Data structures and Algo in Java - Day 25

import java.util.Arrays;

public class day25

{

public static void main (String [] args)

{

// int arr [] = {2,3,4,7,11};

// int k = 5;

// findNthMissingNumber(arr,k);

int arr [] = {0,3,4,7,10,9};

int cows = 4;

aggressiveCows(arr,cows);

}

public static void findNthMissingNumber(int arr[] ,int k )

{

int low = 0;

int high = arr.length - 1;

while(low<=high)

{

int mid = (low+high)/2;

int missing = arr[mid] - (mid+1);

if(missing<k)

{

low = mid +1;

}

else

{

high = mid -1;

}

}

System.out.println("The "+k+"th Missing Number is "+(low+k));

}

public static void aggressiveCows ( int arr [] ,int cows)

{

Arrays.sort(arr);

int low = 0;

int high = arr[arr.length-1] - arr[0] ;

while(low<=high)

{

int mid = (low+high)/2;

if(canWePlaceCows(arr,mid,cows)==true)

{

low = mid +1;

}

else

{

high = mid - 1;

}

}

System.out.println(high+" is the maximum distance in the minimum distance where "+cows+" cows can be placed");

}

public static boolean canWePlaceCows(int arr[], int distance , int cows) // 5

{

int cowsCount = 1;

int lastCow = arr[0];

for(int i=1;i<arr.length;i++)

{

if(arr[i]-lastCow>=distance) // 3-0 = 3>=5 NO 4-0 = 4>=5 NO 7-0 = 7>=5 YES so cow is added there and last cow is 7

{

cowsCount ++;

lastCow = arr[i];

}

}

if(cowsCount>= cows)

{

return true;

}

return false;

}

}