

B. Born This Way

time limit per test: 1 second

memory limit per test: 256 megabytes

input: standard input

output: standard output

Arkady bought an air ticket from a city A to a city C. Unfortunately, there are no direct flights, but there are a lot of flights from A to a city B, and from B to C.

There are  $n$  flights from A to B, they depart at time moments  $a_1, a_2, a_3, \dots, a_n$  and arrive at B  $t_a$  moments later.

There are  $m$  flights from B to C, they depart at time moments  $b_1, b_2, b_3, \dots, b_m$  and arrive at C  $t_b$  moments later.

The connection time is negligible, so one can use the  $i$ -th flight from A to B and the  $j$ -th flight from B to C if and only if  $b_j \geq a_i + t_a$ .

You can cancel at most  $k$  flights. If you cancel a flight, Arkady can not use it.

Arkady wants to be in C as early as possible, while you want him to be in C as late as possible. Find the earliest time Arkady can arrive at C, if you optimally cancel  $k$  flights. If you can cancel  $k$  or less flights in such a way that it is not possible to reach C at all, print  $-1$ .

Input

The first line contains five integers  $n, m, t_a, t_b$  and  $k$  ( $1 \leq n, m \leq 2 \cdot 10^5$ ,  $1 \leq k \leq n + m$ ,  $1 \leq t_a, t_b \leq 10^9$ ) — the number of flights from A to B, the number of flights from B to C, the flight time from A to B, the flight time from B to C and the number of flights you can cancel, respectively.

The second line contains  $n$  distinct integers in increasing order  $a_1, a_2, a_3, \dots, a_n$  ( $1 \leq a_1 < a_2 < \dots < a_n \leq 10^9$ ) — the times the flights from A to B depart.

The third line contains  $m$  distinct integers in increasing order  $b_1, b_2, b_3, \dots, b_m$  ( $1 \leq b_1 < b_2 < \dots < b_m \leq 10^9$ ) — the times the flights from B to C depart.

Output

If you can cancel  $k$  or less flights in such a way that it is not possible to reach C at all, print  $-1$ .

Otherwise print the earliest time Arkady can arrive at C if you cancel  $k$  flights in such a way that maximizes this time.

Examples

input

4 5 1 1 2  
1 3 5 7  
1 2 3 9 10

Copy

output

11

Copy

input

2 2 4 4 2  
1 10  
10 20

Copy

output

-1

Copy

input


4 3 2 3 1  
1 999999998 999999999 1000000000

Copy

Codeforces Global Round 3

Finished

Practice



→ Virtual participation

Virtual contest is a way to take part in past contest, as close as possible to participation on time. It is supported only ACM-ICPC mode for virtual contests. If you've seen these problems, a virtual contest is not for you - solve these problems in the archive. If you just want to solve some problem from a contest, a virtual contest is not for you - solve this problem in the archive. Never use someone else's code, read the tutorials or communicate with other person during a virtual contest.

Start virtual contest

→ Practice

You are registered for practice. You can solve problems unofficially. Results can be found in the contest status and in the bottom of standings.

→ Clone Contest to Mashup

You can clone this contest to a mashup.

Clone Contest

→ Submit?

Language: GNU G++11 5.1.0

Choose file: 

选择文件

 未选择任何文件

Be careful: there is 50 points penalty for submission which fails the pretests or resubmission (except failure on the first test, denial of judgement or similar verdicts). "Passed pretests" submission verdict doesn't guarantee that the solution is absolutely correct and it will pass system tests.

Submit

→ Last submissions

Submission	Time	Verdict
<a href="#">54934593</a>	Jun/01/2019 18:46	Accepted
<a href="#">54931620</a>	Jun/01/2019 18:31	Wrong answer on pretest 8
<a href="#">54928684</a>	Jun/01/2019 18:18	Wrong answer on pretest 7
<a href="#">54928324</a>	Jun/01/2019 18:16	Wrong answer on pretest 7

→ Problem tags

codeforces.com/contest/1148/problem/B

1/2

2019/6/2

Problem - B - Codeforces

3 4 1000000000

output

1000000003

Copy

binary search

brute force

two pointers

\*1700

No tag edit access

**Note**

Consider the first example. The flights from A to B depart at time moments 1, 3, 5, and 7 and arrive at B at time moments 2, 4, 6, 8, respectively. The flights from B to C depart at time moments 1, 2, 3, 9, and 10 and arrive at C at time moments 2, 3, 4, 10, 11, respectively. You can cancel at most two flights. The optimal solution is to cancel the first flight from A to B and the fourth flight from B to C. This way Arkady has to take the second flight from A to B, arrive at B at time moment 4, and take the last flight from B to C arriving at C at time moment 11.

In the second example you can simply cancel all flights from A to B and you're done.

In the third example you can cancel only one flight, and the optimal solution is to cancel the first flight from A to B. Note that there is still just enough time to catch the last flight from B to C.

→ Contest materials

• Announcement #1 (en)

• Announcement #2 (ru)

[Codeforces](#) (c) Copyright 2010-2019 Mike Mirzayanov  
The only programming contests Web 2.0 platform  
Server time: Jun/02/2019 21:43:51<sup>UTC+8</sup> (f1).  
Desktop version, switch to [mobile version](#).  
[Privacy Policy](#)

