The University of Sydney
School of Mathematics and Statistics

Solutions to Assignment 1

MATH2068/2988: Number Theory and Cryptography — Semester 2, 2017

Web Page: http://www.maths.usyd.edu.au/u/UG/IM/MATH2068/
Lecturer: Dzmitry Badziahin

Non-computer part:

1. **This question is for students enrolled in the mainstream unit MATH2068 only. Do not answer this if you are in the advanced unit MATH2988.**

(a) Using the extended Euclidean algorithm, or by some other method, find an inverse of 21 modulo 37.

Solution: The Euclidean algorithm, when applied to the numbers 37 and 17, produces the following results:

$$\begin{aligned} 37 &= 1 \times 21 + 16, \\ 21 &= 1 \times 16 + 5, \\ 16 &= 3 \times 5 + 1. \end{aligned}$$

This confirms that $\gcd(37, 21) = 1$, and it also gives us the information necessary to write 1 as an integer linear combination of 37 and 21:

$$\begin{aligned} 16 &= 37 - 1 \times 21, \\ 5 &= 21 - 1(37 - 1 \times 21) = (-1) \times 37 + 2 \times 21, \\ 1 &= 16 - 3 \times 5 = (37 - 1 \times 21) - ((-1) \times 37 + 2 \times 21) \\ &= 4 \times 37 + (-7) \times 21. \end{aligned}$$

This whole calculation can alternatively be expressed in tabular form using the extended Euclidean algorithm as described in the textbook:

$$\begin{array}{cccccc} 37 & 21 & 16 & 5 & 1 & 0 \\ & & 1 & 1 & 3 & 5 \\ 0 & 1 & 1^{-} & 2 & 7^{-} & \\ 1 & 0 & 1 & 1^{-} & 4 & \end{array}\quad .$$

The equation $4 \times 37 + (-7) \times 21 = 1$ tells us that -7 is an inverse of 21 modulo 37, and so is 30 or any other number congruent to -7 modulo 37.

(b) Solve the following system of simultaneous congruences.

$$\left\{ \begin{array}{rcll} 21x & \equiv & 6 & \pmod{37} \\ x & \equiv & 6 & \pmod{11} \\ x & \equiv & 8 & \pmod{13} \end{array} \right.$$

Solution: We start by finding x which solves the last two congruences. By trying small numbers or by the extended Euclidean algorithm we find that

$$1 = 6 \times 11 - 5 \times 13.$$

therefore the application of the CRT gives us

$$x \equiv 8 \times 6 \times 11 - 6 \times 5 \times 13 \equiv 138 \pmod{11 \times 13}.$$

Alternatively one can notice that $x \equiv -5 \pmod{11}$ and $x \equiv -5 \pmod{13}$ which implies that $x \equiv -5 \pmod{11 \times 13}$.

Now we add the first equation to the system.

Since 30 is an inverse of 21 mod 37, and $30 \times 6 \equiv -5 \pmod{37}$, the first congruence is equivalent to $x \equiv -5 \pmod{37}$. therefore we end up with the system of congruences

$$\begin{cases} x & \equiv & -5 \pmod{37} \\ x & \equiv & -5 \pmod{143} \end{cases}$$

Its solution is $x \equiv -5 \pmod{5291} \equiv 5286$.

2. **This question is for all students in both MATH2068 and MATH2988.**

(a) Find $2015^{24195} \pmod{2017}$. You may use that 2017 is prime.

Solution: Since 2017 is prime, Fermat Little Theorem tells that $a^{2016} \equiv 1 \pmod{2017}$ for any a coprime with 2017. Notice that $2015 \equiv -2 \pmod{2017}$ and $24195 = 12 \times 2016 + 3$. Therefore we finally get that

$$2015^{24195} \equiv (-2)^{2016 \cdot 12+3} \equiv ((-2)^{12})^{2016} \cdot (-2)^3 \equiv -8 \pmod{2017}.$$

(b) Let m and n be two coprime positive integer numbers. Let a be integer such that $\gcd(a, mn) = 1$. Show that

$$a^{\operatorname{lcm}(\phi(m),\phi(n))} \equiv 1 \pmod{mn},$$

where $\operatorname{lcm}(\phi(m), \phi(m))$ is the least common multiple of $\phi(m)$ and $\phi(n)$. Deduce that for any a which is coprime with 10,

$$a^{20} \equiv 1 \pmod{100}.$$

Solution: Since $\gcd(a, mn) = 1$ then $\gcd(a, m) = \gcd(a, n) = 1$. Therefore we can apply Euler's theorem to get that

$$a^{\phi(n)} \equiv 1 \pmod{n} \quad \text{and} \quad a^{\phi(m)} \equiv 1 \pmod{m}.$$

For every multiple $M = d\phi(n)$ of $\phi(n)$ we have

$$a^M \equiv (a^{\phi(n)})^d \equiv 1 \pmod{n}.$$

The same is true for any multiple M of $\phi(m)$: $a^M \equiv 1 \pmod{m}$. Since $M = \operatorname{lcm}(\phi(n), \phi(m))$ is a multiple of both $\phi(n)$ and $\phi(m)$ we have

$$a^M \equiv 1 \pmod{n} \quad \text{and} \quad a^M \equiv 1 \pmod{m}.$$

Therefore $a^M \equiv 1 \pmod{mn}$.

Finally, $100 = 4 \times 25$ and $\operatorname{lcm}(\phi(4), \phi(25)) = \operatorname{lcm}(2, 20) = 20$. Therefore $a^{20} \equiv 1 \pmod{100}$.

(c) With help of the previous section or otherwise find the last two digits of $7^{(7^4)}$ and of $7^{(7^{400})}$.

Solution: We know that $7^{20} \equiv 1 \pmod{100}$. Therefore in order to find the last two digits of the numbers 7^{7^n} we firstly need to find the remainder after division of 7^n by 20, that is to find $7^n \pmod{20}$. $20 = 5 \cdot 4$, so we can use the result from the previous part again to get that $\operatorname{lcm}(\phi(5), \phi(4)) = \operatorname{lcm}(4, 2) = 4$ and

$$7^4 \equiv 1 \pmod{20}.$$

(The last congruence can also be confirmed by direct computation). We end up with

$$7^{7^4} \equiv 7^1 \equiv 7 \pmod{100}$$

The same congruence applies to $7^{7^{400}}$. Therefore the last two digits of these two numbers are 07.

3. This question is for students enrolled in the advanced unit MATH2988 only. Do not answer this if you are in the mainstream unit MATH2068.

(a) You are given that $p = 737279$ is prime and that

$$2^{2p+1} \equiv 2 \pmod{2p+1}.$$

Deduce that $2p + 1$ is also prime.

Solution: Assume that $2p + 1$ is composite. Consider a prime $q|2p + 1$, $q < 2p + 1$. Then we have $2^{2p+1} \equiv 2 \pmod{q}$. Since q is obviously not 2 then $2^{2p} \equiv 1 \pmod{q}$. Then we have $\operatorname{ord}_q(2)|2p$. Since p is prime we have only two possibilities:

- $\operatorname{ord}_q(2) = 2$. Then $2^2 \equiv 1 \pmod{q}$ and $q = 3$. However one can easily check that $3 \nmid 2 \cdot 737279 + 1$.
- $\operatorname{ord}_q(2) = p$ or $\operatorname{ord}_q(2) = 2p$. Then $p|q - 1$. But this is impossible because $q \leq (2p + 1)/2$ or $q \leq p$.

In both cases we get contradiction.

(b) Let $n = 2^{131} - 1$. Show that

$$2^{n-1} \equiv 1 \pmod{n}.$$

Solution: Consider $n - 1 = 2^{131} - 2 = 2 \cdot (2^{130} - 1)$. One can check that 131 is prime (For that one can try to divide 131 by primes between 1 and $[\sqrt{131}] = 11$). Therefore by Fermat little theorem we have $(n - 1) = 131 \cdot k$ where $k \in \mathbb{N}$. Then

$$2^{p-1} \equiv 2^{131 \cdot k} \equiv 1 \pmod{p}.$$

(c) Show that 263 divides $n = 2^{131} - 1$. In particular, it means that n is not prime.

Solution: We compute 2^{131} modulo 263 by the method of successive squaring.

$$131 = 2^7 + 2^1 + 2^0.$$

Next, create the table of $2^{2^n} \pmod{131}$:

n	0	1	2	3	4	5	6	7
$2^{2^n} \pmod{131}$	2	4	16	256	49	34	104	33

Finally, $2^{131} \equiv 33 \cdot 4 \cdot 2 \equiv 1 \pmod{263}$. Therefore $263 \mid 2^{131} - 1$.

Computer part:

4. In this question you will do a simple "experimental test" of the Euler–Fermat Theorem using your student ID as the modulus. In MAGMA give the name `sid` to your student ID (a nine-digit number). Use MAGMA commands from Computer Tutorial 1 to select random nine-digit numbers until you find one that is coprime to `sid`, and call this number `num`.

Now ask MAGMA to compute the order of `num` modulo `sid` and the Euler phi-function of `sid`, using the commands

```
Modorder(num,sid);    and    EulerPhi(sid);
```

Finally, use MAGMA to verify that the first of these numbers divides the second.

Solution: Obviously, your answers will depend on your SID and the random numbers generated by MAGMA. Suitable commands to define `num` are

```
num:=Random(10^9);    num;    GCD(num,sid);
```

repated as necessary until `num` has nine digits and the gcd of `num` and `sid` is 1. One way to verify that the order of `num` modulo `sid` divides the Euler phi-function of `sid` is simply to type

```
EulerPhi(sid)/Modorder(num,sid);
```

and observe that the result is an integer (not a rational number).

5. Type the command `load "asst1ciphertexts.txt";` The file you have loaded defines three ciphertexts called `sct1`, `sct2` and `sct3` (all of type `String`). The original plaintexts were all in English, and all concerned the military applications of mathematics. One of the three was enciphered using a Vigenère cipher, and the other two were enciphered using simple substitution ciphers.

You need to determine which of the three is the Vigenère ciphertext, and decipher it (you can ignore the others). To find the period and decryption key of the Vigenère cipher, you can use either the javascript Vigenère key finder on the MATH2068 web page or the MAGMA methods of Computer Tutorial 3. Beware that the plaintexts were relatively short, so the most frequent letter in a decimation is not guaranteed to be `E`; you will need to check other letters, or use the correlation data provided by the javascript Vigenère key finder.

Once you have printed the plaintext in capitals, you have finished the question.

Solution: The commands `CoincidenceIndex(sct1);` etc. show that the coincidence index of `sct1` is 0.0417..., noticeably lower than those of `sct2` and `sct3` which are in the range typical of English text. This strongly suggests that `sct1` is the Vigenère ciphertext, since simple substitution ciphers don't change the coincidence index. Using the javascript Vigenère key finder, and paying attention to the correlation data to get the best fits, suggests that the period of the Vigenère cipher is 8 and the decryption key is `IYTEWNYQ`. Then the commands

```
V:=VigenereCryptosystem(8);
dk:=V!"IYTEWNYQ";
Enciphering(dk, Encoding(V,sct1));
```

reveal the plaintext, which is the first verse of the Major-General's song from Gilbert and Sullivan's *The Pirates of Penzance*.