





Integer Factorization

Chinese Remainder Theorem


 **Reading:** Remainers for Two Modules
10 min


 **Reading:** Chinese Remainder Theorem
10 min


 **Quiz:** Remainers
4 questions


 **Quiz:** Chinese Remainder Theorem: Code
1 question


Modular Exponentiation


 **Reading:** Modular Exponentiation
10 min

 **Reading:** Fast Modular Exponentiation
7 min

 **Quiz:** Fast Modular Exponentiation: Code
2 questions

 **Reading:** Fermat's Little Theorem
10 min

 **Reading:** Euler's Theorem
10 min

 **Quiz:** Modular Exponentiation
4 questions

Remainders for Two Modules

When dividing numbers $0, 1, 2, 3, \dots$ by (say) 5, we get remainders

$0, 1, 2, 3, 4, 0, 1, 2, 3, 4, 0, 1, 2, \dots$

for $0 \leq m < 5$ the remainder is equal to m itself ($m = 0 \cdot 5 + m$), and then remainders repeat along the cycle: 5 has remainder 0, then 6 has remainder 1, and so on. In general, m and $m + 5$ have the same remainder ($m \equiv m + 5 \pmod{5}$), their difference is divisible by 5).

It is like driving on a circular road of length 5 miles: you start near the mark 0, then you see mark 1, then 2, 3, 4, but then instead of 5 you see again 0, then 1, 2, etc. (A similar idea appears in "Groundhog day" movie.)

What if we consider the remainders for two different modules at the same time?

```
1 for i in range(15):
2     print(f'{i}: {i} mod 2={i % 2}, {i} mod 3={i % 3}')
```

Run
Reset

```
0: 0 mod 2=0, 0 mod 3=0
1: 1 mod 2=1, 1 mod 3=1
2: 2 mod 2=0, 2 mod 3=2
3: 3 mod 2=1, 3 mod 3=0
4: 4 mod 2=0, 4 mod 3=1
5: 5 mod 2=1, 5 mod 3=2
6: 6 mod 2=0, 6 mod 3=0
7: 7 mod 2=1, 7 mod 3=1
8: 8 mod 2=0, 8 mod 3=2
9: 9 mod 2=1, 9 mod 3=0
10: 10 mod 2=0, 10 mod 3=1
11: 11 mod 2=1, 11 mod 3=2
12: 12 mod 2=0, 12 mod 3=0
13: 13 mod 2=1, 13 mod 3=1
14: 14 mod 2=0, 14 mod 3=2
```

```
1 0: 0 mod 2=0, 0 mod 3=0
2 1: 1 mod 2=1, 1 mod 3=1
3 2: 2 mod 2=0, 2 mod 3=2
4 3: 3 mod 2=1, 3 mod 3=0
5 4: 4 mod 2=0, 4 mod 3=1
6 5: 5 mod 2=1, 5 mod 3=2
7 6: 6 mod 2=0, 6 mod 3=0
8 7: 7 mod 2=1, 7 mod 3=1
9 8: 8 mod 2=0, 8 mod 3=2
10 9: 9 mod 2=1, 9 mod 3=0
11 10: 10 mod 2=0, 10 mod 3=1
12 11: 11 mod 2=1, 11 mod 3=2
13 12: 12 mod 2=0, 12 mod 3=0
14 13: 13 mod 2=1, 13 mod 3=1
15 14: 14 mod 2=0, 14 mod 3=2
```

The first columns contains i , and two other columns are $i \bmod 2$ and $i \bmod 3$. What can we observe looking at this table?

- In the second column the remainders modulo 2 (i.e., 0 and 1) alternate (even values of i alternate with odd values of i).
- In the third column the remainders modulo 3 (i.e., 0, 1, 2) appear in a 3-loop (0, 1, 2, 0, 1, 2, ...)
- When we come to six, both cycles return to their original position (since 6 is divisible both by 2 and 3, and then the pairs start to repeat, thus forming a loop of length 6).
- All six possible combinations of remainders (two possible remainders modulo 2 combined with three possible remainders modulo 3) appear in this sequence (once per a 6-loop).

Stop and think! Consider some other pair of numbers instead of 2 and 3, for example, 2, 4, or 3, 4, or 6, 4. Will the behavior of remainders be similar?

```
1 for i in range(15):
2     print(f'({i}:2d): ({i % 2}, {i % 4}) '
3         f'({i % 3}, {i % 4}) '
4         f'({i % 6}, {i % 4})')
```

Run
Reset

```
0: (0, 0) (0, 0) (0, 0)
1: (1, 1) (1, 1) (1, 1)
2: (0, 2) (2, 2) (2, 2)
3: (1, 3) (0, 3) (3, 3)
4: (0, 0) (1, 0) (4, 0)
5: (1, 1) (2, 1) (5, 1)
6: (0, 2) (0, 2) (0, 2)
7: (1, 3) (1, 3) (1, 3)
8: (0, 0) (2, 0) (2, 0)
9: (1, 1) (0, 1) (3, 1)
10: (0, 2) (1, 2) (4, 2)
11: (1, 3) (2, 3) (5, 3)
12: (0, 0) (0, 0) (0, 0)
13: (1, 1) (1, 1) (1, 1)
14: (0, 2) (2, 2) (2, 2)
```

```
1 0: (0, 0) (0, 0) (0, 0)
2 1: (1, 1) (1, 1) (1, 1)
3 2: (0, 2) (2, 2) (2, 2)
4 3: (1, 3) (0, 3) (3, 3)
5 4: (0, 0) (1, 0) (4, 0)
6 5: (1, 1) (2, 1) (5, 1)
7 6: (0, 2) (0, 2) (0, 2)
8 7: (1, 3) (1, 3) (1, 3)
9 8: (0, 0) (2, 0) (2, 0)
10 9: (1, 1) (0, 1) (3, 1)
11 10: (0, 2) (1, 2) (4, 2)
12 11: (1, 3) (2, 3) (5, 3)
13 12: (0, 0) (0, 0) (0, 0)
14 13: (1, 1) (1, 1) (1, 1)
15 14: (0, 2) (2, 2) (2, 2)
```

Here the fancy **format** mechanism is used just to get a more nice-looking table.

Let us look at the column for 2 and 4: we see a cycle of length 4 in this column formed by pairs $(0, 0)$, $(1, 1)$, $(0, 2)$, $(1, 3)$ - and, indeed, 4 is divisible both by 2 and 4, so we get again the pair $(0, 0)$. Note that *not all combinations of remainders appear in the cycle*: for example, pair $(1, 2)$ does not appear.

Stop and think! Why pair $(1, 2)$ does not appear? In other words, why there is no integer m such that $m \equiv 1 \pmod{2}$ and $m \equiv 2 \pmod{4}$?

It is easy to see: if $m \equiv 2 \pmod{4}$, then $m = 4q + 2$ for some q , so m is even, and $m \equiv 0 \pmod{2}$ (the remainder is 0).

For the pair 3, 4 things are different: we return to 0, 0 remainders after 12 steps, and all the possible combinations of remainders appear in the cycle.

Finally, for pair 4, 6 the loop has also length 12, but now not all the possible combinations appear (e.g., you do not see the pair 0, 1).

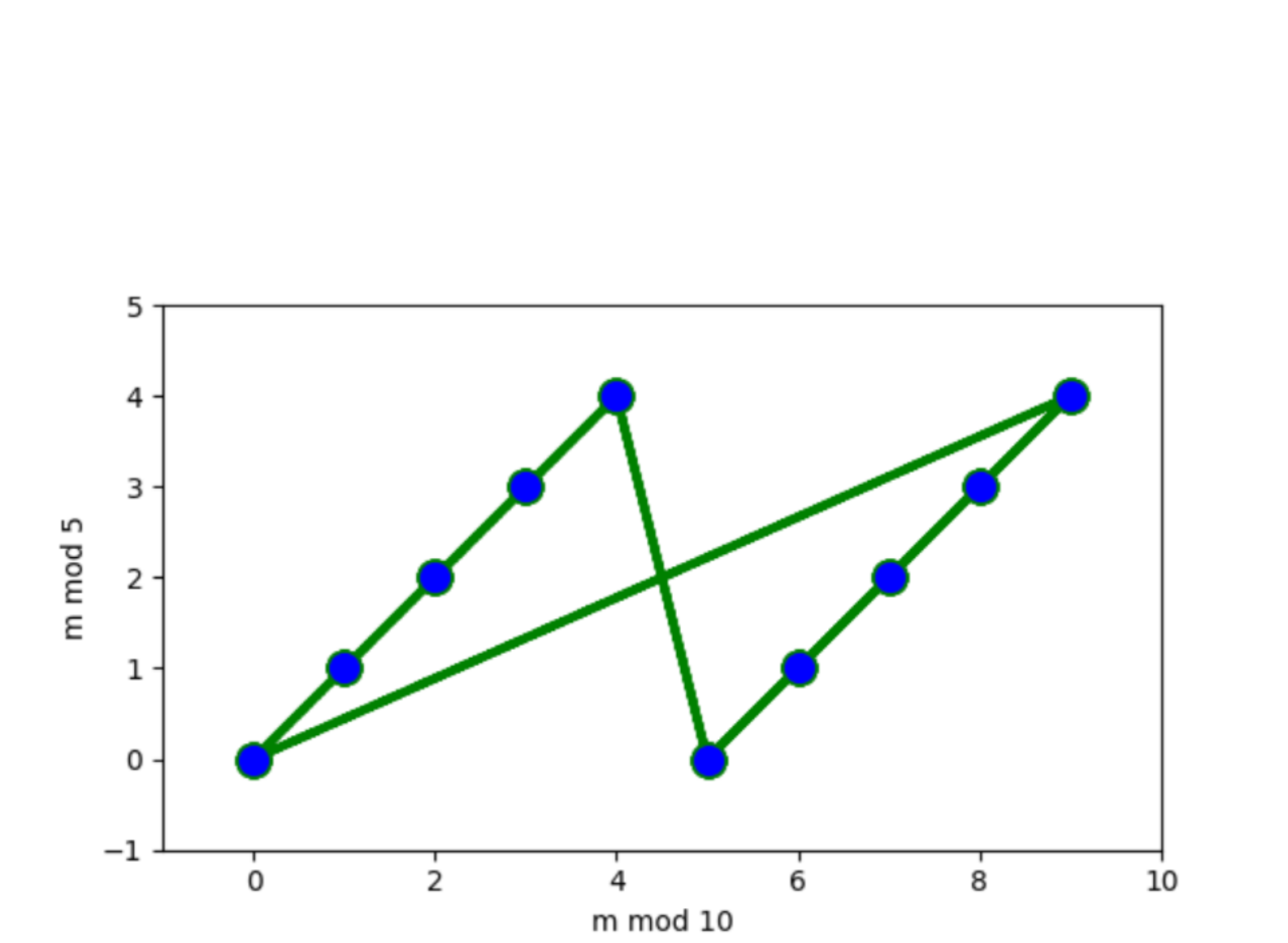
Stop and think!

Can you explain why this pair does not appear: why there are no m such that $m \equiv 0 \pmod{4}$ and $m \equiv 1 \pmod{6}$?

Again, the reason is simple: numbers of the form $4u + 0$ are even, while numbers of the form $6v + 1$ are odd.

To understand the behavior of remainders, it is useful to look at bigger examples, and draw the pairs of remainders on a coordinate plane. This can be done with the following code (do not worry if you do not understand the options, they are needed to make nice plots).

```
1 import matplotlib.pyplot as plt
2
3 a, b = 10, 5
4 n = a * b
5
6 plt.plot([i % a for i in range(n)], [i % b for i in range(n)],
7         color='green', linestyle='dashed', linewidth=3,
8         marker='o', markerfacecolor='blue', markersize=12)
9
10 plt.axis('square')
11 plt.xlim(-1, 0)
12 plt.ylim(-1, 0)
13 plt.xlabel(f'm mod {a}')
14 plt.ylabel(f'm mod {b}')
15
16 plt.savefig('crt-10-5.png')
```



The green line connects the consecutive pairs of remainders (for m and $m + 1$). In most cases, it goes from (x, y) to $(x + 1, y + 1)$, but when one of the remainders reaches the maximal possible value, it jumps to 0 instead.

In the first picture, the remainder modulo 10 determines the remainder modulo 5. Indeed, if $m = 10k + r$, then $10k$ is divisible by 5, so $m \bmod 5 = r \bmod 5$.

Stop and think! What is $m \bmod 5$ if m is a positive integer with last digit 8?

Last digit is remainder modulo 10, so the answer is $8 \bmod 5 = 3$.

A similar picture for $m \bmod 13$ and $m \bmod 7$ looks different: