

Bài này có thể sử dụng hash để giải.

Ta chọn 1 giá trị đủ lớn MOD để lấy phần dư. Ví dụ  $10^{16} + 3$

Việc tính giá trị hash của chuỗi  $s[i..j]$  bằng tổng:  $((s[i] - 'a') * 26^{(j-i)} + \dots + (s[j] - 'a') * 26^0) \% \text{MOD}$

Với  $s_2$  là chuỗi cần tìm các vị trí xuất hiện trong chuỗi gốc ( $s_1$ ),  $l_1$  và  $l_2$  là độ dài của chuỗi  $s_1$  và  $s_2$ , ta tính giá trị hash của chuỗi  $s_2$  trước, và giá trị hash của  $s_1[0..l_2-1]$ . Nếu 2 giá trị hash này bằng nhau thì ta in ra vị trí xuất hiện tại 1. Sau đó sử dụng 1 vòng lặp  $i$  ( $l_2 \leq i < l_1$ ) để tính giá trị hash của chuỗi con kết thúc tại  $i$ , ta có giá trị hash của  $s$  kết thúc tại  $i$  bằng:

$$((\text{giá trị hash kết thúc tại } i - 1) * 26 - (26^{l_2} * (s_1[i - l_2] - 'a') \% \text{MOD} + \text{MOD}) \% \text{MOD}$$

Ở đây trừ kết quả cho  $(26^{l_2} * (s_1[i - l_2] - 'a'))$  để loại bỏ giá trị phần tử  $s_1[i - l_2]$  (vì dãy hash kết thúc tại  $i$  không chứa phần tử  $i - l_2$ ). Việc tính  $26^{l_2}$  ta có thể tính trước khi chạy vòng lặp vì giá trị này là hằng số.

Độ phức tạp thuật toán  $O(\max(l_1, l_2))$

Tham khảo lời giải các bài khác hoặc thảo luận tại blog: <https://icnhoukdsiih.blogspot.com/>