Task Overview Sheet
IOI 2004
Athens
Greece

Day-2

Version 1.1
Overview

TASK OVERVIEW SHEET / DAY-2

TASK		Phidias	Farmer	Empodia
CPU time limit per test		1 second CPU	1 second CPU	1 second CPU
Memory limit		16 MB	16MB	128 MB
Compiler options	**C**	`-pipe –static –O2 –DCONTEST –lm`		
	C++	`-pipe –static –include /usr/include/stdlib.h –O2 –DCONTEST –lm`		
	Pascal	`-So –O1 –XS –dCONTEST`		
Number of tests		20	20	20
Maximum points per test		5	5	5
Maximum total points		100	100	100

Program header comment when using C		`/*`
		`TASK: phidias`
		`LANG: C`
		`*/`
			`/*`
			`TASK: farmer`
			`LANG: C`
			`*/`
				`/*`
				`TASK: empodia`
				`LANG: C`
				`*/`

Program header comment when using C++		`/*`
		`TASK: phidias`
		`LANG: C++`
		`*/`
			`/*`
			`TASK: farmer`
			`LANG: C++`
			`*/`
				`/*`
				`TASK: empodia`
				`LANG: C++`
				`*/`

Program header comment when using Pascal		`(*`
		`TASK: phidias`
		`LANG: PASCAL`
		`*)`
			`(*`
			`TASK: farmer`
			`LANG: PASCAL`
			`*)`
				`(*`
				`TASK: empodia`
				`LANG: PASCAL`
				`*)`

Submission is accepted if:		Example input is solved.	Example input is solved.	Example input is solved.

Empodia

PROBLEM

The ancient mathematician and philosopher Pythagoras believed that reality is mathematical in nature. Present-day biologists study properties of biosequences. A biosequence is a sequence of M integers, which

- contains each of the numbers $0,1,\ldots,M-1$,
- starts with 0 and ends with $M-1$, and
- has no two elements $E,E+1$ in adjacent positions in this order.

A subsequence consisting of adjacent elements of a biosequence is called a *segment.*.

A segment of a biosequence is called a *framed interval* if it includes all integers whose values are between the value of the first element, which must be the smallest element in the segment, and the last element, which must be the largest and different from the first. A framed interval is called an *empodio* if it does not contain any shorter framed intervals.

As an example, consider the biosequence (0,3,5,4,6,2,1,7). The whole biosequence is a framed interval. However, it contains another framed interval (3,5,4,6) and therefore it is not an empodio. The framed interval (3,5,4,6) does not contain a shorter framed interval, so it is an empodio. Furthermore, it is the only empodio in that biosequence.

You are to write a program that, given a biosequence, finds all empodia (plural for empodio) in that biosequence.

INPUT

The input file name is `empodia.in`. The first line contains a single integer M: the number of integers in the input biosequence. The following M lines contain the integers of the biosequence in the order of the sequence. Each of these M lines contains a single integer.

OUTPUT

The output file name is `empodia.out`. The first line in this file is to contain one integer H: the number of empodia in the input biosequence. The following H lines describe all empodia of the input biosequence in the order of appearance of the starting point in the biosequence. Each of these lines is to contain two integers A and B (in that order) separated by a space, where the Ath element of the input biosequence is the first element of the empodio and the Bth element of the input biosequence is the last element of the empodio.

Task Description
IOI 2004
Athens
Greece

Day-2

Version 1.2
empodia

EXAMPLE INPUTS AND OUTPUTS

`empodia.in`

```
8
0
3
5
4
6
2
1
7
```

`empodia.out`

```
1
2 5
```

CONSTRAINTS

In one input, $1000000 \le M \le 1100000$. In all other inputs, $1 \le M \le 60000$.
Additionally, in 50% of the inputs, $M \le 2600$.

Farmer

PROBLEM

A farmer has a set of fields, each of which is surrounded by cypress trees. Also, the farmer has a set of strips of land, each of which has a row of cypress trees. In both fields and strips, between every two consecutive cypress trees is a single olive tree. All of the farmer's cypress trees either surround a field or are in a strip and all of the farmer's olive trees are between two consecutive cypress trees in a field or in a strip.

One day the farmer became very ill and he felt that he was going to die. A few days before he passed away he called his eldest son and told him, "I give you any Q cypress trees of your choice and all the olive trees which are between any two consecutive cypress trees you have chosen." >From each field and from each strip the son can pick any combination of cypress trees. Since the eldest son loves olives he wants to pick the Q cypress trees which will allow him to inherit as many olive trees as possible.

Field 1 has 13 cypress trees   Field 2 has 4 cypress trees   Field 3 has 8 cypress trees

Strip 1 has 4 cypress trees

Strip 2 has 8 cypress trees
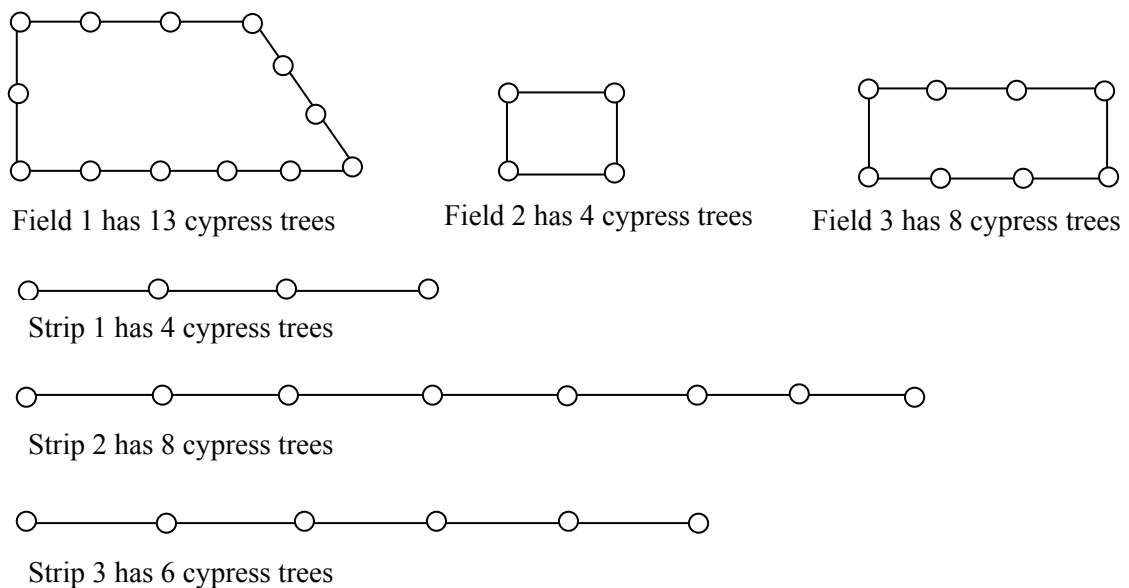
Strip 3 has 6 cypress trees

Figure 1. An example setting of cypress trees; olive trees are not shown.

In Figure 1, assume that the son is given Q=17 cypress trees. To maximize his olive inheritance he should choose all the cypress trees in Field 1 and Field 2, inheriting 17 olive trees.

You are to write a program which, given the information about the fields and the strips and the number of cypress trees the son can pick, determines the largest possible number of olive trees the son may inherit.

INPUT

The input file name is `farmer.in`. The first line contains first the integer Q: the number

---

of cypress trees the son is to select; then the integer M, the number of fields; and then the integer K, the number of strips. The second line contains M integers $N_1, N_2, \ldots N_M,$: the numbers of cypress trees in fields. The third line contains K integers $R_1, R_2, \ldots R_K$: the numbers of cypress trees in strips.

OUTPUT

The output file name is `farmer.out`. The file is to contain one line with one integer: the largest possible number of olive trees the son may inherit.

EXAMPLE INPUTS AND OUTPUTS

`farmer.in`   `farmer.out`

```
17 3 3
13 4 8
4 8 6
```

```
17
```

CONSTRAINTS

In all inputs, $0 \leq Q \leq 150000$, $0 \leq M \leq 2000$, $0 \leq K \leq 2000$, $3 \leq N_1 \leq 150$, $3 \leq N_2 \leq 150, \ldots 3 \leq N_M \leq 150$, $2 \leq R_1 \leq 150$, $2 \leq R_2 \leq 150, \ldots 2 \leq R_K \leq 150$. The total number of cypress trees in the fields and strips is at least Q. Additionally, in 50% of the inputs, $Q \leq 1500$.
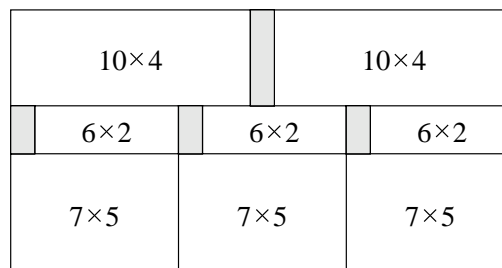
Phidias

PROBLEM

Famous ancient Greek sculptor Phidias is making preparations to build another marvelous monument. For this purpose he needs rectangular marble plates of sizes $W_1 \times H_1$, $W_2 \times H_2$, ..., $W_N \times H_N$.

Recently, Phidias has received a large rectangular marble slab. He wants to cut the slab to obtain plates of the desired sizes. Any piece of marble (the slab or the plates cut from it) can be cut either horizontally or vertically into two rectangular plates with integral widths and heights, cutting completely through that piece. This is the only way to cut pieces and pieces cannot be joined together. Since the marble has a pattern on it, the plates cannot be rotated: if Phidias cuts a plate of size $A \times B$ then it cannot be used as a plate of size $B \times A$ unless $A = B$. He can make zero or more plates of each desired size. A marble plate is wasted if it is not of any of the desired sizes after all cuts are completed. Phidias wonders how to cut the initial slab so that as little of it as possible will be wasted.

As an example, assume that in the figure below the width of the original slab is 21 and the height of the original slab is 11, and the desired plate sizes are 10×4, 6×2, 7×5, and 15×10. The minimum possible area wasted is 10, and the figure shows one sequence of cuts with total waste area of size 10.

Your task is to write a program that, given the size of the original slab and the desired plate sizes, calculates the minimum total area of the original slab that must be wasted.

INPUT

The input file name is `phidias.in`. The first line of input contains two integers: first W, the width of the original slab, and then H, the height of the original slab. The second line contains one integer N: the number of desired plate sizes. The following N lines contain the desired plate sizes. Each of these lines contains two integers: first the width W_i and then the height H_i of that desired plate size $(1 \leq i \leq N)$.

OUTPUT

The output file name is `phidias.out`. The file is to contain one line with a single integer: the minimum total area of the original slab that must be wasted.

EXAMPLE INPUTS AND OUTPUTS

`phidias.in`   `phidias.out`

```
21 11
4
10 4
6 2
7 5
15 10
```

```
10
```

CONSTRAINTS

In all inputs, $1 \leq W \leq 600$, $1 \leq H \leq 600$, $0 < N \leq 200$, $1 \leq W_i \leq W$, and $1 \leq H_i \leq H$. Additionally, in 50% of the inputs, $W \leq 20$, $H \leq 20$ and $N \leq 5$.