Fall 2023 CIS 3362 Homework #5: Number Theory Solution

1) (5 pts) Without the aid of a computer program, determine the prime factorization of 2,427,559,200. Show your work. You may do division on a calculator.

Solution
2,427,559,200 = 100 x 24275592

Now use a calculator to divide out as many copies of 2 as possible from 24275592:

$= 2^2 \times 5^2 \times 2^3 \times 3034449$

Noticing that the sum of digits of the large number is divisible by 9, we can divide out 3^2, and that result has a digit sum divisible by 3, so after dividing out all the 3s we have:

$= 2^5 \times 5^2 \times 3^3 \times 112387$

Continue trial division by primes in increasing order:

$= 2^5 \times 3^3 \times 5^2 \times 11 \times 10217 =$ **$2^5 \times 3^3 \times 5^2 \times 11 \times 17 \times 601$**

Trial division by primes upto 23 shows that 601 must be prime and we can stop. (This is because the next prime after 23 is 29 and $29^2 > 601$.)

2) (5 pts) What is φ(2,427,559,200)?

Solution
$\varphi(2,427,559,200) = \varphi(2^5) \times \varphi(3^3) \times \varphi(5^2) \times \varphi(11) \times \varphi(17) \times \varphi(601)$
$= (2^5 - 2^4)(3^3 - 3^2)(5^2 - 5)(11 - 1)(17 - 1)(601 - 1)$
$= 16 \times 18 \times 20 \times 10 \times 16 \times 600$
$=$ **552,960,000**

In prime factorized form, that's **$2^{15} \times 3^3 \times 5^4$.**

3) (5 pts) Use Fermat's Theorem to calculate the remainder when 12^{7250} is divided by 907?

Solution
Since 907 is prime and gcd(12, 907) = 1, by Fermat's Theorem, we have $12^{906} \equiv 1 \pmod{907}$.

$$12^{7250} = 12^{8(906)+2} = (12^{906})^8 12^2 \equiv 1^8(144) \equiv 144 \ (mod\ 907)$$

It follows that the desired remainder is 144.

4) (5 pts) Use Euler's Theorem to calculate the remainder when 77^{32641} is divided by 26010?

Solution

26010 = 2601 x 10 = 9 x 289 x 10 = 3^2 x 17^2 x 2 x 5

It follows that φ(26010) = φ(2) x φ(3^2) x φ(5) x φ(17^2) = (2 – 1)(3^2 – 3)(5 – 1)(17^2 – 17)
= 6 x 4 x 272
= 6528

Since gcd(77, 26010) = 1, due to Euler's Theorem, it follows that $77^{6528} \equiv 1 \pmod{26010}$.

$$77^{32641} = 77^{5(6528)+1} = (77^{6528})^5 77^1 \equiv 1^5(77) \equiv 77 \ (mod\ 26010)$$

It follows that the remainder when 77^{32641} is divided by 26010 is 77.

5) (10 pts) Show the steps of running the Miller-Rabin algorithm, testing n = 1729 for primality with the randomly chosen value of a = 2. Please use a calculator or computer program to calculate the modular exponents and just show the result of each squaring/mod operations.

Solution

n – 1 = 1728 = 2^6 x 27. (Note: 1728 is twelve cubed, a fact that some of aware of because 1729 is the smallest number that can be expressed in 2 different ways as the sum of two cubes. 1729 = 12^3 + 1^3 = 10^3 + 9^3.) Thus, k = 6, m = 27. (Step 1 from the typed up notes.)

2. Pick a = 2 as dictated by the problem.

3. We start with a = 2, m = 27. $2^{27} \equiv 645 \ (mod\ 1729)$. (This is step 3 from the typed up notes.)

4. Since the number above isn't 1, we continue the algorithm, repeatedly squaring the previous term:

5-6) a) $645^2 \equiv 1065 \ (mod\ 1729)$
b) $1065^2 \equiv 1 \ (mod\ 1729)$
c-f) Future steps will all yield a calculation of 1 (mod 1729)

7) We return composite, because we never achieve a value of 1728 mod 1729.

6) (10 pts) Trace through the Fermat Factoring algorithm to factor 45,241 as the product of two prime numbers. You may use a calculator or computer program to execute each calculation, but print out the result of each number being tested as a perfect square.

Solution

$\sqrt{45241} \sim 212.7$, thus, we can start our algorithm with x = 213

x	$x^2 - 45241$	Perfect Square?
213	128	No
214	555	No
215	984	No
216	1415	No
217	1848	No
218	2283	No
219	2720	No
220	3159	No
221	3600	Yes (60 x 60)

It follows that 45241 = (221 + 60)(221 – 60) = **281 x 161.**

It turns out that 161 is NOT prime. 161 = 7 x 23.

Thus, the full prime factorization of this number is 45241 = **7 x 23 x 281.**

Using Fermat Factorization, we split 45241 into two factors (closest to the square root of the number), and these are 161 and 281.

Note: Just the two factor split assuming the full chart is accurate will earn full credit. It was not necessary to note that 161 was not prime.

7) (10 pts) A primitive root, α, of a prime, p, is a value such that when you calculate the remainders of α, α^2, α^3, α^4 , ... , α^{p-1}, when divided by p, each number from the set {1, 2, 3, ..., p-1} shows up exactly once. Prove that a prime p has exactly φ(p-1) primitive roots. In writing your proof, you may assume that at least one primitive root of p exists. (Normally, this is the first part of the proof.) (Note: This question is difficult, so don't feel bad if you can't figure it out.) **(Note: The solution to this can probably be found on the internet, so I'll be looking for original explanations that show understanding but aren't identical to the book proofs…ie what a normal person would come up with after thinking about the problem on their own)**

Solution

Let α be an arbitrarily chosen primitive root of p.
Let k be an integer in between 1 and p – 1 such that gcd(k, p – 1) = 1.
Now consider the value $\beta \equiv \alpha^k (mod\ p)$. Our goal will be to prove that β is a primitive root as well.

Consider the sequence of values $\beta, \beta^2, \beta^3, \dots, \beta^{p-1}$ mod p. These will be equivalent to

$$\alpha^k, \alpha^{2k}, \alpha^{3k}, \dots, \alpha^{(p-1)k}$$

mod p, respectively.

We know the last of these is equivalent to 1 (mod p) via Fermat's Theorem.

What we would like to show is that these exponents, when taken mod (p – 1) will all be distinct values covering the set {0, 1, 2, …, p – 2}. If we can show that, then we know that the actual list of values itself are distinct mod p and that the first time the value of 1 shows up on the list is at the very end.

Assume to the contrary, that two values on the list of exponents, k, 2k, 3k, …, (p – 1)k are equivalent mod (p – 1). Then there exist distinct integers, i and j, with $1 \leq i < j \leq p - 1$, such that ki and kj are equivalent mod (p – 1):

$$kj \equiv ki\ (mod\ p-1)$$
$$kj - ki \equiv 0\ (mod\ p-1)$$
$$k(j-i) \equiv 0\ (mod\ p-1)$$

Because gcd(k, p – 1) = 1, it follows that $(j-i) \equiv 0\ (mod\ p-1)$. (This uses a rule that was stated in class but not proved. Intuitively, if there are no common factors between k and p – 1, then for p – 1 to divide this product, it has to entirely divide into the other part and not k.)

But, recall that 0 < j – i < p – 1. This means that p – 1 can NOT divide the difference between j and i, resulting in a contradiction. It follows that the original assumption was incorrect, and that each of the exponents to alpha are distinct mod p – 1.

Since α is a primitive root, by definition, this list of values is equivalent to each unique non-zero value mod p. Thus, if α is a primitive root, it follows that β is as well. Thus, the number of primitive

roots is at least the number of values k such that gcd(k, p – 1) = 1. By definition of the phi function, this is $\phi(p-1)$.

We must finally also prove that if gcd(k, p – 1) ≠ 1, then α^k is NOT a primitive root. Once we prove this, then we know the count above is accurate and not below the actual answer. (This is the only if part of the proof.) Let d = gcd(k, p – 1) > 1. Also, let $X = \frac{k}{d}$. We know that *X* must be an integer by definition of gcd.

Let $\beta = \alpha^k$. Consider the exponent $\frac{p-1}{d}$. Since g > 1, this exponent is strictly less than p – 1. Now, calculate the following:

$$\beta^{\frac{p-1}{d}} = \alpha^{\frac{k(p-1)}{d}} = \alpha^{X(p-1)} \equiv 1 \ (mod\ p)$$

This proves that the order of β is less than p – 1. Thus, β is not a primitive root of p.

Thus, we've proved that given one primitive root, α, all other primitive roots must be of the form α^k, where gcd(k, p – 1) = 1. By definition of the phi function, there are precisely $\phi(p-1)$ of these.

8) (10 pts) In class, we made a chart, for p = 7, of the different lengths of cycles produced by exponentiating each of the possible non-zero mod values, mod 7. We found that two of the values (3, 5) have a cycle length of 6, two of the values (2, 4) have a cycle length of 3, 1 value (6) has a cycle length of 2, and 1 value (1) has a cycle length of 1. Based on this example, give a counting/logical argument proving the sum below, for prime numbers, p:

$$\sum_{d \in Divisor(p-1)} \phi(\frac{p-1}{d}) = p - 1$$

Solution

The cycle length of each possible base, by Fermat's theorem, **MUST BE** a divisor of p – 1. Thus, if we were to make a frequency chart of how many bases have each cycle length (as described in the problem statement), the sum of those frequencies must necessarily equal p – 1, since there are p – 1 bases, 1 through p – 1, to consider.

Thus, what remains to be proven is that for any divisor, d, of p – 1, the number of elements with order $\frac{p-1}{d}$ is exactly $\phi(\frac{p-1}{d})$. If we can prove this, then symbolically, the sum on the left will represent the number of elements/bases of each different possible cycle length, and since each element must appear exactly once in the sum, it would then follow that the sum equals p – 1.

In question 7, we proved the fact specifically for the divisor d = 1.

Now, let's generalize that proof for any divisor d.
Consider a primitive root, α, we can generate each possible base mod p by exponentiating it to each power from 1 to p – 1. Every one of these values can be represented as $\beta = \alpha^k$, for some integer k in the range 1 to p – 1. Let d = gcd(k, p – 1).

As previously stated in the proof for #7, $\beta^{\frac{p-1}{d}} = \alpha^{\frac{k(p-1)}{d}} = \alpha^{X(p-1)} \equiv 1 \ (mod\ p)$.

From this statement, it follows that $\beta^{\frac{p-1}{d}(i)} \equiv 1 \ (mod\ p)$, for each integer in between 1 and d, inclusive. Thus, each of these values (there are d of them), have order, $\frac{p-1}{d}$, because the list of values, $\beta^1, \beta^2, \dots, \beta^{\frac{p-1}{d}}$ are all unique mod p, with the last value equivalent to 1 mod p. (To fully prove this, we can do another proof by contradiction rewriting each of these in terms of α, and proving that the list of exponents is unique mod (p – 1).

This proof works for each divisor, d, of p – 1. It follows that there are $\phi(\frac{p-1}{d})$ bases with an order of $\frac{p-1}{d}$. As previously discussed, this means that the sum on the left adds exactly 1 for each unique base modulo p. Thus, the sum must equal exactly p – 1.

9) (40 pts) Write a program that will take in as input a prime number, p ($2 \leq p < 10^9$) and will calculate the **sum of the cycle lengths for each possible base, 1 through p-1, inclusive for exponentiation mod p.** More formally, the input format for the program is as follows:

The first line contains a single integer, *n*, representing the number of input cases.
The input cases follow, one per line. Each of these lines has a single integer, p ($2 \leq p < 10^9$), representing the input for the case. It is guaranteed that p will be prime.

The output for each test case (on a line by itself) should simply be a single integer equal to the value described above. **(Note: This answer can be quite a bit larger than 10^9, so please use a long long in C/C++ or long in Java to store the result.)**

You may write your program in C, C++, Java or Python.

Sample Input	**Sample Output**
4	1
2	21
7	77
13	473
29	

Note: Brute force, where you manually run all cycles will earn ½ credit (20 pts out of 40).

An efficient solution requires a calculation of phi in $O(\sqrt{n})$ for calculating phi(n) and an overall run time of the summation code should be $O(\tau(n)\sqrt{n})$, where $\tau(n)$ equals the number of divisors of n.

There are two attached programs: cyclesumbf.cpp and cyclesum.cpp. The former simply tries exponentiating each base until the value of 1 is reached and adds up the requested values. This is the intended solution for half credit.

The latter utilizes the fact that the desired value is simply $\sum_{d \in Divisor(p-1)} \phi(\frac{p-1}{d}) \times \frac{p-1}{d}$. This directly follows from the result in question 8, where we proved that the number of bases with cycle length $\frac{p-1}{d}$ is $\phi(\frac{p-1}{d})$. Basically, if we know for each possible cycle length, how many times it occurs, we can just multiply each cycle length by its frequency.

Our solution can simply search for each divisor of p - 1 upto the square root of the input value and process divisors in pairs. (For example, if p = 101, and d = 2, process both 100/2 = 50 and 100/50 = 2 in the same loop iteration.)