大家好，我是来自福建省福州第一中学的余林韵。我今天给大家演讲的题目是《运用化归思想解决信息学中的数列问题》。

在日常生活中，我们经常会碰到许多按一定规律排列的数，比如：

军训时喊得口号：1，1，1，2，1，1，1，1，2，1.....

年份：

1980，1981，1982，1983，1984，1985，1986，...

像上面的这些例子，都是按某种法则排列着的一列数，这样的一列数就叫做数列。

作为数学学科的一个重要分支，从小我们就开始了对数列的学习：从小学数学的找规律填数，到高中学习中专门被列为一个大章节，而全国高中数学联赛中与数列有关的问题也是层出不穷。到了工作生活当中，数列仍然有着举足轻重的地位，比如说：在股价分析中就运用到了 fibbonacci 数列引申出来的黄金分割率。

而近几年来，在各类的信息学竞赛中，数列问题也是屡次出现。本文选取了其中的 4 道比较有代表性的题目，结合各题的特性进行解题，并试图从中找出一些技巧，希望能够起到抛砖引玉的作用。

解决数列问题，一方面需要解题者有着丰富的数学知识，另一方面需要解题者能够有着开拓性的思维，充分运用化归思想，大胆猜想，不断挖掘问题的本质。

先说说化归。所谓化归，就是转化和归结，在解决问题时，人们常常将待解决的问题甲，通过某种转化过程，归结为一个已经解决或者比较容易解决的问题乙，然后通过乙问题的解答返回去求得原问题甲的解答，这就是化归方法的基本思想。它的实质就是将新问题转化为已掌握的旧知识，然后进一步理解并解决新问题。

化归方法的要素有：化归对象，即对什么东西进行化归；化归目标，即化归到何处去；化归途径，即如何进行化归。

在用化归思想解决数列问题的时候，我们需要牢牢抓住数列的特性，诸如数列的周期性、阶段性等等都可能成为我们化归的途径。

让我们看看例题一，Dispute。

题目描述如下：

数列 F_n 满足：

$F_0 = 0$

$F_n = g_{n,F_{n+1}}$

其中：$g_{x,y} = ((y-1)x^5 + x^3 - xy + 3x + 7y) \mod 9973$

给定 n，求解 F_n。[1]

数据规模：0<=N<=10^8

观察本题，题目给出了一个数列的递推公式，需要我们去求数列的某一特定项。由于数据规模太大，我们显然不能简单的从 1 到 N 进行递推。

怎么办呢？简单递推遇到的瓶颈在于数据规模过大，那么我们何不尝试着把数据规模减小到一个合适的范围，再进行递推?

于是我们萌生了算法一：将整个数列分而治之，分成若干段来解决。

通过预处理，求出数列 F 的所有项，然后将所有的 F_100000k 保存下来，这样，当给定 N 的时候，我们可以先找到离他最接近的一项 F_100000k，而后再利用简单的递推即可解决。由于递推的次数最多为 100000 次，因此算法的复杂度仅为 O(100000)，是一个非常高效的算法。

拓展：如果将数据规模继续扩大，变为 10^10，10^11，甚至 10^15 的话，如何能够解决呢?

我们遗憾的发现，在这种情况下，算法一不能胜任，因为数据规模实在太大了，我们根本无法进行一个预处理：假设递推 10^7 次需要 1s 的时间，那么如果数据达到 10^15 的话，预处理就需要运行 10^8s=27777h=1157d>3 年（等到黄花菜都凉了）。

回到算法一的思想上来。算法一化归的目标是将规模减小到一个可以承受的范围当中，化归的过程是先运用预处理的方法，然后将数据分割成有限段。当数据规模扩大的时候，我们在预处理上遇到了一道难以逾越的鸿沟。那为什么算法一会失败呢？我想，原因就在于：

数列是很优美的，然而算法一仅仅是很武断、很暴力地把数列分成若干部分。能否利用数列本身的特性、“温柔”地对它分段呢?

再次观察数列：函数 g(x,y)虽然有非常多项，看起来也十分可怕，但是仔细观察，这个式子中有两个特点：1、所有项 y 的指数非 0 即 1；2、式子的最后取模了一个质数 9973。

对于性质 1，也就是说如果把 x 看作一个常数的话，那么整理后可以得到：g(x,y) =((x^5-x+7)y +(-x^5+x^3+3x)) mod 9973;所以 g(x,y) 关于 y 是线性的。进而言之，数列 F_n 它是个 1 阶线性递推数列。联想到对于计算常系数线性递推数列的第 N 项可以利用矩阵乘法的方法，于是我将化归的目标稍作调整：看看能否通过合理的分段，将数列变成常系数线性递推数列。

幸运的，性质 1 给了我解题的方向，而性质 2 给了我实现这个化归的可能。通过进一步分析，我们可以发现：令 M=9973，则有 F_km=A*F_(k-1)m+B，于是我们便可以通过前面所说的矩阵乘法的方法迅速求出离 N 最接近的一项 F_km，再通过 O(M)的递推算出答案。整个算法的时间复杂度为 O(M+log2(N/M))。

鉴于时间关系，后面的三道例题不能在这里与大家一同分享了，有兴趣的同学可以参看我的论文与我一起讨论。

总结:

数列问题千变万化，化归不是万能的，但是只要我们能牢牢抓住问题的本质，认真仔细观察，敢于大胆猜想，富有创新精神，发现数列的美，我们就能跨越艰难险阻，到达成功的彼岸。而这，也是解决其他信息学问题一样需要的。

感谢大家的聆听！欢迎大家进行提问。