New Year's '19 P6 - Buffer Zone

Time limit: 1.0s **Memory limit:** 256M

Having finished a stressful school term, you finally arrive home, kick back, and relax! It is at this time that your friend calls you, and tells you about a new algorithm he's found!

This algorithm allows for sharing the data of two stacks within a single array of length W. Any data belonging to stack A must be placed at the beginning of the array, starting at index 0. Any data belonging to stack B must be placed at the end of the array, counting downwards from index $W-1$. To prevent the stacks from colliding with each other, the algorithm uses a buffer zone of some width $0 \leq w < W$.

This buffer zone is represented by two integers l and r, with $r = l + w$. These values l and r are permitted to be any integers, not necessarily nonnegative or less than W. The buffer zone can only be changed by the operation of incrementing l and r together, or decrementing l and r together. If an index i of the array is to be associated with stack A, then the condition $i < l$ must hold. If an index i of the array is to be associated with stack B, then the condition $r \leq i$ must hold.

There will be $2N$ queries where one of the stacks temporarily requests use of some number of array indices. This may require performing operations to change the buffer zone. After each operation, the stack will be cleared and no query immediately requires any array indices. The stacks responsible for the queries will alternate between A and B in the order A, B, A, B, and so on. The number of requested indices for each query will be given by the integers a_1, b_1, a_2, b_2, and so on.

For a given width w, you want to minimize the number of operations performed on the buffer zone to keep it in a valid state throughout the $2N$ queries. The initial state of the buffer zone can be determined freely without any extra cost.

However, he needs your help to find the minimum number of operations for every $0 \leq w < W$. You'd help your friend, even over the break...right?

Constraints

$1 \leq N \leq 10^5$

$1 \leq W \leq 10^6$

$1 \leq a_i, b_i \leq W$

Input Specification

The first line contains two space-separated integers N and W.

N lines follow; the i-th one contains two space-separated integers a_i and b_i.

Output Specification

Output W lines, each containing a single integer: the minimum number of operations for a buffer width of w. Output the lines for $0 \leq w < W$ in increasing order of w.

Sample Input 1

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

Output for Sample Input 1

```
0
1
3
8
13
```

Explanation for Sample Output 1

Consider $w = 2$, for which the answer is 3. To achieve this, use an initial buffer state of $(l, r) = (2, 4)$. This state is valid for the first two queries. Before the third query, perform one operation to change the state to $(3, 5)$. Before the fourth query, perform two operations to change the state to $(1, 3)$. This state remains valid for the remaining queries.