Pointer Sequence

Input file:	`standard input`
Output file:	`standard output`
Time limit:	3 seconds
Memory limit:	512 megabytes

Give you a pointer sequence of length n, which is initialized as follows:

```
using integer = long long;
integer *A[MAXN]; //MAXN is a constant larger than n
for (int i = 1; i <= n; ++i)
    A[i] = new integer(0);
```

You need to maintain five types of operations, as shown below:

- **1 L1 R1 L2 R2** $(R1 - L1 = R2 - L2, [L1, R1]$ and $[L2, R2]$ are disjoint intervals falling in $[1, n])$
- **2 L R k b** $(1 \le L \le R \le n$ and $0 \le k, b \le 10^4)$
- **3 L R v** $(0 \le L \le R \le 10^9$ and $0 \le v \le 10^9)$
- **4 p v** $(1 \le p \le n$ and $0 \le v \le 10^9)$
- **5 p** $(1 \le p \le n)$

For each type of the operations, you need to efficiently handle the corresponding procedure:

Operation 1

```
for (int i = 0; i <= R1 - L1; ++i) //copy A[L1, R1] to A[L2, R2]
    A[L2 + i] = A[L1 + i];
```

Operation 2

```
for (integer i = 0; i <= R - L; ++i) //assign an arithmetic sequence to A[L, R]
    A[L + i] = new integer(k * i + b);
```

Operation 3

```
for (int i = 1; i <= n; ++i) //change the value in [L, R] to v
    if (*A[i] >= L && *A[i] <= R)
        *A[i] = v;
```

Operation 4

```
*A[p] = v; //single point modification
```

Operation 5

```
cout << *A[p] << endl; //Output the value of A[pos]
```

Input

The first line of the input contains two integers $n(1 \le n \le 10^6)$ and $m(1 \le m \le 10^5)$, indicating the length of the sequence and the number of operations. Then each of the next m lines contains a query in the form described above.

Output

For all operations of type 5, output an integer indicating the answer.

Example

standard input	standard output
5 10 5 1 2 1 5 1 1 1 1 2 3 4 4 1 8 5 3 3 7 8 2 3 2 5 6 5 1 5 2 5 5	0 8 6 6 6