Первый курс, весенний семестр
Практика по алгоритмам #15:
Теория Чисел

Contents

Новые задачи

1. **Числа**
   a) За сколько можно посчитать $\varphi(m)$?
   b) Посчитайте $a_1^{a_2^{a_3^{\cdot^{\cdot^{\cdot^{a_n}}}}}} \mod m$.
   c) Зная, что простых чисел $p \leqslant n$ всего $\Theta(\frac{n}{\log n})$, оцените количество чисел вида $p^k \leqslant n$.

2. **Решето Эратосфена**
   a) Докажите оценку времени работы $\mathcal{O}(n \log\log n)$.
   b) Используя решето Эратосфена, найдите все простые числа от n^2 до n^2+n за $\mathcal{O}(n \log\log n)$.
   c) Найдите все простые от 1 до n за $\mathcal{O}(n \log\log n)$, используя $\sqrt{n}$ памяти.

3. **Применяем решето**
   Для каждого числа от 1 до n найти
   a) Количество различных простых делителей
   b) Количество делителей
   c) Сумму делителей
   d) Функцию Мёбиуса

4. **RSA**
   RSA – криптосистема с открытым ключом. n, d, e: $de \equiv 1 (\mod \varphi(n))$.
   (n, e) – открытый ключ. (n, d) – закрытый ключ.
   Кодирование: $m \to m^e (\mod n)$.
   Декодирование: $m^e \equiv y \to y^d \equiv m (\mod n)$.
   a) Пусть n простое, взломайте RSA.
   b) Пусть известно $\varphi(n)$, взломайте RSA.
   c) Пусть $n = pq$, известно $\varphi(n)$, разложите n на множители.
   d) Пусть у нас есть "волшебный" оракул. Для любого открытого ключа (n, e) оракул может взломать 1% из всех возможных зашифрованных сообщений. Придумайте алгоритм, который взламывает любое сообщение. Матожидание времени работы $\mathcal{O}(poly(\log n))$.

5. **Расширенный Евклид**
   a) Докажите, что в строке $ax_i + by_i = r_i$: $(x_i, y_i) = 1$
   b) Докажите, что $\max |x_i| \leqslant |b|$ и $\max |y_i| \leqslant |a|$
   c) Найдите класс решений диофантового уравнения $ax \equiv b (\mod m)$
   d) Найдите x, y: $ax + by = c, |x| + |y| \to \min$

6. (*) **Магия**
   Поймите, что делает код:

```
f[1] = 1;
for (int i = 2; i < p; i++)
  f[i] = (p - f[p % i]) * (p / i) % p;
```

Домашнее задание

Обязательная часть

1. **(3) Подсчёт p^α в лоб.**
   Пусть $p[x]$ – минимальный простой делитель x, мы его уже насчитали за $\mathcal{O}(n)$, запустив решето Эратосфена. $\alpha[x]$ – степень вхождения $p[x]$ в x. Докажите, что код 'for (x=2; x<=n; x++) alpha[x] = 0; for (y = x; p[y] == p[x]; y /= p[x]) alpha[x]++; ' так же работает за $\mathcal{O}(n)$.
2. **(3) Взлом RSA при малой $|p-q|$.**
   Пусть известно, что $n = pq, |p-q| \leqslant 10^6$. Взломайте `RSA`-ключ (n, e).
3. **(3) Взлом RSA при малом d.**
   Пусть известно, что $d \leqslant 10^6$, n произвольно. Взломайте `RSA`-ключ (n, e).
4. **(3) $\varphi(n)$ на отрезке.**
   Для каждого $x \in [l..r]$ посчитать $\varphi(x)$ за $\mathcal{O}(r)$.
5. **(3) $\varphi(n)$ на отрезке.**
   Для каждого $x \in [1..n]$ построить `vector<int> divisors[x]`.
   Оценить время работы решения, доказать оптимальность.
   Существует простое решение в три строчки кода.

Дополнительная часть

1. **(5) $\pi(n)$ для больших n.**
   $\Phi(n, d)$ – количество чисел от 1 до n, все простые делители которых больше d. $\pi(n)$ – количество простых от 1 до n. С помощью решета Эратосфена мы умеем считать $\pi(n)$ за $\mathcal{O}(n)$. Можно быстрее. Эта задача даёт вам подсказку и предлагает придумать алгоритм. $\pi(n) = \Phi(n, \sqrt{n})$. p_i – i-е простое. $\Phi(n, p_{i+1}) = \Phi(n, p_i) - \#\{\text{числа кратные} p_i\} = \Phi(n, p_i) - \Phi(\frac{n}{p_i}, p_i)$. Это реккурентная формула пересчёта. Осталось добавить правильную базу и доказательство. Баллы будут ставиться за любое решение за $\mathcal{O}(n^{1-\varepsilon})$. Существует решение за $\mathcal{O}(n^{2/3})$.
2. **(5) $\binom{n}{k} \bmod m$.**
   Рассмотрим алгоритм подсчёта $\binom{n}{k} \bmod m$ за $\mathcal{O}(n \log m) + \texttt{FACT}(m)$. Разложим $m = \prod_i p_i^{\alpha_i}$. $\binom{n}{k} = \frac{n!}{k!(n-k)!} \equiv \frac{f_p(n)}{f_p(k) f_p(n-k)} p^{cnt_p(n) - cnt_p(k) - cnt_p(n-k)} \bmod p^\alpha$. Далее используем КТО.
   Слабое место этого алгоритма – факторизация. Придумайте аналог за $\mathcal{O}(poly(n, \log m))$.