Kick Start 2022 - Round B

Unlock the Padlock

Problem

Imagine you have a padlock, which is a combination lock consisting of $\mathbf{N}$ dials, set initially to a random combination. The dials of the padlock are of size $\mathbf{D}$, which means that they can have values between 0 and $\mathbf{D} - 1$, inclusive, and can be rotated upwards or downwards. They are also ordered from left to right, with the leftmost and rightmost dials at positions 1 and $\mathbf{N}$, respectively. The padlock can be unlocked by setting the values of all its dials to 0.

You can perform zero or more operations of this kind:

- Pick any range $[l, r]$ such that $1 \leq l \leq r \leq \mathbf{N}$ and rotate all the dials in $[l, r]$ together, upwards or downwards. Rotating up increases the value of each dial in the range $[l, r]$ by 1, and rotating down decreases its value by 1. Note that a dial with value $\mathbf{D} - 1$ becomes 0 when increased (rotated up) and a dial with value 0 becomes $\mathbf{D} - 1$ when decreased (rotated down).

The series of operations must satisfy the following condition:

- The range $[l_{i-1}, r_{i-1}]$ chosen in the $(i - 1)$-th operation needs to be completely contained within the range $[l_i, r_i]$ chosen in the i-th operation; that is, $l_i \leq l_{i-1} \leq r_{i-1} \leq r_i$. The initial range ($[l_1, r_1]$) can be chosen arbitrarily.

Example of a valid sequence of operations to unlock a padlock with initial combination $[1, 1, 2, 2, 3, 3]$:

1. Rotate range $[5, 6]$ downwards.
2. Rotate range $[3, 6]$ downwards.
3. Rotate range $[1, 6]$ downwards.

The following are some operations that cannot be performed:

1. Rotating range $[1, 4]$ after $[6, 9]$, because $[6, 9]$ is not completely contained in $[1, 4]$ (does not satisfy $r_{i-1} \leq r_i$ where $r_{i-1} = 9$ and $r_i = 4$).
2. Rotating range $[3, 6]$ after $[2, 7]$.

The goal for you is to output the *minimum* number of valid operations needed to make all dials in the padlock set to 0.

Input

The first line of the input contains the number of test cases, $\mathbf{T}$. $\mathbf{T}$ test cases follow.

Each test case consists of two lines.

The first line of each test case contains two integers $\mathbf{N}$ and $\mathbf{D}$, representing the number of dials in the padlock and the size of the dials, respectively.

The second line of each test case contains $\mathbf{N}$ integers $\mathbf{V_1}, \mathbf{V_2}, \ldots, \mathbf{V_N}$, where the i-th integer represents the value of the i-th dial in the initial combination of the padlock.

Output

For each test case, output one line containing `Case #`x`: `y, where x is the test case number (starting from 1) and y is the minimum number of operations needed to unlock the padlock as described in the statement.

Limits

Time limit: 30 seconds.
Memory limit: 1 GB.
$1 \leq \mathbf{T} \leq 100$.
$0 \leq \mathbf{V_i} \leq \mathbf{D} - 1$, for all i.

Test Set 1

$1 \leq \mathbf{N} \leq 40$.
$\mathbf{D} = 2$.

Test Set 2

$1 \leq \mathbf{N} \leq 40$.
$2 \leq \mathbf{D} \leq 10$.

Test Set 3

$1 \leq \mathbf{N} \leq 400$.
$2 \leq \mathbf{D} \leq 10^9$.

Sample

Note: there are additional samples that are not run on submissions down below.

Sample Input

```
2
6 2
1 1 0 1 0 1
6 2
0 1 0 0 1 1
```

Sample Output

```
Case #1: 3
Case #2: 2
```

In Sample Case #1, the minimum number of operations needed to unlock the padlock is 3. We can unlock it using the following operations:

1. Rotate range $[4, 4]$ downwards.
2. Rotate range $[3, 5]$ downwards.
3. Rotate range $[1, 6]$ downwards.

In Sample Case #2, the minimum number of operations needed to unlock the padlock is 2. We can unlock it using the following operations:

1. Rotate range $[3, 4]$ upwards.
2. Rotate range $[2, 6]$ downwards.

Additional Sample - Test Set 2

The following additional sample fits the limits of Test Set 2. It will not be run against your submitted solutions.

Sample Input

```
2
6 10
1 1 2 2 3 3
6 10
1 1 9 9 1 1
```

Sample Output

```
Case #1: 3
Case #2: 3
```

In Sample Case #1, the minimum number of operations needed to unlock the padlock is 3. We can unlock it using the following operations:

1. Rotate range $[5, 6]$ downwards.
2. Rotate range $[3, 6]$ downwards.
3. Rotate range $[1, 6]$ downwards.

In Sample Case #2, the minimum number of operations needed to unlock the padlock is 3. We can unlock it using the following operations:

1. Rotate range $[3, 4]$ upwards.
2. Rotate range $[3, 4]$ upwards.
3. Rotate range $[1, 6]$ downwards.