nLFSR:

目的:state 是亂數產生的,我們的目標是回推這個 state

教材:可以參考 crypto 1 P.30 左右

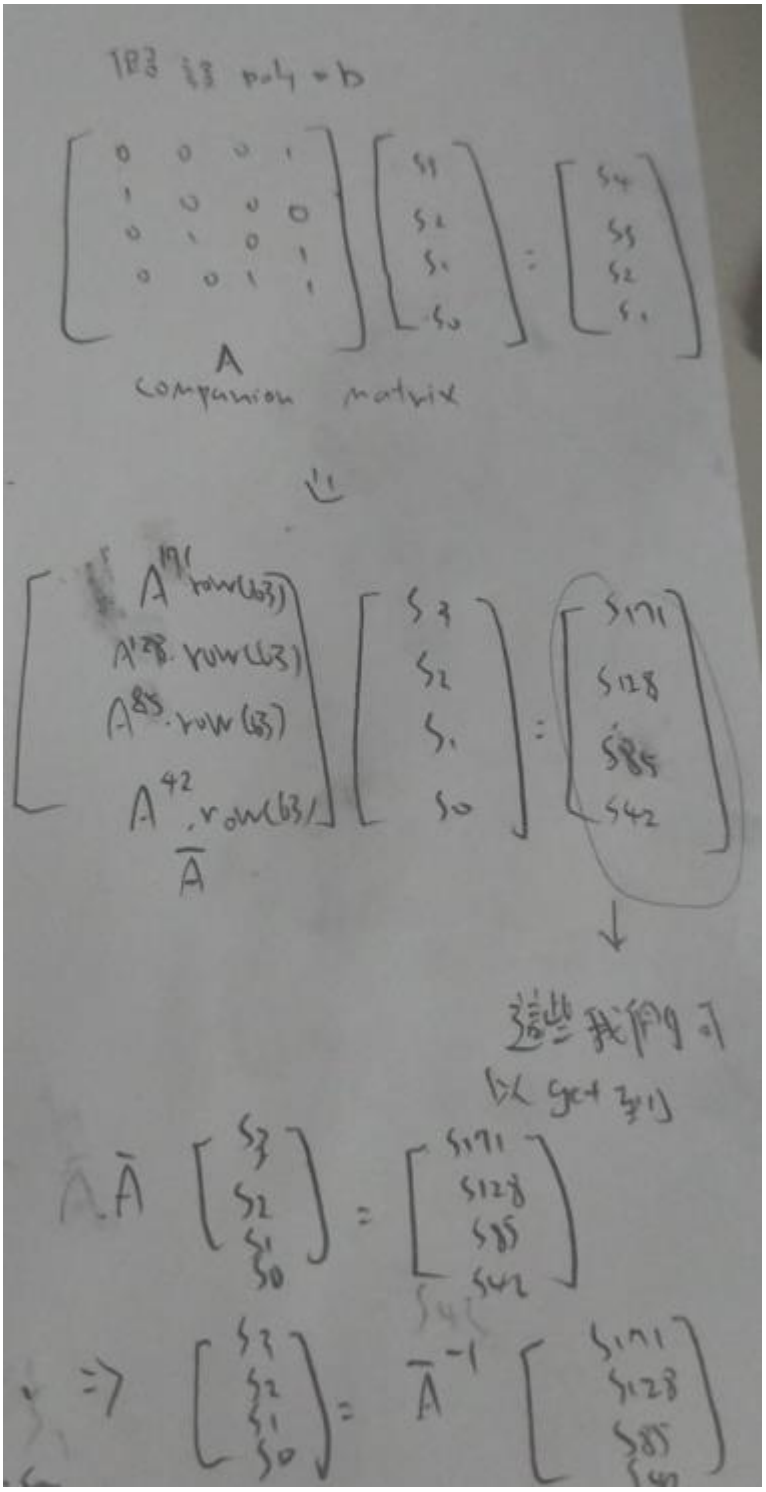
解法:

1. 透過 sever 給的 poly 可以建構出一個 companion matrix A,這個 companion matrix,是每次做 step() 的狀態轉移函數,ste()裡面是用 poly 對 64 bit 的 state 做 xor,我們可以理解為現在有 64 個狀態 ,然後在 GF(2)的 space 下面作加法 ,注意還要把 left shift 1bit 考慮進去
2. 可以發現我們可以透過 money 的變化知道 random() 回傳的是 1 還是 0(我們全都傳 1 過去,減少代表是 0),而這個值會是上一個 state 的 first bit,我們可以計算這個 bit 是經過幾次的變化才得到的,相當於 start state 乘上幾次 companion matrix 會得到 ( 注意我們只關注 first bit,因為我們只能 get 到這個資訊)
3. 知道是 start state 乘上幾次(i)companion matrix 會得到這個 bit(另這個值是 si),這個動作做 64 次,就可以建構一個 matrix 64*64,每一個 row 是 A^i.row(63)令這個矩陣為 B

   B * Start_state = we_get_state

   => Start_state = B^(-1) * we_get_state

   就可以得到 start_state 了
4. 得到 start_state 之後可以得知現在的 state 長怎樣,就可以很好的每一步都知道要傳甚麼


A
companion matrix