Elementary Number Theory

Theorem (Polynomial Remainder Theorem):

If a polynomial $f(x)$ is divided by $(x-a)$, the remainder of this division is $f(a)$.

Proof: When we divide a polynomial $f(x)$ by $(x-a)$, we can express $f(x)$ in the form:

$$f(x) = (x-a)q(x) + r$$

where $q(x)$ is the quotient and r is the remainder. Here, r must be a constant (degree 0) because the degree of $(x-a)$ is 1.

Next, we evaluate $f(x)$ at $x = a$:

$$f(a) = (a-a)q(a) + r$$

Simplifying, we get:

$$f(a) = 0 \cdot q(a) + r$$
$$f(a) = r$$

Thus, the remainder r when $f(x)$ is divided by $(x-a)$ is $f(a)$. Note that we also know r is degree 0 because $f(a)$ outputs a real number. This makes sense as the degree of the remainder is always less than the degree of the divisor. Additionally note that if we divide by x, the remainder is given by $f(0)$. Finally, note the case where we take $a = -b$.

Theorem (Factor Theorem):

A polynomial $f(x)$ has a factor $(x-a)$ if and only if $f(a) = 0$.

Proof:

1. If $(x-a)$ is a factor of $f(x)$:

Suppose $(x-a)$ is a factor of $f(x)$. This means there exists a polynomial $q(x)$ such that:

$$f(x) = (x-a)q(x)$$

Evaluate $f(x)$ at $x = a$:

$$f(a) = (a-a)q(a) = 0 \cdot q(a) = 0$$

Therefore, if $(x-a)$ is a factor of $f(x)$, then $f(a) = 0$.

2. If $f(a) = 0$:

Suppose $f(a) = 0$. According to the Polynomial Remainder Theorem, when $f(x)$ is divided by $(x-a)$, the remainder is $f(a)$. So:

$$f(x) = (x-a)q(x) + f(a)$$

Given $f(a) = 0$, we have:

$$f(x) = (x-a)q(x) + 0$$
$$f(x) = (x-a)q(x)$$

Therefore, if $f(a) = 0$, then $(x-a)$ is a factor of $f(x)$.

By proving both directions, we have shown that $(x-a)$ is a factor of $f(x)$ if and only if $f(a) = 0$. This completes the proof of the Factor Theorem.

Proposition:

If $a \equiv b \pmod{m}$ and $c \equiv d \pmod{m}$, then $(a+c) \equiv (b+d) \pmod{m}$.

Proof:

$$\begin{aligned}
\text{Given:} \quad & a \equiv b \pmod{m} \quad \text{and} \quad c \equiv d \pmod{m}, \\
\text{we have:} \quad & a - b = k_1 m \quad \text{and} \quad c - d = k_2 m \quad \text{for some integers } k_1, k_2. \\
\text{Adding these equations, we get:} \quad & (a-b) + (c-d) = k_1 m + k_2 m, \\
& a + c - b - d = (k_1 + k_2) m, \\
\text{which implies:} \quad & a + c \equiv b + d \pmod{m}.
\end{aligned}$$

Proposition:

If $a \equiv b \pmod{m}$, then for any integer k, $ka \equiv kb \pmod{m}$.

Proof:

$$\begin{aligned}
\text{Given:} \quad & a \equiv b \pmod{m}, \\
\text{from the initial congruence,} \quad & a - b = \lambda m \quad \text{for some integer } \lambda. \\
\text{Multiplying both sides by } k, \quad & k(a-b) = k\lambda m, \\
& ka - kb = k\lambda m, \\
\text{which implies:} \quad & ka \equiv kb \pmod{m}.
\end{aligned}$$

Theorem:

If $a \equiv \tilde{a} \pmod{n}$ and $b \equiv \tilde{b} \pmod{n}$, then $ab \equiv \tilde{a}\tilde{b} \pmod{n}$.

Proof:

Given:

$$\begin{aligned}
a \equiv \tilde{a} \pmod{n} &\implies a = np + \tilde{a} \quad \text{for some integer } p, \\
b \equiv \tilde{b} \pmod{n} &\implies b = nq + \tilde{b} \quad \text{for some integer } q.
\end{aligned}$$

We want to show:

$$ab \equiv \tilde{a}\tilde{b} \pmod{n}$$

Start by substituting the expressions for a and b:

$$\begin{aligned}
ab &= (np + \tilde{a})(nq + \tilde{b}) \\
&= npnq + np\tilde{b} + \tilde{a}nq + \tilde{a}\tilde{b} \\
&= n(npq + p\tilde{b} + \tilde{a}q) + \tilde{a}\tilde{b}.
\end{aligned}$$

Here, $npq + p\tilde{b} + \tilde{a}q$ is clearly an integer as it is a sum of products of integers, so we can rewrite ab as:

$$ab = n \cdot \text{integer} + \tilde{a}\tilde{b}$$

Thus, by the definition of modulo:

$$ab \equiv \tilde{a}\tilde{b} \pmod{n}$$

Conclusion: The product of two integers modulo n is equivalent to the product of their respective equivalences modulo n, completing the proof.

Corollaries:

If $a \equiv b \pmod{m}$ and $c \equiv d \pmod{m}$, then this implies:

1. $ka \equiv kb \pmod{m}$
2. $ac \equiv bd \pmod{m}, \quad ad \equiv bc \pmod{m}$
3. $a^k \equiv b^k \pmod{m}$

These swiftly follow from the congruence product rule. We leave the formal proofs to the reader.

Note on Converse: If $ab \equiv \tilde{a}\tilde{b} \pmod{n}$, does it necessarily follow that $a \equiv \tilde{a} \pmod{n}$ and $b \equiv \tilde{b} \pmod{n}$? We give a counterexample. Consider $n = 6$, and let:

- $a = 4$ and $\tilde{a} = 10$
- $b = 9$ and $\tilde{b} = 3$

Here, we calculate:

$$ab = 4 \times 9 = 36$$
$$\tilde{a}\tilde{b} = 10 \times 3 = 30$$

Both 36 and 30 give the same remainder when divided by 6, i.e., $36 \mod 6 = 0$ and $30 \mod 6 = 0$. Thus, $ab \equiv \tilde{a}\tilde{b} \pmod{6}$. However:

- $a \not\equiv \tilde{a} \pmod{6}$ since $4 \not\equiv 10 \pmod{6}$
- $b \not\equiv \tilde{b} \pmod{6}$ since $9 \not\equiv 3 \pmod{6}$

Addition Property in Modulo Arithmetic

Property: If $a \equiv \tilde{a} \pmod{n}$ and $b \equiv \tilde{b} \pmod{n}$, then $(a+b) \equiv (\tilde{a}+\tilde{b}) \pmod{n}$.

Proof

Given:

- $a \equiv \tilde{a} \pmod{n}$ implies $a = \tilde{a} + kn$ for some integer k.
- $b \equiv \tilde{b} \pmod{n}$ implies $b = \tilde{b} + jn$ for some integer j.

We need to show that $(a+b) \equiv (\tilde{a}+\tilde{b}) \pmod{n}$.

Calculation:

$$a+b = (\tilde{a}+kn) + (\tilde{b}+jn) = (\tilde{a}+\tilde{b}) + (k+j)n$$

Since $(k+j)$ is an integer, $(k+j)n$ is a multiple of n, thus:

$$a+b \equiv \tilde{a}+\tilde{b} \pmod{n}$$

This completes the proof of the addition property.

Note on Converse:

If $(a+b) \equiv (\tilde{a}+\tilde{b}) \pmod{n}$, does it necessarily follow that $a \equiv \tilde{a} \pmod{n}$ and $b \equiv \tilde{b} \pmod{n}$?

We give a counterexample. Consider $n = 5$ and let:

- $a = 1$, $\tilde{a} = 6$ (Note $a \not\equiv \tilde{a} \pmod{5}$)
- $b = 4$, $\tilde{b} = -1$ (Note $b \not\equiv \tilde{b} \pmod{5}$)

Calculating the sums:

$$a+b = 1+4 = 5$$

$$\tilde{a}+\tilde{b} = 6-1 = 5$$

Both sums modulo 5 yield:

$$5 \mod 5 = 0$$

Thus:

$$(a+b) \equiv (\tilde{a}+\tilde{b}) \pmod{5}$$

However, individually:

$$a \not\equiv \tilde{a} \pmod{5}, \quad b \not\equiv \tilde{b} \pmod{5}$$

This shows that $(a+b) \equiv (\tilde{a}+\tilde{b}) \pmod{n}$ does not imply $a \equiv \tilde{a} \pmod{n}$ and $b \equiv \tilde{b} \pmod{n}$.

Divisibility Rules and Their Proofs

Divisibility by 3 and 9

Rule: A number is divisible by 3 if and only if the sum of its digits is divisible by 3. Similarly, a number is divisible by 9 if and only if the sum of its digits is divisible by 9.

Proof: Consider a number n represented by the digits $d_k d_{k-1} \ldots d_1 d_0$, which can be written as:

$$n = d_k \cdot 10^k + d_{k-1} \cdot 10^{k-1} + \ldots + d_1 \cdot 10^1 + d_0 \cdot 10^0$$

Since $10 \equiv 1 \pmod 3$ and $10 \equiv 1 \pmod 9$, raising 10 to any power yields $10^m \equiv 1^m \equiv 1 \pmod 3$ and $\pmod 9$. Thus:

$$n \equiv d_k \cdot 1 + d_{k-1} \cdot 1 + \ldots + d_1 \cdot 1 + d_0 \cdot 1 \equiv (d_k + d_{k-1} + \ldots + d_1 + d_0) \pmod 3$$

and similarly for modulo 9. Therefore, n is divisible by 3 or 9 **if and only if** the sum of its digits is divisible by 3 or 9, respectively.

Divisibility by 4 and 8 (and 2^k)

Rule: A number is divisible by 4 if and only if its last two digits form a number that is divisible by 4. It is divisible by 8 if and only if its last three digits form a number divisible by 8.

Proof: Since $10^2 = 100 \equiv 0 \pmod 4$ and $10^3 = 1000 \equiv 0 \pmod 8$, any higher powers of 10 also result in 0 mod 4 and 8. Therefore:

$$n = d_k \cdot 10^k + \ldots + d_2 \cdot 10^2 + d_1 \cdot 10^1 + d_0 \equiv d_1 \cdot 10 + d_0 \pmod 4$$

$$n \equiv d_2 \cdot 100 + d_1 \cdot 10 + d_0 \pmod 8$$

This shows that only the last two digits for divisibility by 4 and the last three digits for divisibility by 8 matter. This pattern holds for any power of 2 as well because 10^j contains j factors of 10, each contributing a factor of 2, and thus 10^j has at least j factors of 2. Therefore, if $j \geq k$, 10^j will contain at least k factors of 2, making it divisible by 2^k.

This is illustrated below.

$$n = \underbrace{d_m \cdot 10^m + \ldots + d_{k+1} \cdot 10^{k+1}}_{\equiv 0 \pmod{2^k}} + d_k \cdot 10^k + \ldots + d_1 \cdot 10^1 + d_0$$

$$n \equiv d_k \cdot 10^k + \ldots + d_1 \cdot 10 + d_0 \pmod{2^k}$$

Divisibility by 11

Rule: A number is divisible by 11 if and only if the difference between the sum of its digits in odd positions and the sum of its digits in even positions is divisible by 11.

Proof: Consider a number represented in decimal form as $n = d_k \cdot 10^k + d_{k-1} \cdot 10^{k-1} + \ldots + d_1 \cdot 10^1 + d_0 \cdot 10^0$. The properties of modular arithmetic, particularly the behavior of powers of 10 modulo 11, play a crucial role in establishing the divisibility rule for 11. We note that:

$$10 \equiv -1 \pmod{11}$$

Property: In modular arithmetic, the product rule states that for any integers a, b, and m, the following holds:

$$(a \times b) \mod m = [(a \mod m) \times (b \mod m)] \mod m$$

Thus, the power 10^n will be congruent to $(-1)^n$ modulo 11. This equivalence implies that raising 10 to any power will yield alternating signs:

$$10^1 \equiv -1 \pmod{11}, \quad 10^2 \equiv 1 \pmod{11}, \quad 10^3 \equiv -1 \pmod{11}, \quad \text{and so on.}$$

Applying this property to each digit's contribution in the number n, we get:

$$n \equiv d_k \cdot (-1)^k + d_{k-1} \cdot (-1)^{k-1} + \ldots + d_1 \cdot (-1)^1 + d_0 \cdot (-1)^0 \pmod{11}$$

The expression $d_k \cdot (-1)^k + d_{k-1} \cdot (-1)^{k-1} + \ldots + d_0$ represents the alternating sum of the digits, where each digit is alternatively added or subtracted depending on whether its position is even or odd. This means:

$$n \equiv d_0 - d_1 + d_2 - d_3 + \ldots + (-1)^k d_k \pmod{11}$$

Hence, n is divisible by 11 if and only if the alternating sum of its digits is divisible by 11.

Division in Modular Arithmetic

First, we will build some intuition as to division and its limitations. Consider $8 \equiv 2 \pmod{6}$. Dividing both sides by 2: $4 \equiv 1 \pmod{6}$. This is false. We will see a notion of division is only guaranteed to hold when the divisor and the mod are coprime.

Claim: If $ax \equiv b \pmod{c}$ and $\frac{b}{a}$ is an integer, then $x = \frac{b}{a} \pmod{c}$ is a valid solution.

Given:

- $ax \equiv b \pmod{c}$
- $\frac{b}{a} = d$ where d is an integer

Objective:

To prove that $x = d$ is a valid solution to $ax \equiv b \pmod{c}$.

Proof:

1. Starting from the given modular equation:

$$ax \equiv b \pmod{c}$$

2. Substituting $x = d$:

$$a\left(\frac{b}{a}\right) \equiv b \pmod{c}$$

3. Simplifying the equation:

$$b \equiv b \pmod{c}$$

Limited Solution Set: The simplification $b \equiv b \pmod{c}$ is always true, thereby confirming that $x = \frac{b}{a}$ is indeed a valid solution to the original equation under the modulus c. But, this does not always give all solutions.

Problem:

Solve $4x \equiv 12 \pmod{20}$. Given that $4x \equiv 12 \pmod{20}$, we observe that $\frac{12}{4} = 3$ is an integer. Thus, one solution is:

$$x = 3$$

$$4 \times 3 = 12 \equiv 12 \pmod{20}$$

This shows that $x = 3$ is indeed a solution. But, this is not the form of all solutions. Consider $x = 8$...

Since $x = 3$ is a solution, other solutions can be derived by adding the modulus divided by the gcd of 4 and 20, which is 4:

$$x = 3 + \frac{20}{4}k$$

$$x = 3 + 5k$$

where k is any integer. We will prove this in the next pages. For now, note that if the gcd was 1 then the "standard" notion of division would work (the same one solution repeats every cycle). Does that mean the gcd is the number of solutions per cycle? Yes - assuming a solution exists. We will revisit this.

Proposition:

If $ka \equiv kb \pmod{m}$ and k, m are coprime, then $a \equiv b \pmod{m}$.

Proof:

$$\begin{aligned}
\text{Given:} \quad & ka \equiv kb \pmod{m}, \\
\text{This implies:} \quad & ka - kb = m\lambda \quad \text{for some integer } \lambda. \\
\text{Rewrite:} \quad & k(a-b) = m\lambda. \\
\text{Since } \gcd(k,m) = 1, \quad & \text{it follows that } k \text{ and } m \text{ share no common factors.} \\
& \text{Therefore, } k \text{ cannot divide } m.
\end{aligned}$$

Assume for contradiction that k does not divide λ.

$$\text{Then, } \frac{\lambda}{k} \text{ is not an integer, and thus } m\left(\frac{\lambda}{k}\right) \text{ is not an integer.}$$

However, this contradicts the fact that $a - b$ is an integer

$$\text{because we know } m\left(\frac{\lambda}{k}\right) = a - b.$$

Thus, our assumption must be wrong, and k must divide λ.

$$\begin{aligned}
& \text{Therefore, we can write } \lambda' = \lambda/k \text{ for some integer } \lambda'. \\
\text{After division, we get:} \quad & a - b = m\lambda', \\
\text{Which implies:} \quad & a \equiv b \pmod{m}.
\end{aligned}$$

Proposition:

Consider the sequence $a, 2a, 3a, \ldots, (m-1)a$ where a and m are integers such that $\gcd(a, m) = 1$ (i.e., a and m are coprime). We make the following claims about this sequence:

Claims:

1. None of those are divisible by m.
2. No two of those leave the same remainder when divided by m.

Proof of Claim 1: Suppose m divides ja for some j where $1 \leq j \leq m-1$. Since a and m are coprime, this implies m divides j. But since $j \leq m-1$, j is less than m and cannot be divisible by m, leading to a contradiction. Therefore, none of $a, 2a, 3a, \ldots, (m-1)a$ are divisible by m.

Proof of Claim 2: Suppose there exist i and j such that $1 \leq i, j \leq m-1$ and $ia \equiv ja \pmod{m}$. This can be rewritten as:

$$(i-j)a \equiv 0 \pmod{m}$$

Since a and m are coprime, m must divide $i-j$. However, the possible values of $i-j$ range from $-(m-2)$ to $m-2$, which are all less than m and greater than $-m$ (except for $i - j = 0$), implying $i = j$. Hence, no two of $a, 2a, 3a, \ldots, (m-1)a$ leave the same remainder when divided by m, contradicting the assumption that they do.

Fermat's Little Theorem

Theorem: If p is a prime number and a is any integer such that p does not divide a (i.e., $\gcd(a, p) = 1$), then:

$$a^{p-1} \equiv 1 \pmod{p}$$

Proof:

Consider the set of integers $\{a, 2a, 3a, \ldots, (p-1)a\}$. Since $\gcd(a, p) = 1$, none of these integers are congruent to each other modulo p. This means that the set $\{a, 2a, 3a, \ldots, (p-1)a\}$ forms a complete residue system modulo p, except for zero.

Multiplying all the elements of this set, we get:

$$a \cdot 2a \cdot 3a \cdots (p-1)a$$

This product can be written as:

$$a^{p-1} \cdot (p-1)!$$

Since the set $\{a, 2a, 3a, \ldots, (p-1)a\}$ have distinct remainders modulo p, and there are $p-1$ of them, we know there exists exactly one integer in the set that is congruent to each of $1, 2, ...p-1$. Thus, by the commutative property of multiplication and modulo product rules, we can state the following:

$$a \cdot 2a \cdot 3a \cdots (p-1)a \equiv 1 \cdot 2 \cdot 3 \cdot \cdots \cdot (p-1) \pmod{p}$$

$$a^{p-1} \cdot (p-1)! \equiv (p-1)! \pmod{p}$$

We are not claiming that p has a remainder of 1 and so on, just that there is a one-to-one correspondence between the p_i and components in the factorial.

Since $(p-1)!$ and p are coprime (because p is a prime and none of the terms in $(p-1)!$ share factors with p), we can divide both sides of the congruence by $(p-1)!$. This is because the only factor they could share is p (p is prime) and they are all less than p.

Said simply, $(p-1)!$ cannot share a factor because there is no way to construct a prime with multiplication, and none of the individual factors could be p - they were all less than p. giving:

$$a^{p-1} \equiv 1 \pmod{p}$$

This completes the proof of Fermat's Little Theorem.

Theorem:

If $dx \equiv c \pmod{n}$, letting $g = \gcd(d, n)$, then: $\frac{d}{g}x \equiv \frac{c}{g} \pmod{\frac{n}{g}}$.

Proof

Part 1: Checking for Solutions Assume $dx \equiv c \pmod{n}$.
Given $g = \gcd(d, n)$:

- **Divisibility**: If $dx \equiv c \pmod{n}$, by the definition of modulo, n divides $dx - c$. Therefore, there exists an integer k such that:

$$dx - c = kn$$

- **Divisibility by** g: Since g divides both d and n, g also divides dx and thus kn. It follows that g must divide c because we can write c as a linear combination of terms divisible by g.

$$dx - kn = c$$

Part 2: Simplifying the Congruence We now know g divides c. Let $c = gm$ for some integer m.

- **Expression Substitution**: Substitute $d = ga$ and $n = gb$. Since g is the gcd, a and b are integers where $\gcd(a, b) = 1$). If $\gcd(a, b) \neq 1$ then d and n would share yet another factor on top of the gcd, which is an absurdity.

$$gax \equiv gm \pmod{gb}$$

We can write:

$$g(ax - m) = gbk$$
$$ax - m = bk$$

Part 3: Transforming to New Modulo We recognize that $ax - m = bk$ can be equivalently written in the following ways:

$$ax - m \equiv 0 \pmod{b}$$

$$ax \equiv m \pmod{b}$$

This simplifies to:

$$\frac{d}{g}x \equiv \frac{c}{g} \pmod{\frac{n}{g}}$$

by recognizing that $a = \frac{d}{g'}$, $m = \frac{c}{g'}$, and $b = \frac{n}{g'}$.

Conclusion: This shows that $\gcd(d, n)$ divides c and the equation $dx \equiv c \pmod{n}$ simplifies to $\frac{d}{g}x \equiv \frac{c}{g} \pmod{\frac{n}{g}}$. Note: the converse also holds, as you will see.

Theorem:

Let $a, b, c \in Z$ and d be a non-zero integer. If $a \equiv b \pmod{c}$, then it is always true that $da \equiv db \pmod{dc}$.

$$a \equiv b \pmod{c} \implies da \equiv db \pmod{dc}$$

Proof:

Given that $a \equiv b \pmod{c}$, by definition, this means there exists an integer k such that:

$$a = b + kc$$

Multiplying through by d (where d is non-zero), we have:

$$da = db + dkc$$

$$da = db + k(dc)$$

Since dc is also an integer (as the product of two integers), this can be rewritten as:

$$da \equiv db \pmod{dc}$$

Corollary: A consequence of this proof is a fact we established earlier.

$$a \equiv b \pmod{c} \implies da \equiv db \pmod{c}$$

This is because if it has the same remainder when divided by a scaled version of a number, it will have the same remainder when you divide by that number (a factor).

Note that we could substitute $k' = kd$ and use associativity in the below expression to prove this algebraically.

$$da = db + k(dc)$$

This leads to the interesting fact that if $d > 1$ then $k' > k$. Intuitively, if we go from $da \equiv db \pmod{dc}$, if we made the step size c instead of dc, it would take longer to get to the intersection.

Prime Factor Algorithm:

We now present an algorithm programmed in Java to efficiently find the prime factors of a number. This allows us to express any number we desire as a product of primes. We also present an algorithm to check if a number is prime. We first must establish two principles used in these algorithms.

1. To determine if a number is prime, it is only necessary to consider potential factors up to $\sqrt{n}$.

Assume for the sake of contradiction that both a and b are factors of n and are greater than $\sqrt{n}$. Then we have:

$$a > \sqrt{n} \quad \text{and} \quad b > \sqrt{n}$$

Multiplying these inequalities together:

$$a \times b > \sqrt{n} \times \sqrt{n}$$

$$a \times b > n$$

But this contradicts our original statement that $n = a \times b$. Thus, our assumption must be wrong, and at least one of a or b must be less than or equal to $\sqrt{n}$. We are only interested in integers, so we can stop at the floor of $\sqrt{n}$: $\lfloor \sqrt{n} \rfloor$. In our code, we achieve this without the floor function because our step size will run at the floor (required), increment, and then stop because the step size is not less than 1.
2. To determine if a number is prime, if it is not 2 or 3, it suffices to check for i only numbers that have a remainder of 5 or 1 under modulo 6.

- If the remainder is 0 or 2 or 4 under modulo 6, it is even and divisible by 2.
- If the remainder is 0 or 3 under modulo 6, it is divisible by 3

Thus, with the exception of 2 and 3, the only numbers that can be prime under modulo 6 are numbers that are congruent to 1 or 5.

Thus, for i starting at 5 modulo 6 and increasing by 6 each iteration, only i and $i + 2$ need to be checked for further divisibility tests, as all other numbers in this sequence are divisible by either 2 or 3.

```
import java.util.TreeMap;

public class PrimeFactors {

    //method to check if a number is prime
    //this is standalone and not called by primeFactors
    public static boolean isPrime(int number) {
        if (number <= 1) {
            return false;
        }
        if (number <= 3) {
            return true;
        }
        if (number % 2 == 0 || number % 3 == 0) {
            return false;
        }
        int i = 5;
        //we start at 5. divisibility by 2 rules out 6,8,10.
        //divisibility by 3 rules out 6,9.
        //we must check 7, 11, 13. and so on.
        while (i * i <= number) {
            if (number % i == 0 || number % (i + 2) == 0) {
                return false;
            }
```

```
            i += 6;
        }
        return true;
    }

    //method to return a TreeMap of prime factors and their multiplicity
    public static TreeMap<Integer, Integer> primeFactors(int number) {
        TreeMap<Integer, Integer> factors = new TreeMap<>();
        //pull out all factors of 2
        while (number % 2 == 0) {
            factors.put(2, factors.getOrDefault(2, 0) + 1);
            number /= 2;
        }
        //pull out all factors of 3
        while (number % 3 == 0) {
            factors.put(3, factors.getOrDefault(3, 0) + 1);
            number /= 3;
        }
        //we checked for factors of 2 and 3 already, so we can use a faster step
        //when we reach an i value and it divides, we know it is prime
        //this is because it is was composite, then there would be an earlier pr
        //but, we aren't missing any primes.
        for (int i = 5; i <= Math.sqrt(number); i += 6) {
            //pulling out factors of i - as many as we can
            while (number % i == 0) {
                factors.put(i, factors.getOrDefault(i, 0) + 1);
                number /= i;
            }
            //if i+2 divides num, it is prime because both factors have to be le
            //if i+2 is greater than sqrt, that is okay, it will either not be a
            while (number % (i + 2) == 0) {
                    factors.put(i + 2, factors.getOrDefault(i + 2, 0) + 1);
                    number /= (i + 2);
            }
        }

        //once we've checked to the sqrt, we now know the remaining number is pr
        //we must add the remaining number if it is not 1 (it cannot be two or t
        if (number > 4) {
            factors.put(number, 1);
        }
        return factors;
    }
    //output: Prime factors of 360 are: {2=3, 3=2, 5=1}

}
```