# Miller

1811

#include <iostream>

#include <cstring>

#include <algorithm>

#include <cmath>

#include <map>

#include <cstdlib>

#include <cstdio>

using namespace std;

#define Times 10

map<long long, int>m;

long long mi;

long long random(long long n){

return ((double)rand() / RAND\_MAX\*n + 0.5);

}

long long multi(long long a, long long b, long long mod){

long long ans = 0;

while (b){

if (b & 1) ans = (ans + a) % mod;

b >>= 1;

a = (a << 1) % mod;

}

return ans;

}

long long Pow(long long a, long long b, long long mod){

long long ans = 1;

while (b){

if (b & 1) ans = multi(ans, a, mod);

b >>= 1;

a = multi(a, a, mod);

}

return ans;

}

bool witness(long long a, long long n){

long long d = n - 1;

while (!(d & 1))

d >>= 1;

long long t = Pow(a, d, n);

while (d != n - 1 && t != 1 && t != n - 1){

t = multi(t, t, n);

d <<= 1;

}

return t==n-1||d&1;

}

long long P[5]={2, 3, 7, 61,24251};

bool miller\_rabin(long long n){

if (n == 2)

return true;

if (n<2 || !(n & 1))

return false;

for (int i = 1; i <= Times; i++){//随机测试

long long a = random(n - 2) + 1;

if (!witness(a, n))

return false;

}

//return 1;若超时可以去掉加强测试或者调整Times

for (int i = 1; i <= 5; i++){//加强测试

if(n<=P[i-1]) break;

long long a=P[i-1];

if (!witness(a, n))

return false;

}

return true;

}

long long gcd(long long a, long long b){

return a&&b ? gcd(b, a%b) : a + b;

}

long long pollard\_rho(long long n, int c){

long long x,y,d,i=1,k=2;

x=random(n - 2)+1;

y=x;

while (1){

i++;

x = (multi(x, x, n) + c) % n;

d = gcd(y - x, n);

if (1<d&&d<n)return d;

if (y == x)return n;

if (i == k){

y = x;

k <<= 1;

}

}

return 0;

}

int find(long long n, int c){//c是个很迷的东西

if (n == 1)return 0;

if (miller\_rabin(n)){

//printf("%lld\n",n);//此处可以输出所有素因子

m[n]++;

mi = min(mi, n);//本题求的是最小素因子

return 0;

}

long long p = n;

while (p >= n)

p = pollard\_rho(p, c--);

find(p, c);

find(n / p, c);

return 0;

}

int main(){

//freopen("A.in","r",stdin);

//freopen("A.out","w",stdout);

int t;

scanf("%d", &t);

while (t--){

long long n;

scanf("%lld", &n);

mi = n;

if (miller\_rabin(n))

cout << "Prime" << endl;

else{

find(n, 12312);

cout << mi << endl;

}

}

return 0;

}