**Max Min**

You will be given a list of integers arr, and a single integer k . You must create an array of length k  from elements of arr such that its *unfairness* is minimized. Call that array  subarr. Unfairness of an array is calculated as

Max(subarr)-min(subarr)

Where:   
- *max* denotes the largest integer in  subarr  
- *min* denotes the smallest integer in subarr

As an example, consider the array [1,4,7,2] with a k of 2. Pick any two elements, test subarr=[4,7].   
 unfairness = max(4,7)- min(4,7)=7-4=3

Testing for all pairs, the solution [1,2] provides the minimum unfairness.

**Note**: Integers in arr may not be unique.

**Function Description**

Complete the *maxMin* function in the editor below. It must return an integer that denotes the minimum possible value of *unfairness*.

maxMin has the following parameter(s):

* *k*: an integer, the number of elements in the array to create
* *arr*: an array of integers.

**Input Format**

The first line contains an integer n, the number of elements in array arr.   
The second line contains an integer k.   
Each of the next n lines contains an integer arr[i] where0<=i<=n .

**Constraints**

 2<=n<=10^5

2<=k<=n

0<=arr[i]<=10^9  
 

**Output Format**

An integer that denotes the minimum possible value of *unfairness*.

**Sample Input 0**

7

3

10

100

300

200

1000

20

30

**Sample Output 0**

20

**Explanation 0**

Here k=3; selecting the 3 integers 10,20,30, unfairness equals

max(10,20,30) - min(10,20,30) = 30 - 10 = 20

**Sample Input 1**

10

4

1

2

3

4

10

20

30

40

100

200

**Sample Output 1**

3

**Explanation 1**

Here k=4; selecting the 4 integers 1,2,3,4, unfairness equals

max(1,2,3,4) - min(1,2,3,4) = 4 - 1 = 3

**Sample Input 2**

5

2

1

2

1

2

1

**Sample Output 2**

0

**Explanation 2**

Here k=2.  Subarr=[2,2]or subarr=[1,1]  give the minimum unfairness of 0.

using System.CodeDom.Compiler;

using System.Collections.Generic;

using System.Collections;

using System.ComponentModel;

using System.Diagnostics.CodeAnalysis;

using System.Globalization;

using System.IO;

using System.Linq;

using System.Reflection;

using System.Runtime.Serialization;

using System.Text.RegularExpressions;

using System.Text;

using System;

class Solution {

// Complete the maxMin function below.

static int maxMin(int k, int[] arr) {

Array.Sort(arr);

int min=arr[k-1]-arr[0];

for(int i=k-1;i<arr.Length;i++){

min=min>arr[i]-arr[i+1-k] ?arr[i]-arr[i+1-k]:min;

}

return min;

}

static void Main(string[] args) {

TextWriter textWriter = new StreamWriter(@System.Environment.GetEnvironmentVariable("OUTPUT\_PATH"), true);

int n = Convert.ToInt32(Console.ReadLine());

int k = Convert.ToInt32(Console.ReadLine());

int[] arr = new int [n];

for (int i = 0; i < n; i++) {

int arrItem = Convert.ToInt32(Console.ReadLine());

arr[i] = arrItem;

}

int result = maxMin(k, arr);

textWriter.WriteLine(result);

textWriter.Flush();

textWriter.Close();

}

}

**Congratulations**

You solved this challenge. Would you like to challenge your friends?

[Next Challenge](https://www.hackerrank.com/challenges/reverse-shuffle-merge?h_l=interview&playlist_slugs%5B%5D=interview-preparation-kit&playlist_slugs%5B%5D=greedy-algorithms&h_r=next-challenge&h_v=zen)

* **Test case 0**
* **Test case 1**
* **Test case 2**
* **Test case 3**
* **Test case 4**
* **Test case 5**
* **Test case 6**
* **Test case 7**
* **Test case 8**
* **Test case 9**
* **Test case 10**
* **Test case 11**
* **Test case 12**
* **Test case 13**
* **Test case 14**
* **Test case 15**
* **Test case 16**

Compiler Message

**Success**

Input (stdin)

Download

* **7**
* **3**
* **10**
* **100**
* **300**
* **200**
* **1000**
* **20**
* **30**

Expected Output

Download

* **20**