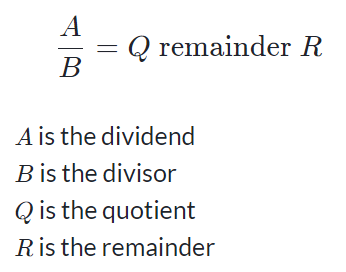
**Modular arithmetic**

## **An Introduction to Modular Math**

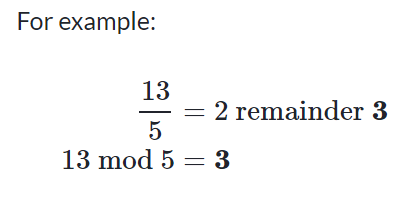
When we divide two integers we will have an equation that looks like the following:

****

Sometimes, we are only interested in what the **remainder** is when we divide A by B.  
For these cases there is an operator called the modulo operator (abbreviated as mod).

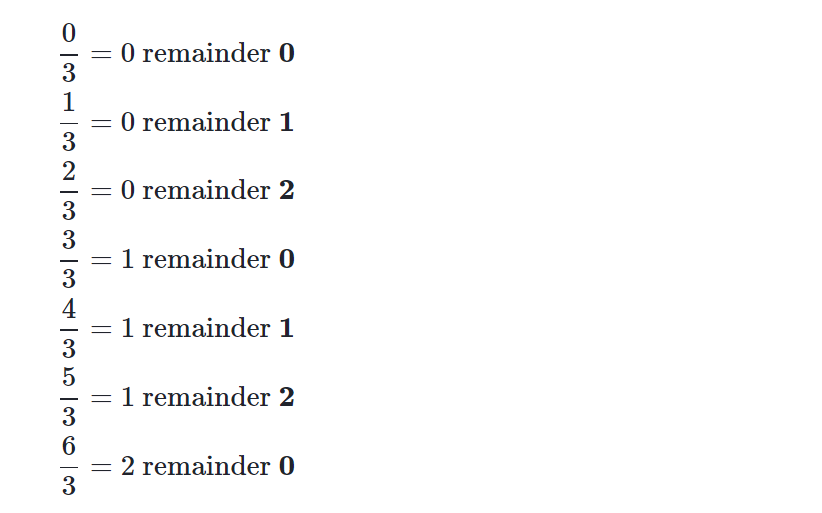
Using the same A, B, Q, and R as above, we would have: A mod B = R.

We would say this as A *modulo* B *is equal to* R. Where B is referred to as the **modulus**.

****

## **Visualize modulus with clocks**

Observe what happens when we increment numbers by one and then divide them by 3.

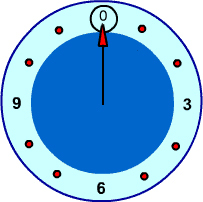
****

The remainders start at 0 and increases by 1 each time, until the number reaches one less than the number we are dividing by. After that, the sequence **repeats**.

By noticing this, we can visualize the modulo operator by using circles.

We write 0 at the top of a circle and continuing clockwise writing integers 1, 2, ... up to one less than the modulus.

For example, a clock with the 12 replaced by a 0 would be the circle for a modulus of 12.



To find the result of A mod Bwe can follow these steps:

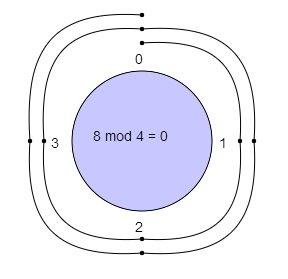
1. Construct this clock for size B
2. Start at 0 and move around the clock A steps
3. Wherever we land is our solution.

(If the number is positive we step clockwise, if it's **negative** we step **counter-clockwise**.)

## **Examples**

### **8 mod 4=?**

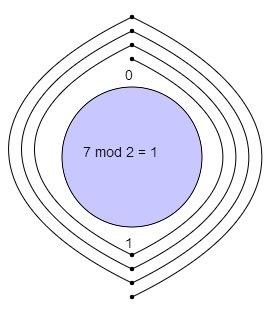
With a modulus of 4 we make a clock with numbers 0, 1, 2, 3.  
We start at 0 and go through 8 numbers in a clockwise sequence 1, 2, 3, 0, 1, 2, 3, 0.



We ended up at **0** so 8 mod 4 = **0**

### **7 mod 2=?**

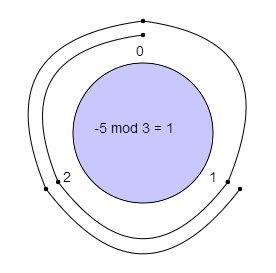
With a modulus of 2 we make a clock with numbers 0, 1.  
We start at 0 and go through 7 numbers in a clockwise sequence 1, 0, 1, 0, 1, 0, 1.



We ended up at **1** so 7 mod 2=**1**

### **−5 mod 3=?**

With a modulus of 3 we make a clock with numbers 0, 1, 2.  
We start at 0 and go through 5 numbers in **counter-clockwise** sequence (5 is **negative**) 2, 1, 0, 2, 1.



We ended up at **1** so −5 mod 3=**1**

## **Conclusion**

If we have A mod B, and we increase A by a **multiple of B**, we will end up in the same spot, i.e.

A mod B =(A+K⋅B) mod B for **any integer K**.

**For example:**

3 mod 10=3

13 mod 10=3

23 mod 10=3

33 mod 10=3

## **Notes to the Reader**

### **mod in programming languages and calculators**

Many programming languages, and calculators, have a mod operator, typically represented with the % symbol. If you calculate the result of a negative number, some languages will give you a negative result.  
e.g.

-5 % 3 = -2.

### **Congruence Modulo**

You may see an expression like:

**A≡B (mod C)**

This says that A is **congruent** to B modulo C. It is similar to the expressions we used here, but not quite the same.

**RELEVANT READING MATERIAL AND REFERENCES:**

**Source Notes:**

1. https://www.khanacademy.org/computing/computer-science/cryptography/modarithmetic/a/what-is-modular-arithmetic

**Lecture Video:**

1. https://youtu.be/vX3ROTkk\_Jg

**Online Notes:**

1. <http://vssut.ac.in/lecture_notes/lecture1428551222.pdf>

**Text Book Reading:**

1. Cormen, Leiserson, Rivest, Stein, “*Introduction to Algorithms*”, Prentice Hall of India, 3rd edition 2012. problem, Graph coloring.

**In addition: PPT can be also be given.**