Greetings From Globussoft

- Given below are 5 Programming questions, you have to solve any 3 out of 5 questions.

- These 5 questions you can attempt in any technology like C/C++, java, .Net

- To solve these 5 questions you've max. 3 hours.

- While solving these questions you are not allowed to use any **Search Engine** like Google, Yahoo, Bing …

All the best for your test

Globussoft

Question: - 1

Assume that you are a manager and there are m types of worker (numbered from 1 to m) and n types of task (numbered from 1 to n). There are a(i) workers of type #i and b(j) postitions for task #j. C(i, j) is the cost of hiring a worker of type #i to do the task of type #j. Your job is to minimize the cost of hiring workers to fill all the positions given that the total number of workers is equal to the total number of positions.

Input

The first line of input contains the number of test cases nTest (1<= nTest <= 10). Each test case contains:

- The first line contains the number of worker types - m and number of task types - n.
- The second line contains m positive integers: a(1), a(2), ..., a(m).
- The third line contains n positive integers: b(1), b(2), ..., b(n).
- Each of the next m lines contains n integers describing matrix C(i, j).

Notes:
1 <= m, n <= 200;
1 <= a(i), b(i) <= 30000;
1 <= C(i, j) <= 10000.
Sum of a(i) equals to sum of b(j).

Output

For each test case write the minimum cost in a separate line (it will fit in a signed 32-bit integer).

Example:

Input

```
2
3 4
3 6 7
2 5 1 8
1 2 3 4
8 7 6 5
9 12 10 11
4 4
1 3 5 7
2 4 2 8
```

1 4 7 3
4 7 5 3
5 7 8 3
5 3 6 8

Output
110
54

Question: - 2

A major cosmic battle was getting over. The Inter Galactic Super Power had been under attack, but it had defended itself quite well. It was about to launch its final retaliatory assault. But the number of enemy ships was quite large and they could scatter very easily. Their only hope, or so their Space Warfare expert said, was to bomb the enemies (who happened to be lined up in a long line!) using the strategy described below.

Because the number of ships will be a power of 2, to bomb all the ships (numbered 0 to 2N -1), the strategy to be used, which we will call Bomb Strat, goes like this:

1. Bomb it's first half, [0 to 2N-1 -1], in the left to right direction.
2. Of the remaining half, bomb its latter half part in reverse direction, i.e., bomb ships 2N-1, 2N-2,...., 2N-1+2N-2 in that order.
3. Then use Bomb Strat on the remaining ships: [2N-1 to 2N-1 + 2N-2 -1]

For example, when N=3, i.e., with ships numbered from 0 to 23 -1, this is what happens:

Step 1: Ships 0,1,2,3 get bombed in that order.
Step 2: Ships 7, 6 go down next.
Step 3: Now, the remaining ships [4, 5] are destroyed using the same strategy.

So the bombing is done in the order 0 -> 1 -> 2 -> 3 -> 7 -> 6 -> 4 -> 5. To make the job easier for the Inter Galactic Super Power's ships' pilots, they want to find which ship should be bombed when. This is your task. Given N, and the description of a ship, return the 0-based serial number of the bomb will blast it.

Input

T – the number of test cases, T<=50.

For each test case:

One line containing a binary number, describing the number of the place. The length of this string will equal N (it will be padded with leading zeroes if necessary). N<=30000.

Output

For each test case, output the index of a bomb, represented in the same format, as binary digits, whose length is exactly N.

Example:

Input

3
111
100
1100

Output

100
110
1011

Question: - 3

Inexperienced in the digital arts, the cows tried to build a calculating engine (yes, it's a cowmpouter) using binary numbers (base 2) but instead built one based on base negative 2! They were quite pleased since numbers expressed in base -2 do not have a sign bit.

You know number bases have place values that start at 1 (base to the 0 power) and proceed right-to-left to base^1, base^2, and so on. In base -2, the place values are 1, -2, 4, -8, 16, -32, ... (reading from right to left). Thus, counting from 1 goes like this: 1, 110, 111, 100, 101, 11010, 11011, 11000, 11001, and so on.

Eerily, negative numbers are also represented with 1's and 0's but no sign. Consider counting from -1 downward: 11, 10, 1101, 1100, 1111, and so on.

Please help the cows convert ordinary decimal integers (range -2,000,000,000 .. 2,000,000,000) to their counterpart representation in base -2.

Input

A single integer to be converted to base -2

Output

A single integer with no leading zeroes that is the input integer converted to base -2. The value 0 is expressed as 0, with exactly one 0.

Example:

Input
-13

Output
110111

Question: - 4

FJ has purchased N (1 <= N <= 2000) yummy treats for the cows who get money for giving vast amounts of milk. FJ sells one treat per day and wants to maximize the money he receives over a given period time. The treats are interesting for many reasons:

The treats are numbered 1..N and stored sequentially in single file in a long box that is open at both ends. On any day, FJ can retrieve one treat from either end of his stash of treats.

Like fine wines and delicious cheeses, the treats improve with age and command greater prices.

The treats are not uniform: some are better and have higher intrinsic value. Treat i has value v(i) (1 <= v(i) <= 1000).

Cows pay more for treats that have aged longer: a cow will pay v(i)*a for a treat of age a.

Given the values v(i) of each of the treats lined up in order of the index i in their box, what is the greatest value FJ can receive for them if he orders their sale optimally?

The first treat is sold on day 1 and has age a=1. Each subsequent day increases the age by 1.

Input

Line 1: A single integer, N
Lines 2..N+1: Line i+1 contains the value of treat v(i)

Output

The maximum revenue FJ can achieve by selling the treats.

Example:

Input
5
1
3
1
5
2

Output
43

Question: - 5

Technicians in a pathology lab analyze digitized images of slides. Objects on a slide are selected for analysis by a mouse click on the object. The perimeter of the boundary of an object is one useful measure. Your task is to determine this perimeter for selected objects.

The digitized slides will be represented by a rectangular grid of periods, '.', indicating empty space, and the capital letter 'X', indicating part of an object. Simple examples are

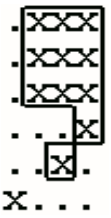
XX Grid 1 .XXX Grid 2
XX .XXX
.XXX
...X
..X.
X...

An X in a grid square indicates that the entire grid square, including its boundaries, lies in some object. The Xin the center of the grid below is adjacent to the X in any of the 8 positions around it. The grid squares for any two adjacent X's overlap on an edge or corner, so they are connected.

XXX
XXX Central X and adjacent X's
XXX

An object consists of the grid squares of all X's that can be linked to one another through a sequence of adjacent X's. In Grid 1, the whole grid is filled by one object. In Grid 2 there are two objects. One object contains only the lower left grid square. The remaining X's belong to the other object.

The technician will always click on an X, selecting the object containing that X. The coordinates of the click are recorded. Rows and columns are numbered starting from 1 in the upper left hand corner. The technician could select the object in Grid 1 by clicking on row 2 and column 2. The larger object in Grid 2 could be selected by clicking on row 2, column 3. The click could not be on row 4, column 3.



One useful statistic is the perimeter of the object. Assume each X corresponds to a square one unit on each side. Hence the object in Grid 1 has perimeter 8 (2 on each of four sides). The perimeter for the larger object in Grid 2 is illustrated in the figure at the left. The length is 18.

Objects will not contain any totally enclosed holes, so the leftmost grid patterns shown below could NOT appear. The variations on the right could appear:

Impossible Possible

XXXX XXXX XXXX XXXX
X..X XXXX X... X...
XX.X XXXX XX.X XX.X
XXXX XXXX XXXX XX.X
....
..X.. ..X.. ..X.. ..X..
.X.X. .XXX. .X... .....
..X.. ..X.. ..X.. ..X..
....

The input will contain one or more grids. Each grid is preceded by a line containing the number of rows and columns in the grid and the row and column of the mouse click. All numbers are in the range 1-20. The rows of the grid follow, starting on the next line, consisting of '.' and 'X' characters. The end of the input is indicated by a line containing four zeros. The numbers on any one line are separated by blanks. The grid rows contain no blanks.

For each grid in the input, the output contains a single line with the perimeter of the specified object.

Example:

Input

```
2 2 2 2
XX
XX
6 4 2 3
.XXX
.XXX
.XXX
...X
..X.
X...
5 6 1 3
.XXXX.
X....X
..XX.X
.X...X
..XXX.
7 7 2 6
XXXXXXX
XX...XX
X..X..X
X..X...
X..X..X
X.....X
XXXXXXX
7 7 4 4
XXXXXXX
XX...XX
X..X..X
X..X...
X..X..X
X.....X
XXXXXXX
0 0 0 0
```

Output

8
18
40
48
8