Solution 8.3

T1 等式

题面 $(\sqrt{A}+\sqrt{B})^2=C$ ，我们将式子展开，得：$A+B+2\sqrt{AB}=C$

我们的任务就是统计有多少对 (A,B) 满足 AB 是平方数。

因为原题的数据满足 $n,m\le 2\times 10^6$，所以如果我们使用 O(nm) 的算法进行枚举，就寄了。

我们考虑枚举 A，然后计算每一个 A 有多少个 B 满足 $B\le m$ 并且 AB 为平方数。
我们想想，平方数有什么性质，如果 x 是一个平方数，则 x 一定可以写成 y^2（y 为整数）的形式。
对于每一个 的质因子，如果它的指数为奇数，就把他乘到 t 里，最后对于 A 满足条件答案就是 $\left\lfloor\sqrt{\frac{m}{t}}\right\rfloor$。

如果我们对于每一个 A 都去分解质因数，那么时间复杂度就是 $O(n\sqrt{n}\log n)$，也就寄了。

我们需要类似前几场快速分解质因子的方法来分解质因子

最后时间复杂度 $O(2\times 10^6+n\log n)$，期望得分 100 分。

T2 变换

由于 m 不超过 1000000， % m 的情况下数值为 0 ~ m - 1, 会出现循环节。
假设 h1 第一次变成 a1 需要 t1 次，从 a1 变换到 a1 的循环节是 B1
假设 h2 第一次变成 a2 需要 t2 次，从 a2 变换到 a2 的循环节是 B2
那我们需要求出一个 最小的 X
(X * B1 + t1 - t2) % B2 = 0，大力枚举一下 X 即可。

T3 计数

背包问题，我们可以把 1 ~ n 看成是物品从小到大依次放入，保证数据底层，dp[j][k] 表示总和为 j 情况下，已经放入了 k 个元素的方案数。最终答案是 f[m][n]
【核心代码】

```
dp[0][0]=1;
for(int i=0;i<=n;i++)
  for(int j=i;j<=m;j++)
     for(int k=1;k<=n;k++)
        dp[j][k]=(dp[j][k]+dp[j-i][k-1])%100000007;
```

T4 距离

思维题，首先我们求出两个串长度的最大公约数 g
然后对第一个串每位位置 % g 进行分类，统计每一类每种字母有几个记录为 sum[i% g]，以及每一类的字母总数为 cnt[i % g][s1[i] - ‘a’]。

然后我们对于第二个串每个位置 % g 进行分类。那么这个位置对答案在每个周期内的贡献为 sum[i % g] - cnt[s2[i] - ‘a’]的贡献。

最终答案为 tot *= gcd(n, m)

【核心代码】

```
length1=strlen(num);length2=strlen(num2);
int g=(int)gcd((int)length1,(int)length2);
for(int i=0;i<length1;++i) {
	++cnt[i%g][num[i]-'a'];
	++final[i%g];}
}
for(int i=0;i<length2;++i) total+=final[i%g]-cnt[i%g][num2[i]-'a'];

total=total*(gcd(n,m));
printf("%lld\n",total);
```