

Colorful Knapsack

Problem Code: **COLORINT**



Tweet

Like

Share

Be the first of your friends to like this.

You are given N stones, labeled from 1 to N . The i -th stone has the weight $W[i]$. There are M colors, labeled by integers from 1 to M . The i -th stone has the color $C[i]$ (of course, an integer between 1 to M , inclusive).

You want to fill a Knapsack with these stones. The Knapsack can hold a total weight of X . You want to select exactly M stones; one of each color. The sum of the weights of the stones must not exceed X . Since you paid a premium for a Knapsack with capacity X (as opposed to a Knapsack with a lower capacity), you want to fill the Knapsack as much as possible.

Write a program that takes all the above values as input and calculates the best way to fill the Knapsack - that is, the way that minimizes the unused capacity. Output this unused capacity. See the explanation of the sample test cases for clarity.

[All Submissions \(/DI15R075/status/COLORINT\)](#)

Successful Submissions

+

Input

The first line of input contains the integer T , the number of test cases. Then follows the description of T test cases. The first line of each test case contains three integers, N , M and X , separated by single space. The next line contains N integers, $W[1]$, $W[2]$, $W[3]$... $W[N]$, separated by single space. The next line contains N integers $C[1]$, $C[2]$, $C[3]$... $C[N]$, separated by single space.

Output

An optimal way of filling the Knapsack minimizes unused capacity. There may be several optimal ways of filling the Knapsack. Output the unused capacity of the Knapsack (a single integer on a line by itself) for an optimal way. If there is no way to fill the Knapsack, output -1. Output T lines, one for each test case.

Constraints

```
1 ≤ T ≤ 10
1 ≤ M ≤ 100
M ≤ N ≤ 100
1 ≤ W[i] ≤ 100
1 ≤ C[i] ≤ M
1 ≤ X ≤ 10000
```

Sample Input

```
4
9 3 10
2 3 4 2 3 4 2 3 4
1 1 1 2 2 2 3 3 3
9 3 10
1 3 5 1 3 5 1 3 5
1 1 1 2 2 2 3 3 3
3 3 10
3 4 4
1 2 3
3 3 10
3 3 3
1 2 1
```

Sample Output

```
0
1
-1
-1
```

Explanation

In the first test case you can select stone 2, stone 5 and stone 9. The knapsack will be completely full. Of course, there are several other ways to select stones such that the knapsack is full. The unused capacity in all such ways is 0.

In the second test case you cannot select stones such that the knapsack is completely full. You can select stones {1, 4, 9}, such that the unused capacity is $10 - 1 - 5 = 3$. But there is a better way. Select stones {2, 5, 8}. The unused capacity is $10 - 3 - 3 = 1$. This is the optimal way. There is another way that is optimal. Select stones {1, 5, 9}. The unused capacity is $10 - 1 - 3 - 5 = 1$.

In the third test case there is only one option. Select stones {1, 2, 3}. The total weight will be 11. This is more than what the knapsack can hold.

In the fourth test case there is no stone of color 3. Thus, there is no valid selection of stones possible. The answer will be -1.

Author: [directi_campus \(/users/directi_campus/\)](/users/directi_campus/)

Tags: [directi_campus \(/tags/problems/directi_campus/\)](/tags/problems/directi_campus/)

Date Added: 18-10-2014

Time Limit: 1 secs

Source Limit: 50000 Bytes

Languages: C, CPP 4.3.2, CPP 6.3, CPP14, JAVA, PYTH, PYTH 3.5

Comments ▾

[CodeChef is a non-commercial competitive programming community](#)

[About CodeChef \(http://www.codechef.com/aboutus/\)](http://www.codechef.com/aboutus/) [About Directi \(http://www.directi.com/\)](http://www.directi.com/) [CEO's Corner \(http://www.codechef.com/ceoscorner/\)](http://www.codechef.com/ceoscorner/)
[C-Programming \(http://www.codechef.com/c-programming/\)](http://www.codechef.com/c-programming/) [Programming Languages \(http://www.codechef.com/Programming-Languages/\)](http://www.codechef.com/Programming-Languages/) [Contact Us \(http://www.codechef.com/contactus/\)](http://www.codechef.com/contactus/)

© 2009 Directi Group (<http://directi.com>). All Rights Reserved. CodeChef uses SPOJ © by Sphere Research Labs (<http://www.sphere-research.com>)
In order to report copyright violations of any kind, send in an email to copyright@codechef.com (<mailto:copyright@codechef.com>)

Directi (<http://directi.com>)
Intelligent People. Uncommon Ideas.

The time now is: 10:24:53 PM
Your IP: 123.201.210.10

<http://www.codechef.com>) - A Platform for Aspiring Programmers

CodeChef was created as a platform to help programmers make it big in the world of algorithms, **computer programming** and **programming contests**. At CodeChef we work hard to revive the geek in you by hosting a **programming contest** at the start of the month and another smaller programming challenge in the middle of the month. We also aim to have training sessions and discussions related to **algorithms**, **binary search**, technicalities like **array size** and the likes. Apart from providing a platform for **programming competitions**, CodeChef also has various algorithm tutorials and forum discussions to help those who are new to the world of **computer programming**.

<https://www.codechef.com/problems/easy>) - A Place to hone your 'Computer Programming Skills'

Try your hand at one of our many practice problems and submit your solution in a language of your choice. Our **programming contest** judge accepts solutions in over 35+ programming languages. Preparing for coding contests were never this much fun! Receive points, and move up through the CodeChef ranks. Use our practice section to better prepare yourself for the multiple **programming challenges** that take place through-out the month on CodeChef.

<https://www.codechef.com/problems/easy>) - Monthly Programming Contests and Cook-offs

Here is where you can show off your **computer programming skills**. Take part in our 10 day long monthly coding contest and the shorter format Cook-off **coding contest**. Put yourself up for recognition and win great prizes. Our **programming contests** have prizes worth up to INR 20,000 (for Indian Community), \$700 (for Global Community) and lots more CodeChef goodies up for grabs.

[Programming Tools](#)

[Online IDE \(https://www.codechef.com/ide\)](https://www.codechef.com/ide)

[Upcoming Coding Contests \(http://www.codechef.com/contests#FutureContests\)](http://www.codechef.com/contests#FutureContests)

[Practice Problems](#)

[Easy \(https://www.codechef.com/problems/easy\)](https://www.codechef.com/problems/easy)

[Medium \(https://www.codechef.com/problems/medium\)](https://www.codechef.com/problems/medium)

[Contest Hosting \(http://www.codechef.com/hostyourcontest\)](http://www.codechef.com/hostyourcontest)

[Problem Setting \(http://www.codechef.com/problemsetting\)](http://www.codechef.com/problemsetting)

[CodeChef Tutorials \(http://www.codechef.com/wiki/tutorials\)](http://www.codechef.com/wiki/tutorials)

[CodeChef Wiki \(https://www.codechef.com/wiki\)](https://www.codechef.com/wiki)

[Hard \(https://www.codechef.com/problems/Hard\)](https://www.codechef.com/problems/Hard)

[Challenge \(https://www.codechef.com/problems/challenge\)](https://www.codechef.com/problems/challenge)

[Peer \(https://www.codechef.com/problems/extcontest\)](https://www.codechef.com/problems/extcontest)

[School \(https://www.codechef.com/problems/school\)](https://www.codechef.com/problems/school)

[FAQ's \(https://www.codechef.com/wiki/faq\)](https://www.codechef.com/wiki/faq)

Initiatives

[Go for Gold \(http://www.codechef.com/goforgold\)](http://www.codechef.com/goforgold)

[CodeChef for Schools \(http://www.codechef.com/school\)](http://www.codechef.com/school)

[Campus Chapters \(http://www.codechef.com/campus_chapter/about\)](http://www.codechef.com/campus_chapter/about)

[Domain Registration in India \(http://www.bigrock.in/\)](http://www.bigrock.in/) and [Web Hosting \(http://www.bigrock.com/web-hosting/\)](http://www.bigrock.com/web-hosting/) powered by BigRock