Given two arrays of equal size N and an integer K. The task is to check if after permuting both arrays, we get sum of their corresponding element greater than or equal to k i.e Ai + Bi >= K for all i (from 0 to N-1). Return true if possible, else false.

Example 1:

Input :
a[] = {2, 1, 3},
b[] = { 7, 8, 9 },
k = 10.
Output :
True
Explanation:
Permutation a[] = { 1, 2, 3 }
and b[] = { 9, 8, 7 }
satisfied the condition a[i] + b[i] >= K.

Example 2:

Input :
a[] = {1, 2, 2, 1}, b[] = { 3, 3, 3, 4 }, k = 5.
Output :
False
Explanation:
Since any permutation won't give the answer.

Your Task:

You don't need to read input or print anything. Your task is to complete the function isPossible() which takes the array A[], B[], its size N and an integer K as inputs and returns the answer.

Expected Time Complexity: O(N. Log(N))

Expected Auxiliary Space: O(1)

Constraints:

$1 \leq N \leq 10^5$

$1 \leq K \leq 10^{18}$

$1 \leq A_i, B_i \leq 10^{17}$

Solution:

```
class Solution {
   public boolean isPossible(long A[], long B[], int N, long K) {
       // Sort A in ascending order
       Arrays.sort(A);

       // Sort B in descending order
       Long[] BLong = Arrays.stream(B).boxed().toArray(Long[]::new);
       Arrays.sort(BLong, Collections.reverseOrder());

       // Check if A[i] + B[i] >= K for all i
       for (int i = 0; i < N; i++) {
           if (A[i] + BLong[i] < K) {
               return false;
           }
       }

       return true;
       // Your code goes here
   }
}
```