Mr Snake's Data Structures Lecture (Output Only)

It is the middle of December Camp and you are currently in the ANU computer labs listening to Mr Snake's Data Structure's II Lecture. You are very tired as you attempted to catch the local ANU ducks with bed sheets the entire night prior. As such, you doze off for a few moments. However, Mr Snake does not approve. While you are sleeping, he takes your USB and encrypts all your *secret files*. He does so by representing each file as an array A of **8-bit signed integers** and then performs several range updates and queries on the array. As Mr Snake does not want to break the AMT code of conduct (as if he hasn't already), he gives you instructions you can use to decypt each of the files. More precisely, for each encrypted file, there is an associated list of Q operations which you may use to decrypt it. The operations are as follows:

#	*Operation*	*Type*	*Input Format*	*Description*
1	Set	*Update*	1 l r x	Set all values in the range $[l, r]$ to x.
2	Add	*Update*	2 l r x	Add x to all values in the range $[l, r]$.
3	Sum	*Query*	3 l r	Find the sum of all elements in the range $[l, r]$.
4	Negate	*Update*	4 l r	Negate all values in the range $[l, r]$.
5	Longest Maximum Repetition	*Query*	5 l r	Find the longest contiguous repetition of the maximum element in the range $[l, r]$.
6	Mex Threshold	*Query*	6 l x	Find the leftmost r, $r \geq l$ such that all values ≥ -128 and $< x$ exist in the range $[l, r]$ at least once.
7	2Sum	*Query*	7 l r x	Does there exist a pair $(i, j), l \leq i < j \leq r$ such that $A[i] + A[j] = x$? If so, find the pair which maximises $abs(A[i] - A[j])$. This difference between the two values will be the result of the query. If no pair exists, the result is -1.
8	Erase	*Update*	8 l r	Erase the range $[l, r]$, reducing the size of A by $r - l + 1$.

#	*Operation*	*Type*	*Input Format*	*Description*
9	Shift	*Update*	9 l r x	Shift the position of the subsequence $[l, r]$ by x such that the subsequence ends up at position $[l + x, r + x]$.
10	Expand	*Update*	10 l r x	Take every value in the range $[l, r]$ and replace it with a contiguous block of x copies of itself.

You cannot stand having your *secret files* hidden from you. You absolutely must decrypt them!

Input

There are 5 test cases numbered 1 - 5. For the i-th test case, there will be two input files, `i.encrypted.ext` (where .ext is some file extension) and `i.op.txt`.

The first file is a binary blob representing the initial array A. Each byte is to be interpretted as an 8-bit signed integer.

The second file contains two integers N (The initial size of A) and Q (the number of operations to be performed on A). Q lines follow. The j-th such line describes the j-th operation to be perfomed. The input format of each operation is described above and all indexes into the array are 0-based.

To ensure you perform all the queries and updates as intended, Mr Snake has xor'd every integer in the list of operations with the answer of the most recent query. If there was no last query, the interger is xor'd with 0.

See included template.cpp for an implementation of I/O in c++

Output

For each test case i, you must produce an output file `i.out.txt` containing a single integer - the result of the last query operation performed in the i-th test case. **You do not have to submit the decrypted file.**

Sample

You can find the sample input included with the other 5 cases. A brief expanation can be found below:

#	*A*	*Operation*	*Result*
0	$[-65, -128, -112, -127, -121, 92, 32, 99, 34, -56, 89, 94, 56]$	-	_

#	*A*	*Operation*	*Result*
1	$-$	Mex 0 -126	3
2	$[-65, -128, -112, -90, -121, 92, 32, 99, 34, -56, 89, 94, 56]$	Set 3 3 -90	–
3	$[-65, -112, -96, -74, -105, 108, 32, 99, 34, -56, 89, 94, 56]$	Add 1 5 16	–
4	$[-65, -112, -96, -74, -105, 108, 32, 99, 108, 108, 108, 108, 56]$	Set 8 11 108	–
5	$[-65, -112, -96, -74, -105, 108, 32, 99, 108, 67, 108, 108, 56]$	Set 9 9 67	_
6	$-$	Repetition 0 12	2
7	$[-65, -112, -96, -74, -105, 108, 32, 99, 108, 64, 105, 108, 56]$	Add 9 10 -3	–
8	$[-65, -112, -96, -74, -105, 108, 32, 99, 114, 70, 111, 108, 56]$	Add 8 10 6	–
9	$-$	Sum 2 5	-167
10	$[65, 112, 96, 74, 105, 108, 32, 99, 114, 70, 111, 108, 56]$	Negate 0 4	–
11	$[65, 112, 96, 74, 99, 114, 105, 108, 32, 70, 111, 108, 56]$	Shift 7 8 -3	–
12	$-$	2Sum 1 10 210	18
13	$[65, 112, 114, 105, 108, 32, 70, 111, 108, 56]$	Erase 2 4	_
14	$[65, 112, 114, 105, 108, 32, 70, 111, 111, 108, 108, 56]$	Expand 7 8 2	–
15	$[65, 112, 114, 105, 108, 32, 70, 111, 111, 108, 115, 63]$	Add 10 11 7	–

Result of last query: `18`

Decrypted File: `April Fools?`

Constraints

- You are guaranteed all operations performed on a range will be within the bounds of A.
- No update will cause any value to exceed the bounds of a signed 8-bit integer.
- There will always be some r that satisfies the mex query.
- The sum of any subset of A will be within the range $[-10^{18}, 10^{18}]$ at all times.
- The expand operation will not cause the size of A to become greater than 10^{18}.
- When expanding, $0 \leq x \leq 10^{18}$.

Subtask	***Initial N***	***Operations***	***Q***
Sample	13	$1 - 10$	15
1	106078	$1 - 3$	521083
2	15760202	$1 - 5$	24096024
3	5280892	$1 - 7$	18164629
4	14120429	$1 - 9$	17351068
5	10000000	$1 - 10$	23535552