

Решето Эратосфена

C++. Базовые алгоритмы теории чисел. Решето Эратосфена



Пусть нам необходимо проверить на простоту все числа от 1 до n .

Очевидно, мы можем проверить каждое число по отдельности и получить алгоритм с вычислительной сложностью $O(n\sqrt{n})$.

Однако, есть более быстрый алгоритм — решето Эратосфена. Идея проста — запишем ряд чисел $1 \dots n$ и будем вычёркивать сначала все числа, делящиеся на 2, кроме самого числа 2. Перейдём к следующему числу 3, оно не вычеркнуто, значит, является простым. Вычеркнем все числа, делящиеся на 3. Четыре окажется вычеркнутым, значит, следующее простое число — 5 и т.д. В результате невычеркнутыми останутся простые числа в диапазоне от 2 до n .

При этом в алгоритме можно сделать улучшение — вычёркивать все числа, делящиеся на простое i не от $2i$, а от i^2 , так как все числа от $2i$ до $(i-1)i$ заведомо имеют меньший простой делитель и уже были вычеркнуты.

Вычислительная сложность данного алгоритма — $O(n \log \log n)$.

Реализация

```
vector<int> prime(n + 1, 1);  
prime[0] = prime[1] = 0;
```

```
for (int i = 2; i <= n; ++i) {  
    if (!prime[i] || i * 111 * i > n)  
        continue;  
    for (int j = i * i; j <= n; j += i)  
        prime[j] = 0;  
}
```