

PASSWORDS

Lecture 01: Dynamic array & string

I/ TÓM TẮT ĐỀ BÀI

- ▶ Vanya có danh sách n mật khẩu sử dụng cho các website, nhưng mà lại không nhớ cụ thể mật khẩu là mật khẩu đúng.
- ▶ Nên Vanya sẽ thử các mật khẩu trong danh sách đó theo thứ tự tăng dần của chiều dài. Thời gian nhập mỗi mật khẩu là tốn 1 giây, tuy nhiên nếu nhập sai liên tiếp k lần ($k \leq 100$) thì anh ấy phải đợi thêm 5 giây để tiếp tục nhập được.
- ▶ Hỏi: Vanya sẽ tốn bao nhiêu giây trong trường hợp tệ nhất và tốt nhất để nhập đúng mật khẩu của mình?

I/ TÓM TẮT ĐỀ BÀI

- ▶ Input: gồm hai số n, k lần lượt là số mật khẩu trong danh sách của Vanya và số lần sai liên tiếp mà hệ thống sẽ chặn thêm 5 giây sau đó ($1 \leq n, k \leq 100$)
 - ▶ Trong n dòng tiếp theo sẽ có n mật khẩu, mỗi mật khẩu có độ dài không quá 100 kí tự.
 - ▶ Dòng cuối cùng của test sẽ là mật khẩu đúng của Vanya và luôn đảm bảo mật khẩu đúng xuất hiện ít nhất 1 lần trong danh sách n mật khẩu trên.
- ▶ Output: Sẽ bao gồm hai số nguyên lần lượt là thời gian (tính theo giây) tốn để tìm ra mật khẩu đúng trong trường hợp tốt nhất và tệ nhất.

II/GIẢI THÍCH VÍ DỤ

► Test 1: $n = 5$ và $k = 2$ Mật khẩu: abc

cba

abc

bb1

abC

ABC

A series of parallel cyan lines of varying lengths and slopes, located in the bottom right corner of the slide.

II/GIẢI THÍCH VÍ DỤ

► Test 1: $n = 5$ và $k = 2$ Mật khẩu: **abc**

Best case

abc

cba

bb1

abC

ABC

cba

abc

bb1

abC

ABC

Worst case

cba

bb1

abC

ABC

abc

II/GIẢI THÍCH VÍ DỤ

► Test 1: $n = 5$ và $k = 2$ Mật khẩu: **abc**

Best case

abc → nhập tốn 1 giây

cba

bb1

abC

ABC

Thử mật khẩu đầu tiên và đúng → chỉ tốn 1 giây nhập

Worst case

cba → nhập tốn 1 giây

bb1 → nhập tốn 1 giây

→ Do sai 2 lần, đợi 5 giây abC → nhập tốn 1 giây

ABC → nhập tốn 1 giây

→ Do sai 2 lần, đợi 5 giây **abc** → nhập tốn 1 giây

Tổng cộng thời gian thử =
 $1 + 1 + 5 + 1 + 1 + 5 + 1 =$
15

II/GIẢI THÍCH VÍ DỤ

► Test 1: $n = 5$ và $k = 2$ Mật khẩu: **abc**

Best case

abc

cba

bb1

abC

ABC

Worst case

cba

bb1

abC

ABC

abc

→ Đáp án: 1 15 (1 của best case, 15 của worst case)

II/GIẢI THÍCH VÍ DỤ

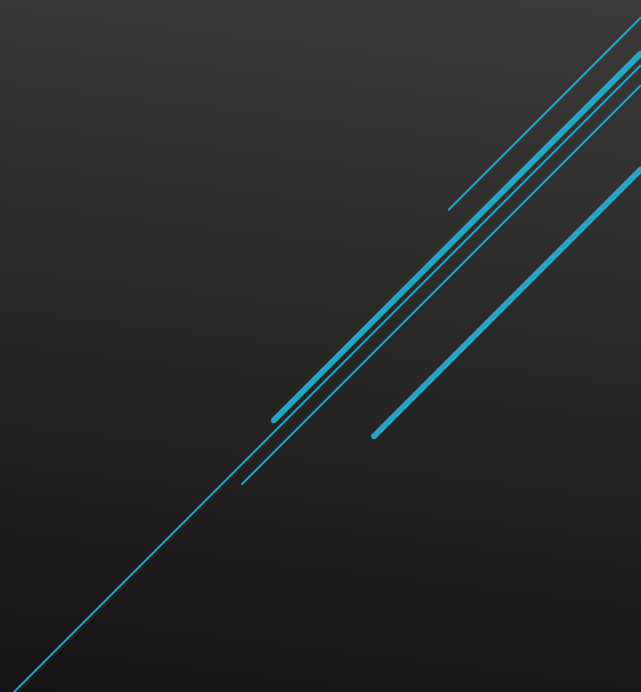
► Test 2: $n = 4$ và $k = 100$ Mật khẩu: 22

11

22

1

2



II/GIẢI THÍCH VÍ DỤ

► Test 2: $n = 4$ và $k = 100$ Mật khẩu: 22

Best case

1	11
2	22
22	1
11	2

Worst case

1
2
11
22

II/GIẢI THÍCH VÍ DỤ

► Test 2: $n = 4$ và $k = 100$ Mật khẩu: 22

Best case

1 → nhập tốn 1 giây

2 → nhập tốn 1 giây

22 → nhập tốn 1 giây

11

Do Vanya nhập theo độ dài kí tự tăng dần, nên sẽ luôn thử 1 và 2 trước, sau đó mới đến 22 → tốn 3 giây nhập

Worst case

1 → nhập tốn 1 giây

2 → nhập tốn 1 giây

11 → nhập tốn 1 giây

22 → nhập tốn 1 giây

Tương tự, Vanya sẽ nhập 1 và 2 trước, nhưng sau đó sẽ thử 11 trước 22 → $1 + 1 + 1 + 1 = 4$

II/GIẢI THÍCH VÍ DỤ

► Test 2: $n = 4$ và $k = 100$ Mật khẩu: 22

Best case

1
2
22
11

Worst case

1
2
11
22

→ Đáp án: 3 4 (3 của best case, 4 của worst case)

III/Ý TƯỞNG THUẬT TOÁN

- ▶ Nhận xét: các mật khẩu có độ dài **lớn hơn** mật khẩu đúng của Vanya thì sẽ không có thử → chỉ xét các mật khẩu có độ dài kí tự bé hơn hoặc bằng.
- ▶ Ngoài ra, các mật khẩu có độ dài **bé hơn** mật khẩu đúng của Vanya sẽ luôn được thử trước → TH tốt nhất và xấu nhất phụ thuộc vào thứ tự thử các mật khẩu có **cùng độ dài** với mật khẩu đúng.

III/Ý TƯỞNG THUẬT TOÁN

- ▶ Gọi số lần thử sai là Wrong_attempt (viết tắt là **WA**)
- ▶ Ở TH tốt nhất (**Best** case):
 - ▶ Mình sẽ thử mật khẩu đúng của Vanya **đầu tiên** trong các mật khẩu có cùng độ dài.
 - ▶ Số lần thử sai (**WA**) sẽ là số mật khẩu có độ dài bé hơn độ dài mật khẩu đúng.
- ▶ Ở TH xấu nhất (**Worst** case):
 - ▶ Mình sẽ thử hết tất cả mật khẩu cùng độ dài mà **khác mật khẩu đúng** sau đó mới đến mật khẩu đúng của Vanya.
 - ▶ Số lần thử sai (**WA**) sẽ là số mật khẩu có độ dài **bé hơn** + số mật khẩu **có cùng độ dài khác** mật khẩu đúng.

III/Ý TƯỞNG THUẬT TOÁN

- ▶ Từ số lần thử sai (**WA**), ta có thể suy ra được số lần phải đợi (**Delay**) thêm 5 giây bằng cách chia lấy nguyên **WA** cho k :

$$\text{Delay} = \frac{\text{WA}}{k}$$

- ▶ **Lưu ý**: phép chia “/” lấy phần nguyên.
- ▶ Vậy tổng thời gian thử của Vanya của mỗi TH được tính theo công thức:

$$\text{Answer} = \text{WA} + \text{Delay} * 5 + 1$$

III/Ý TƯỞNG THUẬT TOÁN

► Test: $n = 9$ và $k = 2$

Mật khẩu: 22

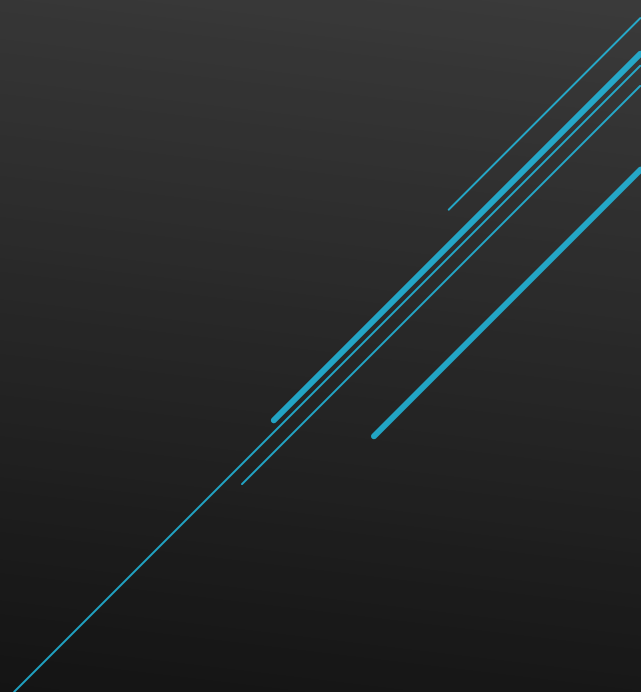
Best case

1
2
3
4
5
22
11
33
69

11
22
33
69
1
2
3
4
5

Worst case

1
2
3
4
5
11
33
69
22



III/Ý TƯỞNG THUẬT TOÁN

► Test: $n = 9$ và $k = 2$

Mật khẩu: 22

Best case

1 → nhập tốn 1 giây
2 → nhập tốn 1 giây, đợi 5 giây
3 → nhập tốn 1 giây
4 → nhập tốn 1 giây, đợi 5 giây
5 → nhập tốn 1 giây
22 → nhập tốn 1 giây (không có đợi
11 do là nhập đúng)
33
69

→ Best case
= 16

Worst case

1 → nhập tốn 1 giây
2 → nhập tốn 1 giây, đợi 5 giây
3 → nhập tốn 1 giây
4 → nhập tốn 1 giây, đợi 5 giây
5 → nhập tốn 1 giây
11 → nhập tốn 1 giây, đợi 5 giây
33 → nhập tốn 1 giây
69 → nhập tốn 1 giây, đợi 5 giây
22 → nhập tốn 1 giây

→ Worst case
= 29

III/Ý TƯỞNG THUẬT TOÁN (ÁP DỤNG CÔNG THỨC)

► Test: $n = 9$ và $k = 2$ Mật khẩu: 22

Best case

1 → WA $WA = 5$
2 → WA
3 → WA $Delay = \frac{WA}{k} = \frac{5}{2} = 2$
4 → WA
5 → WA
22 → accepted
11 → Best case = $WA + Delay * 5 + 1$
33 = $5 + 2 * 5 + 1$
69 = 16

Worst case

1 → WA $WA = 8$
2 → WA
3 → WA $Delay = \frac{WA}{k} = \frac{8}{2} = 4$
4 → WA
5 → WA
11 → WA → Worst case
33 → WA = $WA + Delay * 5 + 1$
69 → WA = 29
22 → accepted

IV/CÁC BƯỚC THUẬT TOÁN

- ▶ Bước 1: Đọc vào n và k
- ▶ Bước 2: Khởi tạo mảng rỗng `arr = []`, biến đếm `Smaller = 0` và `Same = 0`.
Cho i chạy từ 1 đến n và đọc từng mật khẩu cho vào mảng `arr`.
- ▶ Bước 3: Đọc mật khẩu đúng password của Vanya.
- ▶ Bước 4: Duyệt qua từ xâu s trong mảng `arr`:
 - ▶ Bước 4.1: Nếu `len(s) < len(password)` thì `Smaller` tăng 1 đơn vị.
 - ▶ Bước 4.2: Nếu `len(s) == len(password)` và `s != password` thì `Same` tăng 1 đơn vị.

IV/CÁC BƯỚC THUẬT TOÁN

- ▶ Bước 5: Tính theo từng TH tốt nhất ($WA_Best = \text{Smaller}$), và xấu nhất ($WA_Worst = \text{Smaller} + \text{Same}$).
- ▶ Bước 6: tính $Delay_Best = WA_Best / k$ và $Delay_Worst = WA_Worst / k$.
- ▶ Bước 7: Xuất ra đáp án TH tốt nhất và xấu nhất theo công thức.

V/ MÃ GIẢ

```
function main():
    read(n, k)
    arr = []*n
    for i=0 to n-1:
        read(arr[i])
    read(password)

    smaller = same = 0
    for i=0 to n-1:
        if (len(arr[i]) < len(password)):
            smaller = smaller + 1
        elif (len(arr[i]) == len(password) &&
arr[i] != password):
            same = same + 1

    WA_best = smaller
    WA_worst = smaller + same
```

```
delay_best = WA_best / k
delay_worst = WA_worst / k
```

```
print(WA_best+delay_best*5+1)
print(WA_worst+delay_worst*5+1)
```