Cubes and Cylinders

by xennygrimmato

You are in charge of storing some of the inventory in a company. There are different types of packages. The total number of packages and the number of packages of each type is known to you. Also, each package is cube-shaped, with edge length given.

Packages can only be stored in *cylindrical* containers which have a circular opening of given radius. Each container can hold a particular number of packages. We may assume that each container is tall enough that it can accommodate the total height of all the packages.

As part of company policy, you must ensure the following holds:

- Each package can only be kept on top of another package.
- The package must not touch the sides of the cylinder, even at a single point. The packages may touch boundaries with each other or with the base of the cylinder.

You have to choose which packages go into which containers. Discard packages that can't be stored.

What is the maximum number of packages that can be put in the containers?

Complete the function `maximumPackages` which takes in four arrays

- S, denoting the edge-lengths of each package type,
- K, denoting the number of copies of each package type,
- R, denoting the radius of the opening of each container, and
- C, denoting the number of packages each container can contain,

and returns a single integer denoting the answer.

Input Format

The first line contains two space-separated positive integers n, denoting number of types of packages, and m, denoting number of cylindrical containers.

The second line contains n space-separated positive integers S_i, where S_i represents the edge-length of the i^{th} package type.

The third line contains n space-separated non-negative integers K_i, where K_i represents the number of copies of i^{th} package type.

The fourth line contains m space-separated positive integers R_i, where R_i represents the radius of the opening of the i^{th} cylinder.

The fifth line contains m space-separated positive integers C_i, where C_i represents the capacity of the i^{th} cylinder.

Constraints

- $1 \leq n, m \leq 500$
- $1 \leq S_i, R_i \leq 500$
- $1 \leq K_i, C_i \leq 2500$

- All the given S_i and R_i are unique.

Output Format

Print a single integer denoting the maximum number of packages that can be put inside the cylinders, provided they remain intact on one top of another and do not touch the circumference of the cylinder in which they are put.

Sample Input 0

```
2 2
1 2
1 1
1 2
1 1
```

Sample Output 0

```
2
```

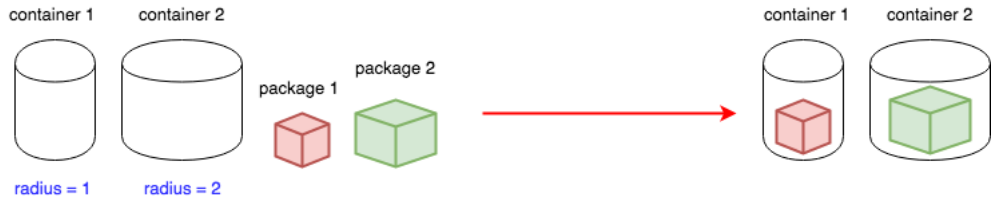
Explanation 0

The edge length of the 1^{st} package = 1. There is 1 such package.
The edge length of the 2^{nd} package = 2. There is 1 such package.

The radius of 1^{st} cylinder = 1.
The radius of 2^{nd} cylinder = 2.

The 1^{st} package can be put inside 1^{st} as well as 2^{nd} cylinder.
The 2^{nd} package can only be put inside the 2^{nd} cylinder.

Since both cylinders can contain 1 package each, we can put 1^{st} package in the 1^{st} cylinder, and the 2^{nd} package in the 2^{nd} cylinder.



Contest ends in a day

Submissions: 1994
Max Score: 30
Difficulty: Medium

Rate This Challenge:

More

Current Buffer (saved locally, editable) C++14

```
#include <cmath>
#include <cstdio>
#include <vector>
#include <iostream>
#include <algorithm>
using namespace std;


int main() {
    /* Enter your code here. Read input from STDIN. Print output to STDOUT */
    return 0;
}

```

Line: 1 Col: 1

Upload Code as File Test against custom input Run Code Submit Code