**MINQUERY**

Cho mảng gồm **n** phần tử. Có **q** truy vấn (**l**, **r**), ta cần tìm phần tử nhỏ nhất trong các phần tử từ **l** đến **r**.

**INPUT**

Dòng đầu tiên gồm hai số **n**, **q**, tức số phần tử của mảng và số truy vấn.

Dòng thứ hai gồm **n** phần tử của mảng.

**q** dòng sau, mỗi dòng gồm hai số **l**, **r** ám chỉ một truy vấn.

**n, q** <= 300000

Các phần tử của mảng <= 1000000000

**OUTPUT**

Với mỗi truy vấn, in ra kết quả cần tìm trên từng dòng.

|  |  |
| --- | --- |
| Sample input | Sample output |
| 3 3  1 2 3  1 2  2 3  1 3 | 1  2  1 |

**GCDLR**

Cho mảng gồm **n** phần tử. Có **q** truy vấn (**l**, **r**), ta cần tìm ước chung lớn nhất của phần tử từ **l** đến **r**.

**INPUT**

Dòng đầu tiên gồm hai số **n**, **q**, tức số phần tử của mảng và số truy vấn.

Dòng thứ hai gồm **n** phần tử của mảng.

**q** dòng sau, mỗi dòng gồm hai số **l**, **r** ám chỉ một truy vấn.

**n, q** <= 300000

Các phần tử của mảng <= 1000000000

**OUTPUT**

Với mỗi truy vấn, in ra kết quả cần tìm trên từng dòng.

|  |  |
| --- | --- |
| Sample input | Sample output |
| 3 3  1 2 4  1 2  2 3  1 3 | 1  2  1 |

**MODULO**

Cho mảng gồm **n** phần tử. Có **q** truy vấn (**l**, **r**), ta cần tìm modulo k lớn nhất có thể sao cho các số từ l đến r đều đồng dư theo modulo k.

**INPUT**

Dòng đầu tiên gồm hai số **n**, **q**, tức số phần tử của mảng và số truy vấn.

Dòng thứ hai gồm **n** phần tử của mảng.

**q** dòng sau, mỗi dòng gồm hai số **l**, **r** ám chỉ một truy vấn.

**n, q** <= 300000

Các phần tử của mảng <= 1000000000

**OUTPUT**

Với mỗi truy vấn, nếu không thể in được k, in ra -1, ngược lại in ra kết quả cần tìm trên từng dòng.

|  |  |
| --- | --- |
| Sample input | Sample output |
| 3 3  2 3 5  1 2  2 3  1 3 | 1  2  1 |

**SUMDIFF**

Cho mảng gồm **n** phần tử. Định nghĩa giá trị của một đoạn con liên tiếp của mảng là hiệu giữa phần tử lớn nhất và phần tử nhỏ nhất của đoạn con đấy.

Ta cần tính tổng của giá trị của tất cả đoạn con liên tiếp của mảng.

**INPUT**

Dòng đầu tiên gồm số **n**, tức số phần tử của mảng.

n dòng tiếp theo, mỗi dòng gồm một phần tử của mảng.

**n** <= 300000

Các phần tử của mảng <= 100000000

**OUTPUT**

In ra kết quả cần tìm.

|  |  |
| --- | --- |
| Sample input | Sample output |
| 4  3  1  7  2 | 31 |

**SORT**

Khánh có một mảng gồm các lũy thừa của hai và các phần tử thì đôi một phân biệt.

Để tiết kiệm bộ nhớ, Khánh chỉ lưu số mũ của chúng, tức nếu có phần tử có giá trị 16 thì sẽ lưu thành 4.

Giá trị của một đoạn con liên tiếp là tổng của các phần tử trong đoạn con đấy.

Cho một tập các đoạn con, ta cần sắp xếp lại tập đó theo thứ tự từ bé đến lớn (giá trị nhỏ hơn thì đứng trước), nếu hai tập cùng giá trị thì sort theo id.

**INPUT**

Dòng đầu tiên gồm số **n**, tức kích thước mảng.

Dòng sau gồm n số mũ của các phần tử của mảng.

Dòng sau gốm số m tức số đoạn con

m dòng sau, mỗi dòng gồm hai số l, r ám chỉ đoạn con thứ i.

n , m <= 200000

Các phần tử của mảng <= 1000000000

**OUTPUT**

In ra các id của các đoạn con sau khi sắp xếp.

|  |  |
| --- | --- |
| Sample input | Sample output |
| 5 5  3 4 2 6 7  1 4  5 5  5 5  1 5  1 2 | 5 1 2 3 4 |

Giải thích: Đoạn một có giá trị là 8 + 16 + 4 + 64, đoạn 2, 3 là 128, đoạn 4 là 8 + 16 + 4 + 64 + 128 và đoạn 5 là 8 + 16