**Function**

Bài 1. Viết một hàm số python có tham số đầu vào là 1 danh sách và các giá trị muốn xóa

khỏi danh sách đó. Kết quả đầu ra là 1 danh sách đã được xóa đi các giá trị nhập vào. Ví

dụ: Truyền vào \_inlist = [5,8,11,9,11,8,8] và giá trị cần xóa là 8 và 11, thì danh sách đầu

ra thu được sẽ là: \_outlist = [5,9].

Bài 2. Viết một hàm Python để tính toán giai thừa của một số (một số nguyên không âm).

Gợi ý: Sử dụng hàm đệ quy n! = (n-1)!\*n.

Bài 3. Viết một hàm Python chấp nhận một chuỗi và tính toán số chữ hoa và chữ thường.

Vd: Chuỗi mẫu: ‘The Brow Fox’  Kết quả mong đợi: Số ký tự chữ hoa: 3; Số ký tự chữ

thường: 7.

**List**

Bài 4: Viết chương trình python tính tổng của các phần tử trong một list sau: \_list= [1,

2, 3, 4, 5, 6, 7, 8, 9].

Bài 5: Nhập vào từ bàn phím số n và list cho trước, tìm các từ có độ dài lớn hơn n từ

list đó.

Bài 6: Viết chương trình tạo một list mới bằng cách loại bỏ các phần tử có giá trị giống

nhau trong list đầu vào. Ví dụ: nhập list đầu vào là \_list= ['abc', 'xyz', 'abc', '12', ‘ii’, ‘12’,

‘5a’]  kết quả list mới là \_new= ['xyz', ‘ii’, ‘5a’]

Problem. Numbers on Whiteboard

time limit per test

2 seconds

memory limit per test

256 megabytes

input

standard input

output

standard output

Cho một dãy số nguyên từ 1 ….. n. Trong một lần thực hiện bạn được phép xóa 2 số a và b bất kỳ và thay thế bằng một số nguyên (a + b)/2 được làm tròn lên.

Bạn được thực hiện n−1 tính toán như vậy sao cho thu được số nhỏ nhất có thể còn lại.

Cho ví dụ: Nếu n=4, Ta sẽ có những hoạt động tối ưu như sau:

Chọn a=4 và b=2, Số mới thu được là  3, Dãy mới thu được sẽ là [1,3,3] ;

Chọn a=3 và b=3, Số mới thu được là  3, Dãy mới thu được sẽ là [1,3] ;

Chọn a=1 và b=3, Số mới thu được là  2, Dãy mới thu được sẽ là [2] ;

choose a=3 and b=3, so the new number is 3, and the whiteboard contains [1,3];

choose a=1 and b=3, so the new number is 2, and the whiteboard contains [2].

Rất dễ dàng nhận thấy sau n−1 lần thực hiện, số còn lại là số duy nhất. Mục tiêu của bạn là tối thiểu nó.

**Input**

Dòng đầu tiên chưa một số nguyên  t (1≤t≤1000) — Số lượng trường hợp xảy ra.

Trên mỗi dòng chứa một trường hợp chứa 1 số nguyên n (2≤n≤2⋅105) — n là số nguyên dương đại diện cho dãy số 1… n.

Đảm bảo rằng tổng 2 số lớn nhất của n không vượt quá 2\*10^5

**Output**

Trong mỗi trường hợp, Dòng đầu tiên in số nguyên nhỏ nhất còn lại sau cùng sau  n−1 lần thực hiện. Mỗi dòng tiếp theo n−1 dòng. Hiển thị 2 số nguyên  a và b được chọn để xóa trong mỗi lần thực hiện.

**Example**

**input**

1

4

**output**

2

2 4

3 3

3 1

Problem. Three Indices

time limit per test

2 seconds

memory limit per test

256 megabytes

input

standard input

output

standard output

Bạn được đưa cho p1,p2,…,pn hoán vị. Lưu ý, dãy n được gọi là hoán vị khi mỗi phần tử xuất hiện duy nhất 1 lần.

Tìm ba vị trí i, j và k như sau:

* 1≤i<j<k≤n;
* pi<pj và pj>pk.

Hoặc ba vị trí đó không tồn tại.

**Input**

Dòng đầu tiên chưa 1 số nguyên T (1≤T≤200) — số lượng test cases.

Hai dòng tiếp theo chứa test cases . Trong đó dòng đầu tiên chưa số lượng dãy số nguyên  n (3≤n≤1000) — Độ dài của dãy hoán vị p.

Dòng thứ hai chứa n số nguyên p1,p2,…,pn (1≤pi≤n; pi≠pj if i≠j) — hoán vị pi.

**Output**

Mỗi một test cases:

* Nếu tồn tại các vị trí i, j và k, hiện thị YES (viết hoa) và các chỉ số tương ứng.;
* Nếu không tồn tại thì hiện thị NO (viết hoa).

Nếu có nhiều đáp án, in ra 1 trong số đó.

Example

**input**

3

4

2 1 4 3

6

4 6 1 2 5 3

5

5 3 1 2 4

**output**

YES

2 3 4

YES

3 5 6

NO

**Problem. ADD STRINGS**

Cho 2 số không âm, num1 and num2 đại diện dưới 2 chuỗi string, Trả về tổng của  num1 và num2 ở dạng string.

Bài này yêu cầu không sử dụng thư viện sẵn có, hay các hàm có liên quan để

convert str → int.

**Example 1:**

**Input:** num1 = "11", num2 = "123"

**Output:** "134"

**Example 2:**

**Input:** num1 = "456", num2 = "77"

**Output:** "533"

**Example 3:**

**Input:** num1 = "0", num2 = "0"

**Output:** "0"

**Constraints:**

* 1 <= num1.length, num2.length <= 104
* num1 và num2 chỉ toàn các ký tự số.
* num1 và num2 Không có bất kỳ số 0 nào ở đầu ngoại trừ chỉ có 0.