HW 1 CMSC 456. Morally DUE Sep 21
NOTE- THE HW IS EIGHT PAGES LONG

1. (10 points)

   (a) What is the day and time of the midterm?

   (b) What is the day and time of the final?

   (c) What is the *dead-cat policy*?

GOTO NEXT PAGE FOR NEXT PROBLEM

2. (15 points) Klingons use an alphabet of 30 letters. Klingons want to use the affine cipher, so x maps to $ax + b \pmod{30}$. List out all of the values of a can that Klingons can use. List out all of the values of b that Klingons can use. How many pairs (a, b) can be used? (You can use DOT DOT DOT if it is REALLY obvious what you mean.)

GOTO NEXT PAGE FOR NEXT PROBLEM

3. (15 points) Alice did not like doing the last problem! It was tedious! She worries that Dr. Gasarch might ask about mod 100 or mod 1000 and then it will be really boring!

   Write pseudocode for a program that will, on input m, output

   - the list of all a such that a can be used as the coefficient of x in the affine cipher.
   - the list of all b such that b can be used as the constant term in the affine cipher.
   - the number of (a, b) such that $ax + b$ can be used for the affine cipher.

   GOTO NEXT PAGE FOR NEXT PROBLEM

4. (15 points) Alphabet is $\{x, y\}$. $s \in \{0, 1\}$, selected uniformly at random. We use $x + s$ to mean x shifted by s. Same for $y + s$.

   Alice wants to send TWO letters. $m \in \{xx, xy, yx, yy\}$ is the message Alice wants to send. c is the ciphertext Alice sends.

   - If $m = xx$ then Alice sends $c = (x + s)(x + s)$.
   - If $m = xy$ then Alice sends $c = (x + s)(y + s)$.
   - If $m = yx$ then Alice sends $c = (y + s)(x + s)$.
   - If $m = yy$ then Alice sends $c = (y + s)(y + s)$.
   - Let p_{xx}, p_{xy}, p_{yx}, p_{yy} be such that $\Pr(m = xx) = p_{xx}$, etc. Note that $p_{xx} + p_{xy} + p_{yx} + p_{yy} = 1$. WE ASSUME p_{xx} etc are all NONZERO.

   Give expressions for the following in terms of p_{xx}, p_{xy}, p_{yx}, p_{yy}.

   (a) $\Pr[m = xx | c = xx]$

   (b) $\Pr[m = xx | c = xy]$

   (c) $\Pr[m = xx | c = yx]$

   (d) $\Pr[m = xx | c = yy]$

   (e) Use the results above to show that the cipher is insecure.

GOTO NEXT PAGE FOR NEXT PROBLEM

5. (30 points) This is a programming problem. You will write two programs. You will upload the second one for autograding.

   In this problem we will look at the SHIFT cipher when you include not just letters, but also digits. So our alphabet will be $\{a, b, c, \ldots, z, 0, 1, \ldots, 9\}$.

   (a) Your first program should take a text and (1) eliminate all non-alphanumeric symbols and whitespace, (2) replace a and A with 1, ..., replace z and Z with 26, replace 0 with 27, replace 1 with 28, ..., replace 9 with 36.

   (b) Write a program that, will given a text of numbers in $\{1, \ldots, 36\}$ count how many of each symbol above there are. So the output is a 36-long array of natural numbers.

   (c) Write a program that, will, given a 36-long array of natural numbers $(f_1, \ldots, f_{36})$ will compute $F = \sum_{i=1}^{36} f_i$ and then output the 36-long array of reals $(\frac{f_1}{F}, \ldots, \frac{f_{36}}{F})$.

   (d) Run your code on the sample.tex file (not hw01.tex) provided on the course webpage next to this homework, and have it calculate $\vec{f} = (\frac{f_1}{F}, \ldots, \frac{f_{36}}{F})$. Put the vector that you get as your answer to problem 5.

   (e) Your second program should take a shifted text and also calculate $\vec{f}$. You got a vector for unshifted text from part (d), call this f'. For $0 \leq i \leq 35$ let $\vec{f_i}$ be $\vec{f}$ shifted by i mod 36. Print out a table of the 36 values.
   $\vec{f'} \cdot \vec{f_0}$
   $\vec{f'} \cdot \vec{f_1}$
   $\vdots$
   $\vec{f'} \cdot \vec{f_{35}}$

   (f) Calculate the max of $\vec{f'} \cdot \vec{f_i}$ as $1 \leq i \leq 35$ (it should be much larger than $\vec{f'} \cdot \vec{f_0}$. Don't print this.

   (g) Use this to break the shift cipher. You should print the shift used to encrypt the input. Then, you should convert your values back into characters, and print the plaintext.

   (h) Run your program on other math texts that have been shifted and see how well it works.

GO TO NEXT PAGE FOR MORE ABOUT THIS PROBLEM!

(a) This problem will be autograded. There will be a separate assignment for this problem – upload your program here. Specifically, the program that is uploaded should take in a shifted text from standard input and print parts (e) and (g) to standard output. You only need to run parts (a) through (d) once, so you can hardcode the result from part (d) in the program you upload. But don't forget it to include part (d) in your manually-graded homework!

(b) You should upload a single file ending in .java, .py, .ml, .rb, or .c, corresponding to Java, Python3, OCaml, Ruby, and C respectively. Ask on Piazza if you want more options.

Whatever you do, don't attack the server or exfiltrate the tests.

(c) If for whatever reason you decide to handwrite this homework, it is ok to just put down the first few values. I'm not going to make you write down all 36. If you're typing this, you can just copy-paste your results from your program.

(d) Separate each value with spaces, tabs, or newlines. Do not include symbols or whitespace in part (g). The autograder is case-insensitive. Floats should be printed to as many digits as you have: they will be rounded internally. Finally, please don't try to leak the test cases.

The input will be everything that goes into stdin. It will contain newlines, spaces, and irrelevant punctuation. Be sure that your program can handle these.

(e) The test cases are the shifted LaTeX sources for real papers, up to 200 kilobytes each.

GOTO NEXT PAGE FOR NEXT PROBLEM

6. (15 points) How many $x \in \{0, \ldots, 11\}$ satisfy the equation

$$x^2 + 3x + 2 \equiv 0 \pmod{12}$$

No justification needed. (Hence your grade will be either 0 or 15.)

GOTO NEXT PAGE

7. (0 points but this is the most important problem on this entire HW set!!!!!) Given a, b we want to find if $a^{-1} \pmod{b}$ exists.

   (a) Look up *The Euclidean Algorithm* which is for this problem.

   (b) Code up the algorithm (it will be used in many later assignments).

   DO NOT hand anything in, but DO THIS and I will, in the future, assume that you did this and can use it within other assignments.