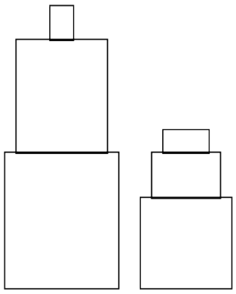
Problem A
A Towering Problem

Time limit: 1 second
Memory limit: 256 megabytes

Problem Description

You've been put in charge of an art exhibit from the famous minimalist sculptor J (even his name is minimalist!). J's work involves the careful layout of vertically dispositioned orthogonal parallelpipeds in a set of tapering obelisks – in other words, he puts smaller boxes on top of larger boxes. His most recent triumph is called "2 by 3's Decreasing," in which he has various sets of six boxes arranged in two stacks of three boxes each. One such set is shown below:



J has sent you the art exhibit and it is your job to set up each of the six-box sets at various locations throughout the museum. But when the sculptures arrived at the museum, uncultured barbarians (i.e., delivery men) simply dropped each set of six boxes on the floor, not realizing the aesthetic appeal of their original layout. You need to reconstruct each set of two towers, but you have no idea which box goes on top of the other! All you know is the following: for each set of six, you have the heights of the two towers, and you know that in any tower the largest height box is always on the bottom and the smallest height box is on the top. Armed with this information, you hope to be able to figure out which boxes go together before tomorrow night's grand opening gala.

Input Format

The input consists of eight positive integers. The first six represent the heights of the six boxes. These values will be given in no particular order and no two will be equal.

The last two values (which will never be the same) are the heights of the two towers.

Output Format

Output the heights of the three boxes in the first tower (i.e., the tower specified by the first tower height in the input), then the heights of the three boxes in the second tower. Each set of boxes should be output in order of decreasing height. Each test case will have a answer, and you may output any solution if there are multiple ones.

Technical Specification

All box heights will be at most 100 and the sum of the box heights will equal the sum of the tower heights.

Sample Input 1

```
12 8 2 4 10 3 25 14
```

Sample Output 1

```
12 10 3 8 4 2
```

Sample Input 2

```
12 17 36 37 51 63 92 124
```

Sample Output 2

```
63 17 12 51 37 36
```

Problem B
B is The Easiest

Time limit: 1 second

Memory limit: 256 megabytes

Problem Description

You are given n positive integers $a_1, a_2, \ldots, a_n$. Consider the multisets $S_1, \ldots, S_n$ where $S_i = \{a_1, \ldots, a_{i-1}, a_{i+1}, \ldots, a_n\} = \{a_1, \ldots, a_n\} \setminus \{a_i\}$. Please write a program to compute

$$\min_{1 \le i \le n} \left(\max_{x \in S_i} x - \min_{y \in S_i} y \right)$$

efficiently.

Input Format

The first line of the input contains one integer n. The second line of the input contains n positie integers $a_1, \ldots, a_n$.

Output Format

Output the answer in a line.

Technical Specification

$2 \le n \le 10^5$, and $a_1, \ldots, a_n$ are positive integers at most 10^5.

Sample Input 1

```
5
1 2 3 4 7
```

Sample Output 1

```
3
```

Sample Input 2

```
3
1 1 3
```

Sample Output 2

```
0
```

Almost blank page

Problem C
Coloring Socks

Time limit: 1 second
Memory limit: 256 megabytes

Problem Description

Having discolored his white socks in a rather beige shade (as seen on the picture), Luktas Svettocek realised he cannot just throw all his laundry into one machine and expect it to retain its original colors. However, he is also too lazy to do his laundry in several rounds. He would much rather buy more laundry machines!

Each of Luktas' socks have a color D_i which has a number between 0 and 10^9 assigned to it. After some experimentation, he found that he could wash any socks with a maximum absolute color difference of K in the same machine without any discoloring. The color difference of two socks i and j is $|D_i - D_j|$.

Luktas now needs to know how many washing machines he needs to wash his S socks, given that each machine can take at most C socks a time.

Input Format

The first line consists of three integers S, C and K, the number of socks, the capacity of a laundry machine and the maximum color difference, respectively. Then follow one line with S integers; these are the color values D_i of every sock.

Output Format

Output a single integer; the number of machines Luktas needs to wash all his socks.

Technical Specification

- $1 \leq S \leq 10^5$
- $1 \leq C \leq 10^5$
- $0 \leq K \leq 10^9$
- $0 \leq D_i \leq 10^9$ for $i \in \{1, \ldots, S\}$.

Sample Input 1

```
5 3 0
0 0 1 1 2
```

Sample Output 1

```
3
```

Sample Input 2

```
5 3 1
0 0 1 1 2
```

Sample Output 2

```
2
```

Almost blank page

Problem D
Dolls

Time limit: 2 seconds
Memory limit: 256 megabytes

Problem Description

Matryoshka dolls (Russian dolls) are a set of wooden dolls of decreasing size placed one inside another. Alice is now manufacturing a new kind of Matryoshka dolls. The new Matryoshka dolls can contain multiple doll sets at once (while not nested). To clearify the problem, we assign a size value for each type of dolls. For a doll with size k, it can contain a multiple doll sets if the sum of their outermost layers' size are strictly smaller than k. We also assign a pretty value for every type of dolls. The pretty value of a doll set is the sum of the pretty value of each dolls inside it.

Alice wants to make a new doll with size x, if the pretty value of this doll is zero, can you tell her the maximum pretty value of the doll set if this doll is the outermost doll? Assuming that she have infinite numbers of every kind of existing dolls.

Input Format

First line contains a testcase number T. For each testcase, the first line contains the number of existing doll types n and the new doll's size s. The second line contains n numbers as their size s_i. The third line contains n numbers as their pretty value p_i.

Output Format

For each testcase, output a number indicating the maximum pretty value.

Technical Specification

- $1 \leq T \leq 10^2$
- $1 \leq n \leq 10^2$
- $1 \leq s \leq 10^4$
- $1 \leq s_i \leq 10^4$ and $1 \leq p_i \leq 10^9$ for $i \in \{1, \ldots, n\}$

Sample Input 1

```
2
5 6
1 2 3 4 5
1 2 3 4 5
5 6
1 1 1 1 1
1 2 3 4 5
```

Sample Output 1

```
15
25
```

Almost blank page

Problem E
Exhausted `mangojunior`

Time limit: 2 seconds
Memory limit: 256 megabytes

Problem Description

One of `mangojunior`'s jobs is to deal with the management affairs of a mysterious classroom. The classroom is a $r \times c$ grid, so there are totally $r \times c$ blocks. There are n craft chairs numbered from 1 to n in the classroom. They are designed to be stacked on a certain magic block which lies on the intersection of row i and column j. Moreoever, chair k can be put over the top of some chair numbered with an integer less than k, and chairs are not allowed to be stacked on all the other blocks.

`mangojunior` plans to clean up the classroom. He observed that there is no chair on the magic block, so he can clean it first. Then, he can stack all the chairs on the magic block and clean the other blocks. In every step, `mangojunior` can select chair k and move it to an adjacent block which must be a block without chair or the magic block with a stack of chairs of number less than k. During the final exam period, `mangojunior` is exhausted. Once a chair moved into the magic block, `mangojunior` does not have enough energy to move it to another block. Please help `mangojunior` determine how many steps does he need to move all chairs to the magic block?

Input Format

The first line contains an integer T indicating the number of test cases. The first line of each test case contains three integers r, c, and n describing the size of the classroom and number of chairs. The second line of each test case contains two integers i and j denote the position of the magic block. The following n lines describe the positions of the chairs. The k-th line of them contains two integers i_k and j_k denote the position of chair k. No two given positions are the same and there is no chair on the magic block initially.

Output Format

For each test case, output one line containing the minimum number of moves which `mangojunior` needs to perform. The answer is guaranteed to be less than 13.

Technical Specification

- $1 \le T \le 10$
- $1 \le r, c$ and $r \times c \le 30$
- $0 \le i < r$ and $0 \le j < c$
- $0 \le i_k < r$ and $0 \le j_k < c$ for $k \in \{1, 2, ..., n\}$
- $1 \le n \le 8$

Sample Input 1

```
3
3 3 3
0 0
1 1
0 1
1 0
3 3 5
0 0
1 1
0 1
0 2
1 0
2 0
3 3 5
0 0
1 1
0 2
0 1
1 0
2 0
```

Sample Output 1

```
6
12
12
```

Problem F
Fortune

Time limit: 1 second
Memory limit: 256 megabytes

Problem Description

The mayor of Takao City wants to make money! He decides to invest in some stocks. By the power of the city mayor, he can know the best stock of each day. In day i, he can earn b_i dollars by investing one unit of stock. He can invest up to k_i units of stock in that day. However, he is not allowed to invest a fractional unit. He must invest an integral number of units.

The mayor can simply invest k_i unit in day i earning the maximum amount of money. But he wants to create a illusion of making more and more money. He decided to make the amount of the money earned increase everytime he invests. To keep the amount of earnings increasing, he may need to not invest somedays. What is the maximum days he can achieve?

Input Format

First line contains the number of testcases T. For each testcase, the first line contains the number of days n. The second line contains n numbers from b_1 to b_n. The third line contains n numbers k_1 to k_n.

Output Format

For each testcase, output a number indicating the maximum day keeping the illusion.

Technical Specification

- $1 \leq T \leq 10^2$
- $1 \leq n \leq 10^3$
- $1 \leq b_i \leq 10^9$ and $1 \leq k_i \leq 10^2$ for $i \in \{1, \ldots, n\}$.

Sample Input 1

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

Sample Output 1

```
2
1
```

Almost blank page

Problem G
Greatness of Subarrays

Time limit: 4 seconds
Memory limit: 256 megabytes

Problem Description

$s_0, \ldots, s_{n-1}$ is a sequence of integers. In this problem, a subarray of a sequence is an array containing some consecutive elements of the sequence. `harryooooooooooo` defines the greatness of a continuous segment $s_\ell, s_{\ell+1}, \ldots, s_{r-1}$ as its longest subarray which forms an arithmetic sequence. In other words, the greateness of $s_\ell, s_{\ell+1}, \ldots, s_{r-1}$ is the length of its longest subarray which owns the following feature: there exist two integers a and d such that the subarray can be described as $a, a+d, a+2d, \ldots, a+(m-1)d$, where m denotes the length of subarray. To make this problem harder, `harryooooooooooo` sometimes modifies the sequence. He chooses a subarray and increases or decreases all elements with an integer v. Please find the answers for him.

Input Format

The first line contains an integer n which is the length of the sequence. The second line contains n integers denoting the elements of the sequence $s_0, \ldots, s_{n-1}$. The third line contains an integer q, the number of operations. Then the following q lines are operations in one of following forms:

- $1\ \ell\ r\ v$: increase the elements in range $[\ell, r)$(including ℓ and excluding r) by v. A negative v means decreasing by $-v$.
- $2\ \ell\ r$: query the greatness of $s_\ell, \ldots, s_{r-1}$.

Output Format

For every type-2 operations, output an integer in one line.

Technical Specification

- $1 \le n \le 10^5$
- $1 \le q \le 10^6$
- $-10^9 \le s_i \le 10^9$ for $i \in \{0, \ldots, n-1\}$
- $0 \le \ell < r \le n$
- $-10^3 \le v \le 10^3$

Sample Input 1

```
10
1 2 3 4 1 2 3 4 5 6
5
2 0 10
2 0 5
2 2 7
1 4 9 4
2 0 10
```

Sample Output 1

```
6
4
3
9
```

Problem H
Hank's Challenge

Time limit: 1 second
Memory limit: 256 megabytes

Problem Description

Hank likes colorful triangles. He collects many sticks in various colors and wants to use them to build as many triangles as possible under the following constraints.

1. Each triangle must consist of exactly three sticks of distinct colors.
2. Every pair of triangles must consist of sticks of at least five distinct colors.

For example, if Hank builds a triangle with the color red, blue, and green, he cannot to build another triangle with color red, blue, and purple. This violate the second constraint.

You just discovered that Hank's collection of sticks has exactly 3^n different color in total. And there are at least 2^{100} sticks of each color. Hank is too tired during the final exam period, so he asks you to help him to build the triangle.

Input Format

The input has only one line with a number n. Hank's collection has exactly 3^n colors.

Output Format

The first line contain a number m – the number of triangle you can build. If m is no greater than 10^5, then you have to output m more lines to show how to build the triangles. Each of the m lines will contain three number indicate the colors of sticks building the corresponding triangle. Use only the integers between 1 to 3^n to indicate the different color. If there are multiple answer, you can print any of them.

Technical Specification

n is no more than 20.

Sample Input 1

```
1
```

Sample Output 1

```
1
1 2 3
```

Sample Input 2

```
2
```

Sample Output 2

```
12
1 2 3
1 4 9
1 5 8
1 6 7
2 4 7
2 5 9
2 6 8
3 4 8
3 5 7
3 6 9
4 5 6
7 8 9
```