```cpp
// --------------以下ライブラリ-------------------------------//

#include <iostream>
#include <utility>
#include <map>
#include <string>
#include <vector>
#include <algorithm>
#include <functional>
#include <array>
#include <math.h>
#include <numeric>
#include <sstream>
typedef long long ll;
using namespace std;

vector<int> primes;
map<int, int> factors;

void erathosthenes(ll n) {
    vector<bool> primeFlags(n+10, true);
    primeFlags[0] = primeFlags[1] = false;
    vector<int> sqrtprimes;
    for (int i = 2; i*i <= n; i++) {
        bool mod0 = false;
        for (int j = 2; j*j <= i; j++) {
            if (i%j == 0) mod0 = true;
        }
        if (!mod0) sqrtprimes.push_back(i);
    }
    for (auto &&e: sqrtprimes) {
        for (int i = 2; i <= n; i++) {
            if (i%e == 0) primeFlags[i] = false;
        }
    }
    for (auto &&e: sqrtprimes) primeFlags[e] = true;
    for (int i = 1; i <= n; i++) if (primeFlags[i]) primes.push_back(i);
    return;
}

void factoring(ll n) {
    ll m = sqrt(n);
    erathosthenes(m);
    for (auto &&e: primes) {
        while(n%e == 0){
            factors[e]++;
            n /= e;
        }
    }
    factors[n]++;
    factors.erase(1);
}



//------------------以下debug用main関数----------------------//


int main(int argc, char const *argv[]) {
    int n;
```

```
    cin >> n;
    factoring(n);
    cout << "Primes =" << " ";
    for (auto x : primes) cout << x << " ";
    cout << "" << std::endl;
    string factorString = "factoring = ";
    for (auto &&x : factors) factorString +=  to_string(x.first) + "^" + to_string(x.second)
+ " * ";
    for (int i = 0; i < 3; i++) factorString.pop_back();
    cout << factorString << std::endl;
    return 0;
}
```