South African Computer Olympiad
Final Round
Day 2

Overview

Author	**Graham Poulter**	**Marco Gallotta**	**Harry Wiggins**
Problem	**forest**	**cheese**	**roads**
Source	forest.java forest.py forest.c forest.cpp forest.pas	cheese.java cheese.py cheese.c cheese.cpp cheese.pas	roads.java roads.py roads.c roads.cpp roads.pas
Input file	forest.in	cheese.in	roads.in
Output file	forest.out	cheese.out	roads.out
Time limit	1 second	4 seconds	1 second
Number of tests	10	10	20
Points per test	10	10	5
Total points	**100**	**100**	**100**

The maximum total score is 300 points.

How Not To Be Seen

Author

Graham Poulter

Introduction

(Based on the Monty Python sketch, "How Not To Be Seen")

You pride yourself on running a truly challenging survival course. In the last 15 years of the "How Not To Be Seen" survival course, there have been no survivors. Trainees are sent into a forest and given very specific instructions Not To Be Seen. Of course, some stick their heads out as soon as you call their name. That's a useless way not to be seen! Others just don't hide very well. However, there's usually one or two truly obstinate fellows that simply aren't anywhere to be seen, so each year you attach explosives to some of the trees in the forest. If there's no trees there's nowhere to hide, right? You have some ethical reservations about destroying a whole forest, but nevertheless it's a job that needs to be done (cue maniacal laughter).

Task

High explosives are expensive, so you want to use as few packs as possible, but you also don't want to leave any trees standing. Fortunately, exploding a tree knocks down trees one place to the left and right of it, each of which then knocks down its neighbour in a domino effect.

However a tree will only knock down a neighbour that is shorter (not taller or the same height), and neighbour means "one place to the left or right", no further.

Given the heights of a row of trees, your task is to choose which trees to explode and the order in which to explode them so that the entire forest is destroyed when they are detonated.

Example

Take for example the row of trees with heights 1, 2, 5, 5, 4, 6. The minimum number of explosive packs needed is 3, and these should be placed on trees 3, 4 and 6.

The explosives on tree 6 destroys trees 6 and 5. The explosives on tree 4 then knocks down tree 4. The explosives on tree 3 knocks down trees 3, 2 and 1, and the forest is gone.

Input (forest.in)

The first line of input contains N, the number of trees. The next N lines each contain one integer: h_1, h_2, ..., h_N, representing the heights of the trees $1, 2, \ldots, N$.

Sample input

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

Output (forest.out)

If K is the minimum number of trees to explode so that the forest is destroyed, then the output will have K lines, each line containing one integer t_1, t_2, ..., t_K, representing the trees to explode in the order in which you wish to explode them.

Sample output

```
3
4
6
```

Constraints

- $1 \leq N \leq 50000$
- $1 \leq h_i \leq 10000$

50% constraints

- $1 \leq N \leq 10000$

Time limit

1 second.

South African Computer Olympiad
Final Round
Day 2

Scoring

There are 10 test cases. On each test case you score 100% if you explode the minimum number of trees and destroy the forest.

You score 0% if you fail to destroy the forest, explode more trees than is optimal, or have some other error in your output.

South African Computer Olympiad
Final Round
Day 2

Cheese

Author

Marco Gallotta

Introduction

Wensleydale owns a shop which carries the sign:

> Ye Olde Cheese Emporium
> Henry Wensleydale, Purveyor of Fine Cheese
> to the Gentry and the Poverty Stricken
> Licensed for Public Dancing

Mousebender, who has a passion for good cheese, visited Wensleydale's place of purveyance to negotiate the vending of some cheese.

Now Wensleydale has just recently received a new stock of cheesy comestibles. Unfortunately though, the cheese is all packed in a row inside a cupboard with two doors in such a way that only the cheese at the two ends of the row can be reached at any time. Once a block of cheese is removed, the block next to it can be reached. However, once a block is removed it has to be sold that day. Cheese, like liqueur, matures over time. Wensleydale has customers, like Mousebender, that are prepared to pay more for a cheese the older it is. They are so concerned about the age of the cheese that they will increase the price they are prepared to pay for every day that it matures.

Wensleydale has got to know his customers well and it is guaranteed that he will always have exactly one customer purchase cheese from him each day. Wensleydale convinces his customers to purchase the cheese available that day by saying that he has nothing in stock except the cheese he wishes to sell. In this way Wensleydale can always choose which cheese to sell each day, provided the cheese can be reached.

Task

Wensleydale, like most people, would like to make the most money out of selling the cheese. But he has a problem — he cannot handle the complicated calculations. He has asked you to calculate the maximum amount he can sell the cheese for.

He explains to you that he has worked out what his customers are prepared to pay for a block of cheese. He knows that the value of the i^{th} block of cheese on the day he receives the cheese, day one, is v_i. His customers are prepared to pay $v_i \times a$ where a is the age of the cheese in days, with the age of the cheese on day one being equal to 1.

Cheese	v_i
Red Leicester	1
Tilsit	3
Caerphilly	1
Red Windsor	5
Stilton	2

Table 1: Example of the blocks of cheese in the cupboard

Example

Suppose that the blocks of cheese in the cupboard are the ones shown in Table 1. The blocks at the top and bottom are the ones accessible through the two doors.

The maximum amount can be obtained by selling the Red Leicester on day one for $1 \times 1 = 1$, the Stilton on day two for $2 \times 2 = 4$, the Tilsit on day three for $3 \times 3 = 9$, the Caerphilly on day four for $1 \times 4 = 4$ and the Red Windsor on day five for $5 \times 5 = 25$.

This results in a maximum amount of $1+4+9+4+25 = 43$.

Input (cheese.in)

The first line of input contains a single integer N, the number of blocks of cheese. The next N lines each contain a single integer v_i, the value of the i^{th} block of cheese on day 1.

Sample input

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

Output (cheese.out)

The ouptut should contain a single integer, the maximum amount all the blocks of cheese can be sold for.

Sample output

```
43
```

Constraints

- $1 \leq N \leq 2000$
- $1 \leq v_i \leq 1000$
- $1 \leq$ optimal answer $< 2^{31}$

50% constraints

- $1 \leq N \leq 20$

Time limit

4 seconds.

Scoring

An optimal answer scores 100%, while a sub-optimal or invalid answer scores 0%.

Roads

Author

Harry Wiggins

Introduction

In order to get from Camelot to the Castle Aargh, King Arther and the Knights of the Round Table were forced to cross the Bridge of Death, guarded by the old man from scene 24. The knights are now tired of often being forced to take a particular path between two castles, and want to build a number of new paths so that they will always have a choice of at least two separate routes.

Task

You will be given a description of the current layout of England. England consists of M castles, joined together by N roads. Each road directly connects two castles, and the knights travel only from one castle to the next using the roads. It is possible to get between any two castles in England using the roads. Determine the minimum number of new roads (each connecting two castles) that the knights must build so that there are at least two separate routes between any pair of castles. Routes are considered separate if they use none of the same roads, even if they visit the same castle.

There may already exist more than one road between the same pair of castles, and you may also build a new road that connects the same castles as another road.

Example

Figure 1 shows a possible layout of England (circles represent castles and lines represent roads). At present, going from castle 1 to castle 4 forces one to use the road between 1 and 2. However, if new roads are built from 1 to 6 and from 7 to 4, then there is always a choice.

Input (roads.in)

The first line contains two integers M and N, separated by a space. M is the number of castles (which are numbered from 1 to M) and N is the number of existing roads. The next N lines each contain two space-seperated integers, the numbers of the two castles at the ends of a road.

Sample input

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

Output (roads.out)

The output should contain a single integer, the least number of roads that the knights must build.

Sample output

```
2
```

Constraints

- $M - 1 \leq N \leq 10000$
- $3 \leq M \leq 5000$

50% constraints

- $M - 1 \leq N \leq 200$
- $3 \leq M \leq 100$

Time limit

1 second.

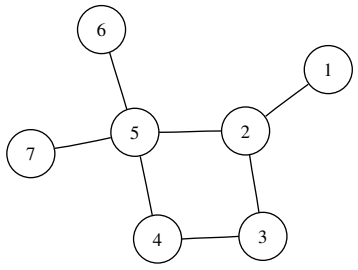

Figure 1: Example layout of the castles and roads in England

South African Computer Olympiad
Final Round
Day 2

Scoring

Each correct answer scores 100%, while an incorrect answer scores 0%.