

Login

Username

Password

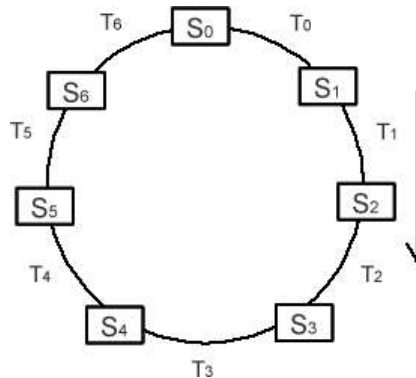
☐ Remember Me[Password forgotten?](#)

Navigation

[Register](#)[Learn C++](#)[Algorithms](#)[Training](#)[Competitions](#)[Forums](#)[Wiki](#)[Help](#)[Report Bug](#)

Mountain running

This year, Ohrid will host an international mountain running competition. Since runners need a lot of water to sustain them, the organizers have prepared a set of locations where refreshment stations may be set up, and have connected them in a large directed cycle via one-directional trails (as shown on the image below). The refreshment stations are named S_0 to S_{N-1} (going along the direction of the cycle), and the trail T_i connects stations S_i and S_{i+1} (directed toward S_{i+1}) - where $S_N = S_0$.



It is known that a runner will spend exactly K_i units of water to finish trail T_i , and each runner will be able to get at most W_i units of water at refreshment station S_i . There is no limit to the amount of water a runner may carry at any time, but the time in minutes a runner needs to finish a trail is equal to the number of units of water he started the trail with - so if a runner starts trail T_1 with 5 units of water, then he will finish it in 5 minutes regardless of the amount of water he spends on the trail.

The organizers are now considering possible start and end points for the race (the race must start from and finish at a refreshment station). They want you to write a program that, for each of the Q pairs of start and end refreshment stations given to it, will calculate and output the minimum time necessary for an optimal runner to finish the given race. Note that the runner may start the race with at most as much water as is available at the starting refreshment station.

Input

The first line of input contains two integers N ($3 \leq N \leq 1000$) and Q ($1 \leq Q \leq 100000$) - the number of trails, and the number of pairs of start and end refreshment stations that the organizers are considering.

The second line of input contains N integers W_i ($1 \leq W_i \leq 1000000$) - the units of water available at each refreshment station (starting with station S_0).

The third line of input contains N integers K_i ($1 \leq K_i \leq 1000000$) - the units of water required to finish each trail (starting with trail T_0).

Each of the following Q lines contains two distinct integers S_i and E_i ($0 \leq S_i, E_i \leq N-1$) - the pairs for starting and ending refreshment stations.

In test cases worth at least 40% of the full score, the number of refreshment stations will be at most 100 ($3 \leq N \leq 100$) and all W_i and K_i will be at most 10 ($1 \leq W_i, K_i \leq 10$).

Output

For each pair of starting and ending locations, in the same order as they are given in the input, your program should output the minimum time necessary for an optimal runner to finish the given race. If the race cannot be completed, your program should output "-1" (quotes for clarity).

Constraints

Time limit: 1 second
Memory limit: 64 megabytes

Examples

input	output
7 4	2
3 2 5 2 5 1 6	-1
2 4 3 1 6 1 2	11
0 1	5
4 5	
2 5	
5 1	

Submit **Test**

Submit your code

Solution language:

Automatic Detection...



Upload solution (option 1):

Select

Paste your code (option 2):



CodeMirror

Submit!

Copyright © 2026 Mendo Judge Team | All Rights Reserved
Supported by [CSM](#) | Powered by [Struts](#) | [XHTML 1.0](#) | [CSS 2.0](#) | [FAMFAMFAM](#)
» This website (project) is dedicated to the memory of [Nikola Postolov](#) «