第十周直播课（下午）
综合模拟/作业讲评

洛谷网校
基础-提高衔接计划
2024-09
disangan233

www.luogu.com.cn

T1 书书（total）

- 给定 2 个正整数 n, k，m 个正整数 2 元组 (a_i, b_i)
- 请你求出对于任意 $i = 1, 2, ..., m$ 都满足

$$|a_i - x| + |b_i - y| \geq k$$

  的有序正整数 2 元组 $(x, y)(x, y \leq n)$ 的个数
- $n, m, k \leq 500$

T1 树树（total）

- 使用两重循环枚举 x, y，因为是有序的，所以都是 $1 \sim n$
- 使用 `abs()` 计算绝对值，判断答案
- 对于任意的 (a_i, b_i)，若 $|a_i - x| + |b_i - y| < k$，则该 (x, y) 不满足条件
- 可以用 `op=0` 来实现

T1 树树（total）

```
scanf("%d%d%d",&n,&m,&k)
for(int i=1;i<=m;i++)
    scanf("%d%d",&a[i],&b[i])
for(int x=1;x<=n;x++)
for(int y=1;y<=n;y++) {
    int op=1;
    for(int i=1;i<=m;i++)
        if(abs(x-a[i])+abs(y-b[i])<k)
            op=0;
    ans+=op;
}
printf("%d",ans);
```

T2 书书（history）

- 需要完成 n 件事，第 i 件事需要 a_i 小时 b_i 分 c_i 秒
- 可以同时完成 2 件事情，但是必须全部完成后再开始下一次行动
- 求完成所有事情最少需要多少小时多少分多少秒
- $n \leq 100$

T2 书书（history）

- 假设从小到大的耗时是 $1,3,5,7$
- $\{1,5\},\{3,7\}$ 的耗时是 $5+7=12$
- $\{1,3\},\{5,7\}$ 的耗时是 $3+7=10$
- 对于两组 $\{a,b\},\{c,d\}$，如果有 $a\leq c\leq b\leq d$，那么 $\{a,c\},\{b,d\}$ 更优
- 所以最优解中耗时小的应该和耗时小的一组
- 这是贪心的一种严谨证明方法：分析最优解

T2 书书（history）

- 在从小到大排序后，将 a_1, a_2 一起做，a_3, a_4 一起做，$\cdots$
- 答案就是 $a_2 + a_4 + \cdots + a_n$
- 如果 n 是奇数呢？
- 可以发现是 $a_1 + a_3 + a_5 + \cdots + a_n$
- 需要模拟一下时: 分: 秒的进位加法

T2 书书（history）

```cpp
bool cmp(int x,int y) {
    return a[x]*3600+b[x]*60+c[x]<a[y]*3600+b[y]*60+c[y];
}
cin>>n;
for(int i=1;i<=n;i++)
    q[i]=i,cin>>a[i]>>b[i]>>c[i];
sort(q+1,q+n+1,cmp);
if(n&1)
    a[0]=a[q[1]],b[0]=b[q[1]],c[0]=c[q[1]];
for(int i=n;i>1;i-=2)
    a[0]+=a[q[i]],b[0]+=b[q[i]],c[0]+=c[q[i]];
if(c[0]>=60)
    b[0]+=c[0]/60,c[0]%=60;
if(b[0]>=60)
    a[0]+=b[0]/60,b[0]%=60;
cout<<a[0]<<" "<<b[0]<<" "<<c[0]<<endl;
```

T3 鼠鼠（math）

- 给定 n 个数 a_n 和 T 组询问，第 i 次询问给定 p_i，询问从第 p_i+1 个数往后第一个不是前 p_i 个数的倍数的数是多少
- $n, T \le 2 \times 10^5$，$a_i \le 10^7$

T3 鼠鼠（math）

- 考虑预处理出所有答案，对于询问 $O(1)$ 回答
- 预处理发现，对于两个询问 $a, b(a < b)$，b 询问的答案位置一定在 a 的答案位置的右侧
- 若 a 的答案是 p_a，如果 p_a 是 $a+1$ 的倍数，那么 p_a 就不能继续作为 $a+1$ 的答案了，需要向右找新的答案
- 那么我们就可以按顺序处理 $1, 2, \ldots, n$ 的询问

T3 鼠鼠（math）

- 由于 $a_i \leq 10^7$，直接从左往右扫描，每次标记倍数的那些数，然后移动答案标记即可
- 2 会标记 $2, 4, 6, \cdots$，3 会标记 $3, 6, 9, \cdots$，i 会标记 $\frac{a}{i}$ 次
- 我们学过了 $n + \frac{n}{2} + \frac{n}{3} + \cdots + \frac{n}{n} < O(n \log n)$
- 那么这样的复杂度是 $O(n \log a)$ 吗？

T3 鼠鼠（math）

- 如果 a 有很多个 2，每一个 2 都会标记 5×10^6 次
- 启发我们一个数最多只能标记一次，重复标记是没有意义的，因为它的所有倍数已经不合法了
- 需要通过每次标记倍数前查看当前数字有无被标记，这样的时间复杂度是 $O(a \log a)$ 的
- 如果数据是 $n, n-1, n-2, \cdots$ 就会卡满复杂度，但是随机数据跑不满

T3 鼠鼠（math）

```
for(int i=1;i<=n;i++)
    mx=max(mx,a[i]);
for(int i=1,r=1;i<=n;i++) {
    if(!vis[a[i]])
        for(int t=a[i];t<=mx;t+=a[i])
            vis[t]=1;
    while(r<=n&&vis[a[r]])
        r++;
    ans[i]=r;
}
for(int p;t--;) {
    scanf("%d",&p);
    printf("%d ",ans[p]==n+1?-1:a[ans[p]]);
}
```

T4 数数（count）

- 给定 n 个数 $\{a_n\}$，满足 $a_1 \text{ and } a_2 \text{ and} \dots \text{and } a_n = 0$，且 $a_1 + a_2 + \dots + a_n = m$
- 求 $\{a_n\}$ 的个数，答案对 998244353 取模
- $n, m \le 2000$

T4 数数（count）

- a_1 and a_2 and ... and $a_n = 0$ 即每个二进制位都不能全是 1
- 那么等价于有 n 个数选 $k(0 \leq k < n)$ 个数的组合数问题
- 但是 $a_1 + a_2 + \ldots + a_n = m$ 的限制如何满足？
- 体积为 $2^0, 2^1, 2^2, \cdots$ 的物品各有 $n - 1$ 个
- 本质上是求背包的方案数

T4 数数（count）

- 考虑按位 dp。令 $f_{i,j}$ 为前 i 位和为 j 的方案数，可得

$$f_{i,j} = \sum_{k=0}^{n-1} f_{i-1,j-k\times 2^i} \binom{n}{k}$$

- 对于 $i < n$，初态为 $f_{0,i} = \binom{n}{i}$
- 预处理组合数，枚举 i, j, k 时间复杂度为 $O\left(m^2 \log m\right)$

T4 数数（count）

```
for(int i=0;i<=n;i++)
    c[i][0]=1;
for(int i=1;i<=n;i++)
    for(int j=1;j<=i;j++)
        c[i][j]=(c[i-1][j]+c[i-1][j-1])%p;
for(int i=0;i<n;i++)
    f[0][i]=c[n][i];
for(int i=1;(1<<i)<=m;i++)
for(int j=0;j<=m;j++)
for(int k=0;k<n&&(j>=k*(1<<i));k++)
f[i][j]=(f[i][j]+1ll*c[n][k]*f[i-1][j-k*(1<<i)])%p;
cout<<f[(int)log2(m)][m]<<'\n';
```

【第十周】小学数学题

- T514263
- 输入一个整数 n，每行输出一个数，表示数字 1 到 i 的和
- $n \le 10^7$

【第十周】小学数学题

- T514263
- 输入一个整数 n，每行输出一个数，表示数字 1 到 i 的和
- $n \le 10^7$
- 直接使用 `for` 循环，`sum += i` 后输出 `sum`
- 也可以使用等差数列求和公式：$\frac{(n+1)\times n}{2}$
- 注意数据范围要 `long long`

【第十周】小学数学题

- 为什么我的代码 TLE 了？

```cpp
ios::sync_with_stdio(0);
cin >> n;
for (int i = 1; i <= n; i++) {
    sum += i;
    cout << sum << endl;
}
```

- std::endl 会在输出换行后调用 os.flush() 刷新缓冲区
- 关了同步流还是很慢，只能使用 cout << "\n";
- 或者老老实实用 printf

【第十周】小学数学题

```
ios::sync_with_stdio(0);
cin >> n;
for (int i = 1; i <= n; i++) {
    sum += i;
    cout << sum << "\n";
    // printf("%lld\n", sum);
}
```

【第十周】约数问题

- T514266
- 求第 k 小的 Aya 数，对 998244353 取模
- $k \le 10^{18}$

定义一个 Aya 数为一个可以经过如下操作变成 1 的正整数 x：

- 第 1 步：令 $x' = x$
- 第 2 步：如果 x' 为 1，则跳出算法步骤并确定 x 为 Aya 数，否则选择任意一个 x' 的约数（不能为 x'）本身的 d，且 d 在之前的操作中并未选择过；
- 第 3 步：令 x' 减去 d 得到新的 x'；
- 第 4 步：返回第 2 步。

【第十周】约数问题

- k 很大，一般是找规律
- 打个表就能发现 Aya 数是 $1, 2, 4, 8, 16, \cdots$
- 问题即求 $2^{k-1} \bmod 998244353$，写一个快速幂即可
- 怎么证？

【第十周】约数问题

- 一个数 n 减去其约数 d 后变为 $n-d$
- 若 $n=kd$，则 $n-d=(k-1)d$
- 从 1 开始倒推，$n-d=1$ 时，$d=\frac{1}{k-1}$
- 因为 d 是整数，$k=2$，$d=1$
- 所以 1 只能从 2 减去 1 得来

【第十周】约数问题

- $n-d=2$ 时，$d=\frac{2}{k-1}$
- 因为 d 是整数，$k=2,3$，$d=1,2$
- 但是 $d=1$ 会被 $2-1=1$ 使用，所以 $d=2$
- 2 只能从 4 减去 2 得来
- 以此类推：4 只能从 8 得来，8 只能从 16 得来，$\cdots$
- 第 k 个 Aya 数即 2^{k-1}

【第十周】约数问题

```
long long x;
cin >> x;
x--;
int ans = 1, c = 2;
while (x) {
    if (x & 1)
        ans = 1ll * ans * c % p;
    c = 1ll * c * c % p;
    x >>= 1;
}
cout << ans;
```

【第十周】公倍数问题

- T514267
- 给定一个正整数 n，请找到另外一个正整数 $m(m < n)$，要求 $\operatorname{lcm}(n, m)$ 最大
- 请输出这个 $\operatorname{lcm}(n, m)$
- $n \le 10^9$

【第十周】公倍数问题

- T514267
- 给定一个正整数 n，请找到另外一个正整数 $m(m < n)$，要求 $\operatorname{lcm}(n, m)$ 最大
- 请输出这个 $\operatorname{lcm}(n, m)$
- $n \leq 10^9$
- 如果 n, m 互质，$\operatorname{lcm}(n, m) = n \times m$
- $\gcd(n, n-1) = \gcd(n-1, 1) = 1$，所以 $m = n-1$ 时 $\operatorname{lcm}(n, m)$ 最大
- 题目所求为 $n \times (n-1)$

【第十周】公倍数问题

```
int main() {
    scanf("%d", &n);
    printf("%lld", (ll)n * (n - 1));
    return 0;
}
```

【第十周】质因数分解

- T514269
- T 组询问，每次给定正整数 n，求 n 的次小质因子和严格次小质因子。若不存在，输出 -1
- $T \leq 10^5, n \leq 10^8$

【第十周】质因数分解

- T514269
- T 组询问，每次给定正整数 n，求 n 的次小质因子和严格次小质因子。若不存在，输出 -1
- $T \le 10^5, n \le 10^8$
- $n \le 10^8$，采用线性筛记录最小质因子来快速质因数分解
- 可以记录所有质因子来求次小和严格次小
- 另一种方法是，x 的次小质因子为 mn[x/mn[x]]，严格次小质因子为 mn[y]，y 为 x 除尽所有最小质因子后的数
- 时间复杂度 $O(T \log n)$

【第十周】质因数分解

```
mn[1]=-1;
for(int i=2;i<N;i++) {
    if(!vis[i]) p[++k]=i,mn[i]=i;
    for(int j=1;j<=k&&i*p[j]<N;j++) {
        vis[i*p[j]]=1;
        mn[i*p[j]]=p[j];
        if(i%p[j]==0) break;
    }
}
scanf("%d",&T);
while(T--) {
    int n,m;
    scanf("%d",&n);
    m=n;while(m%mn[n]==0) m/=mn[n];
    printf("%d %d\n",mn[n/mn[n]],mn[m]);
}
```

【第十周】组合数学

- T514268
- q 次询问，每次给定 n, m，求下列式子的值

$$\sum_{i=0}^{n}\sum_{j=0}^{m}\sum_{k=0}^{j}\binom{i}{k} \bmod 998244353$$

- $q \le 10^6$，$n, m \le 2000$

【第十周】组合数学

$$\sum_{i=0}^{n}\sum_{j=0}^{m}\sum_{k=0}^{j}\binom{i}{k} = \sum_{j=0}^{m}\sum_{k=0}^{j}\sum_{i=0}^{n}\binom{i}{k} \quad \text{(交换独立的枚举顺序)}$$

$$= \sum_{j=0}^{m}\sum_{k=0}^{j}\binom{n+1}{k+1} \quad \text{(固定选的第 } k+1 \text{ 个)}$$

$$= \sum_{j=0}^{m}(m-j+1)\binom{n+1}{j+1}$$

$$= (m+1)\sum_{j=0}^{m}\binom{n+1}{j+1} - \sum_{j=0}^{m} j\binom{n+1}{j+1}$$

【第十周】组合数学

- 预处理 $\sum_{j=0}^{m}\binom{n+1}{j+1}$ 和 $\sum_{j=0}^{m} j\binom{n+1}{j+1}$，单次询问 $O(1)$

```
for(int i=0;i<N;i++) c[i][0]=1;
for(int i=1;i<N;i++) for(int j=1;j<=i;j++)
c[i][j]=(c[i-1][j]+c[i-1][j-1])%p;
for(int i=1;i<N-1;i++) {
    g[i][0]=i+1;
    for(int j=1;j<N-1;j++)
    f[i][j]=(f[i][j-1]+1ll*j*c[i+1][j+1])%p,
    g[i][j]=(g[i][j-1]+c[i+1][j+1])%p;
}
scanf("%d",&q);
while(q--) {
    scanf("%d%d",&n,&m);
    int A=1ll*g[n][min(n,m)]*(m+1)%p;
    printf("%d\n",(A+p-f[n][min(n,m)])%p);
}
```

[yLOI2019] 青原樱

- P5520
- 给定 n 个位置，要求放下 m 个互不相同的东西，东西两两之间不能相邻，求方案数对 p 取模的结果
- $n \leq 2 \times 10^6$，$m \leq 10^6$

[yLOI2019] 青原樱

- m 个东西需要 $m-1$ 个空
- 借走 $m-1$ 个空，相当于在 $n-m+1$ 个位置任意放 m 个不同的东西
- 然后再把空插回前 $m-1$ 个东西之后，两者是等价的
- 方案数即 $n-m+1$ 选 m 的排列

$$\mathrm{A}_{n-m+1}^{m}=\frac{(n-m+1)!}{(n-m+1-m)!}=(n-m+1)\times\cdots\times(n-2m+2)$$

- 循环计算取模

[yLOI2019] 青原樱

```
long long i, n, m, p, ans = 1;
scanf("%d %lld %lld %lld", &t, &n, &m, &p);
for (i = n + 1 - m; i > n + 1 - 2 * m; i--) {
    ans *= i;
    ans %= p;
}
printf("%lld", ans);
```

后缀树

- P6045
- 求有多少种不同的长为 n、只含小写字母的字符串 s 满足：
  - 不存在整数 $i \in [1, n)$ 使得 $s_{1\ldots i}$ 是 $s_{i+1\ldots n}$ 的子串
- 答案对 998244353 取模，$n \leq 10^9$

后缀树

- P6045
- 求有多少种不同的长为 n、只含小写字母的字符串 s 满足：
  - 不存在整数 $i \in [1, n)$ 使得 $s_{1\dots i}$ 是 $s_{i+1\dots n}$ 的子串
- 答案对 998244353 取模，$n \leq 10^9$
- 直接考虑最极端的情况，也就是最容易非法的情况
- 在第一位后面分，使第一个字母单独成子串。那么后面的每一个字母都有机会与它重复，成为非法方案
- 所以最终的答案就是，第一个字母任取，后面的字母不与第一个重复就可以，也就是从剩下的 25 个字母里去选
- 那么答案就是 $26 \times 25^{n-1}$

后缀树

```
cin >> n;
printf("%lld", 26ll * ksm(25,n - 1) % mod);
```

[SHOI2002] N 的连续数拆分

- P6267
- 给定一个正整数 n，求出它可以用几种不同的方法表示成连续正整数之和
- $n \leq 9 \times 10^{14}$

[SHOI2002] N 的连续数拆分

- 从 l 到 r 的连续正整数列求和公式为

$$n = \frac{(l+r)(r-l+1)}{2}$$

- 那么 $2n = (l+r)(r-l+1)$
- 接着我们只要对 $2n$ 分解质因数，当发现某个奇数和某个偶数的乘积为 $2n$ 时结果 $+1$
- $O(\sqrt{n})$ 枚举较小因数 $r-l+1$ 即可

[SHOI2002] N 的连续数拆分

```
scanf("%lld",&n);
n<<=1;
for(ll x=1;x*x<=n;++x) {
    if(n%x) continue;
    if((x&1)^(n/x&1)) ans++;
}
printf("%d\n",ans);
```

[yLOI2020] 金陵谣

- P7094
- 给定四个正整数 a, b, c, d，求有多少对正整数 (x, y) 满足

$$\frac{a}{x} + \frac{b}{c} = \frac{d}{y}$$

- $a, b, c, d \leq 10^6$，$c \times d \leq 10^6$

[yLOI2020] 金陵谣

- 对原式通分

$$\frac{a}{x}+\frac{b}{c}=\frac{d}{y} \implies acy+bxy=cdx \implies x=\frac{acy}{cd-by}$$

- 由于 $c \times d \leq 10^6$，可以枚举 y 判断 x 是不是正整数
- y 从 1 枚举到 $\lfloor \frac{cd}{b} \rfloor$，满足 $(cd-by) \mid acy$

[yLOI2020] 金陵谣

```
while(t--) {
    scanf("%lld%lld%lld%lld",&a,&b,&c,&d);
    for(int y=1;1ll*y*b<c*d;++y) {
        if((a*c*y)%(c*d-b*y) == 0) ans++;
    }
    printf("%d\n",ans);
    ans = 0;
}
```

炼金术（Alchemy）

- P8557
- 有 k 个不同的熔炉，每个熔炉可能炼出 n 种金属中的一些（或者什么都没有）
- 求出一共有多少种情况，使每种金属至少被一个熔炉炼出
- 答案对 998244353 取模，$n, k \le 10^9$

炼金术（Alchemy）

- P8557
- 有 k 个不同的熔炉，每个熔炉可能炼出 n 种金属中的一些（或者什么都没有）
- 求出一共有多少种情况，使每种金属至少被一个熔炉炼出
- 答案对 998244353 取模，$n, k \leq 10^9$
- 对于第 i 种金属，每个熔炉有两种状态：能炼、不能炼
- 每种金属至少被一个熔炉炼出，方案是 $2^n - 1$
- 每种金属互相独立，总方案是 $(2^n - 1)^k$，用快速幂计算

[EGOI2021] Number of Zeros / 零的个数

- P9309
- 给出两个正整数 a 和 b，求 $\operatorname{lcm}(a, a+1, \ldots, b)$ 的后导零个数
- $a, b \le 10^{18}$

[EGOI2021] Number of Zeros / 零的个数

- 后导零的个数取决于它 2 和 5 的个数
- 对于 $2^x \times 5^y \times z$，后导零个数为 $\min(x, y)$
- 现在要求的是 $\operatorname{lcm}(a, a+1, \ldots, b)$ 的后导零个数，那么就是

$$\min\left(\max_{a \le i \le b} x_i, \max_{a \le i \le b} y_i\right)$$

- 如何计算 $\max x_i$？
- 直接把 $a-1, b$ 同时除以 2，除几次能让 $a = b$ 就是答案
- 为什么？

[EGOI2021] Number of Zeros / 零的个数

- 如果一个数可以被 2 整除，那么这个数可以表示为 $2^k \times n$
- 如果这个数在区间 $[a, b]$ 内，那么 n 在区间 $[\lceil \frac{a}{2^k} \rceil, \lfloor \frac{b}{2^k} \rfloor]$ 内
- 所以把 $a - 1$ 拿去整除就可以得到左边界
- 只要区间内有数，那么 $[a, b]$ 就包含 2^k

[EGOI2021] Number of Zeros / 零的个数

```
int f(int x, int y, int k) {
    int cnt = -1;
    while (x != y) {
        x /= k;
        y /= k;
        cnt++;
    }
    return cnt;
}
scanf("%lld%lld", &a, &b);
a--;
printf("%lld", min(f(a, b, 2), f(a, b, 5)));
```

[NOIP2016 提高组] 组合数问题

- P2822
- 给定 k 和 t 组询问，对于每组询问给定 n, m，求对于 $0 \le i \le n, 0 \le j \le \min(i, m)$ 有多少对 (i, j) 满足

$$k \mid \binom{i}{j}$$

- $n, m \le 2000$，$k \le 21$，$t \le 10^4$

[NOIP2016 提高组] 组合数问题

- k 是固定的，提示预处理答案
- $O(n^2)$ 去预处理 $\binom{i}{j} \bmod k$
- 对于 $j \le i$，如果 $\binom{i}{j} \equiv 0 \pmod{k}$，令 $a_{i,j} = 1$
- 否则，令 $a_{i,j} = 0$
- 对 $a_{i,j}$ 求二维前缀和 $f_{i,j}$，答案就是 $f_{n,\min(n,m)}$

[NOIP2016 提高组] 组合数问题

```
for(int i=0;i<N;i++) c[i][0]=1;
for(int i=1;i<N;i++)
for(int j=1;j<=i;j++) {
    c[i][j]=(c[i-1][j-1]+c[i-1][j])%k;
    f[i][j]=f[i-1][j]+f[i][j-1]-f[i-1][j-1]+(!c[i][j]);
}
cin>>n>>m;
cout<<f[n][min(n,m)]<<"\n";
```

[Cnoi2019] 数学作业

- P5390
- T 组询问，每次给定一个集合 $\{a_n\}$，求所有子集异或和之和
- 答案对 998244353 取模，$\sum n \leq 3 \times 10^6$

[Cnoi2019] 数学作业

- 因为是异或运算，所以先拆位
- 考虑每一位，可以想到，要使贡献为 1，那么肯定是选奇数个 1 和偶数个 0
- 如果 n 个数这一位都是 0，那么贡献肯定是 0
- 如果至少有一个 1，什么时候有贡献？
  - 剩下 $n-1$ 个数的子集异或和为 0 且选这个数
  - 剩下 $n-1$ 个数的子集异或和为 1 且不选这个数
- 那么就讨论出了所有 2^n 个子集的贡献：总方案是 2^{n-1}
- 有不止一个 1 和有一个 1 是等价的，答案是 a_i 的或和再乘上 2^{n-1}

[POI 2019] Najmniejsza wspólna wielokrotność

- P6659
- t 组询问，给定整数 M，求一个区间 $[a, b]$ 使得 M 为这个区间所有数的最小公倍数
- 如果有多个区间，输出 a 最小的区间；如果还有多个，输出 b 最小的区间
- $t \leq 10^4$，$M \leq 10^{18}$

[POI 2019] Najmniejsza wspólna wielokrotność

- 区间长度为 2，$\text{lcm} = a(a+1)$，可以二分查找
- 区间长度 $= 3$ 时，$\text{lcm} = \frac{a(a+1)(a+2)}{2}$ 或 $a(a+1)(a+2)$
- 区间长度 ≥ 3 时左端点最多是 $\sqrt[3]{2M} < 1.5 \times 10^6$
- 所以可以预处理，枚举左端点 x，对于 $y = x+2, x+3, \cdots$，一直求 lcm 到 $> 10^{18}$ 时 break
- 对于每一个 (x, y) 记录在 `map[lcm]` 中，如果已有值那么有的值就是最优区间
- map 最多有 $10^6 \times \log_2 10^6$ 个值（实际远小于这个），复杂度不会炸

[POI 2019] Najmniejsza wspólna wielokrotność

```cpp
void init() {
    for (ll l = 1; l <= 1500000; l ++) {
        ll res = (long long)(l * (l + 1));
        for (int r = l + 2; ; r ++) {
            res /= gcd(res, r);
            if (res > inf / r) break;
            res *= r;
            if (!M.count(res)) M[res] = make_pair(l, r);
        }
    }
}
```

[POI 2019] Najmniejsza wspólna wielokrotność

```
pair<int, int> search(ll x) {
    int l = 1, r = 1e9;
    while (l <= r) {
        ll mid = l + r >> 1;
        if (mid * (mid + 1) > x) r = mid - 1;
        else if (mid * (mid + 1) < x) l = mid + 1;
        else return make_pair(mid, mid + 1);
    }
    return make_pair(0, 0);
}
```

[POI 2019] Najmniejsza wspólna wielokrotność

```
scanf("%lld", &m);
pair<int, int> ans = M[m];
if (ans.first) {
    printf("%d %d\n", ans.first, ans.second);
    continue;
}
ans = search(m);
if (ans.first) printf("%d %d\n", ans.first, ans.second);
else puts("NIE");
```

[COCI2019-2020#5] Zapina

- P6870
- 有 n 个不同的人和 n 道不同的题，每个题会被分配到一个人，一个人可以有多个题也可能没有题
- 第 i 个人开心当且仅当他被分配到 i 道题
- 求让至少一个人开心的分配方案数，答案对 10^9+7 取模
- $n \le 350$

[COCI2019-2020#5] Zapina

- 一个很简单的容斥思想：至少一个人开心的方案数等于总方案数 n^n 减去没人开心的方案数
- dp，令 $f_{i,j}$ 为前 i 个人分配了 j 个题且每个人都不开心的方案数，有
$$f_{i,j} = \sum_{j-k \neq i} f_{i-1,k} \times \binom{n-k}{j-k}$$
- 预处理组合数，$O(n^3)$ 转移

[COCI2019-2020#5] Zapina

```
f[0][0] = 1;
for (int i = 1; i <= n; i++)
for (int j = 0; j <= n; j++)
for (int k = 0; k <= j; k++)
if (j - k != i) {
    f[i][j] += 1ll * f[i - 1][k] * c[n - k][j - k] % p;
    f[i][j] %= p;
}
printf("%d", (ksm(n, n) + p - f[n][n]) % p);
```