Fibonacci Words 题解

【题意概述】

斐波那契序列F[0]=0，F[1]=1，F[i]=F[i-1]+F[i-2]这里把加号改成连接。求一个串P在F(n)中出现的次数。(P长度<=100000，n<=100，答案<2^63)

【算法分析】

考虑F(100)的长度，大概是2^64多一点，显然不能用常规匹配算法来做。

注意到fibonacci序列的性质：它是截取自己的一部分来叠加的，所以重复会很多，唯一与之前不同的就是连接的部分，所以我们首先考虑连接处。

下面是P长度为3时的情况，我们只考虑连接处前后3个字符

0

1

10

101

10110

10110101

1011010110110

101101011011010110101

1011010110110101101011011010110110

1011010110110101101011011010110110101101011011010110101

可见连接处完整后其实是循环的，而且是两种交替出现。

事实上，超过P的长度之后，新接入的部分左len(P)个字符不变，而末尾len(P)个字符则是两种轮流出现，所以连接处只有2种。

所以我们考虑递推+特殊处理连接处来解决这题。首先找到大于P长度的第一个串F(x)，如果n<=x暴力KMP即可。否则，用P分别对F(x)、F(x+1)以及两个连接处分别KMP求得出现次数，之后分奇偶线性递推即可求出P在F(n)中的出现次数。

考虑复杂度，匹配4次，之后递推n轮，总复杂度O(n+len(p))。

【附注】

部分选手并未发现连接处循环的规律，而是采用每次重新匹配连接处的方法，这样复杂度是O(n\*len(p))的，但考虑到该算法能够通过现场数据，所以也算是一个编程复杂度相对较低且免去了一定思维难度的算法。最终数据并未卡该算法，但有可能被卡常数。