

Contest Duration: 2025-11-01(Sat) 23:00 (<http://www.timeanddate.com/worldclock/fixedtime.html?iso=20251101T2100&p1=248>) - 2025-11-02(Sun) 00:40 (<http://www.timeanddate.com/worldclock/fixedtime.html?iso=20251101T2240&p1=248>) (local time) (100 minutes)

[Back to Home \(/home\)](/home)

[🏠 Top \(/contests/abc430\)](/contests/abc430)

[📋 Tasks \(/contests/abc430/tasks\)](/contests/abc430/tasks)

[❓ Clarifications \(/contests/abc430/clarifications\)](/contests/abc430/clarifications)

[📊 Results ▼](#)

[🏆 Standings \(/contests/abc430/standings\)](/contests/abc430/standings)

[🏆 Virtual Standings \(/contests/abc430/standings/virtual\)](/contests/abc430/standings/virtual)

[📖 Editorial \(/contests/abc430/editorial\)](/contests/abc430/editorial)

[💬 Discuss \(https://codeforces.com/blog/entry/147960\)](https://codeforces.com/blog/entry/147960)



G - Range Set Modifying Query

[Editorial \(/contests/abc430/tasks/abc430_g/editorial\)](/contests/abc430/tasks/abc430_g/editorial)



Time Limit: 4 sec / Memory Limit: 1024 MiB

Score : 625 points

Problem Statement

There are N sets S_1, \dots, S_N . Initially, all sets are empty.

You are given Q queries in the following formats. Process them in order.

- Type 1: Given as 1 L R x . For each S_i satisfying $L \leq i \leq R$, add x .
- Type 2: Given as 2 L R x . For each S_i satisfying $L \leq i \leq R$, remove x .
- Type 3: Given as 3 L R . Find the maximum number of elements among S_i satisfying $L \leq i \leq R$, and the number of sets that achieve this maximum.

Constraints

- $1 \leq N \leq 3 \times 10^5$
- $1 \leq Q \leq 3 \times 10^5$
- For each query, $1 \leq L \leq R \leq N$.
- For type 1, 2 queries, $1 \leq x \leq 60$.
- All input values are integers.

Input

The input is given from Standard Input in the following format:

```
 $N$   $Q$   
query1  
⋮  
query $Q$ 
```

Here, query _{i} represents the i -th query, and each is given in one of the following formats as shown in the problem statement:

```
1  $L$   $R$   $x$ 
```

```
2  $L$   $R$   $x$ 
```

```
3  $L$   $R$ 
```

Output

Let q be the number of type 3 queries, and print q lines.

The i -th line should contain x , y separated by a space, where x is the maximum number of elements and y is the number of sets that achieve this maximum for the i -th type 3 query.

Sample Input 1

[Copy](#)

```
4 7  
1 1 2 10  
1 2 4 20  
3 1 3  
2 1 2 20  
1 2 3 10  
3 1 2  
3 1 4
```

[Copy](#)

Sample Output 1

[Copy](#)

```
2 1  
1 2  
2 1
```

[Copy](#)

- Initially $(S_1, S_2, S_3, S_4) = (\{\}, \{\}, \{\}, \{\})$.

2026-01-02 (Fri)
05:32:44 +11:00

- The 1-st query adds 10 to S_1, S_2 , resulting in $(S_1, S_2, S_3, S_4) = (\{10\}, \{10\}, \{\}, \{\})$.
- The 2-nd query adds 20 to S_2, S_3, S_4 , resulting in $(S_1, S_2, S_3, S_4) = (\{10\}, \{10, 20\}, \{20\}, \{20\})$.
- For the 3-rd query, the maximum number of elements among S_1, S_2, S_3 is 2 achieved by S_2 , so print 2 1.
- The 4-th query removes 20 from S_1, S_2 , resulting in $(S_1, S_2, S_3, S_4) = (\{10\}, \{10\}, \{20\}, \{20\})$.
- The 5-th query adds 10 to S_2, S_3 , resulting in $(S_1, S_2, S_3, S_4) = (\{10\}, \{10\}, \{10, 20\}, \{20\})$.
- For the 6-th query, the maximum number of elements among S_1, S_2 is 1 achieved by S_1, S_2 , so print 1 2.
- For the 7-th query, the maximum number of elements among S_1, S_2, S_3, S_4 is 2 achieved by S_3 , so print 2 1.

'#telegram)

url=https%3A%2F%2Fatcoder.jp%2Fcontests%2Fabc430%2Ftasks%2Fabc430_g%3Flang%3Den&title=G%20-20Query)

[Rule \(/contests/abc430/rules\)](/contests/abc430/rules) [Glossary \(/contests/abc430/glossary\)](/contests/abc430/glossary)

[Terms of service \(/tos\)](/tos) [Privacy Policy \(/privacy\)](/privacy) [Information Protection Policy \(/personal\)](/personal) [Company \(/company\)](/company)
[FAQ \(/faq\)](/faq) [Contact \(/contact\)](/contact)

Copyright Since 2012 ©AtCoder Inc. (<http://atcoder.co.jp>) All rights reserved.