## Ví dụ

Input

5

1 2 3 4 5

* Ở index 1: từ [1..1] chỉ có 1 phần tử ⇒ Xuất **-1**
* Ở index 2: từ [1..2] chỉ có 2 phần tử ⇒ Xuất **-1**
* Ở index 3: từ [1..3] có 3 phần tử: 1, 2, 3 cũng đồng thời là 3 số lớn nhất ⇒ Xuất tích **1\*2\*3 = 6**
* Ở index 4: từ [1..4] có 4 phần tử: 1, 2, 3, 4 có 3 số lớn nhất là 2, 3, 4 ⇒ Xuất tích **2\*3\*4 = 24**
* Ở index 5: từ [1..5] có 5 phần tử: 1, 2, 3, 4, 5 có 3 số lớn nhất là 3, 4, 5 ⇒ Xuất tích **3\*4\*5 = 60**

## Nhận xét

* Ở mỗi index i, mình phải lấy ra được 3 số lớn nhất từ [1..i] ⇒ Áp dụng max heap để giải

## Thuật toán

B1: Đọc dữ liệu, khởi tạo 1 max heap rỗng

B2: Với mỗi i từ 0 đến n - 1:

* Thêm phần tử thứ i vào heap
* Nếu i < 2: xuất -1. Ngược lại:
  + Lần lượt lấy số lớn nhất, lớn nhì, lớn ba từ heap
  + Xuất tích của 3 số đó
  + Thêm lại 3 số đó vào heap

## Mã giả:

main():

read(n)

read(a)

heap = max\_heap()

for i = 0 to n - 1:

heap.push(a[i])

if i < 2:

print(-1)

else:

first = heap.pop()

second = heap.pop()

third = heap.pop()

print(first \* second \*third)

heap.push(first)

heap.push(second)

heap.push(third)

Độ phức tạp: O(nlogn)