CS1020 Take-home Lab #2

Exercise #1: Marking the Ruler

(http://www.comp.nus.edu.sg/~cs1020/3_ca/labs.html)

Objective

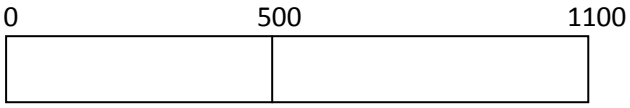
The objective of this problem is to test the understanding of OOP concepts and the use of array.

Task Statement

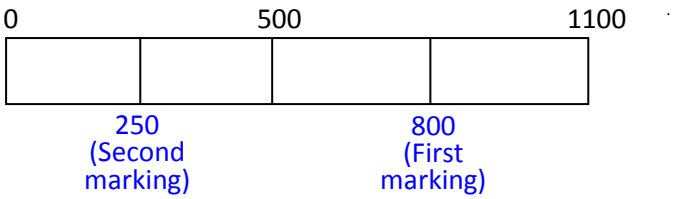
A wooden ruler is **S** centimeters long. The left end of the ruler is represented as 0. A carpenter has already made **N** markings on the ruler, which are randomly distributed along the ruler and each marking is of a certain distance (in cm) from the origin 0. The space between a marking and its adjacent marking is called a segment.

Now, the carpenter is trying to make additional **L** markings on the ruler to minimize the longest segment on the ruler.

For example, given the following ruler of length 1100 cm with one existing marking at 500 cm.



The longest segment is [500, 1100] which is 600 cm long. If we were to add new markings with the purpose of minimizing the longest segment, our first additional marking should be at 800 cm, bringing our longest segment now to 500 cm long, and the second additional marking should be at 250 cm, bringing our longest segment now to 300 cm long.



You are to minimize the length of the longest segment after adding these **L** markings.

Note that the positions of the markings can only be integers. For example, you can make a marking at 2 cm but not 2.5 cm.

Create a class called **Ruler** with an attribute which is an integer array to store the positions of the markings. You may add other attributes for this class. Write a client class **MarkRuler** that contains the main() method. Both classes should be contained in the file **MarkRuler.java**.

You may use the **Arrays.sort()** method in your program. Refer to the API documentation.

Input

The first line of the input contains a positive integer **S** ($1 \leq$ **S** $<$ 2000) which indicates the length of the ruler in centimetre. The second line of the input contains a positive integer **N** ($1 \leq$ **N** $\leq$ 50) which represents the number of existing markings. This is followed by **N** integers indicating the positions of the markings. The last line of the input contains a positive integer **L** ($1 \leq$ **L** $\leq$ 30) which represents the number of new markings.

Note that short symbols such as **S**, **N** and **L** are used in the task description above for convenience. In your program, you are expected to give them descriptive variable names.

Output

The output should contain one positive integer which shows the length of the longest segment.

Sample Input #1

```
1000
3
200 701 800
1
```

Sample Output #1

```
251
```

Explanation: For example, if we make a new marking at 1 cm, the length of the longest segment remains at (701 – 200) = 501 cm. Hence, we should make a new marking at 450 cm or 451 cm which reduces the longest segment to 251 cm.

Sample Input #2

```
800
6
622 411 201 555 755 82
7
```

Sample Output #2

```
70
```

Explanation: The seven new markings can be distributed along the ruler, i.e. make one marking between 0 and 82, one marking between 82 and 201, two markings between 201 and 411, two markings between 411 and 555 and one marking between 622 and 755, a total of 7 markings. The longest segment is 70 cm because in segment [201, 411], the best we can do is to split it into three equal length segments in this case.