

Contest Duration: 2025-08-02(Sat) 22:00 (<http://www.timeanddate.com/worldclock/fixedtime.html?iso=20250802T2100&p1=248>) - 2025-08-02(Sat) 23:40 (<http://www.timeanddate.com/worldclock/fixedtime.html?iso=20250802T2240&p1=248>) (local time) (100 minutes)

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

[🏠 Top \(/contests/abc417\)](/contests/abc417)

[📋 Tasks \(/contests/abc417/tasks\)](/contests/abc417/tasks)

[❓ Clarifications \(/contests/abc417/clarifications\)](/contests/abc417/clarifications)

[📊 Results ▼](#)

[🏆 Standings \(/contests/abc417/standings\)](/contests/abc417/standings)

[🏆 Virtual Standings \(/contests/abc417/standings/virtual\)](/contests/abc417/standings/virtual)

[📖 Editorial \(/contests/abc417/editorial\)](/contests/abc417/editorial)

[💬 Discuss \(https://codeforces.com/blog/entry/145214\)](https://codeforces.com/blog/entry/145214)



## G - Binary Cat

[Editorial \(/contests/abc417/tasks/abc417\\_g/editorial\)](/contests/abc417/tasks/abc417_g/editorial)



Time Limit: 6 sec / Memory Limit: 1024 MiB

Score : 625 points

### Problem Statement

Define strings  $S_0$  and  $S_1$  as  $S_0 = \emptyset$  and  $S_1 = 1$ .

You are given  $Q$  queries, so process them in order.

In the  $i$ -th query ( $1 \leq i \leq Q$ ), you are given a triple of integers  $(L_i, R_i, X_i)$ .

Let  $S_{i+1}$  be the string obtained by concatenating  $S_{L_i}$  and  $S_{R_i}$  in this order. Then, find the  $X_i$ -th character of  $S_{i+1}$ .

It is guaranteed that  $X_i$  is at most the length of  $S_{i+1}$ .

### Constraints

- $1 \leq Q \leq 5 \times 10^5$
- $0 \leq L_i, R_i \leq i$
- $1 \leq X_i \leq 10^{18}$
- $X_i$  is at most the length of  $S_{i+1}$ .
- All input values are integers.

2026-01-02 (Fri)

05:28:09 +11:00

# Input

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

```
Q
L1 R1 X1
L2 R2 X2
⋮
LQ RQ XQ
```

# Output

Output  $Q$  lines. The  $i$ -th line ( $1 \leq i \leq Q$ ) should contain the answer to the  $i$ -th query.

## Sample Input 1

[Copy](#)

```
7
0 1 1
0 0 2
1 1 1
2 3 2
2 4 3
5 4 2
6 7 6
```

[Copy](#)

## Sample Output 1

[Copy](#)

```
0
0
1
1
1
1
1
```

[Copy](#)

Each query is processed as follows:

- In the first query, concatenate  $S_0 = 0$  and  $S_1 = 1$  to get  $S_2 = 01$ . The first character of  $S_2$  is 0, so output 0 on the first line.
- In the second query, concatenate  $S_0 = 0$  and  $S_0 = 0$  to get  $S_3 = 00$ . The second character of  $S_3$  is 0, so output 0 on the second line.
- In the third query, concatenate  $S_1 = 1$  and  $S_1 = 1$  to get  $S_4 = 11$ . The first character of  $S_4$  is 1, so output 1 on the third line.

- In the fourth query, concatenate  $S_2 = 01$  and  $S_3 = 00$  to get  $S_5 = 0100$ . The second character of  $S_5$  is 1, so output 1 on the fourth line.
- In the fifth query, concatenate  $S_2 = 01$  and  $S_4 = 11$  to get  $S_6 = 0111$ . The third character of  $S_6$  is 1, so output 1 on the fifth line.
- In the sixth query, concatenate  $S_5 = 0100$  and  $S_4 = 11$  to get  $S_7 = 010011$ . The second character of  $S_7$  is 1, so output 1 on the sixth line.
- In the seventh query, concatenate  $S_6 = 0111$  and  $S_7 = 010011$  to get  $S_8 = 0111010011$ . The sixth character of  $S_8$  is 1, so output 1 on the seventh line.

---

'#telegram)

url=https%3A%2F%2Fatcoder.jp%2Fcontests%2Fabc417%2Ftasks%2Fabc417\_g%3Flang%3Den&title=G%20-

---

[Rule \(/contests/abc417/rules\)](/contests/abc417/rules) [Glossary \(/contests/abc417/glossary\)](/contests/abc417/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.