Written by Zachariah Abueg
10/19/20

Fall 2020 CIS 3362 Homework #5: Number Theory Solutions

1) Without the aid of a computer program, determine the prime factorization of 12180168000. Show your work. You may do division in a calculator.

Solution

Of course, there are a few ways to prime factorize this number. The way I chose to do it was to keep dividing it by prime numbers until we reach 1 and creating the prime factorization from there. In fact, this is equivalent to making those prime factorization trees we made in middle school. What I'm doing is just the formalization of that method (read: the more pretentious way).

Notice that our number, $12,180,168,000$, ends in three zeros, so we know that it is a multiple of 1,000. Now, $1,000 = 10^3 = (2 \times 5)^3 = 2^3 \times 5^3$, so right off the bat, we have 2^3 and 5^3 in our prime factorization. Dividing $12,180,168,000$ by $1,000$, we have $12,180,168$.

From here, we can keep dividing by 2 until we reach a non-integer. Using the calculator, we can divide 12,180,168 by 2 three times to get $1,522,521$, after which dividing by 2 gives 761,260.5. After having divided by 2 three times, we know have an additional 2^3 factor, giving 2 in our prime factorization the exponent $2^{3+3} = 2^6$. So far, our prime factorization is $2^6 \times 5^3$.

Now we are at 1,522,521 after dividing out all of the 2s from our original number. From here, let's divide 1,522,521 by 3 until we reach a non-integer. Using the calculator, we can divide by 3 two times to get 169,169, after which dividing by 3 gives 56389.666… repeating. After having divided by 3 two times, we know have an additional 3^2 factor. So far, our prime factorization is $2^6 \times 3^2 \times 5^3$.

Now we are at 169,169 after dividing out all of the 2s and 3s from our original number. Where do we go next? We don't divide by 4, because dividing out all of the 2s means we've already divided out all of the 4s, as $4 = 2^2$. Similarly, we wouldn't divide by 9, because dividing out all of the 3s means we've already divided out all of the 9s, as $9 = 3^2$. In fact, in this method, we will only divide by prime numbers, since all composite numbers are made up of prime factors.

So remember, we are at 169,169. Clearly, dividing by 5 is futile, as we know that all multiples of 5 end in either 0 or 5 and we can see that 169,169 does not. Thus, we have divided out all 5s from our original number. We then try dividing 169,169 by the next prime number, 7, until we reach a non-integer, which we can do one time to reach 24,167. Thus, 7^1 is a prime factor. After that, we try dividing 24,167 by 11, which we can do one time to reach 2,197. Thus, 11^1 is a prime factor. Finally, we divide 2,197 by 13 three times to reach 1, giving us 13^3 as a prime factor.

We have now reached 1 after dividing by prime numbers, so combining all of our prime factors, we have

$$12,180,168,000 = 2^6 \times 3^2 \times 5^3 \times 7^1 \times 11^1 \times 13^3.$$

2) What is φ(12180168000)?

Solution

We use the fact that $\varphi(n)$ is multiplicative and that for a prime p and positive integer k, $\varphi(p^k) = p^k - p^{k-1}$ to get

$$\varphi(12180168000) = \varphi\left(2^6 \times 3^2 \times 5^3 \times 7^1 \times 11^1 \times 13^3\right) =$$
$$\varphi\left(2^6\right) \times \varphi\left(3^2\right) \times \varphi\left(5^3\right) \times \varphi\left(7^1\right) \times \varphi\left(11^1\right) \times \varphi\left(13^3\right)$$
$$\left(2^6 - 2^5\right) \times \left(3^2 - 3^1\right) \times \left(5^3 - 5^2\right) \times \left(7^1 - 7^0\right) \times \left(11^1 - 11^0\right) \times \left(13^3 - 13^2\right) =$$
$$32 \times 6 \times 100 \times 6 \times 10 \times 2028 = 2,336,256,000.$$

3) Prove that $\varphi(n) = n \prod_{p \in P(n)} \frac{p-1}{p}$, where the set P(n) represents the set of unique prime factors of n. For example, P(96) = {2, 3} and P(7000) = {2, 5, 7}. Use the formula shown in class for the starting point of your proof.

Solution

Note: Here, the notation $\prod_{x \in X}$, called product notation, denotes the product iterating over all x in a set X. It is similar to $\sum_{x \in X}$, which denotes the sum iterating over all x in a set X.

Let $n = {p_1}^{a_1} \times {p_2}^{a_2} \times ... \times {p_k}^{a_k}$, where each $p_i \in P(n)$ is a prime factor of n and each a_i is the exponent of its associated prime factor p_i. Then

$$\varphi(n) = \varphi\left({p_1}^{a_1} \times {p_2}^{a_2} \times ... \times {p_k}^{a_k}\right).$$

Now, we know that the function φ is multiplicative — that is, if $\gcd(a, b) = 1$, then $\varphi(ab) = \varphi(a)\varphi(b)$. Since each p_i is prime, each ${p_i}^{a_i}$ is prime, which means that all ${p_i}^{a_i}$ are coprime with each other, or $\gcd({p_1}^{a_1}, {p_2}^{a_2}, ..., {p_k}^{a_k}) = 1$. Hence, we can apply multiplicativity:

$$\varphi\left({p_1}^{a_1} \times {p_2}^{a_2} \times ... \times {p_k}^{a_k}\right) = \varphi\left({p_1}^{a_1}\right) \varphi\left({p_2}^{a_2}\right) ... \varphi\left({p_k}^{a_k}\right).$$

Now, we know that for any prime number p and a positive integer j, we have $\varphi(p^j) = p^j - p^{j-1} = p^j\left(1 - \frac{1}{p}\right) = p^j\left(\frac{p-1}{p}\right)$. Thus, for each ${p_i}^{a_i}$, we have

$$\varphi\left({p_1}^{a_1}\right) \varphi\left({p_2}^{a_2}\right) ... \varphi\left({p_k}^{a_k}\right) = \left({p_1}^{a_1}\left(\frac{p_1 - 1}{p_1}\right)\right)\left({p_2}^{a_2}\left(\frac{p_2 - 1}{p_2}\right)\right) ... \left({p_k}^{a_k}\left(\frac{p_k - 1}{p_k}\right)\right).$$

Now, as multiplication is commutative, we can arrange the previous product as follows:

$$\left({p_1}^{a_1}\left(\frac{p_1 - 1}{p_1}\right)\right)\left({p_2}^{a_2}\left(\frac{p_2 - 1}{p_2}\right)\right) ... \left({p_k}^{a_k}\left(\frac{p_k - 1}{p_k}\right)\right) = \left({p_1}^{a_1} \times {p_2}^{a_2} \times ... \times {p_k}^{a_k}\right)\left(\frac{p_1 - 1}{p_1}\right)\left(\frac{p_2 - 1}{p_2}\right) ... \left(\frac{p_k - 1}{p_k}\right)$$

Finally, we know that $n = p_1^{a_1} \times p_2^{a_2} \times ... \times p_k^{a_k}$, and we can rewrite $\left(\frac{p_1 - 1}{p_1}\right)\left(\frac{p_2 - 1}{p_2}\right)...\left(\frac{p_k - 1}{p_k}\right)$ in product notation to obtain the desired result:

$$\varphi(n) = n \prod_{i=1}^{k} \frac{p_i - 1}{p_i} = n \prod_{p \in P(n)} \frac{p - 1}{p}$$

3) Using Fermat's Theorem, determine the remainder when 7^{11596} is divided by 967.

Solution

How will we use Fermat's Theorem to find the remainder of this huge power?

First note that the remainder when x is divided by y is equivalent to $x \pmod{y}$, so we want to find $7^{11596} \pmod{967}$. Now, recall that Fermat's Theorem states that if p is prime and $\gcd(a, p) = 1$, then $a^{p-1} \equiv 1 \pmod{p}$.

We know from exponent rules that

$$(a^b)^c = a^{b \times c} \tag{1}$$

or

$$a^{b \times c} = (a^b)^c \tag{2}$$

If we have from Fermat's Theorem that $a^{p-1} \equiv 1 \pmod{p}$ for some a and p, then we can exploit this congruence to raise higher powers of a:

Say we had a number greater than $p - 1$ that is a multiple of $p - 1$, let's say $k \times (p - 1)$ for some positive integer k, and we wanted to raise a to that power, $k \times (p - 1)$, and take the result mod p. That is, we want to know $a^{k \times (p-1)} \pmod{p}$. Then we use (2) to get

$$a^{k \times (p-1)} = a^{(p-1) \times k} = (a^{p-1})^k$$

and now, since we know from Fermat's Theorem that $a^{p-1} \equiv 1 \pmod{p}$, we have

$$(a^{p-1})^k \equiv 1^k = 1 \pmod{p}.$$

[Aside: Why can we do this? Why can we "replace" $(a^{p-1})^k$ with 1^k when working (mod p)? It is due to the rules of modular arithmetic. Suppose $a \equiv b \pmod{c}$. Then the rules of modular arithmetic tell us that multiplying a and b by the same integer are equivalent (mod c) - that is, $d \times a \equiv d \times b \pmod{c}$ for any integer d. So if we wanted to find $d \times a \pmod{c}$, we could simply find $d \times b \pmod{c}$ if it's easier to calculate instead.

Now, what would happen if we raised a and b to the same positive integer? Say we raise a and b to a positive integer d. We know that a^d is the same as multiplying a by itself d times and b^d is the same as multiplying b by itself d times. Now by the above statement, we have that

multiplying a by itself d times is equivalent (mod c) to multiplying b by itself d times - that is, $a \times a \times ... \times a \equiv b \times b \times ... \times b \pmod{c}$, or $a^d \equiv b^d \pmod{c}$. And this is why we are able to "replace" $(a^{p-1})^k$ with 1^k when working (mod p): because $a^{p-1} \equiv 1 \pmod{p}$.]

So we see that if we have a multiple of $p-1$ as a power of a, then we can easily break it down by exponent rules to get that $a^{k \times (p-1)} = a^{(p-1) \times k} = (a^{p-1})^k \equiv 1^k = 1 \mod p$.

Finally, what if our exponent (greater than $p-1$) isn't a perfect multiple of $p-1$? What then? We can no longer break it down as an integer times $p-1$, *but* we can still break it down as an integer times $p-1$ *plus* another integer. For instance, if $p-1=5$ but our exponent is 7, we can write 7 as $7 = 1 * 5 + 2$; if our exponent is 33, we can write $33 = 6 * 5 + 3$. Knowing this, we use exponent rules once again. Exponent rules tell us that

$$a^{b+c} = a^b \times a^c \tag{3}$$

Thus, if we have an exponent $k > p-1$ that can be written as $k = m \times (p-1) + n$, then we can break down a^k using (3) as follows:

$$a^k = a^{m \times (p-1) + n} = a^{m \times (p-1)} \times a^n = a^{(p-1) \times m} \times a^n = (a^{p-1})^m \times a^n$$

and now, using Fermat's Theorem and the rules of modular arithmetic, we have

$$(a^{p-1})^m \times a^n \equiv 1^m \times a^n = a^n \pmod{p}.$$

And hence the motivation for using Fermat's Theorem to find remainders of large powers. Now we will solve the problem at hand. We want to find $7^{11596} \pmod{967}$ using Fermat's Theorem: $a^{p-1} \equiv 1 \pmod{p}$ for prime p and integers a coprime to p.[1]

Take the prime $p = 967$ and $a = 7$. Since $\gcd(7, 967) = 1$, we can apply Fermat's Theorem:

$$7^{966} \equiv 1 \pmod{967}.$$

We then write 11596 as as integer times 966 plus another integer: $11596 = 12 \times 966 + 4$. Finally, using exponent rules we have

$$7^{11596} = 7^{12 \times 966 + 4} = 7^{12 \times 996} \times 7^4 = 7^{996 \times 12} \times 7^4 = \left(7^{966}\right)^{12} \times 7^4 \equiv 1^{12} \times 2401 \equiv 467 \pmod{967}.$$

Thus, the remainder when 7^{11596} is divided by 967 is 467.

[1] Notice: since p is prime, almost every integer is coprime to p: the only integers that are *not* coprime to p are multiples of p. This gives you an easy way to check if an integer a is coprime to p for using Fermat's Theorem: if it's not divisible by p, then it's coprime to p and Fermat's Theorem applies. Wala!

4) Using Euler's Theorem, determine $99^{10754} \bmod 3104$.

Solution

Euler's Theorem states that for any positive integer n and an integer a that is coprime to n, we have $a^{\varphi(n)} \equiv 1 \pmod{n}$. We apply the same reasoning here that we did with Fermat's Theorem.

Take $n = 3104 > 0$ and $a = 99$. We first determine whether a and n are coprime by using their prime factorizations: $3104 = 2^5 \times 97$ and $99 = 3^2 \times 11$. Since a and n do not share any prime factors, they are coprime and we can apply Euler's Theorem:

$$99^{\varphi(3104)} \equiv 1 \pmod{3104}.$$

Now, we calculate $\varphi(3104)$. As $3104 = 2^5 \times 97$, we can use the statement in question 3 to get

$$\varphi(3104) = 3104 \times \frac{2-1}{2} \times \frac{97-1}{97} = 2^5 \times 97 \times \frac{1}{2} \times \frac{96}{97} = 2^4 \times 96 = 1536.$$[2]

Hence,

$$99^{1536} \equiv 1 \pmod{3104}.$$

We then write 10754 as an integer times 1536 plus another integer: $10754 = 7 \times 1536 + 2$.

Finally, using exponent rules we have

$$99^{10754} = 99^{7 \times 1536 + 2} = 99^{7 \times 1536} \times 99^2 = 99^{1536 \times 7} \times 99^2 = \left(99^{1536}\right)^7 \times 99^2 \equiv 1^7 \times 9801 \equiv 489 \pmod{3104}.$$

Thus, $99^{10754} \equiv 489 \pmod{3104}$.

[2] Alternatively, we could have used the multiplicativity of $\varphi(n)$ and the fact that for a prime p and positive integer k, $\varphi(p) = p - 1$ and $\varphi(p^k) = p^k - p^{k-1}$:

$$\varphi(3104) = \varphi\left(2^5 \times 97\right) = \varphi\left(2^5\right) \times \varphi(97) = \left(2^5 - 2^4\right) \times 96 = 2^4 \times 96 = 1536.$$

5) Write a program that reads in an integer entered by the user (in between 2 and 1000) and determines if the integer is prime. If it is NOT prime, just report a proper divisor of the number less than 1 and end the program. If the number entered is prime, list out each primitive root of the prime number in between 2 and the number minus one, in numerical order.

Here are a couple sample runs of the program:

Sample 1

Enter n.

143

143 is not prime. It's smallest non-trivial divisor is 11.

Sample 2

Enter n.

17

17 is prime.
Its primitive roots are: 3 5 6 7 10 11 12 14

Solution
I wrote the program **primitiveRoots.java** for this problem. Here I will go through it.

First, we determine whether the input, n, is prime using *smallestPrimeFactor(int n)*, which divides n by every number from 2 to $\sqrt{n}$. If it finds a non-trivial divisor[3] of n, then n is composite and it returns that non-trivial divisor of n. Otherwise, it didn't find a non-trivial divisor of n - this means that the only divisors of n are 1 and n. Thus, n is prime.

Now, why do we check up to $\sqrt{n}$? Because larger factors of n are multiples of smaller factors that have already been checked. For instance, take $n = 30$. The factors of 30 are 1, 2, 3, 5, 6, 10, 15, 30, and they are paired up as 1×30, 2×15, 3×10, and 5×6. We only need to go up to $\sqrt{30} \approx 5.4$, or up to 5, because everything after 5 is already a multiple of a smaller factor: 6×5, 10×3, 15×2, and 30×1. Thus, if we have checked through 1, 2, 3, and 5, then we have implicitly checked through 6, 10, 15, and 30. This observation allows us to cut down the number of integers we have to check through: instead of checking for non-trivial divisors from 2 all the way to n, we need only check for divisors from 2 to $\sqrt{n}$. Efficiency, baby!

Now, if n is prime, then it must have primitive roots. From here, we find those primitive roots. Recall that a primitive root of a prime p is a number a (with $1 \leq a \leq p-1$) such that the smallest power of a that is equivalent to 1 (mod p) is $p-1$. Going back to the program, we first let $p = n$ just for readability, since we now know n is prime. To find primitive roots of p, we iterate through all numbers a from 1 to $p-1$ and for each number a, we raise it to all powers from 1 to $p-1$, taking the result (mod p) each time.[4] If we reach 1 (mod p) before we raise a to $p-1$ (that is, the smallest power of a that is equivalent to 1 (mod p) is *less* than $p-1$), then we know a is not a primitive root of p. Otherwise, the smallest power of a that is equivalent to 1 (mod p) is $p-1$, so a is a primitive root of p. To keep track of the primitive roots of p, we store them in an ArrayList, which is the Java equivalent of a dynamically allocated array in C.

I know Intro to C is the only requirement for this course, so I have made some comments throughout the program on some Java syntax and their equivalents in C. Also, if you want to run my program, there are many Java compilers online you can easily use. **Note: Python and C solutions have been included as well.**

[3] A non-trivial divisor means a divisor that isn't 1, since 1 is trivially a divisor of every number.

[4] Notice that to exponentiate a in my program, I used repeated multiplication and took the result (mod p) every time instead of straight up exponentiating and *then* taking the result (mod p). What might I have chosen to do this? (And with that I leave you. *Swipes cape over face and disappears into the darkness* You hear me faintly in the distance: "Happy Halloweeeeeeeeen.")