

Contents

1	Math	1
1.1	FindPrime	1

1 Math

1.1 FindPrime

```

1 #include <bits/stdc++.h>
2 using namespace std;
3
4 //查找[0,2^15]中的所有質數 共有3515
5
6 const int MAXN = 32768; //2^15=32768
7 bool primes[MAXN];
8 vector<int> p; //3515
9
10 //質數篩法 Sieve of Eratosthenes
11 inline void findPrimes() {
12     for (int i = 0; i < MAXN; i++) {
13         primes[i] = true;
14     }
15     primes[0] = false;
16     primes[1] = false;
17     for (int i = 4; i < MAXN; i += 2) {
18         //將2的倍數全部刪掉(偶數不會是質數)
19         primes[i] = false;
20     }
21     //開始逐個檢查--->小心i*i會有overflow問題--->使用long
22     long
23     for (long long i = 3; i < MAXN; i += 2) {
24         if (primes[i]) {
25
26             //如果之前還未被刪掉 才做篩法
27             for (long long j = i * i; j < MAXN; j += i) {
28                 //從i*i開始(因為i*2,i*3...都被前面處理完了)
29                 primes[j] = false;
30             }
31         }
32     }
33     //蒐集所有質數
34     for (int i = 0; i < MAXN; i++) {
35         if (primes[i]) {
36             p.emplace_back(i);
37         }
38     }
39 }

```