

## BÀI TẬP CHƯƠNG 2

### I. Bài tập về cấu trúc tập hợp (set)

**Bài 1.** Cho một vector chứa các số nguyên. Hãy đưa ra số lượng phần tử khác nhau trong vector đó.

Ví dụ: Với `inputVector = [1, 3, 3, 2]`, kết quả là 3.

**Bài 2.** Anh Việt đang thống kê số liệu cho công ty, anh muốn giá trị nhỏ thứ hai ở trong một dãy số nguyên. Hãy giúp anh Việt tìm ra giá trị đó, nếu không có kết quả như yêu cầu thì trả về "NO". Ví dụ: Với `arr = [1, 2, 3, 1, 1]` thì kết quả sẽ là 2.

**Bài 3.** Tính số lượng tối thiểu các ký tự cần thay đổi trong chuỗi `str` để chuỗi đó có `k` ký tự khác nhau hoặc xuất ra điều đó là không thể. Chuỗi `str` chỉ bao gồm những chữ cái latin viết thường và cũng chỉ có thể thay thành những chữ cái latin viết thường.

Ví dụ:

- Với `str = "yandex"`, `k = 6` thì kết quả sẽ là `diversity(s, k) = "0"`
- Với `str = "google"`, `k = 7` thì kết quả sẽ là `diversity(s, k) = "impossible"`.
- Với `str = "codelearn"`, `k = 9` thì kết quả sẽ là `diversity(s, k) = "1"`

**Bài 4.** An có một danh sách các sản phẩm trong siêu thị nhưng thật không may trong danh sách lại có những sản phẩm xuất hiện nhiều lần. Bạn hãy giúp An làm lại danh sách sao cho một sản phẩm chỉ xuất hiện một lần trong danh sách và các sản phẩm xuất hiện theo thứ tăng dần trong từ điển.

Ví dụ:

- Với `products = ["watermelon", "grapes", "grapes", "apple", "grapes"]` thì `getDistinctProducts(products) = ["apple", "grapes", "watermelon"]`.

### II. Bài tập về cấu trúc hàng đợi (queue)

**Bài 5.** Số siêu nguyên tố là số:

- Bản thân nó là số nguyên tố.
- Khi xóa đi lần lượt các chữ số sau cùng của nó thì số mới vẫn là số nguyên tố.

Ví dụ 2393 là số siêu nguyên tố vì 2393, 239, 23, 2 là số nguyên tố.

Cho một số `n`, hãy đưa số dãy số siêu nguyên tố nhỏ hơn hoặc bằng `n` đã được sắp xếp tăng dần.

Ví dụ:

- Với `n = 30`; thì `superPrimeNumber(n) = [2, 3, 5, 7, 23, 29]`;

Vì các số 2, 3, 5, 7, 23 và 29 đều là số siêu nguyên tố và nhỏ hơn hoặc bằng 30.

**Bài 6.** Cho một mảng các số nguyên `arr` và một số nguyên dương `k`. Hãy viết hàm trả về các phần tử âm đầu tiên trong cửa sổ `k`. Nếu không tồn tại số âm nào output 0.

Ví dụ:

- Với `arr = [-8, 2, 3, -6, 10]`, `k = 2` thì các cửa sổ lần lượt chứa các giá trị: `[-8, 2]`, `[2, 3]`, `[3, -6]`, `[-6, 10]` nên kết quả trả về sẽ có dạng `firstNegative(arr, k) = [-8, 0, -6, -6]`

**Bài 7.** Cho một ma trận trong đó mỗi ô trong ma trận có thể có các giá trị 0, 1 hoặc 2 có nghĩa sau:

- 0: Là một ô trống.
- 1: Là một quả cam tươi.
- 2: Là một quả cam bị hỏng.

Chúng ta phải xác định thời gian tối thiểu mà tất cả các quả cam bị hỏng. Một quả cam hỏng ở chỉ số  $[i, j]$  có thể làm hỏng quả cam tươi khác ở các chỉ số  $[i-1, j]$ ,  $[i+1, j]$ ,  $[i, j-1]$ ,  $[i, j+1]$  (lên, xuống, trái và phải). Nếu không thể bị hỏng tất cả quả cam thì trả về -1.

Ví dụ:

- Với matrix = [  
[2, 1, 0, 2, 1],  
[1, 0, 1, 2, 1],  
[1, 0, 0, 2, 1]].

Thì đầu ra có dạng `rotOrange(matrix) = 2`

- Với matrix = [  
[2, 1, 0, 2, 1],  
[0, 0, 1, 2, 1],  
[1, 0, 0, 2, 1]].

Thì đầu ra có dạng `rotOrange(matrix) = -1`

**Bài 8.** Do những ngày hè quá nóng bức và nhàm chán nên Tide đã nghĩ ra một trò chơi khá thú vị với queue. Ban đầu trong queue có 5 số 1, 2, 3, 4, 5 với mỗi lượt chơi Tide sẽ xóa phần tử ở đầu queue và cho 2 phần tử đó xuống cuối của queue và cứ tiếp tục cho đến khi Tide cảm thấy mệt và không chơi được nữa.

Ví dụ tại lượt chơi thứ nhất trạng thái của queue là 1, 2, 3, 4, 5

Tại lượt chơi thứ 2 trạng thái của queue là 2, 3, 4, 5, 1, 1

Các bạn hãy giúp Tide xác định xem số đầu tiên của queue tại lượt chơi thứ N nhé.

**Bài 9.** Giả sử có một đường tròn. Có  $n$  trạm xăng trên vòng tròn đó. Bạn được cho hai tập dữ liệu.

1. Lượng xăng mà mọi trạm xăng có.
2. Khoảng cách từ trạm xăng đó đến trạm xăng kế tiếp.

Tính điểm đầu tiên mà từ đó một xe tải có thể kết thúc hành trình trên vòng tròn (Xe tải sẽ dừng lại tại mỗi trạm xăng và nó có dung tích vô hạn). Độ phức tạp thời gian kỳ vọng là  $O(n)$ . Giả sử với một lít xăng, xe tải có thể đi được một đơn vị quãng đường.

Ví dụ, cho 4 trạm xăng với khối lượng xăng và khoảng cách tới trạm kế tiếp là các cặp giá trị  $\{4, 6\}$ ,  $\{6, 5\}$ ,  $\{7, 3\}$  và  $\{4, 5\}$ . Điểm đầu tiên mà từ đó xe tải có thể thực hiện chuyến đi vòng tròn là trạm xăng thứ hai. Đầu ra phải là “start = 1” (chỉ số của trạm xăng thứ hai).

**Bài 10.** Một hiệu trưởng muốn đi thăm các ký túc xá (có tất cả  $n$  ký túc). Ông ta rất bận nên chỉ thăm được  $k$  ( $k \leq n$ ) nơi gần nhất. Các ký túc xá được tọa lạc trên một mặt phẳng hai chiều. Hãy giúp ông ấy, biết rằng bạn có tọa độ của  $n$  ký túc và vị trí hiện tại của ông ấy có tọa độ là  $O(0,0)$ .

Chú ý: khoảng cách giữa điểm  $P(x,y)$  tới gốc  $O(0,0)$  là khoảng cách Euclidean.

Ví dụ:

Input:  $\{(1, 0), (2, 1), (3, 6), (-5, 2), (1, -4)\}$ ,  $k = 3$

Output:  $\{(1, 0), (2, 1), (1, -4)\}$

**Bài 11.** Monk hiện có một mảng các số nguyên  $A$ . Với mỗi chỉ số  $i$ , anh ta muốn tìm tích của các số lớn nhất, số lớn thứ hai và số lớn thứ ba trong khoảng  $[1, i]$ .

Chú ý: các số có thể giống nhau nhưng chúng phải khác nhau về chỉ số.

**Bài 12.** Cho một mảng số nguyên và một số nguyên  $k$ . Tìm số lớn nhất trong từng mảng con liên tiếp kích thước  $k$ .

Ví dụ, với mảng:  $arr[] = \{1, 2, 3, 1, 4, 5, 2, 3, 6\}$ ,  $k = 3$

Kết quả là: 3 3 4 5 5 5 6, vì

Số lớn nhất của  $\{1, 2, 3\}$  is 3

Số lớn nhất của  $\{2, 3, 1\}$  is 3

Số lớn nhất của  $\{3, 1, 4\}$  is 4

**Bài 13.** Cho một mảng số nguyên và một số nguyên  $k$ . Tính tổng các số nhỏ nhất và lớn nhất trong từng mảng con liên tiếp kích thước  $k$ .

Ví dụ, với mảng:  $arr[] = \{2, 5, -1, 7, -3, -1, -2\}$ ,  $k = 4$

Kết quả là: 18, vì

$\{2, 5, -1, 7\}$ ,  $\min + \max = -1 + 7 = 6$

$\{5, -1, 7, -3\}$ ,  $\min + \max = -3 + 7 = 4$

$\{-1, 7, -3, -1\}$ ,  $\min + \max = -3 + 7 = 4$

$\{7, -3, -1, -2\}$ ,  $\min + \max = -3 + 7 = 4$

Tổng tất cả  $\min$  &  $\max = 6 + 4 + 4 + 4 = 18$

### III. Bài tập về cấu trúc ánh xạ (map)

**Bài 14.** Cho dãy số nguyên  $n$  phần tử  $a_1, a_2, \dots, a_n$ , hãy thống kê tần suất xuất hiện của từng phần tử có trong dãy theo thứ tự từ bé đến lớn.

**Bài 15.** Cho dãy số nguyên  $n$  phần tử  $a_1, a_2, \dots, a_n$ , hãy tìm những phần tử có tần suất xuất hiện nhiều nhất trong dãy. Ví dụ:  $a = \{4\ 7\ 2\ 8\ 4\ 8\ 3\ 2\}$ . Kết quả là các số  $\{2\ 4\ 8\}$  có tần suất lớn nhất là 2.

**Bài 16.** Cho một chuỗi  $s$ , hãy đưa ra một dãy lần lượt là các ký tự và số lần xuất hiện của nó, các ký tự sắp xếp theo thứ tự từ điển.

Ví dụ:

- Với  $s = "aaccdd"$  thì  $countChar = ["a\ 2", "c\ 3", "d\ 1"]$ .
- Với  $s = "aabbbca"$  thì  $countChar = ["a\ 3", "b\ 3", "c\ 1"]$ .

**Bài 17.** Bạn được cung cấp một chuỗi  $s$  làm đầu vào và bạn phải in biểu mẫu đã sửa đổi.

Chuỗi được sửa đổi theo cách sau:

- Những ký tự giống nhau chỉ lấy duy nhất một ký tự.
- Ký tự nào có tần số xuất hiện trong chuỗi  $s$  nhiều hơn được sắp xếp trước.

- Những ký tự có cùng tần số xuất hiện thì sắp xếp theo thứ tự từ điển từ nhỏ đến lớn.

Ví dụ:

- Với  $s = \text{"codelearn"}$  thì  $\text{modifyString} = \text{"eacdlnor"}$ .
- Với  $s = \text{"helloworld"}$  thì  $\text{modifyString} = \text{"lodehrw"}$ .

**Bài 18.** Cho hai dãy số nguyên  $\text{arr1}$  và  $\text{arr2}$ , hãy tính tổng những số xuất hiện trong cả hai dãy, lưu ý là mỗi số chỉ được tính một lần.

Ví dụ:

- Với  $\text{arr1} = [6, 7, 5, 4, 6, 8]$ ,  $\text{arr2} = [2, 5, 7, 5, 3]$  thì  $\text{sumOfCommon}(\text{arr1}, \text{arr2}) = 12$ .
- Với  $\text{arr1} = [5, 6, 7]$ ,  $\text{arr2} = [2, 3, 4]$  thì  $\text{sumOfCommon}(\text{arr1}, \text{arr2}) = 0$ .

**Bài 19.** Cho một dãy số nguyên  $\text{arr}$  và một số nguyên dương  $\text{sum}$ .

Hãy kiểm tra xem dãy số có tồn tại hai số có tổng bằng  $\text{sum}$  hay không.

Ví dụ:

- Với  $\text{arr} = [2, 4, -1, 9, 8]$ ,  $\text{sum} = 6$  thì  $\text{checkSum}(\text{arr}, \text{sum}) = \text{true}$ .
- Với  $\text{arr} = [2, 5, 3, 8, 9]$ ,  $\text{sum} = 3$  thì  $\text{checkSum}(\text{arr}, \text{sum}) = \text{false}$ .
- Với  $\text{arr} = [4, 7, 3, 5]$ ,  $\text{sum} = 6$  thì  $\text{checkSum}(\text{arr}, \text{sum}) = \text{false}$ .

**Bài 20.** Cho một dãy gồm các số nguyên, hãy đưa ra số thỏa mãn là số lớn nhất trong những số có tần số bé nhất.

Ví dụ:

- Với  $\text{arr} = [2, 2, 4, 4, 7, 7, 7]$  thì  $\text{largestElement}(\text{arr}) = 4$ .
- Với  $\text{arr} = [1, 3, 4, 5, 5]$  thì  $\text{largestElement}(\text{arr}) = 4$ .

**Bài 21.** Cho một danh bạ điện thoại và danh sách các tên. Ứng với mỗi tên trong danh sách hãy đếm xem tên này là bắt đầu của bao nhiêu tên trong danh bạ điện thoại, hay nói cách khác hãy đếm xem tên này là prefix của bao nhiêu tên trong danh bạ.

Ví dụ:

- Với  $\text{contacts} = [\text{"Codelearn"}, \text{"Codewar"}]$ ,  $\text{names} = [\text{"Code"}, \text{"Codel"}, \text{"io"}]$  thì  $\text{countPrefix}(\text{contacts}, \text{names}) = [2, 1, 0]$ .

**Bài 22.** Cho danh sách các sản phẩm của 2 kho hàng A và B. Do chiến lược kinh doanh bạn được giao nhiệm nhập các sản phẩm từ kho B vào kho A sao cho những sản phẩm nào đã có trong kho A thì không nhập.

Ví dụ:

- Với  $A = [\text{"Banana"}, \text{"Banana"}, \text{"Apple"}]$ ,  $B = [\text{"Orange"}, \text{"Apple"}, \text{"Banana"}, \text{"Watermelon"}]$  thì  $\text{mergeProducts}(A, B) = [\text{true}, \text{false}, \text{false}, \text{true}]$ .

Lưu ý: Sản phẩm thứ 2 và 3 trong kho B là "Apple" và "Banana" sẽ không được nhập vì trong kho A đã có sản phẩm như vậy.