**TRƯỜNG ĐẠI HỌC GIAO THÔNG VẬN TẢI**

**KHOA CÔNG NGHỆ THÔNG TIN**

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



BÀI TẬP LỚN

**THUẬT TOÁN VÀ ỨNG DỤNG**

**BÀI TOÁN: HOÁN VỊ LỚN NHẤT**

Giảng viên hướng dẫn: thầy Phạm Xuân Tích

Sinh viên thực hiện: Nguyễn Đức Phú (181201867)

**NHẬN XÉT GIẢNG VIÊN**

………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………

Hà Nội, Ngày …. Tháng … Năm.....

Giảng viên

Mục Lục:

[**I.** **PHÁT BIỂU BÀI TOÁN.** - 3 -](#_Toc57968703)

[1. Bài Toán - 3 -](#_Toc57968704)

[2. Phân Tích Bài Toán - 3 -](#_Toc57968705)

[3. Miền Ràng Buộc - 3 -](#_Toc57968706)

[4. Ví Dụ (In, Out) - 3 -](#_Toc57968707)

[**II.** **NÊU Ý TƯỞNG.** - 3 -](#_Toc57968708)

[1. Ý Tưởng - 3 -](#_Toc57968709)

[2. Cách Tính - 3 -](#_Toc57968710)

[**III.** **MINH HỌA BÀI TOÁN.** - 3 -](#_Toc57968711)

[**IV.** **CODE.** - 4 -](#_Toc57968712)

[1. Code - 4 -](#_Toc57968713)

[2. Đánh Giá Bài Toán - 4 -](#_Toc57968714)

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

Cho một mảng gồm **n** phần tử các số là duy nhất tăng dần từ 1 và chưa được sắp xếp và một số tự nhiên **k**, hãy hoán vị các số trong mảng không vướt quá  **k** lần. Xác định giá trị từ lớn nhất có thể tạo được bằng cách kết hợp các số trong mảng đã hoán vị.

1. Phân Tích Bài Toán

**k :** số nguyên biểu thị số lần hoán đổi

**n :** số phần tử trong mảng

**arr :** một mảng các số nguyên

Dòng đầu tiên chứa 2 số nguyên **n** và **k**

Dòng thứ 2 là **arr[i]** gồm **n** phần tử ( 1 ≤ **arr[i]** ≤  **n** )

1. Miền Ràng Buộc

1 ≤ **n** ≤ 105

1 ≤ **k** ≤ 109

1. Ví Dụ (In, Out)

* Input: **n** = 5, **k** =3, **arr**[] = [ 4 , 5 , 2 , 1 , 3 ]
* Output: 5 4 3 2 1
* Input: **n** = 5, **k** = 1, **arr**[] = [ 4 , 5 , 2 , 1 , 3 ]
* OutPut: 5 4 2 1 3

1. **NÊU Ý TƯỞNG.**
2. Ý Tưởng

Ta dùng tham lam để giải bài toán. Hoán vị lớn nhất của bài toán là các phần tử lớn nhất nằm phía trước của mảng, tức là các phần tử lớn nhất được sắp xếp theo thứ tự giảm dần

Nếu số phần tử bằng bằng số lần hoán đổi vị trí thì ta chỉ cần đảo ngược lại mảng.

1. Cách Tính

* Tạo một mảng bổ trợ có độ dài **n** để lưu trữ vị trí của các phần tử
* Bây giờ ta sử dụng một vòng lặp **k** lần

+ Nếu **k** = 0 thì break

+ Nếu phần tử **arr**[**i**] = n – i thì quay lại vòng lặp và tiếp tục

* Tìm giá trị lớn nhất của các phần tử thứ **i** và phần tử **n** – **i**. Sau đó hoán đổi vị trí phần tử cho nhau
* Giảm **k** sau mỗi lần hoán đổi
* In ra mảng đã sắp xếp

1. **MINH HỌA BÀI TOÁN.**

* Input: **n** = 5, **k** =2, **arr**[] = [ 1 , 2 , 3 , 4 , 5]
* **k** = 2
* **arr**[**i**] = 1
* Swap : **arr**[] = [ **5** , 2 , 3 , 4 , **1** ]
* **pos** {5:0 ; 2:1 ; 3:2 ; 4:3 ; 1:4 }
* **k** = 1
* **arr**[**i**] = 5
* **arr**[**i + 1**] = 4
* Swap : **arr**[] = [ 5 , **4** , 3 , **2** , 1 ]
* **pos** {5:0 ; 4:1 ; 3:2 ; 2:3 ; 1:4 }
* **k** = 0 **=>** **Break**
* Output: 5 4 3 2 1
* Input: **n** = 5, **k** = 1, **arr**[] = [ 4, 5, 2, 1, 3 ]
* **k** = 1
* **arr**[**i**] = 4
* Swap : **arr**[] = [ **5** , **4** , 3 , 2 , 1]
* **pos** { 5:0 ; 4:1 ; 3:2 ; 2:3 ; 1:4 }
* **k** = 0 **=>** **Break**
* OutPut: 5 4 2 1 3

1. **CODE.**
2. Code

**def KswapPermutation(n,k,arr):** pos = {}  
 **for i in range(n):** pos[arr[i]] = i  
 **for i in range(n):** if (k ==0):  
 break  
 if (arr[i] == n-i):  
 continue  
 temp = pos[n - i]  
 pos[arr[i]] = pos[n - i]  
 pos[n - i] = i  
 arr[temp], arr[i] = arr[i], arr[temp]  
 k = k - 1  
**if \_\_name\_\_ == '\_\_main\_\_':** n,k = map(int, input().split())  
 arr = list(map(int, input().split()))  
 **KswapPermutation(n,k,arr)** print (' '.join(map(str, arr)))

1. Đánh Giá Bài Toán

* Độ phức tạp: **O**(n).

Để lưu trữ mảng mới **O**(n) khoảng trống bắt buộc