HW Review

October 15, 2019
1. Why is it easier for Vulcans to use PLAYFAIR than Klingons or Romulans? ANSWER: Vulcans: 36 is a square, so they do not need to fiddle with the letters. CAVEAT: Klingons COULD arrange letters in $5 \times 7$ grid. Romulans are out-of-luck.

2. How can Klingons use Playfair? ANSWER: Klingons need to add one dummy character to their alphabet so it has 36, a square. CAVEAT: See above.

3. How can Romulans use Playfair? ANSWER: Romulans select 2 characters to merge into 1, perhaps the 2 least used. so that their alphabet will have 36. CAVEAT: Could add one and use $2 \times 19$ grid. Bad idea?
Alice wants to compute $7^{81} \pmod{101}$. Do this using repeated squaring. Show all work. How many multiplications does it take?

**ANSWER** all arithmetic is mod 101.

$7^2 \equiv (7^1)^2 \equiv 49$
$7^4 \equiv (7^2)^2 \equiv 49^2 \equiv 78$
$7^8 \equiv (7^4)^2 \equiv 78^2 \equiv 24$
$7^{16} \equiv (7^8)^2 \equiv 24^2 \equiv 71$
$7^{32} \equiv (7^{16})^2 \equiv 71^2 \equiv 92$
$7^{64} \equiv (7^{32})^2 \equiv 92^2 \equiv 81$

Each of the above took one mult for a total of 6 mults so far.

$7^{81} = 7^{64} \times 7^{16} \times 7^1 \equiv 81 \times 71 \times 7 \equiv 59$ (2 mults)

**TOTAL:** 8 mults.
Alice notices that $81 \equiv 3^4$. So instead of using repeated *squaring* she decides to use repeated *cubing*. Each cubing takes two multiplications but there are less iterations. Compute $7^{81}$ using this, show your work. How many multiplications does it take?

**ANSWER** All arith is mod 101.

\[
\begin{align*}
7^3 &\equiv 40 \\
7^9 &\equiv (7^3)^3 \equiv 40^3 \equiv 67 \\
7^{27} &\equiv (7^9)^3 \equiv 67^3 \equiv 86 \\
7^{81} &\equiv (7^{27})^3 = 86^3 \equiv 59
\end{align*}
\]

Each of the above took two mult for a total of 8 mults.

**TOTAL:** 8 mults.
Hw 3, Problem 2c

Give algorithm for repeated cubing method for: given \(a, n, p\), find \(a^n \pmod{p}\). Give upper bound on numb. of mults as function \(n\).

**ANSWER**

All arithmetic is \(\pmod{p}\).

1. Input \((a, n, p)\)
2. \(n = (n_L \cdots n_0)_3\). \((n_i \in \{0, 1, 2\}, L = \lceil \log_3(n) \rceil \).)
3. \(x_0 = a\)
4. For \(i = 1\) to \(L\), \(x_i = x_{i-1}^3\). (Note that \(x_i = a^{3^i}\).)
5. (Now have \(a^{n_03^0}, \ldots, a^{n_L3^L}\) ) Answer is \(a^{n_03^0} \times \cdots \times a^{n_L3^L}\)

\(L\) iters, 2 mults per iter: \(\leq 2L \leq 2 \lceil \log_3(n) \rceil \leq 2 \log_3(n)\) mults.

Mults after iterations, : \(\leq L = \lceil \log_3(n) \rceil \leq \log_3(n)\).

Total: \(\leq 3 \log_3(n)\) mults.
Repeated squaring: the number of multiplications is

$$\leq \lg(n) + (\text{Number of 1's in binary rep of } n) - 1.$$ 

Give three examples of an $n \geq 99$ where the repeated-cubing algorithm takes less mults than the repeated-squaring algorithm.
**ANSWER**

We look at powers of 3. $7^{3^5}$. First do by repeated cubing:

$x_0 = 7$

$x_1 \equiv x_0^3$ (which is $7^3$)

$x_2 \equiv x_1^3$ (which is $7^{3^2}$)

$x_3 \equiv x_2^3$ (which is $7^{3^3}$)

$x_4 \equiv x_3^3$ (which is $7^{3^4}$)

$x_5 \equiv x_4^3$ (which is $7^{3^5}$)

Each line takes 2 mults, so 10 mults total.
Hw 3, Problem 2d- Squaring

We now look at repeated squares.
Need to look at $3^5$ in binary: $3^5 = 243 = 11110011$ in binary.

$$11110011 = 2^7 + 2^6 + 2^5 + 2^4 + 2^1 + 2^0$$

$x_0 = 7$

$x_1 \equiv x_0^2$ (which is $7^2$)

$x_2 \equiv x_1^2$ (which is $7^2^2$)

$x_3 \equiv x_2^2$ (which is $7^2^3$)

$x_4 \equiv x_3^2$ (which is $7^2^4$)

$x_5 \equiv x_4^2$ (which is $7^2^5$)

$x_6 \equiv x_5^2$ (which is $7^2^6$)

$x_7 \equiv x_6^2$ (which is $7^2^7$)

7 multiplications so far. And now we do:

$$7^{3^5} = 7^{2^0} \times 7^{2^1} \times 7^{2^4} \times 7^{2^5} \times 7^{2^6} \times 7^{2^7}$$

This is 5 mults. Total Number of mults: 12, more than 10. We OMIT the other two examples, but they are both powers of 3.
Why isn’t repeated cubing used more often?

**ANSWER**

- Sometimes it takes more steps.
- But even when it takes less, multiplying by powers of 2 is a very easy shift of bits, so the type of mult is easier for powers of 2.
Alice and Bob are going to use the Affine Cipher. They get to choose their alphabet size! If the alphabet size is \( n \) then they will pick a number \( a \in \{1, \ldots, n\} \) at random and then test if \( a \) will work to be the coefficient of \( x \). If not, then try again. If so then they will use \( a \) as the coefficient for \( x \). (We are not concerned with the picking of \( b \).)
Hw 3, Problem 3a

Assume the alphabet size is 1000. What is the probability that the letter they pick will work? Call this $p_{1000}$. (Think about but do not hand in: what is the expected number of times they will need to pick an $a$?)

We need to know how many elements of $\{1, \ldots, 1000\}$ are relatively prime to 1000. This is $\phi(1000) = \phi(2^3 \times 5^3) = \phi(2^3) \times \phi(5^3) = (2^3 - 2^2)(5^3 - 5^2) = 4 \times 100 = 400$. Hence the probability is $p_{1000} = \frac{400}{1000} = 0.4$. 
Assume the alphabet size is 1000. What is the probability that the letter they pick will work? Call this $p_{1000}$. (Think about but do not hand in: what is the expected number of times they will need to pick an $a$?)

**ANSWER** We need to know how many elements of $\{1, \ldots, 1000\}$ are rel prime to 1000. This is

$$\phi(1000) = \phi(2^3 \times 5^3) = \phi(2^3) \times \phi(5^3) =$$

$$= (2^3 - 2^2)(5^3 - 5^2) = 4 \times 100 = 400$$

Hence the probability is

$$p_{1000} = \frac{400}{1000} = 0.4.$$
Think about but do not hand in: what is the expected number of times they will need to pick an $a$?
Hw 3, Problem 3a, the THINK ABOUT part

Think about but do not hand in: what is the expected number of times they will need to pick an a?

What is the expected number of times they will need to pick an a?

\[
\sum_{i=1}^{\infty} \text{(Prob that takes } i \text{ tries)} \times i
\]

The prob it takes } i \text{ tries is

(Prob the first } i - 1 \text{ tries don’t work) } \times \text{(Prob the } i\text{th try does work)}

\[
= (1 - 0.4)^{i-1} \times 0.4 = (0.6)^{i-1}(0.4)
\]

So answer is

\[
\sum_{i=1}^{\infty} (0.6)^{i-1} \times 0.4 \times i = (0.4) \sum_{i=1}^{\infty} (0.6)^{i-1}i
\]

How to evaluate this sum?
Hw3, Problem 3a- The Weird Sum

\[ \sum_{i=1}^{\infty} (0.6)^{i-1}i = \]

Let's generalize this: \[ \sum_{i=1}^{\infty} ix^{i-1} = \]
Does this remind you of anything?
Hw3, Problem 3a- The Weird Sum

\[ \sum_{i=1}^{\infty} (0.6)^{i-1}i = \]

Lets generalize this: \[ \sum_{i=1}^{\infty} ix^{i-1} = \]

Does this remind you of anything? \[ \frac{d}{dx} x^i = ix^{i-1}. \]

\[ \sum_{i=1}^{\infty} x^i = \frac{x}{1-x} \]

Differentiate both sides:

\[ \sum_{i=1}^{\infty} ix^i = \frac{(1-x)+x}{(1-x)^2} = \frac{1}{(1-x)^2} \]

\[ (0.4) \sum_{i=1}^{\infty} i(0.6)x^i = \frac{2}{5} \frac{1}{(2/5)^2} = 2.5 \]

SO, not that many tries to get the proper a.
Assume the alphabet size is 1001. What is the probability that the \( a \) they pick will work?
Assume the alphabet size is 1001. What is the probability that the \( a \) they pick will work?

**ANSWER** We need to know how many elements of \( \{1, \ldots, 1001\} \) are rel prime to 1001. This is

\[ \phi(1001) = \phi(7 \times 11 \times 13) = \phi(7) \times \phi(11) \times \phi(13) = 6 \times 10 \times 12. \]

Hence the probability is

\[ p_{1001} = \frac{6 \times 10 \times 12}{1001} = \frac{720}{1001} \sim 0.72. \]
Which of $p_{1000}$ and $p_{1001}$ is bigger? Based on this give some general advice on what alphabet size to use if a prime size is not available.
Which of $p_{1000}$ and $p_{1001}$ is bigger? Based on this give some general advice on what alphabet size to use if a prime size is not available.

**ANSWER** $p_{1001}$ is bigger. Good to pick an alphabet size that has no square factors.
Alice and Bob are using the cipher on the Sept 9 slides, title *Awesome Vig or Psuedo One-Time Pad* EXCEPT that the mod is 2 digits long instead of 4 digits long.
Eve is sure that the word ERIK will be in the plaintext. Eve looks at every 4-long sequence in the ciphertext and guesses that they decode to ERIK and sets up equations.
Eve sees ABCD.
The following table will help you:

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
<th>G</th>
<th>H</th>
<th>I</th>
<th>J</th>
<th>K</th>
<th>L</th>
<th>M</th>
</tr>
</thead>
<tbody>
<tr>
<td>01</td>
<td>02</td>
<td>03</td>
<td>04</td>
<td>05</td>
<td>06</td>
<td>07</td>
<td>08</td>
<td>09</td>
<td>10</td>
<td>11</td>
<td>12</td>
<td>13</td>
</tr>
<tr>
<td>N</td>
<td>O</td>
<td>P</td>
<td>Q</td>
<td>R</td>
<td>S</td>
<td>T</td>
<td>U</td>
<td>V</td>
<td>W</td>
<td>X</td>
<td>Y</td>
<td>Z</td>
</tr>
<tr>
<td>14</td>
<td>15</td>
<td>16</td>
<td>17</td>
<td>18</td>
<td>19</td>
<td>20</td>
<td>21</td>
<td>22</td>
<td>23</td>
<td>24</td>
<td>25</td>
<td>26</td>
</tr>
</tbody>
</table>
Write down (but do not solve) the equations she will try to solve to find how the key is generated. Show all work.
Write down (but do not solve) the equations she will try to solve to find how the key is generated. Show all work.

ERIK is (05,18,09,11). ABCD is (01,02,03,04) Here is how Eve finds her guess for the key:
The first two digits:
\[0 + x \equiv 0 \pmod{10}\]
\[5 + y \equiv 1 \pmod{10}\]
Hence \(x = 0\) and \(y = 6\).

Keep doing this to find that the guess for the key for this part is (06, 94, 04, 93)

Just for my own sanity I’ll rewrite this to check it.

<table>
<thead>
<tr>
<th>ERIK</th>
<th>05</th>
<th>18</th>
<th>09</th>
<th>11</th>
</tr>
</thead>
<tbody>
<tr>
<td>KEY</td>
<td>06</td>
<td>94</td>
<td>04</td>
<td>93</td>
</tr>
<tr>
<td>ABCD</td>
<td>01</td>
<td>02</td>
<td>03</td>
<td>04</td>
</tr>
</tbody>
</table>
The conjecture is that the key for this part is (06, 94, 04, 93). We need to test that.

RECALL that the key is formed by a recurrence of the form

\[ x_i = A x_{i+1} + B \pmod{M} \]

So if the sequence (06, 94, 04, 93) is part of the key then we must have:

\[ 94 \equiv 6A + B \pmod{M} \]
\[ 4 \equiv 94A + B \pmod{M} \]
\[ 93 \equiv 4A + B \pmod{M} \]
What are the bounds on $M$?
What are the bounds on $M$?

**ANSWER** We know that $M$ is 2-digits long so $M \leq 99$. We know that one of the numbers in the key is 94, so $95 \leq M$. Hence

$$95 \leq M \leq 99$$
Solve the equations or show they can’t be solved:

\[ 94 \equiv 6A + B \pmod{M} \]
\[ 4 \equiv 94A + B \pmod{M} \]
\[ 93 \equiv 4A + B \pmod{M} \]

AND \( 95 \leq M \leq 99 \). All \( \equiv \) are mod \( M \).

Subtract the second from the first equation to get \( \text{EQ1} \) \( 90 \equiv -88A \)
Subtract the third from the first equation to get \( \text{EQ2} \) \( 1 \equiv 2A \)
Multiply \( \text{EQ2} \) by 44 to get \( \text{EQ3} \) \( 44 \equiv 88A \)
Add \( \text{EQ1} \) and \( \text{EQ3} \) to get \( 134 \equiv 0 \). Hence \( M \) divides 134.

So \( M \) has to be one of 1, 2, 67, 134.
NONE of these are between 95 to 99 so NO value of \( M \) works.
Alice and Bob are going to use Diffie-Hellman. Bob wants to save some time so instead of picking a RANDOM $b \in \{ \frac{p}{3}, \frac{2p}{3} \}$ he picks a $b$ that is a power of 2 because he thinks that for such $b$, $g^b$ will be easier to compute. (Alice still picks $a \in \{ \frac{p}{3}, \frac{2p}{3} \}$ at random.)
Bob is right! Computing $g^b$ IS easier if $b$ is a power of 2. Explain why.

**ANSWER**
Recall that repeated squaring for $g^b$ takes

$$\left\lfloor \log_2(b) \right\rfloor + \text{(number of 1's in } b \text{ in binary)}-1$$
mults. But if $b$ is a power of 2 then there is only one 1 in $b$ in binary. So only $\left\lfloor \log_2(b) \right\rfloor$ mults.
Eve can now find the shared secret in time $O(\log p)^c$. Show how.

What is $c$?

**ANSWER**

Eve knows that $b \in X = \{2^0, 2^1, \ldots, 2^{\lfloor \log_2 p \rfloor} \}$. $X$ IS A VERY SMALL SET!

Eve sees $p, g, g^a, g^b$.

Eve computes $g^x$ for all $x \in X$. Each of these takes $O(\log p)$ mults, and there are $O(\log p)$ elements of $X$, so that's $O(\log p)^2$ mults. For one of those $x$ you will see that $g^x = g^b$, so Eve finds out what $b$ is. Once Eve knows $b$, she computes $(g^a)^b = g^{ab}$, so she has the secret. This last step took another $O(\log p)$ mults, so still

$$O(\log p)^2 \text{ mults, so } c = 2.$$ 

One can be a bit cleverer and get $c = 1$. 
We need to compute $g^x$ for every $x \in X$. But note that

$$X = \{2^0, 2^1, \ldots, 2^{|\log_2 p|}\}.$$ 

Hence we need

$g^{2^0}$

$g^{2^1}$

$g^{2^2}$

$\vdots$

$g^{2^{|\log_2 p|}}$

By repeated squaring an do this in $O(\log_2 p)$ steps.
Compute the following and show your work. (You may use a calculator for simple operations such as multiplication.)

1. (5 points) $7^{999,999,999,999,999} \pmod{100}$
2. (5 points) $7^{999,999,999,999,999} \pmod{101}$
3. (5 points) $7^{999,999,999,999,999} \pmod{102}$
We need \( \phi(100) \). This is

\[
\phi(100) = \phi(2^2 \times 5^2) = \phi(2^2) \phi(5^2) = (2^2 - 2)(5^2 - 5) = 2 \times 20 = 40
\]

Therefore, \( 7^{999,999,999,999,999} \equiv 7 \pmod{40} \pmod{100} \)
\textbf{hw05, 6a}

\[ 7^{999,999,999,999,999} \pmod{100} \]

\textbf{ANSWER:} We need \( \phi(100) \). This is

\[ \phi(100) = \phi(2^2 \times 5^2) = \phi(2^2) \phi(5^2) = (2^2 - 2)(5^2 - 5) = 2 \times 20 = 40 \]

\[ 7^{999,999,999,999,999} \equiv 7^{999,999,999,999,999} \mod{40} \pmod{100} \equiv 7^{39} \pmod{100} \]

We need 39 as a sum of powers of 2. By taking the highest power-of-2 that is \( \leq \) current value, we get:

\[ 39 = 2^5 + 2^2 + 2^1 + 2^0 \]

All \( \equiv \) are mod 100.

\[ 7^0 \equiv 1 \]
\[ 7^{2^0} \equiv 7 \]
\[ 7^{2^1} \equiv (7^{2^0})^2 \equiv 7^2 \equiv 49 \]
\[ 7^{2^2} \equiv (7^{2^1})^2 \equiv 49^2 \equiv 1 \]
\[ 7^{2^3} \equiv (7^{2^2})^2 \equiv 1^2 \equiv 1 \]
We need $\phi(101)$. This one is easy $\phi(101) = 100$.

$7^{999,999,999,999,999} \ (mod\ 101)$
$7^{999,999,999,999,999} \pmod{101}$

**ANSWER**

We need $\phi(101)$. This one is easy $\phi(101) = 100$. 
$7^{999,999,999,999,999} \pmod{102}$

Use $\phi(102) = \phi(2 \times 3 \times 17) = 1 \times 2 \times 16 = 32$. 