# Ordine online (ristorante)

Edoardo, being busy implementing new features for CMS, has not realized that it is getting late and that restaurants are going to close in a short while! To avoid risking to go to sleep without having dinner, he decides to order online some takeaway pizzas. On the pizzeria's website, the list of the $N$ appetizers is shown on the first page, the list of the $N$ pizzas on the second one and the list of the $N$ desserts on the third one. In order to not lose any time scrolling the pages up and down, Edoardo would like to choose, in the order in which they are shown, the first $K_A$ appetizers, the first $K_P$ pizzas and the first $K_D$ desserts.

However, because of a bug in the website, it is not possible to add to the order two dishes that have the same price. Edoardo knows that the $i$-th appetizer costs $A_i$, the $j$-th pizza costs $P_j$ and the $q$-th dessert costs $D_q$. Can you help him choose $K_A$, $K_P$ and $K_D$ in order to maximize the total number of dishes (that is $K_A + K_P + K_D$), without triggering the bug?

## Implementation

You should submit a single file, with either a `.c` or `.cpp` extension.

> ☞ Among the attachments in this task you will find a template `ristorante.c` or `ristorante.cpp` with a sample implementation.

You will have to implement the following function:

| | |
|---|---|
| C | `int conta(int N, int *A, int *P, int *D);` |
| C++ | `int conta(int N, vector<int> &A, vector<int> &P, vector<int> &D);` |

- The integer $N$ is the amount of appetizers, pizzas and desserts.
- The array $A$, indexed from 0 to $N-1$, contains the prices of the appetizers.
- The array $P$, indexed from 0 to $N-1$, contains the prices of the pizzas.
- The array $D$, indexed from 0 to $N-1$, contains the prices of the desserts.
- The function must return the maximum number of dishes Edoardo can order without triggering the website's bug.

The grader calls the `conta` function and prints the returned value in the output file.

## Sample grader

Among this task's attachments you will find a simplified version of the grader used during evaluation, which you can use to test your solutions locally. The sample grader reads data from `stdin`, calls the functions that you should implement and writes back on `stdout` using the following format.

The input file is formed by four lines, containing:

- Line 1: the integer $N$.
- Line 2: the $N$ integers $A_0, \ldots, A_{N-1}$.
- Line 3: the $N$ integers $P_0, \ldots, P_{N-1}$.
- Line 4: the $N$ integers $D_0, \ldots, D_{N-1}$.

The output file is formed by a single line, containing the value returned by `conta`.

## Constraints

- $1 \leq N \leq 100\,000$.
- $1 \leq A_i, P_i, D_i \leq 10^9$ for each $i = 0 \dots N-1$.

## Scoring

Your program will be tested on a number of testcases grouped in subtasks. In order to obtain the score associated to a subtask, you need to correctly solve all the testcases it contains.

- **Subtask 1 [ 0 points]**: Sample testcases.
- **Subtask 2 [ 6 points]**: $N \leq 5$, $1 \leq A_i, P_i, D_i \leq 20$ for each $i = 0 \dots N-1$.
- **Subtask 3 [10 points]**: $N \leq 40$, $1 \leq A_i, P_i, D_i \leq 1000$ for each $i = 0 \dots N-1$.
- **Subtask 4 [ 9 points]**: No appetizer has the same price of a pizza or dessert, no pizza has the same price of a dessert.
- **Subtask 5 [11 points]**: $N \leq 150$, $1 \leq A_i, P_i, D_i \leq 1000$ for each $i = 0 \dots N-1$.
- **Subtask 6 [16 points]**: $N \leq 2\,000$, $1 \leq A_i, P_i, D_i \leq 500\,000$ for each $i = 0 \dots N-1$.
- **Subtask 7 [17 points]**: $N \leq 5\,000$.
- **Subtask 8 [31 points]**: No limits.

## Examples

| stdin | stdout |
|---|---|
| 4<br>10 4 8 6<br>7 9 5 5<br>2 1 5 4 | 9 |
| 3<br>1 2 5<br>7 2 3<br>5 1 4 | 6 |

## Explanation

In the **first sample testcase** Edoardo can order at most 9 dishes. For example, he can order all the appetizers, the first three pizzas and the first two desserts.

| Appetizers | Pizzas | Desserts |
|---|---|---|
| 10,00 € ✓ | 7,00 € ✓ | 2,00 € ✓ |
| 4,00 € ✓ | 9,00 € ✓ | 1,00 € ✓ |
| 8,00 € ✓ | 5,00 € ✓ | 5,00 € |
| 6,00 € ✓ | 5,00 € | 4,00 € |

In the **second sample testcase** Edoardo can order at most 6 dishes. For example, he can order all the pizzas and all the dessert, without ordering any appetizer.

| Appetizers | Pizzas | Desserts |
|---|---|---|
| 1,00 € | 7,00 € ✓ | 5,00 € ✓ |
| 2,00 € | 2,00 € ✓ | 1,00 € ✓ |
| 5,00 € | 3,00 € ✓ | 4,00 € ✓ |