# 1482. Minimum Number of Days to Make m Bouquets

You are given an integer array bloomDay, an integer m and an integer k.

You want to make m bouquets. To make a bouquet, you need to use k adjacent flowers from the garden.

The garden consists of n flowers, the ith flower will bloom in the bloomDay[i] and then can be used in exactly one bouquet.

Return the minimum number of days you need to wait to be able to make m bouquets from the garden. If it is impossible to make m bouquets return -1.

## SOLUTION IN JAVA

class Solution {

public int minDays(int[] bloomDay, int m, int k) {

if (bloomDay.length < (long) m \* k)

return -1;

int l = Arrays.stream(bloomDay).min().getAsInt();

int r = Arrays.stream(bloomDay).max().getAsInt();

while (l < r) {

final int mid = (l + r) / 2;

if (getBouquetCount(bloomDay, k, mid) >= m)

r = mid;

else

l = mid + 1;

}

return l;

}

// Returns the number of bouquets (k flowers needed) can be made after the

// `waitingDays`..

private int getBouquetCount(int[] bloomDay, int k, int waitingDays) {

int bouquetCount = 0;

int requiredFlowers = k;

for (final int day : bloomDay)

if (day > waitingDays) {

// Reset `requiredFlowers` since there was not enough adjacent flowers.

requiredFlowers = k;

} else if (--requiredFlowers == 0) {

// Use k adjacent flowers to make a bouquet.

++bouquetCount;

requiredFlowers = k;

}

return bouquetCount;

}

}