1. $\phi(8500)$을 구하시오.

8500을 소인수분해 하면 $2^2 \times 5^3 \times 17$이다.

오일러 피 함수는 multiplicative function 임을 이용하면 $\phi(8500) = \phi\left(2^2\right)\phi\left(5^3\right)\phi(17)$

소수 p 에 대하여, $\phi\left(p^k\right) = p^{k-1}(p-1)$ 임을 이용하면 $\phi\left(2^2\right)\phi\left(5^3\right)\phi(17) = 2 \times 100 \times 16 = 3200.$

2. $\phi(n) = 14$ 를 만족하는 자연수 n은 존재하지 않음을 보이시오.

$\phi(n) = 14$ 를 만족하는 자연수 n이 존재한다고 가정하자.

만약 소수 p가 n을 나눈다면, $p-1$이 $\phi(n)$을 나눠야 한다.
$\Longleftrightarrow$ 만약 소수 p에 대하여 $p-1$이 $\phi(n)$을 나누지 않는다면, p가 n을 나누지 않는다.

$p-1$의 후보는 $\phi(n) = 14$의 약수들인 $\{1,\ 2,\ 7,\ 14\}$이므로, p의 후보는 $\{2,\ 3,\ 8,\ 15\}$이다.

이중 소수는 $2,\ 3$ 뿐이므로 $n = 2^a 3^b$ 꼴로 나타낼 수 있다. (단, $a,\ b \geq 0$)

만약 $a = 0,\ b = 0$ 이라면, $n = 1$ 인데, $\phi(1) = 1$ 이므로 모순이 발생한다.
만약 $a = 0,\ b = 1$ 이라면, $n = 3$ 인데, $\phi(3) = 2$ 이므로 모순이 발생한다.
만약 $a = 1,\ b = 0$ 이라면, $n = 2$ 인데, $\phi(2) = 1$ 이므로 모순이 발생한다.
만약 $a = 1,\ b = 1$ 이라면, $n = 6$ 인데, $\phi(6) = 4$ 이므로 모순이 발생한다.
만약 $a \geq 2,\ b \geq 2$ 이라면, $n = 2^a 3^b$ 이고, $\phi\left(2^a 3^b\right) = 2^a \times 3^{b-1} = 14$ 인데, 3은 $\phi\left(2^a 3^b\right)$를 나누지만 14는 나누지 않으므로 조건을 만족하는 n이 존재하지 않아 모순이 발생한다.

모든 경우에 모순이 발생하므로, 가정이 잘못되었음을 알 수 있다.

따라서 $\phi(n) = 14$ 를 만족하는 자연수 n은 존재하지 않는다.

3. n이 홀수일 때, 5가 n을 나누지 않으면 5가 $n^4 + 4^n$을 나눔을 보이시오.

모든 자연수 k에 대하여, $4^{2k-1} \equiv 4 \pmod 5$가 성립한다. $\Longrightarrow$ 모든 홀수 n에 대하여 $4^n \equiv 4 \pmod 5$가 성립한다.

> $k = 1$일 때, $4^1 \equiv 4 \pmod 5$.
>
> $k = i$일 때 $4^{2i-1} \equiv 4 \pmod 5$가 성립한다고 가정하면,
> $k = i+1$일 때 $4^{2i+1} = 16 \times 4^{2i-1} = 4 \pmod 5$.
>
> 수학적 귀납법에 의해 모든 자연수 k에 대하여, $4^{2k-1} \equiv 4 \pmod 5$.

페르마 소정리에 의해, 5가 n을 나누지 않으면, $n^4 \equiv 1 \pmod 5$.

따라서 n이 홀수이고 5가 n을 나누지 않으면 $n^4 + 4^n \equiv 0 \pmod 5$,즉 5가 $n^4 + 4^n$을 나눈다.

4. $p \equiv 5 \pmod 6$을 만족하는 소수 p가 무한히 많다는 것을 보이시오.

5. 밀러-라빈 판정법을 사용하여 다음 자연수 $n = 118901521$이 합성수임을 보이시오.

```cpp
/**
 *    author:  pizzaroot
 *    created: 2023-10-05 20:15:47
**/
#include <bits/stdc++.h>
#define inf 0x3f3f3f3f
#define linf 0x3f3f3f3f3f3f3f3f
#define all(x) (x).begin(), (x).end()
#define rall(x) (x).rbegin(), (x).rend()
#define pb push_back
using namespace std;
typedef long long ll;
typedef vector<ll> vi;
typedef pair<ll, ll> pi;
ll mul(ll x, ll y, ll mod) {
    return (__int128)x * y % mod;
}
ll fexp(ll x, ll y, ll p) {
  ll ret = 1, piv = x % p;
  while (y) {
    if (y & 1) ret = mul(ret, piv, p);
    piv = mul(piv, piv, p);
    y >>= 1;
  }
  return ret;
}
bool miller_rabin(ll x, ll a) {
  if (x % a == 0) return true;
  ll d = x - 1;
  while (true) {
    ll tmp = fexp(a, d, x);
    if (d & 1) return (tmp != 1 && tmp != x - 1);
    else if (tmp == x - 1) return false;
    d >>= 1;
  }
}
bool isprime(ll n) {
  if (n < 2 || (n % 2 == 0)) return (n == 2);
  vector<ll> seeds;
  if (n < (1 << 30)) seeds = {2, 7, 61};
  else seeds = {2, 325, 9375, 28178, 450775, 9780504};
  for (auto &i : seeds) {
    if (n == i) return true;
    if (miller_rabin(n, i)) return false;
  }
  return true;
}
int main() {
    ios::sync_with_stdio(0); cin.tie(0);
    ll n; cin >> n;
    cout << (isprime(n) ? "prime₩n" : "not prime₩n");
    return 0;
}
```

실행 결과:

```
118901521
not prime
```