**TRƯỜNG ĐẠI HỌC GIAO THÔNG VẬN TẢI**

**KHOA CÔNG NGHỆ THÔNG TIN**

**-----------------**



BÀI TẬP LỚN

**THUẬT TOÁN VÀ ỨNG DỤNG**

**BÀI TOÁN:**

**XÓA K CHỮ SỐ ĐƯỢC SỐ LỚN NHẤT**

Giảng viên hướng dẫn: thầy Phạm Xuân Tích

Sinh viên thực hiện: Lê Quang Thọ (181202289)

**NHẬN XÉT GIẢNG VIÊN**

……………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………..…………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………

Hà Nội, Ngày …. Tháng … Năm.....

Giảng viên

Mục Lục:

[**I.** **PHÁT BIỂU BÀI TOÁN.** - 3 -](#_Toc57668609)

[1. Bài Toán - 3 -](#_Toc57668610)

[2. Phân Tích Bài Toán - 3 -](#_Toc57668611)

[3. Miền Ràng Buộc - 3 -](#_Toc57668612)

[4. Ví Dụ (In, Out) - 3 -](#_Toc57668613)

[**II.** **NÊU Ý TƯỞNG.** - 3 -](#_Toc57668614)

[1. Ý Tưởng - 3 -](#_Toc57668615)

[2. Cách Tính - 4 -](#_Toc57668616)

[**III.** **MINH HỌA BÀI TOÁN.** - 4 -](#_Toc57668617)

[1. Minh Họa - 4 -](#_Toc57668618)

[**IV.** **CODE.** - 5 -](#_Toc57668619)

[1. Code - 5 -](#_Toc57668620)

[2. Đánh Giá Bài Toán - 5 -](#_Toc57668621)

1. **PHÁT BIỂU BÀI TOÁN.**
2. Bài Toán

Cho số tự nhiên **n**, bạn hãy xóa đi **k** chữ số đó của **n** những số còn lại giữ nguyên thứ tự mà được số lớn nhất.

**Biết rằng:** **k** luôn nhỏ hơn số chữ số của **n**. Tức, sau khi xóa luôn còn lại số có **n-k** chữ số.

1. Phân Tích Bài Toán

**n :** một số tự nhiên

**k:** là số trường hợp

**t[i] :** số chữ số cần phải xóa trong **n**

Dòng đầu tiên số tự nhiên nn không chứa các chữ số 0 vô nghĩa ở đầu có không quá **105** chữ số.

Dòng tiếp theo là chứa **t[i]** là số trường hợp cần tính 1 ≤ **t[i]** ≤ 10

1. Miền Ràng Buộc

**n** , **k, t[i]**≥ 0

1. Ví Dụ (In, Out)

* Input: **n** = 6345423410035456542234234

**k** = 5

**t**[**i**] = {1, 3, 7, 4, 5}

* Output:

645423410035456542234234

6543410035456542234234

654435456542234234

654410035456542234234

65441035456542234234

1. **NÊU Ý TƯỞNG.**
2. Ý Tưởng

Để thực hiện được việc xóa **k** chữ số để được một số tự nhiên lớn nhất thì điều đầu tiên ta phải ưu tiên xóa các chữ số nhỏ hơn trước ưu yên theo chiều từ trái qua phải. Nhận thấy điều đó ta xét lần lượt từ trái qua phải từng chữ số. Nếu thỏa mãn thì đẩy chữ số đó vào stack.

Nếu chữ số đang xét lớn hơn phần tử cuối của stack thì đấy phần tử cuối cùng của stack ra và đưa chữ số đang xét đó vào stack.

Ngược lại ta phải đẩy chữ đang xét đó vào stack.

Xóa tới khi **t[i]** = 0.

Trường hợp **t[i]** chưa về 0 mà đã tới chữ số cuối cùng của **n** thì ta xóa nốt các chữ số ưu tiên từ phải qua trái.

1. Cách Tính

* Chạy 1 vòng while để thực hiện hết **k** trường hợp để thực hiện việc gọi tới hàm xử lí việc xóa **t[i]** chữ số
* Trong hàm xử lí đó ta thực hiện 1 vòng **for** có lần lặp là chiều dài của số tự nhiên **n.** (1)
* Trong vòng lặp (1) ta chạy 1 vòng **while** với điều kiện nếu chữ số đang xét của **n**  lướt hơn phần tử cuối cùng của stack thì bỏ phần tử cuối đó đi cho đến khi không bỏ được nữa thì đẩy chữ số đó vào trong stack và tiếp tục vòng lặp (1)
* Kết thúc vòng (1) ta xét nếu **t[i]** ≥ 0 thì ta lại chạy vòng while để bỏ nốt **t[i]** chữ số ở cuối stack đi
* In ra stack đó ra màn hình 🡪 đó là kết quả cần tìm.
* Out khỏi hàm xử lí để thực hiện tiếp các trường hợp còn lại

1. **MINH HỌA BÀI TOÁN.**
2. Minh Họa

* Input: **n** = 99856, **k** = 1, **t**[**i**] = {2}
* len(n) = 4

+ **n[0]** = 9, len(stack) = 0, **t[i]** = 2

🡪stack.append(9)

+ **n[1]** = 9, len(stack) = 1, **t[i]** = 2

🡪stack.append(9)

+ **n[2]** = 8, len(stack) = 2, **t[i]** = 2

🡪stack.append(8)

+ **n[3]** = 5, len(stack) = 3, **t[i]** = 2

🡪stack.append(5)

+ **n[4]** = 6, len(stack) = 4, **t[i]** = 2

🡪**t[i]--,**stack.pop()

+ **n[4]** = 6, len(stack) = 3, **t[i]** = 1

🡪stack.append(6)

* **t[i]** = 1, len(stack) = 4

🡪stack.pop(), **t[i]--**

* **Output**: 998

1. **CODE.**
2. Code

**def** ketqua(a,k):

stack = []

**for** i **in** range(len(a)):

**while** len(stack) **and** k **and** int(a[i]) > int(stack[len(stack)-1]):

stack.pop()

k-=1

stack.append(a[i])

**while** len(stack) **and** k:

k-=1

stack.pop()

**for** j **in** range(len(stack)):

print(stack[j], end="")

print("")

**if \_\_name\_\_ == '\_\_main\_\_':**

a = input()

k = int(input())

while k:

ketqua(a,int(input()))

k-=1

1. Đánh Giá Bài Toán

* Độ phức tạp thời gian: **O**(n).
* Độ phức tạp không gian: **O**(n).