Sprout 2020 Algorithm - Week 12

Author: 陳楚融

Problem 1

a

令花色 c 依照花色順序(梅花、磚塊、紅心、黑桃)對應至 $0, 1, 2, 3$

點數 n 依 $1, 2, \cdots, 10, J, Q, K$ 對應至 $1, 2, \cdots, 13$

若不是鬼牌則雜湊值為 $13 * c + n$ ，是鬼牌則定為 0

b

將原樹補上節點形成高度不變的完全二元樹，依序由上層至下層，每層由左至右打上編號 $i \in N$

若節點是後來補上的，設點權 $w_i = 0$ ，否則 $w_i = 1$

如題序附圖中的第二個非空樹，補上後有三個節點，權重有 $w_1 = 1, w_2 = 1, w_3 = 0$

對於高度 h 的樹，雜湊值為：

$$H(T) = \sum_{i=1}^{2^{h+1}-1} 2^{i-1}$$

附圖中的樹的雜湊值分別為 $0, 1, 3, 5, 7$ ，皆不相同

Problem 2

a

$$\frac{52 * (52^6 - 1)}{52 - 1} = 20, 158, 268, 676$$

每個可能的字串皆對應至不同雜湊值，因此值域大小至少為可能字串的數量

長度為 n 的字串組合數為 52^n ，得總數為：

$$\sum_{i=1}^{6} 52^i = \frac{52 * (52^6 - 1)}{52 - 1} = 20,158,268,676$$

b

不能

因為能任意使用該雜湊函數僅表示可以求出任意密碼之對應雜湊值，

在無法對雜湊函數逆運算的條件下，僅能透過窮舉可能字串並檢查來找出密碼

c

若使用的是 HTTP 協定，可透過攔截封包得到使用者密碼 x 的雜湊值 $H(x) = Y$

由於雜湊函數在瀏覽器端，因此可以透過原始碼找出雜湊函數

且雜湊函數不滿足 $One - wayness$ 性質，因此可以找出 $H^{-1}(Y) = x$ ，然後透過瀏覽器登入

d

令輸入序列的每項元素 $a_i = 1000000007 * i,\ i \geq 0$ ，則 $\forall i \geq 0,\ H(a_i) = 0$

最後 Hashing Table 中雜湊值為 0 的 list 會有 N 個相異元素

則當查找任意 $q \equiv 0 \mod 1000000007$ 時，需花 $O(N)$ 時間與 list 中所有比較

Problem 3

a

$$\begin{array}{l} t \leftarrow 0 \\ b \leftarrow 1 \\ \textbf{for } i = 1 \textbf{ to } n \textbf{ do} \\ \quad t \leftarrow t + s_i * b \\ \quad b \leftarrow b * C \\ H(s) \leftarrow t \end{array}$$

b

$$\begin{aligned} H(\text{aaaaa}) &= 0 \mod 1000007 = 0 \\ H(\text{vhxec}) &= 1000007 \mod 1000007 = 0 \\ H(\text{qpuje}) &= 2000014 \mod 1000007 = 0 \end{aligned}$$

c

$$y = (x - s_l * C^{k-1}) * C + s_{r+1} \mod M$$

d

先分別以 $O(n)$ 時間計算 $x = H(s),\ y_1 = H(t[1:n])$

然後從 $i = 1$ 開始比較 x 是否與 $y_i = H(t[i:n+i-1])$ 相等，直到 $i = m - n + 1$ ，共 $m - n + 1$ 次

以 $O(n)$ 時間計算 C^{n-1} 後，每次計算 y_{i+1} 時，可透過 Problem 3.c 的結論以 $O(1)$ 時間計算 $y_{i+1} = \frac{y_i - t_i}{C} + t_{n+i} * C^{n-1}$

若 $\exists i \in [1, m-n+1] \implies x = y_i$ ，則 s 為 t 的子字串；反之則否

得總時間複雜度為 $O(n+m)$