C. Little Girl and Maximum Sum

time limit per test: 2 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

The little girl loves the problems on array queries very much.

One day she came across a rather well-known problem: you've got an array of n elements (the elements of the array are indexed starting from 1); also, there are q queries, each one is defined by a pair of integers l_i, r_i $(1 \le l_i \le r_i \le n)$. You need to find for each query the sum of elements of the array with indexes from l_i to r_i, inclusive.

The little girl found the problem rather boring. She decided to reorder the array elements before replying to the queries in a way that makes the sum of query replies maximum possible. Your task is to find the value of this maximum sum.

Input

The first line contains two space-separated integers n $(1 \le n \le 2 \cdot 10^5)$ and q $(1 \le q \le 2 \cdot 10^5)$ — the number of elements in the array and the number of queries, correspondingly.

The next line contains n space-separated integers a_i $(1 \le a_i \le 2 \cdot 10^5)$ — the array elements.

Each of the following q lines contains two space-separated integers l_i and r_i $(1 \le l_i \le r_i \le n)$ — the i-th query.

Output

In a single line print a single integer — the maximum sum of query replies after the array elements are reordered.

Please, do not use the %lld specifier to read or write 64-bit integers in C++. It is preferred to use the cin, cout streams or the %I64d specifier.

Examples

input

```
3 3
5 3 2
1 2
2 3
1 3
```

output

```
25
```

input

```
5 3
5 2 4 1 3
1 5
2 3
2 3
```

output

```
33
```

→ **Attention**

Package for this problem was not updated by the problem writer or Codeforces administration after we've upgraded the judging servers. To adjust the time limit constraint, solution execution time will be multiplied by 2. For example, if your solution works for 400 ms on judging servers, then value 800 ms will be displayed and used to determine the verdict.

Codeforces Round #169 (Div. 2)

Finished

→ **Virtual participation**

Virtual contest is a way to take part in past contest, as close as possible to participation on time. It is supported only ACM-ICPC mode for virtual contests. If you've seen these problems, a virtual contest is not for you - solve these problems in the archive. If you just want to solve some problem from a contest, a virtual contest is not for you - solve this problem in the archive. Never use someone else's code, read the tutorials or communicate with other person during a virtual contest.

Start virtual contest

→ **Problem tags**

data structures | implementation | sortings

No tag edit access

→ **Contest materials**

- Announcement
- Tutorial #1
- Tutorial #2