IT FACULTY

Môn học

Toán Ứng Dụng
Applied mathematics

Giảng Viên: Phạm Minh Tuấn

Giới thiệu

- Phạm Minh Tuấn
- E-mail: pmtuan@dut.udn.vn
- Tel: 0913230910
- Khoa Công nghệ thông tin –Trường ĐHBK – ĐHĐN

Contents

- Number Theory (Lý thuyết số)
  - Basic Number Theory
  - Primality Tests
  - Totient Function
- Combinatorics (Tổ hợp)
  - Basics of Combinatorics
  - Inclusion-Exclusion
- Geometry (Hình học)
- Game Theory (Lý thuyết trò chơi)

Bài 4:

Totient Function

Contents

- Totient Function
  - Introduction
  - Easy Problem
  - What is Euler's Totient Function
  - Algorithm
  - Implementation
  - TEST YOUR UNDERSTANDING

Totient Function

Introduction

- In number theory, Euler's totient function counts the positive integers up to a given integer *n* that are relatively prime to *n*.
- It is written using the Greek letter phi as φ(n) or ϕ(n), and may also be called Euler's phi function
- It can be defined more formally as the number of integers *k* in the range $1 \le k \le n$ for which the greatest common divisor gcd(*n*, *k*) is equal to 1

Easy Problem

- Geeko is in worry now because exam is coming up and he has to know what rank he can get in exams. So he go back into the school records and finds the amazing pattern.
  He finds that if a student is having a current rank **n** than his rank in the final exam will be the count positive numbers between in the range [1,n] which are relatively prime to n
  As being geek he became curious now he want to calculate the rank of all his classmates in final exam, but he finds this task a bit hard , So he ask you programmers to solve this task for him.
- **Input**:
  - The first line of each test file contains a integer t denoting the number of test case. Each test case contains a numbers n representing the current rank of each student
- **Output**:
  - for each test case output single integer the rank of student in new line.
- **Constraints**:
  - $1 <= t <= 2000$, $1 <= n < 10^6$

What is Euler's Totient Function?

- Number theory is one of the most important topics in the field of Math and can be used to solve a variety of problems.
- Many times one might have come across problems that relate to the prime factorization of a number, to the divisors of a number, to the multiples of a number and so on.

What is Euler's Totient Function?

- Euler's Totient function is a function that is related to getting the number of numbers that are coprime to a certain number X that are less than or equal to it. In short , for a certain number X we need to find the count of all numbers Y where $gcd(X,Y) = 1$ and $1 \le Y \le X$.

What is Euler's Totient Function?

- A naive method to do so would be to **Brute-Force** the answer by checking the gcd of Xand every number less than or equal to X and then incrementing the count whenever a GCD of 1 is obtained. However, this can be done in a much faster way using Euler's Totient Function.

What is Euler's Totient Function?

- According to Euler's product formula, the value of the Totient function is below the product over all prime factors of a number. This formula simply states that the value of the Totient function is the product after multiplying the number N by the product of (1−(1/*p*)) for each prime factor *p* of N.

$$\varphi(n) = n \prod_{p \mid n} \left(1 - \frac{1}{p}\right)$$

Why?

The function is multiplicative

- This means that if gcd(m, n) = 1, then $\varphi(mn) = \varphi(m)\,\varphi(n)$.
- Outline of proof:
  - let A, B, C be the sets of nonnegative integers, which are, respectively, coprime to and less than m, n, and mn; then there is a bijection between A × B and C, by the ***Chinese remainder theorem***.

Value for a prime power argument

- If p is prime and $k \geq 1$, then

$$\varphi(p^k) = p^{k-1}(p-1) = p^k\left(1 - \frac{1}{p}\right)$$

- *Proof*:
  - since p is a prime number the only possible values of gcd(p^k, m) are 1, p, p^2, ..., p^k, and the only way for gcd(p^k, m) to not equal 1 is for m to be a multiple of p. The multiples of p that are less than or equal to p^k are p, $2p$, $3p$, ..., $p^{k-1}p = p^k$, and there are p^{k-1} of them. Therefore, the other $p^k - p^{k-1}$ numbers are all relatively prime to p^k.

Proof of Euler's product formula

- The fundamental theorem of arithmetic states that if n > 1 there is a unique expression for n,

$$\boxed{n = {p_1}^{k_1} \cdots {p_r}^{k_r}}$$

- where p1 < p2 < ... < p_r are prime numbers and each $k_i \geq 1$. (The case n = 1 corresponds to the empty product.)
- Repeatedly using the multiplicative property of φ and the formula for $\varphi(p^k)$ gives

Proof of Euler's product formula

- Euler's product formula

$$\begin{aligned}
\varphi(n) &= \varphi\left(p_1^{k_1}\right)\varphi\left(p_2^{k_2}\right)\cdots\varphi\left(p_r^{k_r}\right) \\
&= p_1^{k_1}\left(1-\frac{1}{p_1}\right)p_2^{k_2}\left(1-\frac{1}{p_2}\right)\cdots p_r^{k_r}\left(1-\frac{1}{p_r}\right) \\
&= p_1^{k_1}p_2^{k_2}\cdots p_r^{k_r}\left(1-\frac{1}{p_1}\right)\left(1-\frac{1}{p_2}\right)\cdots\left(1-\frac{1}{p_r}\right) \\
&= n\left(1-\frac{1}{p_1}\right)\left(1-\frac{1}{p_2}\right)\cdots\left(1-\frac{1}{p_r}\right).
\end{aligned}$$

Example

- $\varphi(36) = \varphi(2^2 3^2) = 36(1-1/2)(1-1/3) = 12$
- $\varphi(5) = 5(1-1/5) = 4$
- $\varphi(9) = \varphi(3^2) = 9(1-1/3) = 6$

Problem

- Caculate follow formula:
  - 1. $\varphi(360) = ??$
  - 2. $\varphi(107) = ??$
  - 3. $\varphi(250) = ??$
  - 4. $\varphi(127) = ??$

How to solve the problem?

- Geeko is in worry now because exam is coming up and he has to know what rank he can get in exams. So he go back into the school records and finds the amazing pattern.
  He finds that if a student is having a current rank **n** than his rank in the final exam will be the count positive numbers between in the range [1,n] which are relatively prime to n
  As being geek he became curious now he want to calculate the rank of all his classmates in final exam, but he finds this task a bit hard , So he ask you programmers to solve this task for him.
- **Input**:
  - The first line of each test file contains a integer t denoting the number of test case. Each test case contains a numbers n representing the current rank of each student
- **Output**:
  - for each test case output single integer the rank of student in new line.
- **Constraints**:
  - 1<= t <= 2000, 1<= n <10^6

Solution

- Factorize n in $O(\sqrt{n})$ time
- Euler's Totient Function to calculate output
- The approximate complexity is $O(t\sqrt{n})$

Solution

```
#include <iostream>
using namespace std;
int main() {
        int t; cin >> t;
        while (t-->0){
            long n;
            cin >> n;
            long ans = n;
            for (int i = 2; i * i <= n ; i++){
                if (n%i==0) {
                    ans = ans * (i - 1)/i;
                    while (n%i==0) n/=i;
                }
            }
            if (n!=1) ans = ans * (n - 1) / n;
            cout << ans << endl;
        }
}
```

Problems