiv 0 \pmod 5$, primes $p \in \{7, 11, 31, 101\}$ and $N = 1000$.

(e) (5 points) Note the highest and lowest peragree values, as well as the highest and lowest peragree0 values.

(f) (10 points) Try changing $GENERR$ to use the following probabilities instead:

i. Try (Prob of 0 is $1 - \frac{2}{n^2}$, Prob of 1 and Prob of $p - 1$ are $\frac{1}{n^2}$)

ii. Try (Prob of 0 is $\frac{1}{2}$, Prob of 1 and Prob of $p - 1$ are $\frac{1}{4}$)

iii. Try (Prob of 0 is $\frac{n-4}{n}$, Prob of 1 and Prob of $p - 1$ are $\frac{2}{n}$)

Rerun $GENDATA3$ with the probabilities above. Look at the peragree and peragree0 values for these as well as the original distribution.

Based on this, which distribution do you think works best (has high agreement and peragree0 close to $\frac{1}{2}$)? Give a brief justification why it's better than the other distributions.

SOLUTION TO PROBLEM TWO

Omitted.

GOTO NEXT PAGE

3. (30 points). We assume the secret is of length n. For the problems below explain it so that someone who has never seen secret sharing can understand it (This is not hypothetical. Two of the TAs do not know secret sharing (except what the goal is). Lets call them J1 and J2. J1 is grading this problem and will learn this protocol from you!)

   (a) (15 points) Describe the random-string $(2, 5)$ secret sharing scheme. You must describe both what Zelda gives out, and how any two people can determine the secret. How many strings does each person get?

   (b) (15 points) Describe the polynomial $(2, 5)$ secret sharing scheme. You must describe both what Zelda gives out, and how any two people can determine the secret. How many strings does each person get?

SOLUTION TO PROBLEM THREE

We call the people A_1, A_2, A_3, A_4, A_5.

1) String Method.

Zelda gives out shares:

(a) Zelda has secret s.

(b) Zelda generates random strings

$r_{1,2}$, $r_{1,3}$, $r_{1,4}$, $r_{2,3}$, $r_{2,4}$, $r_{2,5}$, $r_{3,4}$, $r_{3,5}$, $r_{4,5}$.

(c) Zelda gives A_1: $(1, 2, r_{1,2})$ and A_2: $(1, 2, s \oplus r_{1,2})$.

(d) Zelda gives A_1: $(1, 3, r_{1,3})$ and A_3: $(1, 3, s \oplus r_{1,3})$.

(e) Zelda gives A_1: $(1, 4, r_{1,4})$ and A_4: $(1, 4, s \oplus r_{1,4})$.

(f) Zelda gives A_1: $(1, 5, r_{1,5})$ and A_5: $(1, 5, s \oplus r_{1,5})$.

(g) Zelda gives A_2: $(2, 3, r_{2,3})$ and A_3: $(2, 3, s \oplus r_{2,3})$.

(h) Zelda gives A_2: $(2, 4, r_{2,4})$ and A_4: $(2, 4, s \oplus r_{2,4})$.

(i) Zelda gives A_2: $(2, 5, r_{2,5})$ and A_5: $(2, 5, s \oplus r_{2,5})$.

(j) Zelda gives A_3: $(3, 4, r_{3,4})$ and A_4: $(3, 4, s \oplus r_{3,4})$.

(k) Zelda gives A_3: $(3, 5, r_{3,5})$ and A_5: $(3, 5, s \oplus r_{3,5})$.

(l) Zelda gives A_4: $(4, 5, r_{4,5})$ and A_5: $(4, 5, s \oplus r_{4,5})$.

Recovery: If A_i and A_j get together

A_i has $(i, j, r_{i,j})$ and

A_j has $(i, j, s \oplus r_{i,j})$.

They compute:

$$r_{i,j} \oplus s \oplus r_{i,j} = s$$

How many strings does A_1 have: $r_{1,2}$, $r_{1,3}$, $r_{1,4}$, $r_{1,5}$. So FOUR.

2) Poly Method.

Zelda gives out shares:

(a) Zelda has secret s.

(b) Zelda finds a prime p such that $p \sim s$.
Zelda generates random strings r.
Zelda forms polynomial

$$p(x) = rx + s$$

(c) Zelda gives A_1 $p(1)$.

(d) Zelda gives A_2 $p(2)$.

(e) Zelda gives A_3 $p(3)$.

(f) Zelda gives A_4 $p(4)$.

(g) Zelda gives A_5 $p(5)$.

Recovery:

If A_i and A_j get together

A_i has $f(i)$,

A_j has $f(j)$.

Since two points make a line they can determine the line f and obtain its constant term which is s.

Since A_i just gets $f(i)$, each person gets ONE string.

END OF SOLUTION TO PROBLEM THREE

4. (30 points) (This is not something I did in class so it may require some more thought.) For the problems below explain it so that someone who has never seen secret sharing can understand it (This is not hypothetical. Two of the TAs do not know secret sharing (except what the goal is). Lets call them J1 and J2. J2 is grading this problem and will learn this protocol from you!)

   Zelda has a secret $s \in \{0,1\}^n$. She wants to share a secret with Alice, Bob, Carol, Donna, Edgar, Frank (A,B,C,D,E,F) such that the following happens:

   If Alice, Bob and ANY TWO of $\{C, D, E, F\}$ get together then they can find the secret (and of course any superset of that). No other set can find the secret.

   Give a scheme that achieves this. The security must be information theoretic. Both say what Zelda does and what the various combinations of people do. Discuss what happens if any set other than those above gets together.

   SOLUTION TO PROBLEM FOUR

   (a) Zelda generates a random $r_1, r_2 \in \{0,1\}^n$.

   (b) Zelda lets $r_3 = s \oplus r_1 \oplus r_2$.

   (c) Zelda gives A r_1 and B r_2.

   (d) Zelda does a $(2,4)$ Secret Sharing scheme with string r_3 and people $\{C, D, E, F\}$.

   Recovery:

   If A, B, and (say) C and D get together.

   A has r_1.

   B has r_2.

   C and D do their $(2,4)$ secret sharing to get r_3.

   They can find:

$$r_1 \oplus r_2 \oplus r_3 = r_1 \oplus r_2 \oplus s \oplus r_1 \oplus r_2 = s$$

   Similar for A and B and any two of $\{C, D, E, F\}$.

Other sets can't get the secret:

IF A gets together with C, D, E, F all they have is r_1 and $r_3 = s \oplus r_1 \oplus r_2$. If they XOR those together all they get is $s \oplus r_2$ which looks random to them. They can't do anything else!

IF B gets together with C, D, E, F all they have is r_2 and $r_3 = s \oplus r_1 \oplus r_2$. If they XOR those together all they get is $s \oplus r_1$ which looks random to them. They can't do anything else!

IF A, B, C get together all they have is r_1, r_2, and some useless share of r_3 – useless since C can't learn ANYTHING from it.

END OF SOLUTION TO PROBLEM FOUR