Lab09

Hackerrank practices

Missions:

1. Understand and use Python Tuple and Dictionary, and learn some new advanced function. correctly.
2. Complete some tough Hackerrank questions.

Overview

In this week, we will get our hands dirty with some tough Hackerrank practices, some of which can be solved using Python tuple and dictionary. Some questions might need some Python functions not covered in the lecture to produce shorter code. The instructor will demonstrate some of those as the questions are discussed.

There are five Hackerrank problems selected. Here are the names of the problems:

1. The Hurdle Race
2. Apple and Orange
3. Divisible Sum Pairs
4. The Birthday Bar
5. Migratory Birds

Check the next pages for the problems.

The Hurdle Race

Dan is playing a video game in which his character competes in a hurdle race. Hurdles are of varying heights, and Dan has a maximum height he can jump. There is a magic potion he can take that will increase his maximum height by 1 unit for each dose. How many doses of the potion must he take to be able to jump all of the hurdles.

Given an array of hurdle heights $height$, and an initial maximum height Dan can jump, k, determine the minimum number of doses Dan must take to be able to clear all the hurdles in the race.

For example, if $height = [1, 2, 3, 3, 2]$ and Dan can jump 1 unit high naturally, he must take $3 - 1 = 2$ doses of potion to be able to jump all of the hurdles.

Input Format

Complete the function $hurdleRace$ in the editor below. The code stub reads the input at passes it to the function. Inputs are in the following format:

The first line contains two space-separated integers n and k, the number of hurdles and the maximum height Dan can jump naturally.
The second line contains n space-separated integers $height[i]$ where $0 \le i < n$.

Constraints

- $1 \le n, k \le 100$
- $1 \le height[i] \le 100$

Output Format

Print an integer denoting the minimum doses of magic potion Dan must drink to complete the hurdle race.

Sample Input 0

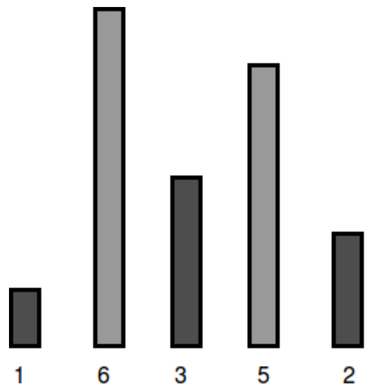
```
5 4
1 6 3 5 2
```

Sample Output 0

```
2
```

Explanation 0

Dan's character can jump a maximum of $k = 4$ units, but the tallest hurdle has a height of $h_1 = 6$:



To be able to jump all the hurdles, Dan must drink $6 - 4 = 2$ doses.

Sample Input 1

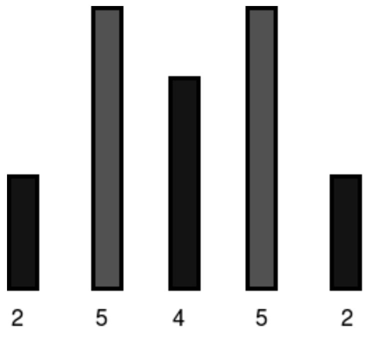
```
5 7
2 5 4 5 2
```

Sample Output 1

```
0
```

Explanation 1

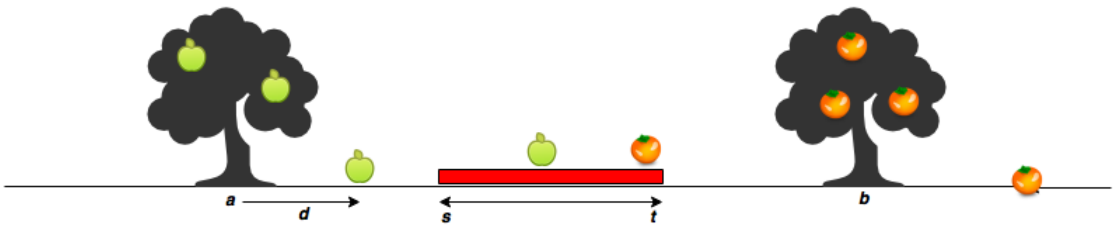
Dan's character can jump a maximum of $k = 7$ units, which is enough to cross all the hurdles:



Because he can already jump all the hurdles, Dan needs to drink 0 doses.

Apple and Orange

Sam's house has an apple tree and an orange tree that yield an abundance of fruit. In the diagram below, the red region denotes his house, where s is the start point, and t is the endpoint. The apple tree is to the left of his house, and the orange tree is to its right. You can assume the trees are located on a single point, where the apple tree is at point a, and the orange tree is at point b.



When a fruit falls from its tree, it lands d units of distance from its tree of origin along the x-axis. A negative value of d means the fruit fell d units to the tree's left, and a positive value of d means it falls d units to the tree's right.

Complete the function `countApplesAndOranges`,

where,

$start$ Starting point of Sam's house location.
end Ending location of Sam's house location.
loc_a Location of the Apple tree.
loc_o Location of the Orange tree.
$size_a$ Number of apples that fell from the tree.
$apples$ Distance at which each apple falls from the tree.
$size_o$ Number of oranges that fell from the tree.
$oranges$ Distance at which each orange falls from the tree.

Given the value of d for m apples and n oranges, can you determine how many apples and oranges will fall on Sam's house (i.e., in the inclusive range $[s, t]$)? Print the number of apples that fall on Sam's house as your first line of output, then print the number of oranges that fall on Sam's house as your second line of output.

Input Format

The first line contains two space-separated integers denoting the respective values of s and t.
The second line contains two space-separated integers denoting the respective values of a and b.
The third line contains two space-separated integers denoting the respective values of m and n.
The fourth line contains m space-separated integers denoting the respective distances that each apple falls from point a.
The fifth line contains n space-separated integers denoting the respective distances that each orange falls from point b.

Constraints

- $1 \leq s, t, a, b, m, n \leq 10^5$
- $-10^5 \leq d \leq 10^5$
- $a < s < t < b$

Output Format

Print two lines of output:

1. On the first line, print the number of apples that fall on Sam's house.
2. On the second line, print the number of oranges that fall on Sam's house.

Sample Input 0

```
7 11
5 15
3 2
-2 2 1
5 -6
```

Sample Output 0

```
1
1
```

Explanation 0

The first apple falls at position $5 - 2 = 3$.
The second apple falls at position $5 + 2 = 7$.
The third apple falls at position $5 + 1 = 6$.
The first orange falls at position $15 + 5 = 20$.
The second orange falls at position $15 - 6 = 9$.
Only one fruit (the second apple) falls within the region between 7 and 11, so we print 1 as our first line of output.
Only the second orange falls within the region between 7 and 11, so we print 1 as our second line of output.

Divisible Sum Pairs

You are given an array of n integers, $a_0, a_1, \ldots, a_{n-1}$, and a positive integer, k. Find and print the number of (i, j) pairs where $i < j$ and a_i + a_j is divisible by k.

Input Format

The first line contains 2 space-separated integers, n and k, respectively.
The second line contains n space-separated integers describing the respective values of $a_0, a_1, \ldots, a_{n-1}$.

Constraints

- $2 \leq n \leq 100$
- $1 \leq k \leq 100$
- $1 \leq a_i \leq 100$

Output Format

Print the number of (i, j) pairs where $i < j$ and a_i + a_j is evenly divisible by k.

Sample Input

```
6 3
1 3 2 6 1 2
```

Sample Output

```
5
```

Explanation

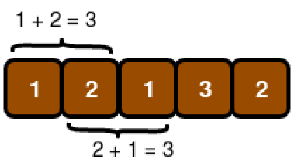
Here are the 5 valid pairs:

- $(0, 2) \rightarrow a_0 + a_2 = 1 + 2 = 3$
- $(0, 5) \rightarrow a_0 + a_5 = 1 + 2 = 3$
- $(1, 3) \rightarrow a_1 + a_3 = 3 + 6 = 9$
- $(2, 4) \rightarrow a_2 + a_4 = 2 + 1 = 3$
- $(4, 5) \rightarrow a_4 + a_5 = 1 + 2 = 3$

Birthday Chocolate

Lily has a chocolate bar with numbered squares. She wants to share it with Ron for his birthday. She decides to share a contiguous segment of the bar selected such that the sum of the integers on the squares is equal to a given value. The length of the segment will match Ron's birth month. The sum of the segments will match his birth day. You must determine how many ways she can divide the chocolate.

Consider the chocolate bar as an array of squares, $s = [1, 2, 1, 3, 2]$. She wants to find segments summing to Ron's birth day, $d = 3$ with a length equalling his birth month, $m = 2$. In this case, there are two segments meeting her criteria.



Input Format

The first line contains an integer n, the number of squares in the chocolate bar.
The second line contains n space-separated integers $s[i]$, the numbers on the chocolate squares where $0 \leq i < n$.
The third line contains two space-separated integers, d and m, Ron's birth day and his birth month.

Constraints

- $1 \leq n \leq 100$
- $1 \leq s[i] \leq 5$, where ($0 \leq i < n$)
- $1 \leq d \leq 31$
- $1 \leq m \leq 12$

Output Format

Print an integer denoting the total number of ways that Lily can portion her chocolate bar to share with Ron.

Sample Input 0

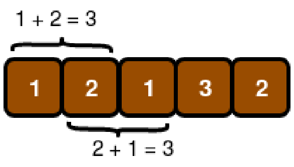
```
5
1 2 1 3 2
3 2
```

Sample Output 0

```
2
```

Explanation 0

Lily wants to give Ron $m = 2$ squares summing to $d = 3$. The following two segments meet the criteria:



Sample Input 1

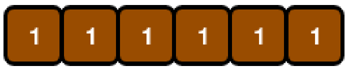
```
6
1 1 1 1 1 1
3 2
```

Sample Output 1

```
0
```

Explanation 1

Lily only wants to give Ron $m = 2$ consecutive squares of chocolate whose integers sum to $d = 3$. There are no possible pieces satisfying these constraints:



Thus, we print 0 as our answer.

Sample Input 2

```
1
4
4 1
```

Sample Output 2

```
1
```

Explanation 2

Lily only wants to give Ron $m = 1$ square of chocolate with an integer value of $d = 4$. Because the only square of chocolate in the bar satisfies this constraint, we print 1 as our answer.

Migratory Birds

You have been asked to help study the population of birds migrating across the continent. Each type of bird you are interested in will be identified by an integer value. Each time a particular kind of bird is spotted, its id number will be added to your array of sightings. You would like to be able to find out which type of bird is most common given a list of sightings. Your task is to print the type number of that bird and if two or more types of birds are equally common, choose the type with the smallest ID number.

For example, assume your bird sightings are of types $arr = [1, 1, 2, 2, 3]$. There are two each of types 1 and 2, and one sighting of type 3. Pick the lower of the two types seen twice: type 1.

Function Description

Complete the *migratoryBirds* function in the editor below. It should return the lowest type number of the most frequently sighted bird.

migratoryBirds has the following parameter(s):

- *arr*: an array of integers representing types of birds sighted

Input Format

The first line contains an integer denoting n, the number of birds sighted and reported in the array arr.
The second line describes arr as n space-separated integers representing the type numbers of each bird sighted.

Constraints

- $5 \leq n \leq 2 \times 10^5$
- It is guaranteed that each type is 1, 2, 3, 4, or 5.

Output Format

Print the type number of the most common bird; if two or more types of birds are equally common, choose the type with the smallest ID number.

Sample Input 0

```
6
1 4 4 4 5 3
```

Sample Output 0

```
4
```

Explanation 0

The different types of birds occur in the following frequencies:

- Type 1: 1 bird

- Type 2: 0 birds
- Type 3: 1 bird
- Type 4: 3 birds
- Type 5: 1 bird

The type number that occurs at the highest frequency is type 4, so we print 4 as our answer.

Sample Input 1

```
11
1 2 3 4 5 4 3 2 1 3 4
```

Sample Output 1

```
3
```

Explanation 1

The different types of birds occur in the following frequencies:

- Type 1: 2
- Type 2: 2
- Type 3: 3
- Type 4: 3
- Type 5: 1

Two types have a frequency of 3, and the lower of those is type 3.