Jelly Flavours (jelly)

Amy is a big fan of jelly, and wishes to buy some for dessert. There are a total of n flavours of jelly, numbered 0 to $n-1$. Store A sells jelly of flavour i for $a[i]$ dollars a piece, whereas Store B sells it for $b[i]$ dollars a piece. Amy can spend up to x dollars in Store A and up to y dollars in Store B.

Help Amy find the **maximum number of unique flavours** of jelly she can purchase.

Implementation details

You should implement the following procedure:

```
int find_maximum_unique(int x, int y, int[] a, int[] b)
```

- x: amount of money that can be spent in Store A.
- y: amount of money that can be spent in Store B.
- a: an array of length n, containing the cost of each jelly in Store A.
- b: an array of length n, containing the cost of each jelly in Store B.
- This procedure will be called exactly once.
- The procedure should return the maximum number of unique flavours of jelly Amy can purchase.

Example

Example 1

Consider the following call:

```
find_maximum_unique(2, 3, [2, 1, 4], [2, 3, 2])
```

This means that Amy can spend up to 2 dollars in Store A and 3 dollars in Store B, and the prices are as follows:

- Jelly 0 costs 2 dollars in both Store A and B,
- Jelly 1 costs 1 dollar in Store A and 3 dollars in Store B,
- Jelly 2 costs 4 dollars in Store A and 2 dollars in Store B.

The maximum number of unique flavours Amy can purchase is 2. This can be done by buying jelly 0 from Store A and jelly 2 from Store B for 2 dollars each.

Therefore, the procedure should return 2.

Example 2

Consider the following call:

```
find_maximum_unique(6, 12, [5, 1, 5, 6, 3], [3, 5, 4, 6, 7])
```

In this case, the maximum number of unique flavours Amy can purchase is 4. This can be done by purchasing jellies 1 and 2 from Store A, costing $1 + 5 = 6$ dollars, as well as jellies 0 and 4 from Store B, costing $3 + 7 = 10$ dollars.

Therefore, the procedure should return 4.

Constraints

- $1 \leq n \leq 2000$
- $0 \leq x, y \leq 10\,000$
- $0 \leq a[i], b[i] \leq 10\,000$ (for all $0 \leq i \leq n-1$)

Subtasks

1. (11 points) $x, y \leq 500$, $n \leq 12$
2. (24 points) $x, y \leq 500$, $n \leq 200$
3. (9 points) $y = 0$
4. (10 points) $b[i] = b[j]$ (for all $0 \leq i, j \leq n-1$)
5. (14 points) $a[i] = b[i]$ (for all $0 \leq i \leq n-1$)
6. (32 points) No additional constraints.

Sample grader

The sample grader reads the input in the following format:

- line 1: $\;n\;\;x\;\;y$
- line $2+i$ ($0 \leq i \leq n-1$): $\;a[i]\;\;b[i]$

The sample grader prints your answers in the following format:

- line 1: the return value of `find_maximum_unique`.